Learning-Oriented Question Recommendation Using Bloom's Learning Taxonomy and Variable Length Hidden Markov Models

Hilda Kosorus(✉) and Josef Küng

Institute of Application Oriented Knowledge Processing,
Johannes Kepler University, Linz, Austria
{hkosorus,jkueng}@faw.jku.at
http://www.faw.jku.at

Abstract. The information overload in the past two decades has enabled question-answering (QA) systems to accumulate large amounts of textual fragments that reflect human knowledge. Therefore, such systems have become not just a source for information retrieval, but also a means towards a unique learning experience. Recently developed recommendation techniques for search engine queries try to leverage the order in which users navigate through them. Although a similar approach might improve the learning experience with QA systems, questions would still be considered as abstract objects, without any content or meaning. In this paper, a new learning-oriented technique is defined that exploits not only the user's history log, but also two important question attributes that reflect its content and purpose: the topic and the learning objective. In order to do this, a domain-specific topic-taxonomy and Bloom's learning framework is employed, whereas for modeling the order in which questions are selected, variable length Markov chains (VLMC) are used. Results show that the learning-oriented recommender can provide more useful, meaningful recommendations for a better learning experience than other predictive models.

Keywords: Recommender systems · Question-answering systems · Learning taxonomy · Topic taxonomy · Collaborative filtering · Variable length markov chain

1 Introduction

In the past two decades, education has become more and more subject to personalization and automation. The success of recommender systems [1] has motivated research on deploying such techniques also in educational environments to facilitate access to a wide spectrum of information [16].

One of the consequences of information overload is the rise of question answering (QA) systems. Over time, QA systems have gathered a large amount of textual fragments – reflections of human knowledge - from a variety of domains

A. Hameurlain et al. (Eds.): TLDKS XVI, LNCS 8960, pp. 29–44, 2014.
DOI: 10.1007/978-3-662-45947-8_3

and, therefore, represent a potential source for learning and establishment of new fields of study. Such systems have become not just a source of information retrieval, but also a medium for online information seeking and knowledge sharing [15], a means towards a unique learning experience. However, the exponential growth in the data volume of QA systems has made the users access to the desired information more difficult and time-consuming [15].

Current QA systems integrate traditional content-based recommendation engines with the goal to identify the most suitable user to answer a question [11,12], but little research aims at filtering out for the user the questions/answers that might be of interest [15]. The drawback of such approaches is that the recommender does not take into account explicitly the user's learning goals or learning process, neither the order in which questions are selected. The goal of this paper is to improve the learning experience of the user in the role of question asker.

Recent research in the field of query recommendation for search engines are based on query search graphs that aim at extracting interesting relations from user query logs [3,4]. Some of these graphs are constructed based on relations between queries, which are explored and categorized according to different sources of information (e.g., words in a query, clicked URLs, links between their answers). Other techniques rely on the co-occurrence frequency of query pairs, which are part of the same search mission [5,7–9]. However, these approaches do not take into account the user's search goal. A recent attempt to tackle this issue is presented in [10].

In [10], the authors propose a general approach to context-aware search using a variable length hidden Markov model (vlHMM). This work is motivated by the belief that the context of a users query, i.e. the past queries and clicks in the same session, may help understand the users information need and improve the search experience substantially. Cao et al. [10] develop a strategy for parameter initialization within the vlHMM learning, which can reduce the number of parameters to be estimated in practice. Additionally, they devise a method for distributed vlHMM learning under the map-reduce model. Within this context, the authors also argue that by considering only correlations between query pairs, the model cannot capture well the users search context. In order to achieve general context-aware search, a comprehensive model is needed that can be used simultaneously for multiple applications (e.g. query suggestion, URL recommendation, document re-ranking). They propose a novel model to support context-aware search and develop efficient algorithms and strategies for learning a very large vlHMM from big log data. The experimental results show that this vlHMM-based context-aware approach is effective and efficient.

Despite the extensive research in this area and the successful application of such methods, they are not suitable for QA systems for at least two reasons. First, the recommendation items are represented by questions as well-formed grammatical units endowed with semantic content, whereas search queries are usually a collection of keywords. Secondly, most QA systems are used with the purpose of learning (e.g., find an explanation for a particular phenomenon, understand a specific concept, etc.), while search engines are usually queried to simply retrieve information.

The work presented in this paper aims at improving question recommendation for QA systems by addressing these two aspects. Our main objective is to leverage the functionality of QA systems towards new learning techniques and use the wisdom of the crowds in order to convey useful information and guide the learner on a meaningful learning journey. For this purpose, a domain-specific topic-taxonomy and Bloom's learning framework is employed, whereas for modeling the order in which questions are selected, variable length Markov chains (VLMC) [6] are used.

