


RESEARCH

Open Access



Prerequisites-based course recommendation: recommending learning objects using concept prerequisites and metadata matching

Abdessamad Chanaa^{1*}  and Nour-eddine El Faddouli¹

*Correspondence:
abdessamad.chanaa@gmail.com

¹RIME Team, MASI Laboratory,
E3S Research Center,
Mohammadia School
of Engineers, Mohammed
V University in Rabat, Rabat,
Morocco

Abstract

The recommendation is an active area of scientific research; it is also a challenging and fundamental problem in online education. However, classical recommender systems usually suffer from item cold-start issues. Besides, unlike other fields like e-commerce or entertainment, e-learning recommendations must ensure that learners have the adequate background knowledge to cognitively receive the recommended learning objects. For that reason, when designing an efficient e-learning recommendation method, these challenges should be considered. To address those issues, in this paper, we first propose extracting pairs concept prerequisites using Linked Open Data (LOD). Then, we evaluate the proposed list of prerequisite relationships using machine learning predictive models. Then, we present the recommendation approach based on matching concept's prerequisites relation and courses metadata through a similarity score. The experimental result of prerequisite identification was evaluated using four well-known machine learning algorithms while achieving an accuracy of 90%. Moreover, using three known evaluation metrics, the final prerequisite-based recommendation demonstrates very good results (NDCG@10 = 86%). This solution will enhance recommendations on online learning platforms. Additionally, it will overcome the cold-start issue and accomplish the needed prerequisites and background knowledge for learners to attain their learning objectives.

Keywords: Content-based recommendation, Course metadata, Prerequisites identification, Machine learning, Cold-start

Introduction

The evolution of the internet and online technology has broadened the range of products available in a variety of fields, including e-commerce, entertainment and e-learning. In distance education, it becomes very difficult for learners to find suitable learning objects that match their preferences and needs due to the big problem of information overload. Therefore, recommender systems are meant to guide learners to different learning objects that might be of interest to them. Recommender systems can automatically match learners with different resources appropriate to their needs, knowledge

background, and learning objectives. In this direction, various approaches have been introduced to improve the recommendation performance of recommender systems in e-learning (De Medio et al., 2020; Khanal et al., 2020; Chanaa and El Faddouli, 2021).

Recommender systems are typically classified into collaborative filtering systems and Content-based filtering. Content-based recommendations promote items to a user based on the item's history and the user's profile. Collaborative recommendations are based on recommending items that are already promoted by users who share similar tastes and interests. These methods are widely used in e-learning. However, their major challenge is the item cold-start (Schein et al., 2002). The system is unable to select new relevant items that have never been selected or rated previously by learners, and keeps recommending old (already seen and rated) ones. Only a few systems can fully exploit the unseen learning objects for personalization. One way to overcome the item cold-start problem is to create a pure content-based recommender system. This kind of system is based only on matching the current studied concepts with potential learning objects without the learner's or item's history intervention (Sun et al., 2017, 2018). Those systems can be integrated into a mixed or switched hybrid recommender system along with a collaborative recommender system to reduce the item cold-start problem.

Although content-based recommender systems that are based on item metadata may provide good recommendation results and overcome the new item cold-start. In the education field, this is still not enough to provide learners with accurate learning objects, recommender systems only rely on learners' past experience without taking into consideration the lack of background knowledge. On the contrary to any other field like entertainment or e-commerce, learners may not fulfil the needed prerequisites or lack the right background knowledge to accept the recommended learning objects. This may significantly reduce learners' motivation. Particularly, the recommendation in e-learning is usually provided to alleviate learners' dropout or demotivation problems due to learner's difficulties in following and understanding the enrolled courseware. To be more precise, when a learner learns a new concept, he or she may need to know a prior concept from another course (Liang et al., 2015). As a result, the two concepts have a prerequisite relationship. Learners should master the concepts in a specific order, which is an important factor that recommender systems should consider while recommending new learning objects.

The E-learning prerequisite-based recommendation system is an essential tool for assisting students throughout their educational path. There are many inspirations behind its creation. First, it improves retention as learners with a solid foundation in prerequisite knowledge are more likely to acquire and understand new concepts, resulting in improved comprehension (Novak, 1990). Also, Instructors can keep students motivated and engaged by offering content and tailoring the learning path to their pace. It is important to consider their knowledge and ensure that they have acquired the prerequisites. This approach allows students to progress comfortably without feeling overwhelmed or bored, ultimately creating an environment, for learning.

Considering that individuals come from backgrounds and possess varying levels of knowledge, the main aim of this research is to ensure that learners have the greatest opportunity, for success by commencing with the appropriate foundation of knowledge (Yang et al., 2015). To accommodate this diversity, it is suitable to incorporate

learning materials based on prerequisites. Such an approach fosters a learning environment, reduces frustration levels, and ultimately improves overall outcomes.

These recommendation algorithms are indeed designed to meet the individual needs of each student, taking into account their distinct prior knowledge and learning foundation and pointing them in the direction of the best place to begin their educational journey. It is now possible to develop systems that, when compared to traditional methods, can analyse and propose required learning items more correctly, thanks to advancements in machine learning algorithms and Open Linked Data (OLD) (Mountantonakis and Tzitzikas, 2017). Machine learning and semantic research will aid in the analysis of learning item information and connecting it with an appropriate learning route. These systems may then continuously improve their suggestions, guaranteeing that students receive the most pertinent and useful guidance that meets their learning basis and background knowledge.

To address the above challenges, we propose an effective content-based recommender systems that recommend learning objects matching concept prerequisites. More precisely, we first suggest a method to extract the potential concept prerequisites using Linked Open Data (LOD), and then identify the real prerequisites using a supervised machine learning algorithm. Second, we propose a method that matches those prerequisites with different metadata contents of learning objects. The matching score will be a factor that decides if a learning object is a good match to the initial learning concept (i.e., learning concept prerequisites). The following research questions have been addressed in this investigation:

RQ1 What is the impact of using OLD and semantic search in identifying potential prerequisites of a course/material?

RQ2 What methodology is most effective in listing potential prerequisites of a course/material?

RQ3 Does the prerequisite-based recommendation system present better results compared to the classical content-based recommendation in education?

RQ4 To what extent do recommended prerequisites impact the overall learning outcomes of learners?

The remainder of this paper is organized as follows. In Section “[Theoretical background](#)”, the background of different theoretical aspects has been presented. Section “[Related works](#)” presents the recent literature review of content-based recommendations as well as prerequisite-based recommendations in e-learning. Section “[Methodology](#)” highlights the methodology of our approach, it includes the problem definition, the system architecture and the recommendation process. We provide exhaustive experiment details in Section “[Experimentation](#)” describing the data, metrics, settings and results’ discussion. Finally, Section “[Conclusion and future work](#)” gives a summary presenting the conclusion and future works.

Theoretical background

For a better understanding of the study, we present in this section the theoretical background of key concepts. This includes an overview of the content-based recommendation system, concept prerequisites in education, and course metadata.

Content-based recommendation

Recommender systems represent an active area of scientific research. They appear in the mid-1990s as a system that was designed to suggest items based on users' past preferences and explicit ratings (Adomavicius and Tuzhilin, 2005). Nowadays, recommender systems are still a problem-rich research area. It is utilized to solve the problem of information overload in various study domains, such as social media, e-commerce, news, movies, retails, etc (Manning et al., 2008).

Basically, there are two classical types of recommender systems: collaborative recommendation (Schafer et al., 2007) and content-based recommendation (Lops et al., 2011). Collaborative recommendations are based on suggesting items already liked by users who share similar preferences and interests. The similarity in preferences of the two users is determined based on their rating history. If two users have the same rate for similar items, then they are presumed to have close tastes. On the other side, the content-based recommendation does not require a rating history by other users. It recommends items that are similar to the ones the user judged relevant in the past by selecting visited, shared, downloaded, or labelled items.

The content-based recommendation can help to avoid item cold-start (a new item that has either none or very few historical interactions in the system). Item-to-item content-based recommender systems (also referred to as the information retrieval problem) can achieve the recommendations by only using similar items' metadata and requiring no historical data about users' experiences. Depending on the recommended items, this metadata usually includes the item's title, type, category, price, source, author, etc.

