Slovak University of Technology in Bratislava
Faculty of Informatics and Information Technologies

FIIT-5208-52628

Bc. Jakub Mačina

Recommendation of New Questions in Online Student Communities

Master's Thesis

Degree Course:	Information Systems
Study field:	9.2.6 Information Systems
Department:	Institute of Informatics, Information Systems and Software Engineering, FIIT STU Bratislava
Supervisor:	Ing. Ivan Srba, PhD.

2017, May

Acknowledgements

I would like to express my gratitude to my supervisor Ivan Srba for his useful remarks, motivation and enthusiasm.

I would also like to thank all members of the Askalot team, PeWe research group lead by Prof. Mária Bieliková, Joseph Jay Williams from Harvard University, course instructors from QuCryptox Quantum Cryptography course and all people involved in deploying and using Askalot at the EdX platform.

Finally, I thank my family for continuous encouragement and support.

Jakub Mačina

Anotácia

Slovenská technická univerzita v Bratislave
FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLÓGIÍ

Študijný program:	Informačné systémy
Autor:	Bc. Jakub Mačina
Diplomová práca:	Odporúčanie nových otázok v online komunitách študentov
Vedúci práce:	Ing. Ivan Srba, PhD.

máj 2017

Výsledky študentov v masívnych otvorených online kurzoch (angl. Massive Open Online Courses - MOOCs) sú podporené participáciou v diskusných fórach alebo najnovšie, v edukačných CQA systémoch (angl. Community Question Answering - CQA). Problémom MOOCs kurzov je nízka angažovanosť študentov o odpovedanie na otázky a s tým súvisiace množstvo nezodpovedaných otázok v diskusných nástrojoch.

Našim cieľom je preto návrh prístupu smerovania nových otázok pre CQA systémy aplikované v doméne vzdelávania. Viaceré existujúce prístupy odporúčajú nové otázky len úzkemu počtu používateľov s vyššou úrovňou znalostí, čo nie je vhodné pre doménu vzdelávania, kde je prospešné zapojiť čo najviac študentov do odpovedania pretože to pozitívne ovplyvňuje ich učenie. Navrhli sme nový prístup k smerovaniu nových otázok, ktorý okrem modelovania znalostí používateľa pre odpovedanie na novú otázku modeluje aj ochotu používateľa odpovedať na danú otázku. Predikcie založené na týchto dvoch modeloch sú skombinované a zoznam odporúčaných používateľov je zoptimalizovaný na základe aktuálneho pracovného zaťaženia študentov. Na modelovanie používateľa boli použité aj dáta z online kurzu, ako napríklad známky študenta a jeho aktivita v kurze, ktoré pomáhajú smerovať nové otázky väčšej časti komunity.

Navrhnutá metóda bola odladená a overená formou offline experimentu a následne bol skúmaný celkový dopad na komunitu pomocou online experimentu. Výsledky online experimentu, ktorý bol realizovaný ako A/B test v CQA systéme v rámci MOOC kurzu na EdX platforme, ukázali zvýšenie presnosti odporúčania nových otázok v porovnaní so všeobecnou metódou smerovania otázok používanou na otvorenom Webe o 4.96% v miere prekliknutia a o 5.30% v metrike S@10.

Annotation

Slovak University of Technology Bratislava
FACULTY OF INFORMATICS AND INFORMATION TECHNOLOGIES

Degree Course:	Information Systems
Author:	Bc. Jakub Mačina
Master's Thesis:	Recommendation of New Questions in Online Student Communities
Supervisor:	Ing. Ivan Srba, PhD.

2017, May

Student's performance in Massive Open Online Courses (MOOCs) is enhanced by participation in discussion forums or recently emerging Community Question Answering (CQA) systems. Nevertheless, the problem is low engagement of students in question answering which leads to many unanswered questions in discussion tools.

The goal of the master's thesis is to propose a new approach for a routing of new questions for CQA systems employed in educational settings. Existing approaches for question routing recommends new questions only to a few experts, which is not suitable in MOOCs because participation in discussions positively influences student's learning outcomes. We proposed a novel approach for question routing which models along user's expertise for a given question also user's willingness to answer a question. The predictions based on these two models are combined and the list of recommended users is optimized by a workload constraint. Furthermore, we incorporated non-QA data from the course for user modelling, such as student's grades and activity in the course, which help in routing new questions to greater part of the student community.

The proposed question routing approach was fine-tuned and evaluated by the offline experiment and the online experiment which measured total impact on the student community. Online experiment was conducted using A/B test in CQA system used by a course at the EdX platform. The proposed question routing method outperformed a baseline question routing method commonly used on the open Web by 4.96% in click-through rate and by 5.30% in S@10.

Diploma thesis proposal

Community question answering (CQA) systems are successful on the open web (e.g. StackOverflow), in enterprise and educational environment. CQA systems have the potential to help mainly student communities, which are getting popular with an increasing number of online courses and where students solve a lot of problems, e.g. related to project elaboration. However, educational domain is specific in several aspects, mainly students can answer only limited number of questions, which must also match their expertise. Furthermore, it is essential to engage as large part of the community as possible. Due to previously stated differences, new approaches for collaboration support of students are required.

Analyze current approaches for collaboration support used in CQA systems. Specifically, focus on the routing of new questions to potential answerers, who are motivated to provide an answer. Target educational domain and discuss, how these approaches are influenced by their employment in educational settings. Propose and implement question answering support method in online student communities. Evaluate the proposed method in CQA system deployed in an educational domain.