The rest of the paper is structured as follows: Section 2 presents the knowledge-base with the domain-specific and learning taxonomies; Section 3 introduces the new learning-oriented recommender model; Section 4 gives an overview of the evaluation results and, finally, Section 5 makes a summary, draws some important conclusions and presents future work objectives.

2 Approach

Aiming at improving the learning experience of users when interacting with a QA system, question recommendation, in the context of this paper, refers to recommending questions to users who ask them and are interested in learning about a particular domain. The question-answer dialog with the system should allow the user to navigate through a meaningful and useful chain of answers that can enrich the users knowledge about a particular domain.

In order to account for the learning process or the order in which questions are selected, a probabilistic graphical model based on variable length Markov chains [6] is constructed and trained on the users question browsing history. This is not a novel approach; it has been successfully adopted for query recommendation [10] as well. In this paper, we attempt to adapt and improve this approach for question recommendation by considering two relevant question features: the questions topic (or subject) and learning objective. The learning objectives, in the context of Blooms learning taxonomy [2], refer to a classification of educational goals (e.g. summarizing, classifying, recognizing, etc.). Current conceptions about learning assume learners as active agents and not passive recipients or simple recorders of information. This shift away from a passive perspective on learning towards more cognitive and constructionist perspectives emphasizes what learners know (knowledge) and how they think (cognitive processes) about what they know [2]. Therefore, the learning taxonomy is defined based on two dimensions: the knowledge and the cognitive process.

For this purpose, two taxonomies are considered: a domain-specific taxonomy that contains possible question topics and Blooms learning taxonomy, a classification of existing learning objectives. In the following, we will present the knowledge base behind the recommendation model.

2.1 Knowledge Base

Let $\mathcal{Q}$ be a set of questions from a particular domain. In general, QA systems can cover several domains of interest, but, for simplicity, we will consider in the following only a single domain.

The Domain-Specific Taxonomy. Let $\mathcal{T}$ be a set of predefined topics from a particular domain. In general, $|\mathcal{Q}| \gg |\mathcal{T}|$. The structure of the corresponding topic taxonomy is given by a generalization-specification relationship between topics:

$$\mathcal{P} \subseteq \mathcal{T} \times \mathcal{T}, (\tau_i, \tau_j) \in \mathcal{P} \iff \tau_i \ parent \ of \ \tau_j, \tag{1}$$

where $\tau_i, \tau_j \in \mathcal{T}$.

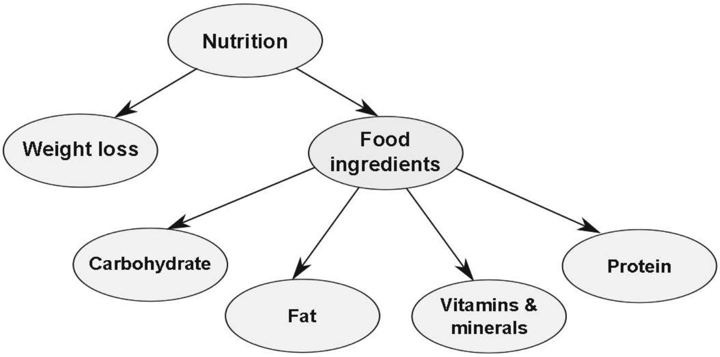

Fig. 1. Snapshot of an example nutrition taxonomy

Consider also a mapping relationship between $\mathcal{Q}$ and $\mathcal{T}$, which maps to each question $q \in \mathcal{Q}$ a topic $\tau \in \mathcal{T}$. A mapping (q, τ) has the following meaning: "question q is about topic τ". We will further refer to this relationship as topic mapping and it is defined in the following way:

$$\mathcal{M}_\tau \subseteq \mathcal{Q} \times \mathcal{T}, (q, \tau) \in \mathcal{M}_\tau \iff q \ is \ mapped \ to \ topic \ \tau. \tag{2}$$

The topic mapping $\mathcal{M}_\tau$ is a surjection with respect to Q (i.e. all questions are mapped to at least one topic). Moreover, one question can be mapped to several topics and one topic can map several questions (see Figure 2).

The Learning Taxonomy. Additionally, we enrich the knowledge base with learning objectives, as defined in Bloom's taxonomy [2]. Let $\mathcal{L}$ be the set of all learning objectives. Every learning objective $\phi \in \mathcal{L}$ is defined as a pair of knowledge and cognitive process instances $(\kappa, \rho) \in \mathcal{K} \times \mathcal{C}$, where

$$\mathcal{K} = \{factual,\ conceptual,\ procedural,\ metacognitive\} \tag{3}$$

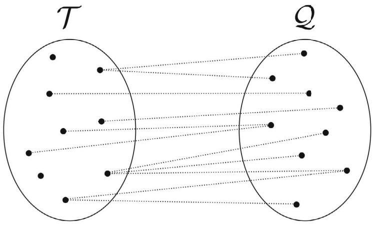

Fig. 2. Question-topic mapping

is the knowledge dimension and

$$\mathcal{C} = \{remember,\ understand,\ apply,\ analyze,\ evaluate,\ create\} \quad (4)$$

is the cognitive process dimension. A more detailed explanation of these concepts can be found in [2]. Similarly to the topic mapping, we define the learning objective mapping as

$$\mathcal{M}_\phi \subseteq \mathcal{Q} \times \mathcal{L}, (q, \phi) \in \mathcal{M}_\phi \iff q\ is\ mapped\ to\ learning\ objective\ \phi. \quad (5)$$