There are two main techniques for content-based recommendation to compute the similarity between items: Term Frequency Inverse Document Frequency (TF-IDF) and Cosine similarity. Based on Eq. 1, TF-IDF weighs a term t in any item's metadata d and assigns a value to it depending on how many times it appears in the whole corpus D (Jones, 1972). The more relevant the term is, the higher is the TF-IDF score.

$$TF - IDF(t, d, D) = TF(t, d) \times IDF(t, D) \tag{1}$$

where

$$TF(t, d) = \log(1 + freq(t, d))$$

$$IDF(t, d) = \log\left(\frac{N}{1 + count(d \in D : t \in d)}\right)$$

where $N = |D|$ is the number of documents in the corpus D . $count(d \in D : t \in d)$ presents the number of documents where the term t appears.

On the other side, The cosine similarity measures the angle between two vectors (Salton and Buckley, 1988). Where each vector presents an item's attributes in an n-dimensional space, the angles between the items' vectors (items) are computed to define the

similarity between those items. It is calculated by Eq. 2. The values of $Simil(A, B)$ range from -1 to 1 , where 1 means that the two items (the two vectors A and B) are totally similar, while -1 means they are completely different.

$$\begin{aligned} Simil(A, B) &= \cos(\theta_{A,B}) = \frac{A \cdot B}{\|A\| \cdot \|B\|} \\ &= \frac{\sum_{i=1}^n A_i \cdot B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}} \end{aligned} \quad (2)$$

Concept prerequisites

From a pedagogical perspective, a prerequisite is a concept or skill that is needed to be learned before proceeding to learn more advanced skills/knowledge (Liang et al., 2017). Prerequisite dependencies exist as natural relations among concepts in cognitive processes when we learn, organize, apply, and generate knowledge (Laurence and Margolis, 1999). Comprehending a prerequisite concept ensures that learners have the prior required knowledge to easily understand the new concept. It ensures a logical sequence of concept understanding and proficiency. Also, it helps learners to build a feeling of ease and confidence about the new concepts and skills (Gaspiretti et al., 2018).

For example, concepts of the course ‘Statistics’ are generally prerequisites to the course ‘Machine Learning’ e.g. The statistic concept ‘Correlation between variables’ is a prerequisite to ‘Logistic regression’ machine learning algorithm.

Course metadata

Metadata can be defined as data about data (Dictionary, 2002). It presents a description and context about that data that constructs an entity. Metadata helps understand, structure and organize data about any entity. In online education, metadata can describe either the concept knowledge or modularized content resources (Fischer, 2001). Concept knowledge describes knowledge ontologies (an ontology is a set of terms and formal definitions), while modularized content contains the actual content that is linked to the concepts of an ontology. There are many open standards to define online course metadata, such as IMS Metadata (Consortium et al., 2003), Dublin Core (Weibel and Koch, 2000) and IEEE LTSC LOM (Robson, 2012).

Those standards include definitions of metadata such as title, identifier, subject, topic, keywords, description, format/type of the content, source, provider, price, authors, tutor’s name, etc. Some standards like IEEE LTSC LOM allow to explicitly specify prerequisites as course metadata. This metadata information can be used in data retrieval applications or content-based recommendation systems to efficiently retrieving and selecting the right learning objects from the online repositories.

Related works

In this section, we review the state-of-the art including content-based recommendation techniques in online education as well as prerequisite-based content recommendation.

Content-based recommendation in online education

The widely used collaborative filtering recommendation algorithm only takes into account user's score history data and ignores the user's identification attributes and, most importantly, the course metadata information. On the other hand, the content-based recommendation algorithm can make better use of the data, recommending the course that is most similar to the one that the user has previously enrolled in (Al-Badarenah and Alsakran, 2016). De Medio et al., (2020), proposed a recommender system plug-in of the Moodle Learning Management System (LMS), it suggests a ranked list of learning objects using keyword-based query and the repository quality. However, this method is relying on tutors' queries rather than learners, and it is not adaptive to any learning objects metadata. Wan and Niu (2018), constructed a spontaneous and autonomous content-based recommendation using learning objects interactions and self-organization theory. Huang and Lu (2018), suggested a content-based course recommender model for MOOCs using a content analyser. However, this recommendation is only based on the TF-IDF weighting algorithm for feature term weighting, and does not consider concepts relation. In the same direction, Zhang et al. (2019) built an accurate resource recommendation model (MOOCRC) in MOOC environments using course content attribute features and learner's behaviour, but this study as most existing recommendation systems suffers from the item cold-start. Shu et al. (2018), adopted a content-based recommendation algorithm based on a Convolutional Neural Network (CNN). CNN is used to predict the latent factors from learning object metadata. Although the efficiency and the significant results of this method, it does not consider the weight/importance of each metadata feature (some metadata are not relevant for the recommendation), also the recommendation is general rather than one concept focused.

Prerequisite-based recommendation in online education

Course prerequisite relationships are generally defined by the educational program designers and domain experts. However, manual labelling cannot answer the requirement of an increasingly massive number of available materials. In open and online education, automatic prerequisite extraction and definition is an important research topic (Manrique et al., 2019; Gasparetti et al., 2018; Li et al., 2020). This new research motivation leads to many prerequisite-based applications, such as recommendations. In this direction, Dai et al. (2021) developed a course-terminology prerequisite relatedness for job-oriented learning goals using Markov Decision Process-Based Ordering (MDPBO). Pang et al. (2019) proposed a locating-based MOOC recommendation method with consideration of prerequisite relationship. However, this method is a basic content-to-content prerequisite-based similarity approach and does not solve the content cold-start problem. Fabbri et al. (2018), adopted a manual collection of good quality resources related to Natural Language Processing (NLP) with topic modelling and prerequisite relations among topics for resource recommendation. Nevertheless, the prerequisite dataset quality, as well as the relevance of each prerequisite concept compared to the learning resources are not well considered. Jing and Tang (2017), conducted an efficient investigation on the student's behaviour modelling problem for course recommendation using course prerequisites to better reveal users' potential choice. Zhao et al. (2020),

presented a new recommendation method named “GuessUNeed” based on extracting concept-level and course-level prerequisite relations. This method computes efficiently the concept’s distribution feature. Although it has great efficiency, it takes implicit feedback of user-course interactions and does not consider metadata importance into content-to-content similarity computation. Most of the methods presented do not solve the problem of learners’ lack of background knowledge and are based on simple correspondence between terms. Moreover, they are limited to considering the weight of each metadata section of the learning object. Our aim is to create a recommendation method based on the correspondence between prerequisite concepts and metadata sections. The particularity of this method is that it takes into account semantically constructed prerequisites based on public semantic datasets in order to adequately define the appropriate prerequisite for a certain concept, and then recommends the learning object based on the defined prerequisite list. This contribution also has a strong point over other methods: it assigns (explicitly and implicitly) a weight to each metadata section of the learning object based on a well-studied equation and according to the desired term. In this way, and unlike other methods, the recommendation will be more precise and have better accuracy, since not all metadata contributes with equal importance to the recommended learning object.

Methodology

In this section, the methodology is conducted, directing the study. The section commences with the introduction of the problem formulation, followed by an explanation of the system architecture and the detailed recommendation method process.

Problem definition

A course corpus D is composed of n courses in the same subject field denoted as $D = \{C_1, \dots, C_i, \dots, C_n\}$ where C_i is the i -th course in the corpus. Each course C consists of m sequential metadata information, as $C_i = \{M_{i,1}, \dots, M_{i,j}, \dots, M_{i,m}\}$ where $M_{i,j}$ is the j -th metadata section of the course C_i . Each Metadata section M is viewed as a document of text where each document (single course metadata) is composed of l terms as $M_{i,j} = \{s_{i,j,1}, \dots, s_{i,j,k}, \dots, s_{i,j,l}\}$ where $s_{i,j,k}$ is the k -th term in the metadata (document) $M_{i,j}$.

A concept corpus is a set of concepts denoted by V where $V = K_1 \cup K_2 \dots \cup K_i \dots \cup K_n$ Where K_i is the set of concepts in the course C_i of the corpus D . K_i can be defined as a p -gram (contiguous sequence of p samples of terms) as $K_i = \{K_{i,1}, \dots, K_{i,q}, \dots, K_{i,p}\}$ where $K_{i,q}$ is the q -th concept of the K_i concept set of the course C_i . The set of concept K_i represents the content of learning metadata of a course C_i , i.e., the p -gram concept terms are represented among the l terms of each of the m metadata sections representing the course C_i .