Table of contents

1 Introduction

Online communities interested in knowledge sharing are an important part of the current World Wide Web. Among the various question answering (QA) systems, community question answering (CQA) services (e.g. StackOverflow[1]) are one of the most successful. CQA services supplement and outperform search engines in answering complex, opinion and conversational based questions.

CQA systems have a great potential to apply in other domain specific environments. Recent boom of MOOCs (Massive Open Online Courses) created online, very large and diverse student communities. MOOCs are online courses, which provide university-like education online for free. However, online student communities in MOOCs environment represent a specific type of community and new approaches for collaboration support need to be proposed. CQA systems are already successfully applied in enterprise domain and they also offer a solution in the educational settings for solving students' problems more easily.

Question routing represents one type of approach that gains an interest in the CQA systems research in the recent years. Question routing refers to a recommendation of new questions to best potential answerers in order to prevent new question of being unanswered for a long time. Previous research in question routing in CQA systems indicates promising results in increasing number of questions answered in a shorter time and in an engagement of larger part of the community in the question answering process.

In contrast to traditional CQA systems, students in educational community are learning about the particular topic throughout the course and therefore they are not experts in the particular field yet. In educational domain it is essential to support whole community of students to ask, answer and discuss about the problems and thus support their learning. While the traditional CQA systems stressed the importance of the question and answer quality, it is not critical part for CQA systems in educational domain. Vital issue of educational domain is limited students' time for contribution. Matching of students' interest and expertise also plays an important role.

In this thesis, a new approach for recommendation of new question specifically for online student communities is proposed. Proposed method is unique in applying question routing within CQA system deployed in educational environment. By taking into account specifics of online student communities, the goal is to effectively utilize resources of the online student community, to decrease information load of users by accurate recommendations and to involve greater part of the community in the question answering process.

The thesis is organized into following sections: section two describes CQA systems, their open problems and current collaboration support approaches; section three discusses MOOCs and university domain communities, their problems and tools for collaboration support; section four analyze question routing. The proposed approach for question routing in online student communities is presented in section five. Section six discuss implementation detail, section seven presents experiment evaluation and section eight concludes with a summary.

[1] http://stackoverflow.com

2 Community Question Answering

It is natural for humans, that people with common goals or interests are grouping together into communities. At these days, it is not only in a real life, but also in the virtual environment. On the Web, there exists huge number of systems, where majority of the content is created by the members of the community, e.g. YouTube or SoundCloud. The purpose of such systems is social networking, discussions and collaborative knowledge sharing.

1.1 Classification of QA Systems

Question-answering (QA) is a broad concept identifying services, that allow people to post a question online and receive responses to the question. QA services are accessible as a website and varies by exchanged content, the way how the content is exchanged and the type of members that are part of the community. Based on the variation, (Shah et al. 2014) proposed a hierarchical structure of QA services.

Within content perspective, we can classify QA services into *horizontal* and *vertical* QA services. Vertical QA services are focused around a specific topic, whereas horizontal contains broad range of various topics. From answering generation perspective, QA services can be classified into an *automatic* and *human-driven* QA services. Human-driven are based on content generated by a community, while automatic QA systems can process a question and extract the answer for a question automatically.

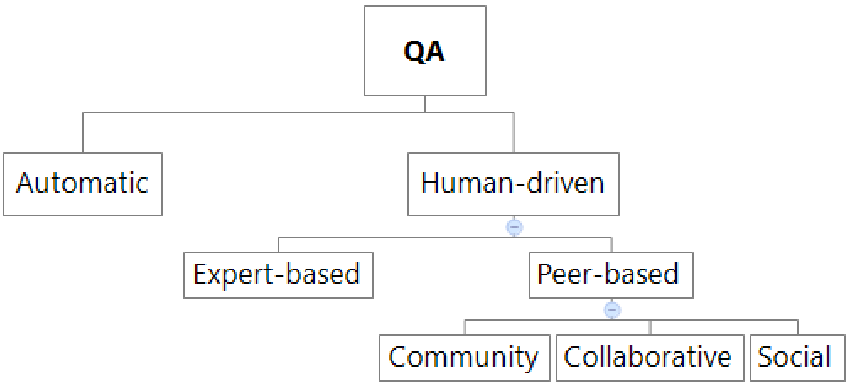


Figure 2-1: Classification of human-driven QA services. (Shah et al. 2014)

The main characteristic of a human-driven service is a community, i.e. members who are actively contributing to the service either by submitting the questions or responses to the questions. The two main distinctions within human QA is whether questions are answered by experts in the topic or by any member of a community can answer a question. We are referring to them as an *expert-based* or *peer-based* respectively.

Peer-based QA is a service on a web platform, where users can seek information by asking a question in a natural language and share a knowledge by answering questions from other participants individually. They can be also considered as a form of a social network where users can interact between each other by asking or answering questions, discussing about topics, voting for answers and even following other members. Some of the peer-based QA systems even motivates their users by gamification mechanism to provide answers.

Peer-based services are classified into:

- *Community QA* – Consist of members of the community, who actively participate in question answering process.
- *Collaborative QA* – Has the same concepts as the CQA, but the main difference is that every member of the community can edit the question and/or answer.
- *Social QA* – It is the newest type of peer-based services, that utilizes the features of social networks (e.g. Facebook, Twitter) to facilitate QA.

This section continues with analysis of two most popular types of QA services for online communities, Community QA and Collaborative QA. They are interrelated and majority of existing QA systems are combination of them. Therefore, it can be referred to both of them by abbreviation CQA.

2.1 Principles of CQA Systems

Nowadays, we can conveniently find information that we seek just by using a search engine. However, there are some needs that search engines cannot satisfy, e.g. complex queries that cannot be easily expressed, the lack of relevant content on the Web, searching for personalized answers or for subjective opinions given by humans (Liu et al. 2012). CQA systems are solving these problems by utilizing the knowledge sharing, wisdom of the crowd and collaboration principles.