In contrast to the topic mapping $\mathcal{M}_\tau$, we allow questions to be mapped to a single learning objective. Intuitively, this means that a question can refer to several topics, but a single learning goal. In general, questions refer also to a single topic. For simplicity, we have only dealt with single topic assignments in our experiments and, in the following, we consider $\mathcal{M}_tau$ to map each question to a single topic.

Question Projections. Based on the mapping relationships, we define the following projection functions:

1. The **topic projection** - a function that projects a question on the topic space using the mapping $\mathcal{M}_\tau$:

$$p_\mathcal{T} : \mathcal{Q} \to \mathcal{T}, p_\mathcal{T}(q) = \tau \iff \exists(q, \tau) \in \mathcal{M}_\tau \quad (6)$$

2. The **learning objective projection** - a function that projects a question on the learning objective space using the mapping $\mathcal{M}_\phi$:

$$p_\phi : \mathcal{Q} \to \mathcal{L}, p_\phi(q) = \phi \iff \exists(q, \phi) \in \mathcal{M}_\phi \quad (7)$$

2.2 The Learning-Oriented Recommendation Model

In the following, a novel learning-oriented question recommendation technique is introduced that aims at improving the user's learning experience based on the following intuition.

Intuition: Question sequences are first influenced by the underlying topics or subject and the order in which these topics are tackled, and then, within each topic, by a particular order of learning objectives. In other words, users tend to ask questions grouped by topics; in a particular order given by the question learning objectives (see Figure 3).

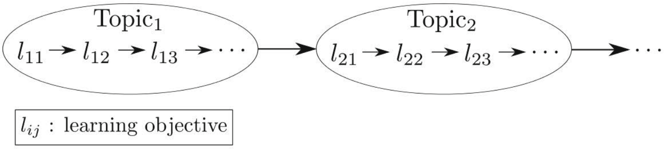
Topic$_1$
$l_{11} \rightarrow l_{12} \rightarrow l_{13} \rightarrow \cdots$
Topic$_2$
$l_{21} \rightarrow l_{22} \rightarrow l_{23} \rightarrow \cdots$
$\rightarrow \cdots$
l_{ij} : learning objective

Fig. 3. Intuition behind the user learning process

This intuition emerged during the evaluation process, where several probabilistic recommendation models were constructed and tested. The results showed that the model based on this intuition performed better than the rest of them. Due to the limited space, only three of the most relevant models will be considered here for comparison.

Preliminaries. Let Q, T and L be random variables taking values in the question set $\mathcal{Q}$, the topic space $\mathcal{T}$ and the set of learning objectives $\mathcal{L}$, respectively.

Consider $\mathcal{H}$ to be the history database which contains, for each user, an ordered sequence of questions representing the user's history log.

A learner is given a training set (usually a subset of the history database $\mathcal{H}$) of question sequences $q_1^n = q_1 q_2 \ldots q_n$, where $q_i \in \mathcal{Q}$ and $q_i q_{i+1}$ means that question q_i was asked before question q_{i+1}.

Given this training set, our goal is to learn a model P that provides a probability assignment for any future outcome given some past. More specifically, given a context of previously asked questions $s \in \mathcal{Q}^*$ (i.e. an ordered sequence of the user's past question selections) and a question q, the learner should generate a conditional probability distribution $P(q|s)$.

We measure the *prediction performance* using the average log-loss [6] $l(P, x_1^t)$ of P with respect to a test sequence $x_1^t = x_1 x_2 \ldots x_t$:

$$l(P, x_1^t) = -\frac{1}{t} \sum_{i=1}^{t} \log(P(x_i|x_1 \ldots x_{i-1})) \tag{8}$$

where x_i represent questions in $\mathcal{Q}$. The average log-loss is directly related to the likelihood $P(x_1^t) = \prod_{i=1}^{t} P(x_i|x_1 \ldots x_{i-1})$ and, therefore, minimizing the average log-loss is equivalent to maximizing the likelihood.