The concept prerequisite relation can be considered as the dependency between two concepts $K_{i,a}$ and $K_{j,b}$. Each single concept $K_{i,a}$ has a concept-prerequisites list $PL_{K_{i,a}}$ that defines r possible prerequisites as $PL_{K_{i,a}} = \{K_{a_1}, K_{a_2}, \dots, K_{a_r}\}$. As explained in Fig. 1, the existence of a prerequisite relation between $K_{i,a}$ and $K_{j,b}$, is denoted by $\langle K_{i,a}, K_{j,b} \rangle$. More precisely, $K_{i,a}$ of the course C_i is a prerequisite concept of $K_{j,b}$ of

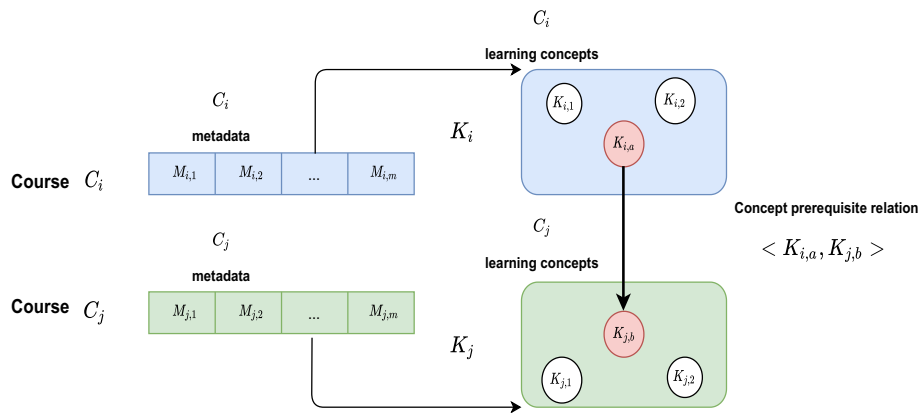


Fig. 1 An example of concept prerequisite relations

the course C_j , i.e., $K_{j,b}$ is a follow-up concept of $K_{i,a}$. Equation 3 presents the function $F(K_{i,a}, K_{j,b})$ that maps two concepts, $K_{i,a}$ and $K_{j,b}$. It is defined as follows:

$$F(K_{i,a}, K_{j,b}) = \begin{cases} 1, & \text{if } K_{j,b} \in PL_{K_{i,a}} \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

Given the course corpus D with the metadata sections M of each course C , the concept corpus V and the prerequisite function F . Our objective is to associate each given concept $K_{i,a}$ with all possible concept-prerequisites list $PL_{K_{i,a}}$, then perform a prerequisite content-based course recommendation of the n courses, by mapping the similarity between course metadata terms M and concept-prerequisites list PL . This recommendation takes into consideration the course difficulty level L for accurate mapping results. More precisely, the proposed system is dedicated to giving an ordered courses list $R_{K,L}$ where courses are ordered by the score function that computes the importance of each concept (and their prerequisites) of each course metadata information. Equation 4 presents the prerequisite-based course recommendation task:

$$G(K, PL_K, L, D) \rightarrow R_{K,L} \tag{4}$$

System architecture

Figure 2 presents the process of prerequisite-based course recommendation. It includes three principal components: prerequisite identification, concept similarity calculation, and course recommendation.

The learner follows a course with a learning concept K and course difficulty level L . First, the prerequisite identification system selects matching prerequisites with the course concept. Then, the system computes the similarity between different metadata sections and concept prerequisites (terms) extracted by the prerequisite identification system. Last, the model selects suitable learning objects that match the course difficulty level and the concept prerequisites.

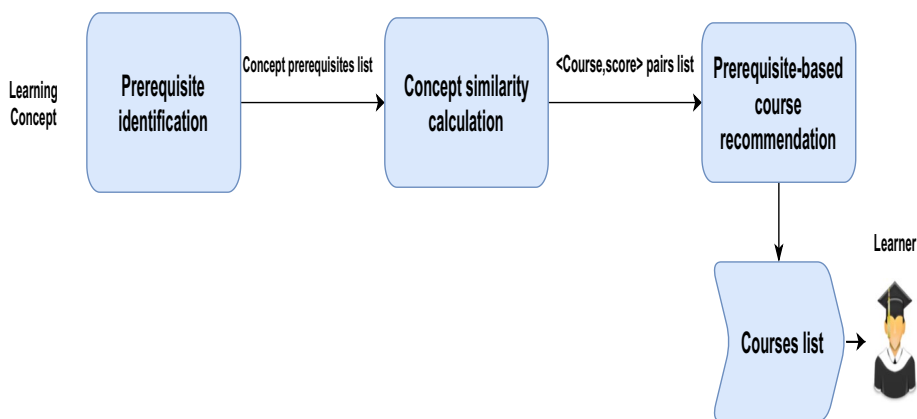


Fig. 2 System architecture

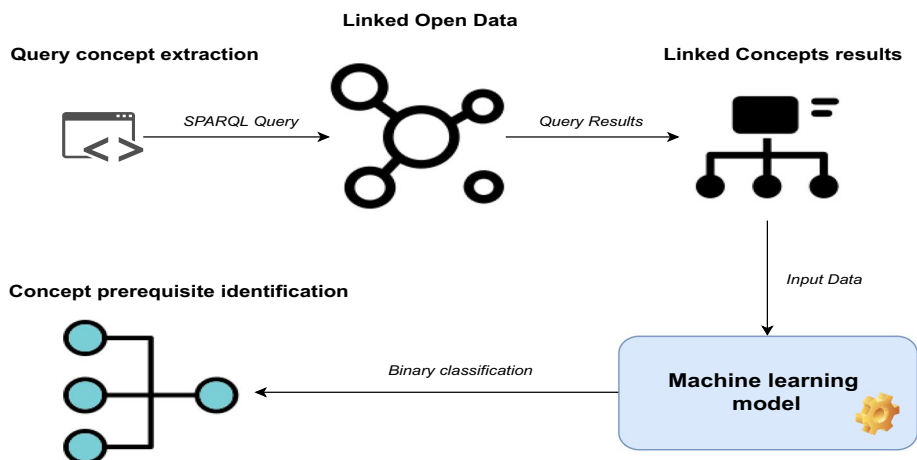


Fig. 3 Prerequisite identification

Prerequisite identification

Inspired by Manrique et al. (2019), The prerequisites’ identification step of our study is based on accurately identifying candidate concept prerequisites using the semantic web. Then, we aim to evaluate the prerequisite relation between the concept target K_a and their concept prerequisites candidates via a supervised machine learning algorithm. This method has proved (theoretically and empirically) an accurate and adequate method to create concepts prerequisites corpus. It is important to note that this is a separate study to extract meaningful concepts and their prerequisites using semantic web data. As shown in Fig. 3, the results will be a prerequisite corpus of all concept candidates.

The strength of semantic web research is to connect all available data on the web using Linked Open Data (LOD) through Resource Description Framework (RDF) standard to present and describe resources on the web, and SPARQL query language to extract those linked data. In our case, linked data are any data that have a semantic relation (maybe a prerequisite relation or not) to the initial concept. Figure 4 is a SPARQL query example to extract concepts semantically related to the «regression» concept. While Fig. 5

```

select ?concept_C ?concept
  where{
    ?concept_C ?relation ?concept .
    filter ( ?relation in (skos:primarySubject,skos:related,skos:broader,skos:topConceptOf)).
    filter (contains(str(?concept_C), "regression")).
  }
limit 500

```

Fig. 4 Example of query to extract prerequisites of concept "Regression"

concept_C	concept
<http://dbpedia.org/resource/Category:Robust_regression>	<http://dbpedia.org/resource/Category:Regression_analysis>
<http://dbpedia.org/resource/Category:Nonparametric_regression>	<http://dbpedia.org/resource/Category:Regression_analysis>
<http://dbpedia.org/resource/Category:Nonparametric_regression>	<http://dbpedia.org/resource/Category:Nonparametric_statistics>
<http://dbpedia.org/resource/Category:People_with_caudal_regression_syndrome>	<http://dbpedia.org/resource/Category:Rare_diseases>
<http://dbpedia.org/resource/Category:People_with_caudal_regression_syndrome>	<http://dbpedia.org/resource/Category:People_by_medical_or_psychological_condition>
<http://dbpedia.org/resource/Category:Categorical_regression_models>	<http://dbpedia.org/resource/Category:Single-equation_methods_(econometrics)>
<http://dbpedia.org/resource/Category:Logistic_regression>	<http://dbpedia.org/resource/Category:Logarithms>
<http://dbpedia.org/resource/Category:Robust_regression>	<http://dbpedia.org/resource/Category:Robust_statistics>
<http://dbpedia.org/resource/Category:Logistic_regression>	<http://dbpedia.org/resource/Category:Categorical_regression_models>
<http://dbpedia.org/resource/Category:Categorical_regression_models>	<http://dbpedia.org/resource/Category:Categorical_variable_interactions>
<http://dbpedia.org/resource/Category:People_with_caudal_regression_syndrome>	<http://dbpedia.org/resource/Category:People_with_disabilities>
<http://dbpedia.org/resource/Category:Categorical_regression_models>	<http://dbpedia.org/resource/Category:Regression_models>
<http://dbpedia.org/resource/Category:Logistic_regression>	<http://dbpedia.org/resource/Category:Statistical_classification>

Fig. 5 Example of query results of prerequisites candidate of the concept "Regression"

presents the concepts' results of the given query (semantically related to the «regression» concept).