Questions in CQA systems are posted in natural language, which is more suitable for humans than searching by keywords in search engines. Time and answer quality trade-off is for information seekers the most essential attribute. By searching in a search engine the answer is retrieved immediately, however it is a presented as a list of links that needs to be further explored to obtain the answer. On a contrary, CQA provides high quality answers even to complex or personalized information needs, but in a longer time period than search engines. Therefore, the main goal of the CQA system is to provide a satisfactory answer for the information seeker in an acceptable time.

The main force behind the CQA systems is a community, i.e. members of the community passionate to ask, discuss, maintain and answer questions about the common interests. According to survey carried by (Shah et al. 2014), more than 50% of the community like to help someone. Furthermore, many CQA systems provide a gamification mechanism, e.g. users can collect badges for activity in the system. Some of the systems use a virtual currency which can be earned by answering questions and spent by asking a question. Other systems just use reputation points to unlock access to more functionality of the system. In general, members of the community consider reputation points as a way of presenting their skills and for making identity and reputation amongst other users.

CQA systems contain variety of questions. Some CQA systems, e.g. StackOverflow, are domain specific and contain factoid or problem solving questions. Other general CQA systems such as Yahoo! Answers, contain questions for discussion, i.e. opinion seeking questions, recommendation or open-ended questions (Dror et al. 2010).

2.1.1 Existing Community and Collaborative QA Systems

StackOverflow

StackOverflow[2] is a domain specific CQA system dedicated to programming. StackOverflow belongs to more general StackExchange[3] platform which groups network of more than 150 communities. These communities are run by experts and enthusiasts in a topic. The main idea behind StackExchange is to build encyclopedias of high-quality question-answer pairs.

To ask a question, user needs to type a title and a text of a question. As questions are organized by tags, user is required to specify at least one tag and at the most five tags. StackOverflow community has rules for asking a question that must be followed. Users must ask a question referring to a specific problem, add details and outline what they have tried so far. StackExchange is just about a question and answers, therefore an opinion or a subjective question are marked by community as inappropriate.

Every member of the community can ask or answer a question. Other members can vote up or down either for questions and answers. Answers for a question are sorted by the difference between number of positive and negative votes. Asker can also choose one answer that satisfied his/her needs as a best answer.

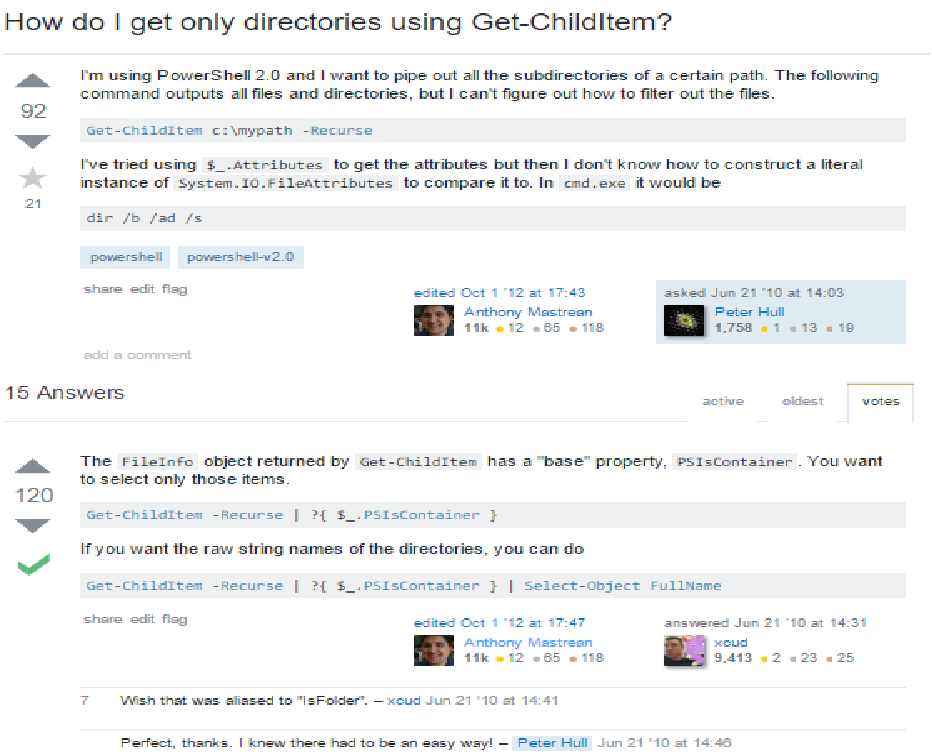


Figure 2-2: Question view in StackOverflow CQA system.

StackOverflow motivates its users by reputation points and badges. Users can earn reputation points for activity in the system. As users are earning reputation points, their privileges in the system are increasing. They can gradually earn privileges to vote up, comment, vote down and at the highest levels even get an access to moderation tools.

[2] http://stackoverflow.com
[3] http://stackexchange.com

Quora

Quora[4] is an example of community QA system with collaborative features. Questions are answered by users individually. However, everybody can suggest an edit to answer or question. Furthermore, every member of the community can collaborate on a question answering process and the community can build the best answer together (called Answer Wiki).

To ask a question, user is required to fill a title and a body of the question. Questions are centered about topics, so it is also necessary to specify topic(s) of the question. Users can vote negatively for questions, while both positively and negatively for answers. Both asking a question and answering a question can be done anonymously.