The Recommendation Model. Based on the learned model P, we define the *learning-oriented recommender* (LoR). For each user with history log $s \in \mathcal{Q}^*$, we want to recommend a set $R(s) \subseteq \mathcal{Q}$ of N questions that satisfies the following:

$$R(s) = \underset{q \in \mathcal{Q} \text{ and } q \notin s}{\arg\overset{N}{\max}} \; P(q|s) \tag{9}$$

where $\arg\max^N$ returns the first N maximal arguments with respect to the given function. In other words, the learning oriented recommender tries to recommend the first N best questions that maximize the user's utility. In this case, the utility is dependent on the learned model P.

P is learned according to a probabilistic graphical model based on hidden VLMCs: one with hidden states T and then, for each $\tau \in \mathcal{T}$ a VLMC with hidden states L. The observation states are given by Q over the question space $\mathcal{Q}$ (see Figure 4).

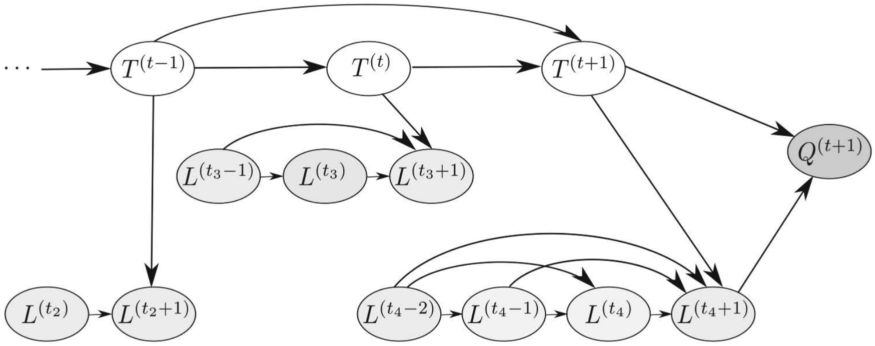

Fig. 4. The learning-oriented recommender

To learn such a model, first, the training sequences are projected on the topic space using p_τ and a VLMC over T is trained on them. As a result, the transition model $P(T^{(t+1)}|T^{(1:t)})$ is obtained.

Then, for each topic τ, a transition probability $P_\tau(L^{(t'+1)}|L^{(1:t')})$ is learned by training a VLMC over L on the projections of the question sub-sequences within topic τ, using the learning objective projection function p_ϕ.

We define the observation model $P(Q^{(t+1)}|T^{(t+1)}, L^{(t+1)}, Q^{(1:t)})$ as the probability of randomly sampling an unvisited question corresponding to topic $T^{(t+1)} = \tau_{t+1}$ and learning objective $L^{(t+1)} = \phi_{t+1}$

$$P(q_{t+1}|\tau_{t+1}, \phi_{t+1}, q_1^t) = \begin{cases} 0 & \textit{if } \nexists(q_{t+1}, \tau_{t+1}) \in \mathcal{M}_\tau \;\vee\; \nexists(q_{t+1}, \phi_{t+1}) \in \mathcal{M}_\phi \\ \frac{1}{S} & \textit{otherwise} \end{cases}, \tag{10}$$

where $S = \{q' \in \mathcal{Q} \backslash \{q_1, \ldots, q_t\} | (q', \tau_{t+1}) \in \mathcal{M}_\tau \wedge (q', \phi_{t+1}) \in \mathcal{M}_\phi\}$.

Prediction. Let $x_1^t = x_1 \dots x_t$ with $x_i \in \mathcal{Q}, i \in \{1, \dots, t\}$ be a context sequence of questions and $x_{t+1} \in \mathcal{Q}$ be the user's next question. Then, we define the probability of observing question x_{t+1} after x_1^t as:

$$P(x_{t+1}|x_1^t) = P(\tau_{t+1}|\tau_1^t) \cdot P(\phi_{t+1}|\phi_1^t) \cdot P(x_{t+1}|\tau_{t+1}, \phi_{t+1}, x_1^t), \tag{11}$$

where $\tau_{t+1} = p_\tau(x_{t+1})$ is the projection of question x_{t+1} on the topic space $\mathcal{T}$, $\phi_{t+1} = p_\phi(x_{t+1})$ is the projection of question q_{t+1} on the space of learning objectives $\mathcal{L}$, $\tau_1^t = p_\tau(x_1^t) = p_\tau(x_1) \dots p_\tau(x_t)$ and $\phi_1^t = p_\phi(x_1^t) = p_\phi(x_1) \dots p_\phi(x_t)$, x_1^t being the last sub-sequence within topic τ.

2.3 Other Models

In order to show the effectiveness of the introduced learning-oriented question recommender, we intend to perform a comparison with other models that use Markov chains. One of the most widely used ones is a simple variable length Markov chain (VLMC) over the question space, which we will further refer to as Simple Recommender (SR). VLMCs were widely used in the literature for prediction purposes in various application domains (e.g. data compression, context-aware search, etc.) [6,10].