After extracting all related concepts (prerequisite candidates), the approach aims to perform a binary classification based on a set of features that are extracted from prerequisites and a corpus of documents. The principal aim of using corpus-based features is to examine the co-occurrence of the concepts in documents. For example, for the given concept pair (K_a, K_b) , if in most of the documents where K_a appears, K_b also occurs, but not vice-versa, it is more likely that K_b is a prerequisite of K_a . The prerequisite candidates are identified using the following features:

- $P(K_a)$: is the probability of finding a document that contains the K_a concept in the corpus. It is defined as: $P(K_a) = \frac{\text{Documents that contains } K_a}{\text{Total documents in Corpus}}$
- $P(K_b)$: is the probability of finding a document that contains the K_b concept in the corpus.
- $P(K_a/K_b)$: is a conditional probability, it examines the occurrences of K_a in the documents where K_b exists.
- $P(K_b/K_a)$: presents the occurrences of K_b in the documents where K_a exists.
- $P(K_b, K_a)$: presents the joint probability. It is defined as: $P(K_b, K_a) = P(K_a/K_b) \times P(K_b)$
- $PD(K_a, K_b)$: presents the portion of documents where both concepts occur in the corpus. It is defined as: $\frac{\text{Documents that contain } k_a \cup \text{Documents that contain } k_b}{\text{Total documents in Corpus}}$

Finally, to evaluate the concept-related corpus, machine learning techniques are used to model the problem as a binary classification. The model output labels are «Prerequisite» and «Not Prerequisite». In such a way, it will be possible to build an accurate corpus of concepts and their prerequisites, properly extracted and processed.

Concept similarity calculation

As stated in Section “Problem definition”, each course contains different metadata sections that describe the course. Each course represents a union of different metadata sections, which represents a single document (a document expresses a bag of words). In the same way, each metadata section presents a contiguous sequence of terms that can also be expressed as a document.

Our purpose is to calculate a concept similarity in each document, i.e., the importance and occurrence of a concept in producing a single relevance score for each course document (union of metadata documents). Based on the Elasticsearch similarity algorithm (Gormley and Tong 2015) and Lucene’s practical scoring function¹, the approach computes the relevance of a concept *K* in a course document *C* (consists of multiple metadata sections) using the similarity function *S(K, C)* as follows:

$$S(K, C) = queryNorm(K) \times coord(K, C) \times \sum_{norm(t, C)(t \text{ in } K)} (TF(t \text{ in } C) \times IDF(t)^2 \times t.getBoost()) \times \tag{5}$$

where *K* presents the query concept to match with, and *C* presents the course document to search for that query. ‘*t*’ presents the terms contained in the concept query (e.g., the concept «Linear regression» has two terms, «Linear» and «regression»). The *TF(t)* and *IDF(t)* present respectively the Term Frequency and the Inverse Document Frequency for the term *t* in course *C* as explained in equations 1 of section . *queryNorm(K)* is the inversed squared sum of the *IDF* of each term in the concept query *K*, denoted as ‘sum Of Squared Weights’. ‘*queryNorm(k)*’ is defined as:

$$queryNorm(k) = \frac{1}{\sqrt{sum \text{ Of Squared Weights}}}$$

Coord(K, C) counts the number of terms from the query concept that appears in the document, it is defined as:

$$Coord(K, C) = \frac{term \ score \times \ number \ of \ matching \ terms}{total \ number \ of \ terms \ in \ the \ query}$$

t.getBoost() is an absolute number (greater than 1) that can be used to explicitly boost one metadata field more than the others. *norm(t, C)* is the inverse square root of the number of terms in the metadata field (which makes a small metadata field like ‘title’ more important than a longer one like ‘description’). It is defined as:

$$norm = \frac{1}{\sqrt{num \ Field \ Terms}}$$

¹ <https://www.elastic.co/guide/en/elasticsearch/guide/current/practical-scoring-function.html>

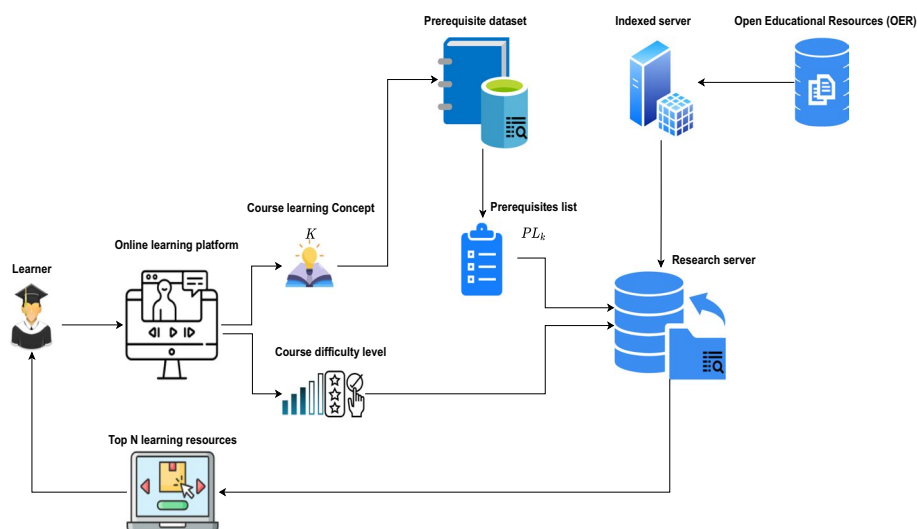


Fig. 6 Prerequisite-based course recommendation workflow

This is very important to our method, as it helps to identify that one metadata field is more important than others.

Prerequisite-based course recommendation

In this section, a new content-based recommendation method for learning resources is based on course prerequisites relation and the course difficulty level is proposed. As shown in Fig. 6, any learner following a course sequence can benefit from a recommendation at any triggered recommendation event(e.g., end of the chapter, end of the learning sequence, etc). The followed course has a difficulty level L and a learning concept K . Based on the study in Section “Prerequisite identification”, all prerequisite concepts of the concept K are extracted from a pre-build dataset to create a prerequisite list PL_k . The prerequisite list, along with the course difficulty level, are input to the metadata research engine. The research results are based on the similarity Eq. 5.

Since content-based recommendation systems are basically information retrieval applications, it is necessary to index the dataset for fast and accurate matching between query (prerequisites list) and metadata fields. This research engine indexes learning objects metadata from external Open Educational Resources (OER). It is important to note that this prerequisite-based content recommendation system is independent of any learner’s metadata or learning object’s history, but only to the course concept and difficulty level. This is very significant as it will help the recommendation system to overcome the learning object (item) cold-start. Also, this system is easy to integrate with any collaborative recommender system as a new switched hybrid recommender system, which could help to enhance recommender systems results in education.

Table 1 Features of concept prerequisite pairs

Conc-ept	Pre-req	$P(K_a)$	$P(K_b)$	$P(K_a/K_b)$	$P(K_b/K_a)$	$P(K_a, K_b)$	$PD(K_a, K_b)$	Label
Linux programming	Video game development	0.0004	0	0	0	0	0	Not prerequisite
Classification algorithms	Machine learning	0.00377	0.18876	0.00358	0.0247	0.00067	0.00355	Not prerequisite
Cloud data-bases	Cloud storage	0.00022	0.00444	0.00288	0.00288	0.00001	0.00022	Prerequisite

Experimentation

In this section, an experiment was conducted to provide concrete details about our approach and verify its feasibility. A full description of the used dataset, experimentation settings and discussions are presented in the following sections.