Quora puts more emphasis on the community itself and has created a kind of a social network. Members of the Quora can follow topics and other members. Every member has a profile, which contains information about the user, his/her followers, following people and following topics. Users in the Quora are usually using their real names which makes Quora unique. Moreover, many famous people are registered and verified by Quora as well.

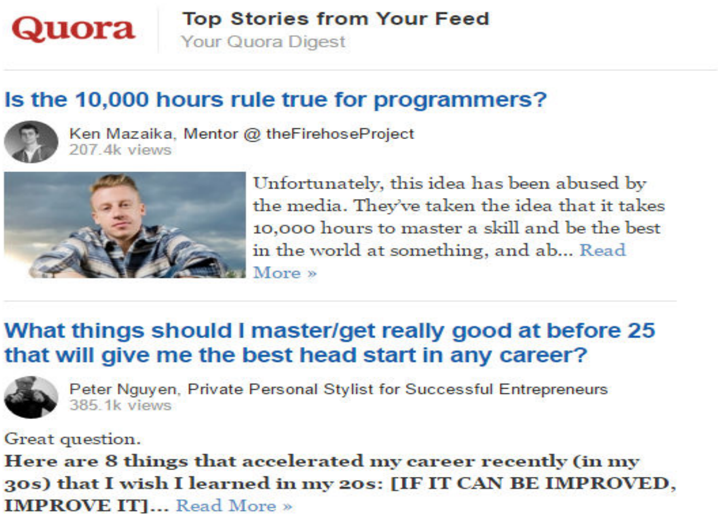


Figure 2-3: Quora weekly email newsletter with most popular questions in the topics following by a user.

Yahoo! Answers

Yahoo! Answers[5] is one of the largest CQA systems. Like StackOverflow, the system is more question-centric rather than user-centric as Quora. One of the main characteristics of Yahoo! Answers is high variance of discussed topics. In comparison to StackOverflow, questions are more discussion based with subjective opinions.

A question thread starts by asking a question with a title and a text of the question. Next, the user chooses a question category from the suggested categories which are automatically generated by the system. The question remains open for four days with an option for extension (Dror et al. 2010).

During this period when the question is in the open state, users can provide answer candidates. Asker can choose the best answer within this period. Finally, the question is marked as resolved.

[4] https://www.quora.com
[5] https://answers.yahoo.com

Yahoo! Answers use points system to motive its users. For example, user can receive one point for answering a question and ten points for answer marked as the best answer. Users spend their points for asking a question, costing five points. As user is earning more points, his/her level is upgrading. Based on the level and number of points, top users will gain recognition by showing their profile on leaderboard on the main page of the system.

2.1.2 Question Lifecycle

Based on the analysis in the previous section, we can generalize question lifecycle into the following phases in existing CQA services as it was first described by (Liu et al. 2008):

1. *Question creation.* User in the role of an asker, asks a question by filling a title of the question and a description of the problem. It is usually necessary to classify the question into the hierarchy of the topics, assign related tags and check related question if the question is not a duplicate.
2. *Question answering.* After the question is posted, other members of the community can find the question in a list of new questions or by searching based on related tags or keywords. These users, in the role of answerers, collaboratively or individually provide answer-candidates for the question. Every member of the community can vote for the answer-candidates to indicate his/her preferences for the best answer.
3. *Best answer selection.* The asker chooses the best answer that satisfies his/her information needs the best. For some of the systems, the asker is required to choose the best answer in a specified time after the question creation. Otherwise, the question with the highest number of votes might be assigned as a best answer. This phase ends by marking the question as answered and moving to the archive.
4. *Question-answer archive.* CQA systems contains vast amount of knowledge encoded in the answered questions in the archive. Other users, who are dealing with the same problem later, can utilize the question-answer archive as a resource of correct answers and solutions for a particular topic. Therefore, systems often facilitate the mechanism for discovering the answered question by full-text search, navigation or faceted search by tags or topics hierarchy.

2.2 Issues in CQA systems

CQA systems have several emerging concerns that need to be solved. Popular CQA sites such as Yahoo! Answers contains hundreds of millions answered question. However, the number of posted question is growing in CQA services. The main goal of the CQA systems might be violated, because new questions might not be resolved in a short period of time (T. C. Zhou et al. 2012). Based on 3 640 randomly sampled questions from Yahoo! Answers, (T. C. Zhou et al. 2012) show, that only 19,95% of new questions in total are resolved within two days.

(Srba & Bieliková 2016b) refers to it as a failure rate, i.e. proportion of deleted or unanswered questions among all new questions. Based on their study on the StackOverflow, failure rate is increasing in average by 0.48% each month. Failure rate is interconnected with the problem of increasing amount of users with low level of expertise asking low-quality questions, while decreasing amount of users with high expertise. For preserving the sustainability of CQA systems, we need to keep or even increase the amount of expert users providing high-quality answers and keeping the system clean.

Due to the openness of the CQA systems, a majority of the users can be categorized as lurkers. Lurkers are members of the CQA community who only consume content but do not actively participate in question answering. According to the analysis on StackOverflow dataset, only 24.8% members of the StackOverflow community have at least one answer[6]. This indicates, that the long tail pattern is present in CQA systems because majority of content is created by minority of users.

All the listed problems negatively affect the main goal of the CQA system, i.e. to get the satisfying answer in a reasonable time. There are two main reasons for this, (1) users are not willing to answer a question, (2) users who are willing to answer are not aware of questions or discussions that are interested for them (Riahi et al. 2012). The first problem of low motivation can be solved by gamification mechanism. The second problem can be solved by approaches that support collaboration between members of the community. In the following section, we are going to analyze collaboration support approaches that are improving the collaboration during the question answering process.