The Simple Recommender (SR) is defined using a VLMC with random variable Q over the question space $\mathcal{Q}$. Consider $P(Q^{(t+1)}|Q^{(1:t)})$ to be the transition model of the VLMC trained over a subset of the history database $\mathcal{H}$. In order to learn a VLMC model, the algorithms presented in [6] were employed.

Then, for a given context of questions $x_1^t = x_1 x_2 \cdots x_t$ with $x_i \in \mathcal{Q}$ and a new question $x_{t+1} \in \mathcal{Q}$, the probability of observing x_{t+1} after x_1^t is given by:

$$\begin{aligned} P(x_{t+1}|x_1^t) &= P(Q(t+1) = x_{t+1}|Q^{(1)} = x_1, \dots, Q^{(t)} = x_t) \\ &= P(Q^{(t+1)} = x_{t+1}|Q^{(t\lambda+1)} = x_{t\lambda+1}, \dots, Q^{(t)} = x_t), \end{aligned} \tag{12}$$

where $\lambda = \lambda(x_t, x_{t1}, \dots)$ is a function of the past determined during the learning process of the VLMC. Let $D = \max_{x_t, x_{t1}, \dots} \lambda(x_t, x_{t1}, \dots)$ be the maximal memory length of the VLMC. Figure 5 represents a simple recommender model with $D = 3$.

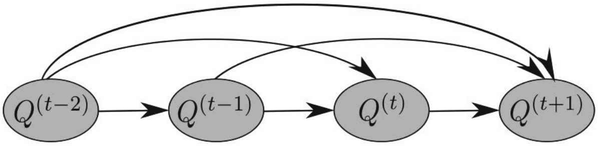

Fig. 5. Simple question recommender based on a VLMC

Intuition: The SR model is based on the intuition that the unique question identifiers are enough to efficiently identify learning patterns within question sequences and use them to produce accurate predictions.

The Random Recommender (RR) is a model that simply recommends questions randomly, without relying on any knowledge or user history log when producing recommendations. This model was considered only as base comparison to show that the previous two recommendation models (LoR and SR) are not random and that they actually outperform greatly the RR model.

The following section will show the performance of each of these recommendation models on three different knowledge bases.

3 Evaluation and Results

3.1 Data

In order to thoroughly test the performance of the LoR model, three data sets of questions, corresponding to three different domains, were collected: earth sciences (from Wiki Answers[1] and MadSci[2]), nutrition (provided by Sasha Walleczek[3]) and homeschooling (from Wiki Answers).

For reasons of robustness, corresponding to each of these data sets, a topic taxonomy with the structure presented in Section 2.1 was manually constructed. Table 1 gives an overview on the size of the data sets and the corresponding taxonomies. None of these taxonomies reflect a unique and complete image of the actual domains. They are merely a snapshot of the domains from a particular perspective. The topic-trees were constructed in a way to cover the question datasets. In this particular case, the structure of the hierarchy does not influence the performance of the recommender models and, therefore, represents no variable in the overall evaluation process. However, the performance of the recommender models does depend on the topic, knowledge and cognitive process mappings.

Table 1. Overview of question data sets

Data set	No. of questions	No. of topics	No. of questions with questions
Earth sciences	313	49	37
Nutrition	318	38	24
Homeschooling	191	42	39

[1] www.wiki.answers.com

[2] www.madsci.org

[3] www.walleczek.at

Similarly, the assignment of questions to the set of topics $\mathcal{T}$ and learning objectives $\mathcal{L}$ was performed manually in order to maintain robustness. However, not all topics or learning objectives were identified within the three question sets. Table 2 gives an overview of the mappings' statistics. Column $avg_{\tau' \in \mathcal{T}}(|\{(q, \tau') \in \mathcal{M}_\tau\}|)$ contains the average number of question per topic, while $avg_{\phi' \in \mathcal{L}}(|\{(q, \phi') \in \mathcal{M}_\phi\}|)$ represents he average number of question per learning objective (12 in total).

Table 2. Statistics on the topic and learning objective mappings

Data set	$avg_{\tau' \in \mathcal{T}}(\|\{(q, \tau') \in \mathcal{M}_\tau\}\|)$	$avg_{\phi' \in \mathcal{L}}(\|\{(q, \phi') \in \mathcal{M}_\phi\}\|)$
Earth sciences	8.46	34.78
Nutrition	13.25	26.5
Homeschooling	4.89	19.1

3.2 Experiment

The evaluation of the recommender models introduced in Subsections 2.2 and 2.3 is not an easy task for several reasons. First, to learn such models, a history of user interactions with the QA system is needed. Without any kind of recommendation engine behind the search or browsing functionality, such interactions would not be possible, or even reliable, since the user is not aware of the possible question choices.