Dataset

Concept prerequisite pairs corpus

The data used for prerequisites extraction are basically extracted from DBpedia. DBpedia is a semantic web dataset that allows users to semantically query properties and relationships of Wikipedia resources. Prerequisites candidates are extracted using Simple Knowledge Organization System (SKOS) predicates (e.g., skos:primarySubject, skos:related, skos:topConceptOf, etc). It is important to note that the corpus presented in the Course corpus section (Section “[Course corpus](#)”) is also used to extract the corpus-based features previously explained in Section “[Prerequisite identification](#)”. The corpus was used to calculate $P(K_a)$, $P(K_b)$, $P(K_a/K_b)$, $P(K_b/K_a)$, $P(K_a, K_b)$ and $PD(K_a, K_b)$. Table 1 exhibits examples of the used corpus to train the model and determine the prerequisite relation between two concepts.

In addition to prerequisites extracted semantically from DBpedia, we gathered public pairs of prerequisites dataset from the following resources:

- *Course Prerequisite Relation dataset* (Pan et al., 2017): it investigates the potential prerequisite relation between knowledge concepts by proposing a representation learning-based method. This dataset contains three MOOC corpus of concept prerequisite pairs with different domains: “Data structure and Algorithm” and “Machine Learning”.
- *CPR-Recover dataset* (Liang et al., 2017): it recovers concept prerequisite relations from 11 U.S. universities and their concept pairs with prerequisite labels. This dataset contains concept prerequisite pairs of the category “Computer science”.
- *LectureBank2.0 dataset* (Li et al., 2020): it contains computer science concept prerequisite pairs data using an unsupervised approach of prerequisite relation prediction.
- *PRET dataset* (Alzetta et al., 2018): it is extracted from a computer science textbook, it is an annotated dataset for prerequisite relations between educational concepts.

Table 2 Course metadata example

Metadata	Content
Id	25
Course type	Data science
Title	Introduction to Data Science in Python
University	University of Michigan
Provider	Coursera
Tutor	Christopher Brooks, Kevyn Collins-Thompson,...
Level	Intermediate
Language	English
Certificate	Available
Price	Free Online Audit Course
Description	This course will introduce the learner to the basics of the python...
Link	https://www.coursera.org/learn/python-data-analysis?...
Num of chapter	4
chapter.name	Fundamentals of Data Manipulation with Python
chapter.desc	In this week you'll get an introduction to the field...
chapter.name	Basic Data Processing with Pandas
chapter.desc	In this week of the course you'll learn the fundamentals of one...
chapter.name	More Data Processing with Pandas
chapter.desc	In this week you'll deepen your understanding of the python pandas...
chapter.name	Answering Questions with Messy Data
chapter.desc	In this week of the course you'll be introduced to a variety of statistical...

- *RefD dataset* (Liang et al., 2015): Reference Distance (RefD) is a dataset of prerequisite relations among concepts. It contains concept prerequisite pairs of the computer science and math categories.

After cleaning, organizing and removing duplicate pairs. A total of 4900 concept prerequisite pairs of the domain «computer science» were constructed.

Course corpus

The course data corpus used in this study is collected by crawling the Class Central website². Class Central is one of the largest publicly available websites that allows users to find and review MOOCs of different learning categories (e.g., computer science, data science, art, business, etc). Class Central provides metadata information about each MOOC such as authors, title, links, description, language, price, etc.

We selected 4503 MOOCs from Class Central. A web crawler was created and launched to automatically extract all the courses' metadata related to these 4503 MOOCs. Following our prerequisites' corpus, we were only interested in crawling courses in the categories «Computer science» and «Data science» and only courses presented in the English language. Table 2 presents an example of an extracted MOOC metadata as well as the related data.

² <https://www.classcentral.com/>

Performance metrics

Concept prerequisite evaluation

Since it is unlikely that we will be able to identify the best performing classification model to evaluate prerequisites identification in advance, we did a guided study to investigate a set of well-known classification models to select the best operating one. We investigated the following candidate machine learning models: Logistic Regression (Cox, 1958), Naive Bayesian (Lewis, 1998), Support Vector Machines (Cortes and Vapnik, 1995), Gradient Boosting Trees (Friedman, 2001), Random Forest (Breiman, 2001), and Neural Network (Dayhoff, 1990). We embraced the following four metrics to measure the proposed concept-level prerequisite relation to check the accuracy of the candidate machine learning models: the precision, the accuracy, the recall and the F1-score.

Content-based recommendation evaluation

We adopted three performance metrics to determine the quality of Top-N recommendation results, namely, Mean Reciprocal Rank (MRR), Mean Average Precision (MAP) and Normalized Discounted Cumulative Gain (NDCG), which are widely used to evaluate Top-N recommendation systems. We describe in detail each of these metrics as follows:

Let $\{C_1, C_2, \dots, C_N\}$ denote the courses (documents) to recommend. The documents are sorted in decreasing order of their similarity measure function value, where N represents the number of retrieved courses. The function $rel(C_i)$ presents the relevance value of a course C_i . If C_i is relevant, $rel(C_i) = 1$; otherwise, $rel(C_i) = 0$. The precision per a concept query q for top-N courses retrieved ($precision_q@N$) is defined as follows:

$$precision_q@N = \sum_{i=1}^N \sum_{j=1}^N \frac{rel(C_i)}{j} \tag{6}$$

The Average Precision (AP) per query is the average precision values over the number of relevant results M for the query q (where $M \leq N$). This can be represented as follows:

$$AP_q@N = \frac{\sum_{q=1}^M precision_q@N}{M} \tag{7}$$

We can finally calculate the Mean Average Precision (MAP) per query set Q , which is the average precision values over all queries q in Q . This can be represented as follows:

$$MAP@N = \frac{\sum_{q=1}^{|Q|} AP_q@N}{|Q|} \tag{8}$$

The Mean Reciprocal Rank (MRR) is defined as the average of the inverse ranks for all Q queries. Where $rank_q$ is the position of the first relevant document (course) among the N recommendation results for the query q . The MRR is defined as follows:

$$MRR = \frac{1}{|Q|} \sum_{q=1}^{|Q|} \frac{1}{rank_q} \tag{9}$$

The Normalized Discounted Cumulative Gain (NDCG) measures the positions of the positive samples in the Top-N recommendation results, it is defined as:

$$NDCG = \frac{DCG}{IDCG} \quad (10)$$

where DCG is the Discounted Cumulative Gain. DCG penalizes the relevant documents that appear in the bottom of the search results by decreasing their relevance value. As for (IDCG), it is Ideal (maximum) Discounted Cumulative Gain of top-N retrieved documents list REL_N (ordered by their relevance) in the corpus up to position N. They are represented as follows:

$$DCG@N = \sum_{i=1}^N \frac{2^{rel(C_i)} - 1}{\log_2(i + 1)} \quad (11)$$

$$IDCG@N = \sum_{i=1}^{REL_N} \frac{2^{rel(C_i)} - 1}{\log_2(i + 1)} \quad (12)$$

Experiment settings

The experiment of extracting concept prerequisites was conducted using SPARQL for concept extraction from the semantic web, most precisely, the DBpedia dataset. We extracted a dataset of 1592 pairs of concepts. After cleaning the data by keeping only pairs related to the domains "Computer science" and "Data science", we built a dataset of 939 pairs of terms. After creating the final prerequisite dataset, two domain expert coders (a full computer science professor and data science engineer) labelled every two concepts of the dataset with a numerical value '1' for the existing prerequisite relation and '0' otherwise. We used Python 3.7 programming language with the Scikit-learn library (Pedregosa et al., 2011) to create the predictive model that predicts whether two concepts have a prerequisite relation or not based on the labelled dataset. The training set and the test set are divided randomly according to 80:20. The overall distribution of the labels in the test set is balanced.

The main courses' corpus for recommendation and prerequisite feature calculation was extracted using Selenium web drive 3.141.0 and SQLite3 database. The data was cleaned and structured using Python 3.7 programming language. While the corpus indexing and search were conducted using Elasticsearch 7.13.2 library. As for the web-based application to visualize results, it was developed using Flask 1.1.4 and Bootstrap 3.

Results and discussion

Many empirical experiments were carried out to demonstrate the efficiency of the prerequisite-based recommendation system in education. In particular, we aim to answer the following Research Questions (RQs):

RQ1 What is the impact of using OLD and semantic search in identifying potential prerequisites of a course/material?