2.3 Current Collaboration Support Approaches in CQA systems

The aim of current collaboration support approaches is to improve collaboration between the members of the community during the question answering process. There exist two main collaboration approaches, which can be analyzed from the question lifecycle perspective:

- *Question retrieval.* Before the new question is posted, the same or very related question-answer pair can be recommended to the asker to answer his/her intended question in order to prevent duplicates.
- *Question routing.* When the answer to the question was not found in the CQA archive, knowledge of the users must be utilized. Question routing represents an approach for recommendation of new questions to the best potential answerers.

Both of the previous approaches are based on content recommendation. To get bigger insight into the current collaboration support approaches, in the following section we are going to analyze the general recommendation approaches that are widely used on the Web in addition to the CQA systems.

2.3.1 Recommendation on the Web

Recommender systems have proven to be powerful and successful in several domains, e.g. products recommendation. Product recommendation tries to recommend products that might be interesting for the user based on his/her shopping history, Web behavior, or based what similar users bought. There are two different strategies for recommendation:

- Content based filtering (CBF)
- Collaborative filtering (CF)

Content based filtering creates a users' and items' profile based on available features. CBF then builds a predictive model of user's preferences based on item profiles that user purchased or viewed. Finally, every item is evaluated by learned model and best matching items are recommended.

Collaborative filtering is based on analyzing relationships between users and interdependencies among products in order to identify new user-item matches (Dror et al. 2010). Input for collaborative filtering is past behavior of users, e.g. product ratings or transactions. The first approach was user-user CF. It computes relationships among users and estimate unknown rating

[6] http://data.stackexchange.com/stackoverflow/query/541760 (as of 19th September 2016)

based on the similarity with other user's ratings (Ekstrand 2011). Later, item-item CF (also called item-based CF) was proposed, which is more scalable approach because user's taste is unstable and it might change frequently. Rather than using similarities between users', item-item CF uses similarities between the items. While CF presents simple, intuitive and working approach, it is still facing cold-start problem as there is insufficient amount of data for recommendation at start.

Both of the recommendation approaches have some drawbacks. However, these drawbacks can be reduced by using combination of CBF and CF, usually referred as hybrid recommenders. For example, CF suffers when a new item without ratings is added, but CBF approaches can still recommend in that case.

CF is not suitable to use in the domain of CQA systems. Main problem of CF in CQA system is the lack of collaborative data, because usually only one answer is needed to completely answer a question. Conversely, a product can be bought by many users which generates more data for CF recommendation. Thus, CBF approaches are used for collaboration support in CQA systems.

2.3.2 Question Retrieval

CQA archives of solved questions are great resources of knowledge and they can be reused. Question retrieval prevents duplicate questions by suggesting answers for a question that user intends to ask. Furthermore, question retrieval can recommend solved questions that extend information about the question or searched keywords, which represents a form of navigation in the CQA system.

The goal of the question retrieval is to find semantically equivalent or relevant questions for the queried question or keywords (Cai et al. 2011). The major challenge for question retrieval is to solve lexical gap, i.e. that language vocabulary is rich and users are expressing similar meanings with diverse words. Because traditional language based models are not suitable for this kind of task, (Cao et al. 2010) applied Translation Model and Translation-Based Language Models. By exploiting latent topics in the query question, (Cai et al. 2011) outperforms models based on translations. Furthermore, (Ji et al. 2012) shows that latent modelling can be further improved by taking into account question along with the answer.

2.3.3 Question Routing

With the rise of CQA systems popularity, an increasing number of questions is being posted every day. In order to prevent new question to remain unanswered for a long time and thus to keep the community healthy, it is important to support question answering process. One active research topic in CQA systems is question routing, which studies new questions recommendation to the best potential answerers.

Most previous studies focus only on the best possible answerers, i.e. experts, to best satisfy the asker needs (e.g. (Dror et al. 2010), (Riahi et al. 2012), (T. C. Zhou et al. 2012), (Tian et al. 2014)). However, to maintain the sustainability of CQA system, it is more essential to satisfy answerers' expectations (Srba & Bieliková 2016b). To improve precision of the recommendation, researchers model various characteristics of users and take into account users' expertise, interest, activity or motivation. For the purpose of matching potential answerers for the question, the most common approach is topic modelling or classification.

Moreover, we need to point out that several research works have aim to engage whole community in question answering process. According to (Szpektor et al. 2013), it is essential to maintain the community ecosystem. (Luo et al. 2014) and (Srba et al. 2015) utilized non-QA data for this task.

The results by (Szpektor et al. 2013) of diversifying and freshening the recommended topics also show the promising results in users' engagement.

2.3.3.1 Discussion

Both, question retrieval and question routing are examples of content based recommendation approaches. Recommender systems are successfully used in product recommendation and current research question in CQA services is how to apply this approach for recommendation of questions.

Question retrieval is more suitable when the CQA systems contain huge amount of answered questions. However, new technologies are emerging and discussed topics of interests are evolving, so it is not possible to find every question in CQA archives.

Question routing utilize the knowledge of the community and therefore has bigger potential to support users' collaboration and thus eliminate the CQA problems. By the proper design of question routing approach, it is possible to engage most of the community into question answering process and by selecting the appropriate user and question features, personalized questions can be recommended for users.

3 University and MOOC Domain

MOOCs (Massive Open Online Courses) expansion in recent years has caused that high-quality education is now easily accessible online for everybody with an internet connection. The idea of MOOCs is to provide university-like education with an open access via the Web. The MOOC platforms offer courses in a wide range of topics. For every online course within the MOOCs domain, thousands of people all around the world are associated into a huge and diverse online learning communities. Each online course provides built-in or external social tools for collaboration of the student's, e.g. discussion board, chat or social network groups.