Secondly, if suggestions are provided, even in their simplest form, the resulting browsing log would not reflect the users natural learning process, but rather a learning process influenced by the capabilities of the used recommendation engine. Therefore, the recorded question sequences would still not be suitable to be used for training a new recommender model which relies on the natural learning process of the user.

In order to evaluate the performance of the LoR, due to the lack of resources, a scenario of user interaction with a QA system was simulated, where recommendations were not provided at all. Having an overview of the available questions is not feasible, given the size of the datasets. Therefore, for each domain, five subsets of 20 questions were randomly generated and users were asked to order each of these 20 question-sets in the sequence that they, personally, would want to ask them or would want **learn** about.

Table 4 shows, for each of the three domains, the number of collected question sequences, i.e. the total number of user responses, the number of distinct questions within the collected sequences and their percentage with respect to the total number of questions.

Overall, about 13 male and female users participated in this survey, but not all of them provided an ordering for each of the question sets. The obtained number of sequences are generally balanced between male and female participants.

Table 3. Statistics of the collected sequences

Data set	No. of sequences	No. of distinct questions	%
Earth sciences	61	90	28.75
Nutrition	46	94	29.56
Homeschooling	56	84	43.98

3.3 Evaluation Metrics

Evaluating a recommender system on its prediction power is crucial, but insufficient in order to deploy a good recommendation engine [17]. There are other measures that reflect various aspects. However, not all of them are desired to perform well for every recommender.

Therefore, the evaluation of the LoR model should not be based on prediction performance (accuracy and average log-loss) alone, but also on other metrics that capture various desired aspects of a learning-oriented recommender within a QA system. Let us briefly define these metrics.

Catalog Coverage. In general, catalog coverage represents the proportion of questions that the recommendation model can recommend. In our case, we define the catalog coverage as the proportion of questions that the model P can recommend with a prediction value higher than a predefined threshold σ.

Overall, all three recommender models introduced in Section 2 can generate recommendations for any user (i.e. full user space coverage) and, eventually, all questions can be recommended, since the recommender repeatedly excludes already visited ones. But, towards the exhaustion of the database, the recommendations will have a very low prediction value. These recommendations are unreliable. Therefore, we introduce the prediction threshold σ.

In our evaluation, we generally set σ to be the lowest prediction value among the questions within the sequences used for training. Since the user space coverage is equal for all recommender models, we will further refer to catalog coverage simply as “coverage”.

Diversity. Generally, diversity is defined as the opposite of similarity. Within this context, we define the diversity as the average dissimilarity among each question pair within a recommendation.

Let s be a question sequence context. Then, the diversity of $R(s)$ is defined as

$$div(R(s)) = \frac{2}{N \cdot (N-1)} \sum_{\substack{(q_i,q_j)\in R(s)\\ i<j}} [1 - sim_q(q_i, q_j)], \tag{13}$$

where $sim_q : \mathcal{Q} \times \mathcal{Q} \rightarrow [0, 1]$ represents the semantic similarity measure between questions.

During the evaluation, we used the simple cosine similarity together with the semantic concept similarity defined by Lin [14]. In order to avoid further dependencies with our topic taxonomy, the Wordnet [18] lexical database was used instead

Learning Utility. The learning utility refers to the learning gain of a user from a recommendation. One way of measuring learning utility is with user ratings. Since such an experiment can only be performed within a user study setting, a comparative metric is introduced instead that shows how good a model reflects the user learning process.

Consider two sets of equal size: S_{learn} a set of user question sequences based on the user's learning process (like the ones collected during our experiment) and S_{rand} a set of randomly generated question sequences. Each of the sequence pairs from $S_{learn} \times S_{rand}$, corresponding to the same user, have the same length. Now let M be a recommendation model. We train this model with each of the two sequence sets using cross-validation and obtain the accuracy values:

$$a_{learn} = acc(M, S_{learn}) \quad \text{and} \quad a_{rand} = acc(M, S_{rand}). \tag{14}$$

We define the learning utility of model P by comparing the normalized accuracy difference:

$$lu(P, S_{learn}, S_{rand}) == \begin{cases} 0 & if\ a_{learn} = 0 \\ \frac{a_{learn}}{a_{learn} - a_{rand}} & \text{otherwise} \end{cases} . \tag{15}$$

This measure works only under the assumption that the set S_{learn} truly reflects the users' learning process. It shows how dependent model P is on receiving as input learning sequences.

3.4 Results

In the first part, an analysis of the survey results was made, in order to have an overview of the generated sequences and to identify early patterns and correlations between user answers. The results show that, in some cases, the users strongly agree on a particular question sequence, yet in other cases major discrepancies were identified (see Figure 6). This can be explained by the unique and personal way humans understand certain concepts, i.e. the unique conceptual world map existing in each human mind. Additionally, some domain-specific questions are rather ambiguous and up for interpretation. The survey also captures user preferences and personal opinions and, therefore, there are no unanimous answers. For our evaluation purposes, this aspect was preferred over highly correlated question sequences, because it reflects real life situations. Hence, the learned models are not highly accurate, but despite the conflicting user opinions, some of them still proved to identify learning process patterns and use them to make useful recommendations.