Table 3 Performance comparison of prerequisites identification machine learning algorithms

	SVM	NB	GB	LR	RF	NN
Acc	0.4	0.7	0.8	0.4	0.9	0.4
F1	0.2285	0.6901	0.8	0.2285	0.9010	0.2285
Prec	0.16	0.6952	0.8666	0.16	0.9199	0.16
Recall	0.4	0.7	0.8	0.4	0.9	0.4

The bold in Table means that the metrics are the highest (in each line)

RQ2 What methodology is most effective in listing potential prerequisites of a course/material?

RQ3 Does the prerequisite-based recommendation system present better results compared to the classical content-based recommendation in education?

RQ4 To what extent do recommended prerequisites impact the overall learning outcomes of learners?

(RQ1): What is the impact of using OLD and semantic search in identifying potential prerequisites of a course/material?

Access to an extensive and linked web of knowledge is made possible by Open Linked Data, allowing for a thorough comprehension of a wide range of subjects and areas. The system can provide more accurate course matching between learners’ prerequisites and desired knowledge by utilizing semantic links between prerequisites through connected data. As previously mentioned in Section “Prerequisite identification”, Semantic search aids in comprehending the context of searches and thus enabling more precise and pertinent results. In this instance, the query context seeks to support the learner by providing recommendations that help him reach his learning objectives and, as a result, get the foundational knowledge required to go on with more advanced learning materials. This will enable a better match between course content and learner requirements.

An easier way to navigate between related subjects or required resources is provided by linked data, which makes learning more organized. Conversely, students may quickly investigate related ideas and develop a deeper comprehension of the material. Furthermore, as discussed earlier, Open Linked Data and semantic search are capable of processing enormous volumes of data (our example uses DBpedia, which is a representation of all content added to Wikipedia), allowing for scalability as the learning material database expands. By enabling interoperability across various learning systems and resources that utilize metadata to identify and describe their course materials, this can also improve cooperation. This partnership based on semantic research mandates resource sharing and gives searchers access to a variety of resources.

In our example dataset (Section “Dataset”), it was demonstrated by how easily we created our training database by combining DBpedia data with other open resources. Learning patterns and preferences will become clear as a result, opening the door to the development of more creative, comprehensive, and rich learning resources. Because OLD is platform-agnostic, it makes integration with other educational platforms and

Table 4 The Performance comparison of the recommendation results

Metricis	Top 10 results			Top 5 results		
	MRR@10	MAP@10	NDCG@10	MRR@5	MAP@5	NDCG@5
Prerequisites-based rec	0.58	0.5601	0.8648	0.58	0.63	0.8576
Content-based rec	0.45	0.472	0.615	0.5	0.498	0.751

Learning Management Systems (LMSs) easier and increases access to a greater variety of resources.

By giving learners personalized, thorough, and contextually relevant content, the integration of Open Linked Data and semantic search in corresponding courses or learning materials not only increases the precision of suggestions but also improves the overall quality of the learning process.

(RQ2): What methodology is most effective in listing potential prerequisites of a course/material?

The results of evaluating the proposed model of predicting concept prerequisite pairs relationship are reported in Table 3. For this experiment, we report the Accuracy (Acc), the Precision (Pre), the Recall (Rec) and the F1-score (F1) as metrics to evaluate the performance of the predictive models.

As shown in Table 3, Random Forest (RF) achieves the best performance (acc=0.9) among all classification models. This proves that we can predict the prerequisite relationship between two learning concepts with a good accuracy of 90% based on the selected features in Section “Prerequisite identification”, which is very plausible. The second-best performance was achieved by Gradient Boosting (GB) with an accuracy of 80%, then Naive Bayes (NB) with an accuracy of 70%. Neural Network (NN), Support Vector Machine (SVM) and Logistic Regression (LR) achieve very low predictive performance. It’s important to remember that algorithms, including Random Forests, might perform differently depending on the particulars of the dataset and the task at hand. However, there are some reasons to determine why the Random Forest outperforms other methods. First, When it comes to numerical variables, other algorithms such as Neural Networks or Logistic Regression can do better; nonetheless, the random forest is the best option for making a decision based on circumstances/conditions. Gradient boosting tree or XGBoost take into account every aspect and provide a single path. Consequently, deeper trees might result from this and thus, overfitting the model. High-dimensional data sets can be handled by Random Forests without overfitting. Also, Unbalanced datasets are well handled by Random Forests. In unbalanced classification issues (the same as our data), the method can nevertheless produce correct predictions for the minority class since it creates trees independently and then combines their predictions. On the other hand, since deciding whether a concept is a prerequisite or not, depends on some convoluted and non-linear features, Random Forest is very useful for capturing patterns in these kinds of situations, where there are complicated, non-linear, or interactive connections between features and the target variable. When compared to some other algorithms, Random Forests exhibit reduced sensitivity to outliers. The effect

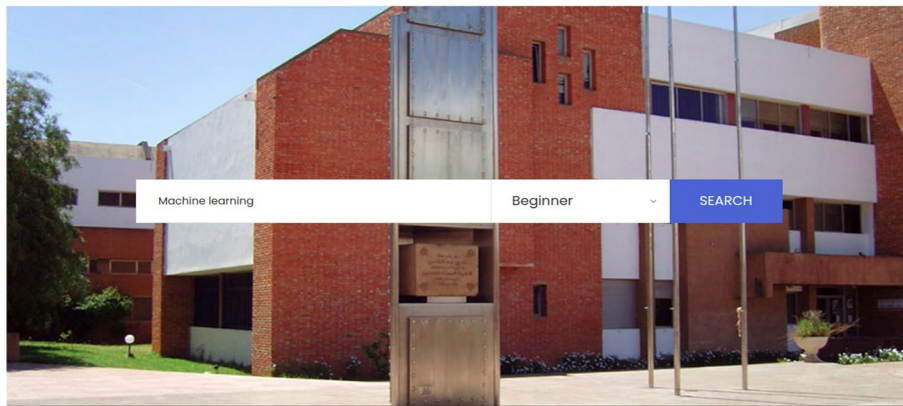


Fig. 7 Query for the concept “Machine learning” and the difficulty level “Beginner”

of a single outlier is typically lessened by the accumulation of predictions from several trees. Finally, because it employs a rule-based methodology, one of the Random Forest’s benefits is that normalization of data is often not necessary. So even if the normalisation of dataset is not well established, the results of prediction using Random forest will still be adequate.

As a result, the supervised learning model carried out using the Random Forest algorithm is the final step of the process to evaluate the prerequisite relation between the selected candidates’ concepts and the target concept. On the other hand, it will be the first step to recommend suitable learning objects that match concept prerequisites.

Compared with the reference study conducted in Manrique et al. (2019), the authors obtained a precision equal to 90% with almost the same number of features using the XGBoost classification algorithm, which is very close to our finding using the Random Forest algorithm. This proves the accuracy of this method in predicting prerequisites among concept pairs with high precision. This also would enhance the results of content-based prerequisite learning materials retrieval/recommendation. The system will be able to recommend accurate resources that match the concept prerequisites with good accuracy.

(RQ3): Does the prerequisite-based recommendation system present better results compared to the classical content-based recommendation in education?

Table 4 shows the performance of our prerequisite-based recommendation system against the classical content-based-recommender system on the educational dataset. We tested the performance on top 10 results using $MRR@10$, $MAP@10$ and $NDCG@10$ metrics. Also, the system was tested on the top 5 results using $MRR@5$, $MAP@5$ and $NDCG@5$ metrics. According to the results, we remark that the Mean Reciprocal Rank has the same value for the 10 best results as well as the 5 best results ($MRR@10=MRR@5=0.58$) for the prerequisites-based recommendation system. Since MRR is generally associated with only one relevant document, it means that the user gets the most accurate recommendation document in the first 5 results, which is very useful. We also remark that our prerequisite-based recommendation system outperformed the classical content-based recommendation system on both $MRR@10$ and $MRR@5$. While

MAP considers all the most relevant documents in the recommended list, a score above 0.5 is considered plausible. $MAP@10=0.5601$ demonstrates that there are some very relevant documents in the top 10 results; Moreover, those documents are also very relevant in the top 5 results ($MAP@5=0.63$). An improvement of $MAP@10=0.5601$ into $MAP@5=0.63$ is very significant, as it shows that the top 5 results have more importance in their pertinence. Also, the content-based recommender system is under 0.5 for $MAP@10$ ($MAP@10=0.472$) and does not present a high improvement for $MAP@5$ ($MAP@5=0.498$), which can demonstrate that the prerequisites-based recommender systems have better performance results that can highly improve the quality of the recommendation as well as the accurate attainment of learning objectives.