CQA systems are successful on the open Web and in various domain-specific environments, moreover they have potential to help online student communities which is worth researching. Student communities are present within MOOCs, but they are also naturally created at universities. Some systems already exist at wide range of universities that support collaboration of students online, e.g. discussion boards for a particular course or at a university or faculty-wide level. However, shortage of educational support is one of the biggest issue in current collaboration support tools.

3.1 MOOC Definition and Principles

According to the Oxford dictionary, MOOC is defined as a "course or study made available over the Internet without charge to a very large number of people". Because of the emerging nature of the concept and ambiguities of letters in MOOC abbreviation, the definition is evolving. Recently OpenUpEd[7], one of the MOOCs providers, tried to propose the more precise definition as: "MOOCs are courses designed for large numbers of participants, that can be accessed by anyone anywhere as long as they have an internet connection, are open to everyone without entry qualifications, and offer a full/complete course experience online for free". The main idea is to enable students to get access to free education provided by universities. Usually, the online course mimics universities, i.e. students are watching video lectures, reading additional papers and doing assignments.

From the university perspective, MOOCs offer a great opportunity for teachers at universities to reach large number of students. Based on the study by (Jordan 2014), an average student enrollment for the three most popular sites (Coursera, EdX, Udacity) is about 43 000 students. Because of such amount of learners, it is impossible for teachers to provide a personalized support for students.

Due to huge number of participants, along with the traditional course materials MOOCs provide build-in or external social tools to support community interactions among students, teaching assistants, and professors. Such tools are usually used for socializing, collaborating in order to get deeper insight into the topic or discuss problematic parts of learning materials.

The courses usually last from 4 to 12 weeks and most of them repeat throughout the year. Assignments are typically assessed by peer review – students anonymously review other student's assignments. Tests and exams are usually in form of a quiz.

One of the main characteristics of current MOOCs is dropout rate among students enrolled. For courses provided by Coursera, one of the most popular MOOCs provider, dropout rate can go up to 94 % (Onah et al. 2014). Researchers (Onah et al. 2014) identified that the most frequent issues

[7] http://www.openuped.eu/images/docs/Definition_Massive_Open_Online_Courses.pdf

are lack of time, course difficulty, wrong expectations and lack of support. Lack of support is the issue that can be solved by utilizing CQA systems.

Other important factor of high dropout is free nature of MOOCs. Therefore, in comparison to university education, the goals of the students enrolled in MOOCs courses are very various. Their main goal is not often to complete the course, but sometimes only to watch few lectures or to learn something new, or for having fun. Several researchers have classified behavior of students seen in the MOOCs. For example, Hill[8] defines four student behavior patterns in MOOCs:

- Lurkers. Enroll for the course, but just observe the content, mostly watches few videos.
- Passive participants. Students who watch videos, take quizzes, but not participating in activities or class discussions.
- Active participants. Fully participate in MOOCs by watching videos, taking assessments and quizzes and actively participating in social tools.
- Drop-ins. Students who are active for selected topic within the course, but did not complete the whole course.

(Grunewald et al. 2013) classifies participants enrolled in MOOCs into five groups based on their communication activity in discussion forum:

- Inactive – Participants who do not visit discussion forum.
- Passive – Only consume information in discussion forum.
- Reacting – Usually add further aspects to the questions but do not answer them.
- Acting – Actively participate to the discussions.
- Supervising/Supporting – Provide overview and summarize gained insight in the discussion forum.

3.2 MOOC Platform

In this section, existing MOOC platforms are analyzed. Main approach to support collaboration in the MOOC platforms is discussion board. Therefore, the discussion boards design for all of the platforms are analyzed as well.

3.2.1 Existing MOOC Platforms

EdX

Platform description: EdX[9] is one of the leading MOOC provider offering courses in more than 30 subjects. EdX was founded by Harvard University and MIT in 2012 as a nonprofit organization. In august 2015, EdX reached 5 million registered students[10]. What is unique about the EdX is that they are nonprofit and their platform is open-source. They are investing earned money to conduct a research of new approaches in MOOCs. EdX courses consist of weekly learning sequences with short video lectures, additional materials and learning exercises. For a reasonable fee, one can earn a verified certificate after successfully completing the course.

Navigation in discussion board: Posts can be filtered by a topic and they are showed on right side as in Figure 3-1. The posts that was pinned by staff team are showed first. Posts by staff members

[8] http://mfeldstein.com/the-four-student-archetypes-emerging-in-moocs/
[9] https://www.edx.org/
[10]https://twitter.com/edXOnline/status/631844606964035588

are distinguished by labels. Users can follow a post to get notifications and can upvote the posts. Replies within a post can be sorted only chronologically.

Creating new post in discussion board: When creating a new post, user must choose the title, body, post type and the topic area of the new post. Post type is either questions or discussion. Question type is about issues that need answers and discussion type is for idea sharing and conversations. Furthermore, user has also the option to post the question anonymously.