In order to show the benefits of the learning-oriented recommender, two other models were considered: a simple recommender (SR) using a VLMC of random variable Q over the question space and a random recommender (RR)

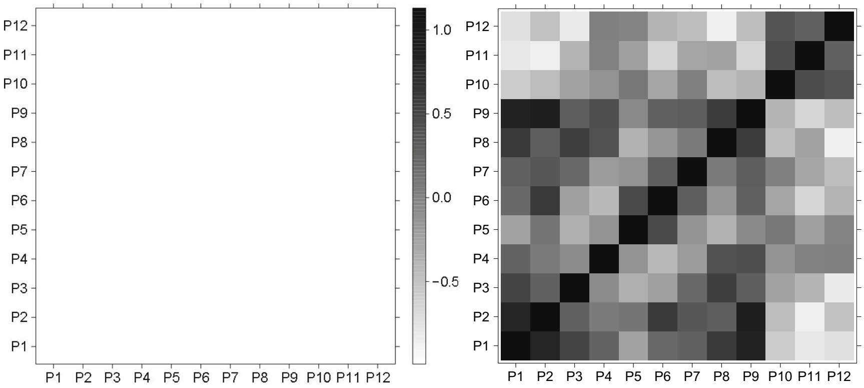

Fig. 6. Correlation matrices of user question orderings for the earth sciences domain

that recommends questions randomly. The SR corresponds to the approach proposed in [10]. Table 4 shows the results obtained using a 10-fold cross validation with input parameters: number of recommendations $N = 5$, maximum order of the VLMCs, $maxOrder = 10$ and $\sigma =$ the lowest prediction value among the questions within the sequences used for training. For testing, the leave-one-out technique was employed.

Table 4. Results

Data set	**Model**	**acc**	**avg-ll**	**cov**	**div**	**lu**
Earth sciences	SR	0.65	93.84	0.29	0.47	0.45
	LoR	0.31	137.69	**0.60**	0.44	**0.69**
	RR	0.01	164.9	1	0.60	0
Nutrition	SR	0.52	112.34	0.30	0.50	0.41
	LoR	0.15	156.23	**0.67**	0.51	**0.49**
	RR	0	165.38	1	0.40	0
Homeschooling	SR	0.50	107.87	0.44	0.36	0.39
	LoR	0.16	145.92	**0.84**	0.47	**0.41**
	RR	0.06	150.06	1	0.42	0

Although the LoR did not achieve an accuracy (acc) and average log-loss ($avg - ll$) as high as the SR, compared to the RR, it still had a good prediction performance (see Table 4). However, the coverage (cov) and learning utility (lu) values of the LoR were much higher, whereas the diversity did not show significant discrepancies among the three models. The coverage values show that the LoR, compared to the SR, can recommend a larger percentage of questions. The increased learning utility of the LoR shows that the prediction performance of this recommendation model is more dependent on receiving as input learning sequences, like the ones collected during our experiments. This means that the

LoR reflects better the learning process depicted in the questions sequences depicted during our experiment.

4 Summary and Future Work

In this paper, a new recommendation technique, called learning-oriented recommender (LoR), is introduced with the goal to improve the user's learning experience while interacting with a QA system.

The evaluation of the learning-oriented recommender is not as easy task for at least two reasons. First, in order to train and learn the recommender model, a substantial history of user learning activity is needed, which is not influenced in any way by other recommenders or other external factors. Secondly, even if such question sequences that reflect the users learning process were to be collected, there is no clear, well established metric to evaluate the performance of the recommender from a learning perspective.

However, a first step was made towards a better understanding of the learning-oriented recommender's capabilities. From each of the above mentioned three datasets of questions (i.e. earth sciences, nutrition and homeschooling), five sets of 20 questions were randomly selected and users were asked to order each set according to their learning preferences in the sequence that they, personally, would ask them or want to learn about.

Five evaluation measures were used to compare the performance of the LoR with a simple VLMC (SR) over the question sequences and a random recommender (RR). Results show that while the SR outperforms the LoR with respect to prediction power, the LoR achieved a much higher coverage and learning utility. The RR was used as a base reference. The obtained results confirm our initial intuition: question sequences are first influenced by the underlying topics and their order, and then, within each topic, by a particular order of learning objectives.

However, further evaluation is required to show that the LoR has great potential in offering an improved user learning experience. To show this in more detail, we intend to conduct an online user study. Additionally, we also plan to analyze the influence of the knowledge-based structure on the recommendation performance.