NDCG is the proportion of the user's score over the ideal ranking's score. A value of 0.8648 for $NDCG@10$ and 0.8576 for $NDCG@5$ means that the results are very close to the perfect score of 1 (close to the ideal order of the recommended items). In other words, almost 85% of the recommendation results (recommended in the right order) present very relevant items to the user, which is very beneficial, especially when the results present prerequisite concepts that are needed to master the initial concept. On the other hand, the classical content-based recommender system presents a value of 0.615 for $NDCG@10$ and 0.751 for $NDCG@5$. This means that almost 61% of the recommendation results deliver appropriate items to the user. This shows that the prerequisite-based recommender system presents high-efficiency results compared to the classical content-based recommender system. This opens great opportunities for learners to enhance their learning outcomes.

(RQ4): To what extent do recommended prerequisites impact the overall learning outcomes of learners?

This research aimed to develop prerequisite-based learning recommendations on e-learning platforms. It automatically predicts suitable learning resources that match concept prerequisites, and that overcome learning object cold-start.

Following the method explained in Section "[Prerequisite-based course recommendation](#)". When a pre-defined recommendation event is launched (end of a sequence, end of a chapter, etc), the learner is provided with the web application recommendation system that contains the studied concept and the difficulty level already defined by educators and instructional designers of online courses. As shown in Fig. 7, the concept is 'Machine learning' and the difficulty level is 'Beginner'. It is important to note that the choice of difficulty level is very important as it filters learning resources based on the course's initial metadata concerning the difficulty that matches learners' knowledge background. When the system launches the query (concept and difficulty level), the results of recommended learning objects are shown graphically in Fig. 8. At the top of the shown interface, all concept prerequisites are automatically shown ('dataset', 'statistical inference', 'probability', 'artificial intelligence', etc). Those concepts are prerequisites of the initial concept 'machine learning'.

On the same screen, recommended courses that match the results of the prerequisite are shown in a data table along with metadata (title, provider, link, tutors and category). The learner can access all metadata, including the external link to the course.

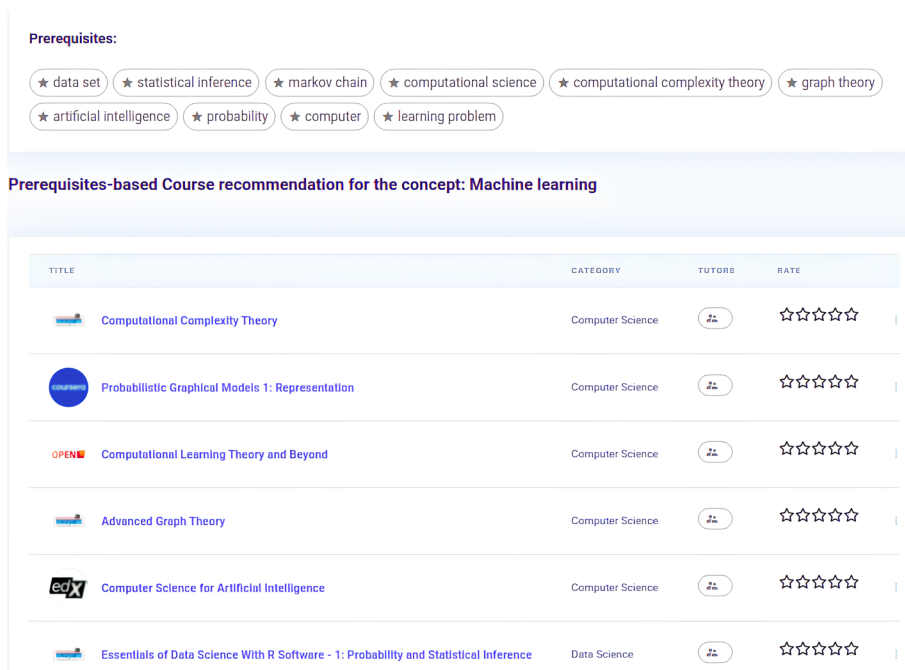


Fig. 8 Query results

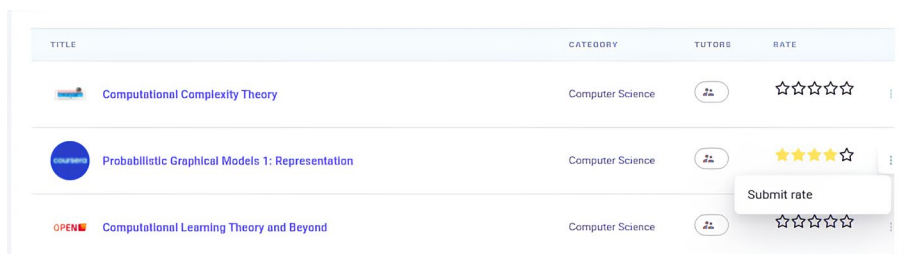


Fig. 9 Rating a learning object

In addition, the learner can rate any learning object at a rate of 1–5 as shown in Fig. 9. The learner’s scores report is automatically provided to an external collaborative recommender system (learners, learning objects and rate) in a switching hybrid recommendation architecture to prevent the items cold-start (learning objects). Since this system is an independent system that is based only on a provided learning concept and a difficulty level, it can be integrated into any hybrid recommendation system architecture. This can enhance and help educators and instructional designers to improve provided learning resources to learners during course build.

It is important to note that in case a concept has no prerequisites in the database, the system will automatically recommend courses that match this exact concept.

Recommendations that are based on prerequisites make sure students have a strong foundation by beginning with basic ideas and working their way up to more difficult subjects. It encourages a methodical learning process, which lessens the possibility of comprehension gaps that may arise from following a classical content-based

recommendation system. The prerequisites-based recommendations are based on customised learning pathways depending on the prior knowledge of each student, making the learning process as effective as possible for each person. By avoiding repetitive or too complex information, learners focus their time and energy on content that is appropriate for their level of ability. On the other hand, by guaranteeing that students possess prerequisite information, they improve their ability to understand and remember new, intricate ideas. Also, this recommendation reduces cognitive stress; By delivering the information gradually, it reduces cognitive overload and improves understanding and memory. Also, recommendations based on prerequisites lessen dissatisfaction and dropout rates by preventing students from being overtaken by overly complex information. Therefore, and more importantly, learners are more likely to remain motivated and interested when they feel suitably challenged without feeling overburdened. Moreover, the gaining required information at the outset will help students grasp advanced subjects at higher levels since they will have a stronger foundation in fundamental ideas, this will improve application skills, as students are more capable of applying newly acquired knowledge to actual situations, which furthers the development of practical skills. and finally, building knowledge on prerequisites facilitates the acquisition of information that extends beyond the current learning environment and promotes continuous skill improvement.

Conclusion and future work

In this paper, we study the course recommendation problem on e-learning platforms. We propose a method to extract course prerequisites using Linked Open Data (LOD) and machine learning. We first construct our final dataset by assembling pertinent datasets from open knowledge bases (DBpedia) and other open datasets. Next, we compile and arrange course-related data, including descriptions, and any applicable metadata. Next, in order to specify the connections between courses-whether they are prerequisites-we created a domain-specific ontology. Using this specification, data will be annotated in accordance with the ontology. Following the extraction of features from the course information, we run machine learning algorithms for binary classification using those features as input and the prerequisite relationships as class output. The results prove the efficiency of our approach, as the Random Forest machine learning algorithm achieves good accuracy in predicting the right prerequisites of a certain concept. Lastly, we used the trained model in a real-case scenario setting to recommend the course's prerequisites-based. Also, the recommended learning objects are very relevant and accurate with the selected prerequisites, as we attain $NDCG@10=0.8684$. Which is very satisfying, since it also outperforms the classical content-based recommendation ($NDCG@10=0.615$).

Through the integration of machine learning techniques and LOD principles, this approach makes it easier to extract and recommend course basic requirements, providing users with precise and customized learning path suggestions. Prerequisite-based recommendations in e-learning ensure that learners have the fundamental information required to successfully grasp more complex ideas, fostering a more organized, customized, and effective learning experience.