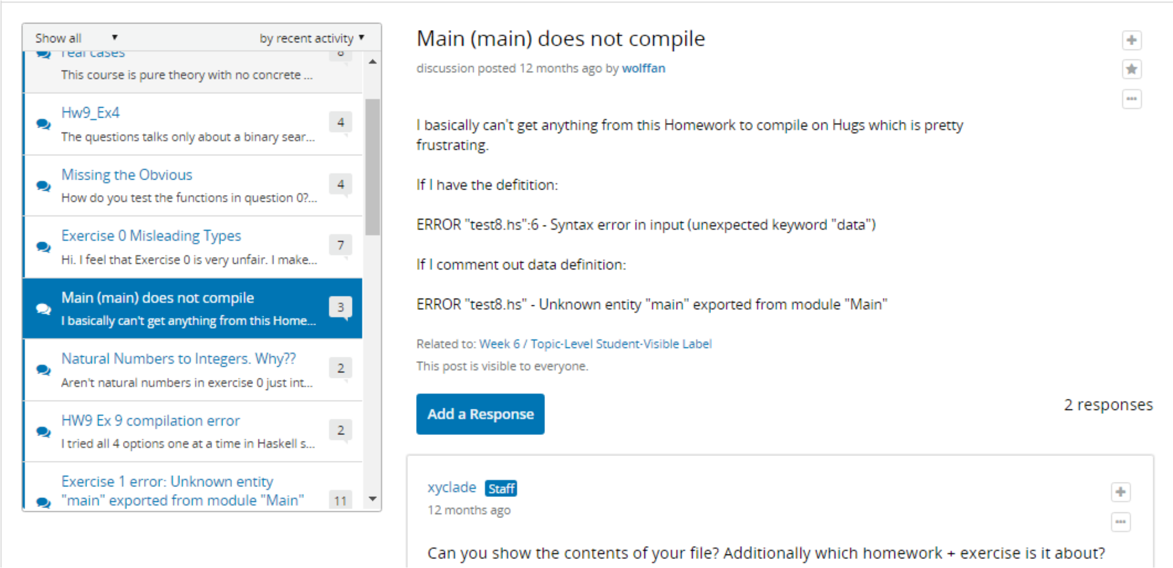


Figure 3-1: Question view of EdX discussion board for Introduction to Functional Programming by Delft University of Technology.

Coursera

Platform description: Coursera[11] is one of the most well-known MOOCs provider. Coursera is based on the same principles as EdX, except that Coursera is a for-profit company. The courses are for free, but if students want to get a verified certificate, they must pay a fee. Furthermore, Coursera also offers an option to apply the credits for the course at the American universities by taking a proctored exam.

[11]https://www.coursera.org/

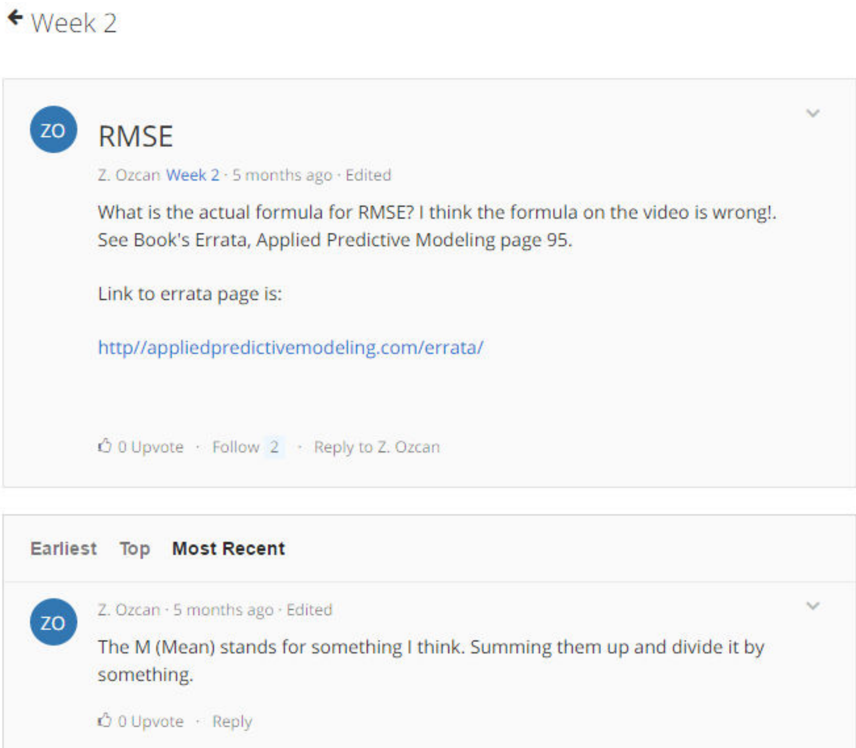


Figure 3-2: Question view of Coursera discussion board for Practical Machine Learning course by Johns Hopkins University.

Navigation in discussion forum: Every module of the course contains a discussion forum. There are also general forums for general discussion, meet and greet and one for creating study groups. Posts within the forum can be filtered to show latest, most popular, or unanswered posts. Users can follow and upvote posts. Replies within a post can be sorted by votes for the reply, most recent or earliest replies as can be seen in Figure 3-2.

Creating new post in discussion forum: User is required to set a title, body and a related module of a new post (called thread).

Udacity

Platform description: Udacity[12] is another big MOOCs provider. Like Coursera, it is a for-profit company and therefore, majority of the courses are not free of charge. The platform originally focused on university-like courses, now it mostly concentrates on professional courses. Therefore, the Udacity platform is collaborating with specialists from global companies like Google, Facebook or Twitter for course content preparation. Udacity is using open source discussion board system called Discourse[13].