Additionally, it would be desirable to investigate the potential of an automatic topic-tree generation, and, more importantly the automatic assignment of questions to topics and to learning objectives.

With the information overload, new aspects of existing disciplines are identified or entirely unknown, unexplored fields of study are discovered. In the first case, a restructuring or extension of the current curriculum is required. The second case demands the settlement of the first building blocks. Learning patterns represent relevant knowledge about these domains. By using the learning patterns derived from our recommender model, we could establish new fields of study and (semi-)automatically generate curricula for these domains. Further research in this direction is expected to answer the question whether and how to exploit the learning-oriented recommender model for this purpose.

References

1. Adomavicius, G., Tuzhilin, A.: Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE Trans. on Knowl. and Data Eng. **17**(6), 734–749 (2005)
2. Anderson, L., Krathwohl, D., Airasian, P., Cruikshank, K., Mayer, R., Pintrich, P., Raths, J., Wittrock, M. (eds.): A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives. Addison Wesley Longman Inc. (2001)
3. Baeza-Yates, R.: Graphs from search engine queries. In: van Leeuwen, J., Italiano, G.F., van der Hoek, W., Meinel, C., Sack, H., Plášil, F. (eds.) SOFSEM 2007. LNCS, vol. 4362, pp. 1–8. Springer, Heidelberg (2007)
4. Baeza-Yates, R., Hurtado, C.A., Mendoza, M.: Query recommendation using query logs in search engines. In: Lindner, W., Fischer, F., Türker, C., Tzitzikas, Y., Vakali, A.I. (eds.) EDBT 2004. LNCS, vol. 3268, pp. 588–596. Springer, Heidelberg (2004)
5. Bai, L., Guo, J., Cheng, X.: Query Recommendation by Modelling the Query-Flow Graph. In: Salem, M.V.M., Shaalan, K., Oroumchian, F., Shakery, A., Khelalfa, H. (eds.) AIRS 2011. LNCS, vol. 7097, pp. 137–146. Springer, Heidelberg (2011)
6. Begleiter, R., El-Yaniv, R., Yona, G.: On prediction using variable order Markov models. Journal of Artificial Intelligence Research **22**, 385–421 (2004)
7. Boldi, P., Bonchi, F., Castillo, C., Donato, D., Gionis, A., Vigna, S.: The query-flow graph: model and applications. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM 2008, pp. 609–618. ACM, New York (2008)
8. Boldi, P., Bonchi, F., Castillo, C., Donato, D., Vigna, S.: Query suggestions using query-flow graphs. In: Proceedings of the 2009 Workshop on Web Search Click Data, WSCD 2009, pp. 56–63. ACM, New York (2009)
9. Bonchi, F., Perego, R., Silvestri, F., Vahabi, H., Venturini, R.: Recommendations for the long tail by term-query graph. In: Proceedings of the 20th International Conference Companion on World Wide Web, WWW 2011 (2011)
10. Cao, H., Jiang, D., Pei, J., Chen, E., Li, H.: Towards context-aware search by learning a very large variable length hidden Markov model from search logs. In: Proceedings of the 18th International Conference on World Wide Web, WWW 2009, pp. 191–200. ACM, New York (2009)
11. Hu, D., Gu, S., Wang, S., Wenyin, L., Chen, E.: Question recommendation for user-interactive question answering systems. In: Proceedings of the 2nd International Conference on Ubiquitous Information Management and Communication, ICUIMC 2008, pp. 39–44. ACM, New York (2008)
12. Kabutoya, Y., Iwata, T., Shiohara, H., Fujimura, K.: Effective Question Recommendation Based on Multiple Features for Question Answering Communities. In: Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010, Washington, DC, USA, May 23–26 (2010)
13. Kosorus, H., Bgl, A., Kng, J.: Semantic similarity between queries in a QA system using a domain-specific taxonomy. In: Proceedings of the 14th International Conference on Enterprise Information Systems, pp. 241–246, Wrocław, Poland (June 2012)

14. Lin, D.: An information-theoretic definition of similarity. In: Proceedings of the 15th International Conference on Machine Learning, pp. 296–304 (1998)
15. Qu, M., Qiu, G., He, X., Zhang, C., Wu, H., Bu, J., Chen, C.: Probabilistic question recommendation for question answering communities. In: Proceedings of the 18th International Conference on World Wide Web, WWW 2009, pp. 1229–1230. ACM, New York (2009)
16. Santos, O., Boticario, J.G. (eds.): Educational Recommender Systems and Technologies: Practices and Challenges. IGI Global (2011)
17. Shani, G., Gunawardana, A.: Evaluating recommendation systems. In: Ricci, F., et al. (eds.) Recommender Systems Handbook. Springer Science+Business Media, LLC (2011)
18. Princeton University. Wordnet. www.wordnet.princeton.edu (2010)