Future studies will focus on integrating this prerequisite-based content recommendation system into a mature switched hybrid recommendation system. Where we can

compute learners' behaviour and rate as well, to enhance classical collaborative recommender systems and overcome their item cold-start weakness.

Acknowledgements

The authors would like to thank the individuals who generously shared their time and expertise for supporting the purpose of this research

Author contributions

Each author contributed equally to each stage of the study. All authors read and approved the final manuscript.

Funding

Not applicable.

Availability of data and materials

Data supporting the findings of this study are available upon request from the first and corresponding author

Declarations

Competing interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Received: 3 April 2023 Accepted: 4 April 2024

Published online: 11 May 2024

References

- Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6), 734–749.
- Al-Badarenah, A., & Alsakran, J. (2016). An automated recommender system for course selection. *International Journal of Advanced Computer Science and Applications*, 7(3), 166–175.
- Alzetta, C., Koceva, F., & Passalacqua, S., et al. (2018). Pret: Prerequisite-enriched terminology. a case study on educational texts. In *Proceedings of the Fifth Italian Conference on Computational Linguistics CLiC-it*, p. 12
- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32.
- Chanaa, A., & El Faddouli, N. (2021). An analysis of learners' affective and cognitive traits in context-aware recommender systems (cars) using feature interactions and factorization machines (fms). *Journal of King Saud University-Computer and Information Sciences*, 34, 4796–4809.
- Consortium, I. G. L., et al. (2003). Ims learning design specification 7:2009. <http://www.imsglobal.org/metadata/index.html> [Accessed: 10-10-2021]
- Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273–297.
- Cox, D. R. (1958). The regression analysis of binary sequences. *Journal of the Royal Statistical Society: Series B (Methodological)*, 20(2), 215–232.
- Dai, Y., Yoshikawa, M., & Sugiyama, K. (2021). Prerequisite-aware course ordering towards getting relevant job opportunities. *Expert Systems with Applications*, 183, 115233.
- Dayhoff, J. E. (1990). *Neural network architectures: An introduction*. Van Nostrand Reinhold Co.
- De Medio, C., Limongelli, C., Sciarone, F., et al. (2020). Moodlerec: A recommendation system for creating courses using the moodle e-learning platform. *Computers in Human Behavior*, 104(106), 168.
- Dictionary, M. W. (2002). Merriam-webster dictionary. Metadata. Merriam-Webster. Retrieved October 10, 2021, from <https://www.merriam-webster.com/dictionary/metadata>
- Fabbri, A. R., Li, I., & Trairatvorakul, P., et al. (2018). Tutorialbank: A manually-collected corpus for prerequisite chains, survey extraction and resource recommendation. [arXiv:1805.04617](https://arxiv.org/abs/1805.04617)
- Fischer, S. (2001). Course and exercise sequencing using metadata in adaptive hypermedia learning systems. *Journal on Educational Resources in Computing (JERIC)*, 1(1es), 5–es.
- Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5), 1189–1232.
- Gasparetti, F., De Medio, C., Limongelli, C., et al. (2018). Prerequisites between learning objects: Automatic extraction based on a machine learning approach. *Telematics and Informatics*, 35(3), 595–610.
- Gormley, C., & Tong, Z. (2015). *Elasticsearch: The definitive guide: A distributed real-time search and analytics engine*. O'Reilly Media Inc.
- Huang, R., & Lu, R. (2018). Research on content-based mooc recommender model. In *2018 5th International Conference on Systems and Informatics (ICSAI)*. IEEE, pp. 676–681.
- Jing, X., & Tang, J. (2017). Guess you like: Course recommendation in moocs. In *Proceedings of the International Conference on Web Intelligence*, pp. 783–789.
- Jones, K. S. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28, 11–21.
- Khanal, S. S., Prasad, P., Alsadoon, A., et al. (2020). A systematic review: Machine learning based recommendation systems for e-learning. *Education and Information Technologies*, 25(4), 2635–2664.
- Laurence, S., & Margolis, E. (1999). Concepts and cognitive science. *Concepts: Core Readings*, 3, 81.

- Lewis, D. D. (1998). Naive (Bayes) at forty: The independence assumption in information retrieval. In *European Conference on Machine Learning*. Springer, pp 4–15.
- Li, I., Fabbri, A., & Hingmire, S., et al. (2020). R-vgae: Relational-variational graph autoencoder for unsupervised prerequisite chain learning. [arXiv:2004.10610](https://arxiv.org/abs/2004.10610)
- Liang, C., Wu, Z., & Huang, W., et al. (2015). Measuring prerequisite relations among concepts. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp 1668–1674.
- Liang, C., Ye, J., & Wu, Z., et al. (2017). Recovering concept prerequisite relations from university course dependencies. In *Thirty-First AAAI Conference on Artificial Intelligence*.
- Lops, P., de Gemmis, M., Semeraro, G. (2011). Content-based recommender systems: state of the art and trends. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P. (eds) *Recommender systems handbook*. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-85820-3_3
- Manning, C. D., Raghavan, P., & Schütze, H. (2008). Introduction to information retrieval <<http://nlp.stanford.edu/IR-book/html/htmledition/irbookhtml>
- Manrique, R., Pereira, B., & Mariño, O. (2019). Exploring knowledge graphs for the identification of concept prerequisites. *Smart Learning Environments*, 6(1), 1–18.
- Mountantonakis, M., & Tzitzikas, Y. (2017). How linked data can aid machine learning-based tasks. In *International Conference on Theory and Practice of Digital Libraries*. Springer, pp. 155–168.
- Novak, J. D. (1990). Concept mapping: A useful tool for science education. *Journal of Research in Science Teaching*, 27(10), 937–949.
- Pan, L., Li, C., & Li, J., et al. (2017). Prerequisite relation learning for concepts in moocs. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1447–1456.
- Pang, Y., Wang, N., Zhang, Y., et al. (2019). Prerequisite-related mooc recommendation on learning path locating. *Computational Social Networks*, 6(1), 1–16.
- Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). Scikit-learn: Machine learning in python. *The Journal of Machine Learning Research*, 12, 2825–2830.
- Robson, R. (2012). IEEE learning technology standards committee (ltsc). Retrieved October 10, 2021, from <https://ieeesa-imeetcentral.com/ltsc/>
- Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5), 513–523.
- Schafer, J. B., Frankowski, D., & Herlocker, J., et al. (2007). Collaborative filtering recommender systems. In *The adaptive web*. Springer, pp. 291–324.
- Schein, A. I., Popescul, A., & Ungar, L. H., et al. (2002). Methods and metrics for cold-start recommendations. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 253–260.
- Shu, J., Shen, X., Liu, H., et al. (2018). A content-based recommendation algorithm for learning resources. *Multimedia Systems*, 24(2), 163–173.
- Sun, G., Cui, T., & Xu, D., et al. (2017). Assisting open education resource providers and instructors to deal with cold start problem in adaptive micro learning: A service oriented solution. In *2017 IEEE International Conference on Services Computing (SCC)*. IEEE, pp. 196–203.
- Sun, G., Cui, T., & Xu, D., et al. (2018). A heuristic approach for new-item cold start problem in recommendation of micro open education resources. In *International Conference on Intelligent Tutoring Systems*. Springer, pp. 212–222.
- Wan, S., & Niu, Z. (2018). An e-learning recommendation approach based on the self-organization of learning resource. *Knowledge-Based Systems*, 160, 71–87.
- Weibel, S. L., & Koch, T. (2000). The dublin core metadata initiative. *D-lib Magazine*, 6(12), 1082–9873.
- Yang, Y., Liu, H., & Carbonell, J., et al. (2015). Concept graph learning from educational data. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, pp. 159–168.
- Zhang, H., Huang, T., Lv, Z., et al. (2019). Moocr: A highly accurate resource recommendation model for use in mooc environments. *Mobile Networks and Applications*, 24(1), 34–46.
- Zhao, Z., Yang, Y., Li, C., et al. (2020). Guessuneeed: Recommending courses via neural attention network and course prerequisite relation embeddings. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 16(4), 1–17.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Chanaa Abdessamad Computer science engineer and PhD student in recommender systems in e-learning. He is interested in machine learning, data mining, deep learning, human-computer interaction, and data science.

Nour-eddine El Faddouli A full computer science professor and PhD. He is an expert in e-learning. He is interested in Information Science Data Mining, Databases, Teacher Education, Educational Technology, Teaching Methods and Pedagogy, and Education.