[12] https://www.udacity.com/

[13] https://github.com/discourse/discourse

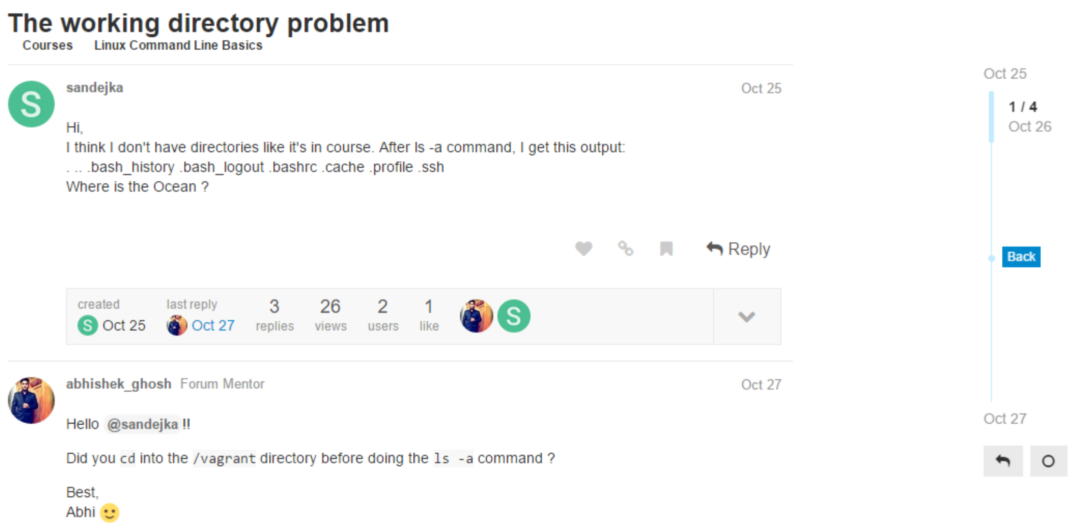


Figure 3-3: Question view of Udacity discussion board for Linux Command Line Basics course.

Navigation in discussion forum: For every course, there is an associated discussion forum. Discussion forum does not contain any categories or tags. Posts in the discussion forums and replies within the posts are sorted only by activity. Furthermore, there is no concept of negative votes; users can express only positive opinion by liking the post.

Creating new post in discussion forum: User is required to set a title, related course and a text of the new post.

3.2.2 Other Collaboration Support Approaches

Besides discussion boards, other collaboration support approaches consist of associating students in the groups based on their similarity, e.g. their learning style, interests or teaching capability.

(Ferschke et al. 2015) implemented a collaborative chat, where pairs of students can work on specified activities within a course in real time. When students enter a chat, the algorithm finds them the best partner according to their learning characteristics. They integrated it into a course in the EdX platform and their results shows reduction of attrition of students who used the chat.

Next approach helps answering question of students by grouping similar students together to solve a question (Rosmalen et al. 2007). When student ask a new question, system sets up a wiki and find most suitable users for the questions. Asker and selected students than collaboratively solve the question through wiki. Authors called proposed approach as type of a peer tutoring. Students are selected based on their competency to be a tutor, availability and similarity to the asker. These features are extracted from students' previous activity in learning platform and personal calendar of students.

Other interesting approach is the concept of the virtual currency proposed by GreenDolphin (Aritajati & Narayanan 2013). Including the activity in discussion forum to the final grade of the online course is another approach to motivate students. However, an example[14] from one course offered on Coursera platform by Duke university shows that students did not like graded discussions.

[14] https://cit.duke.edu/blog/2014/06/coursera-forums-students-dont-like-graded-discussions/

3.3 Issues of Online Student Communities

Discussion forums in MOOCs face similar problem as general CQA systems. Because the average number of the students in the course is very high, the number of the questions asked is proportionally high as well. It leads to the state, where finding interesting question or discussion opportunities for students in the discussion forum can be difficult. According to the (Yang et al. 2014), around half of the posted questions are never resolved.

Questions failure rate in MOOCs collaboration tools can have even bigger impact than in the CQA systems. Students, who do not get their questions answered, might have a problem of understanding the content of the course, which may lead to course dropout. The completion rate for most of the courses is below 13% (Onah et al. 2014), so by decreasing failure rate of question we can help those students to complete the course, who are willing but may need a help sometimes.

The previous problem of unanswered questions is directly related with the problem, that only a small fraction of participants in online course are actively using social collaboration tools. According to the study of (Breslow et al. 2013), based on the data from the first EdX course, only 3% of all 155 000 students participated in discussion forum.

(Klusener & Fortenbacher 2015) tried to predict success based on forum activities in MOOCs and implement a machine learning classifier, which classifies students into risks and non-risks students. Their results have shown, that difference between successful and dropout students is their activity in discussion forum. Moreover, the next most important characteristics of successful students are answer count and number of up votes. (Breslow et al. 2013) show in their work, that 52% of students who completed the course were active in the forum. According to (Alario-Hoyos et al. 2014) it is even more. He claims that 65.4% (298 of 456) who pass the course contributed in any of the social tools and from those who did not pass the course, only 14.3% contributed in any of the available social tools.

3.4 Educational CQA in MOOC and University Domain

The aim of CQA systems used in the educational domain is to support collaboration of students, create social connections and to involve users in online students' communities. By asking questions, students are improving communications skills. Answering a question is beneficial for students' knowledge even more as students are improving their problem solving, critical thinking and deeper understanding about the topic.

3.4.1 CQA in Comparison to Discussion Boards

In general, both discussion boards and CQA systems are services, where users can discuss about various topics, organized in hierarchical structure, by posting messages. The main difference is that CQA systems offer more tools for collaboration of members and they are more community driven.

As seen in section 3.2.1, discussion boards usually contain several topics. Within each topic, new conversation might be started which is called thread. On the other hand, CQA systems are more structured because categories form deeper tree structure, e.g. course at first level, week at second level, topic at the third level and at the last level is a lecture. Moreover, tags can be assigned to posts in CQA systems to describe topic on finer level of detail.

CQA systems allow users to vote for posts, which forms the basis for reputation system. Reputation points can increase privileges in the system and they are visually highlighted in the members' profile. This also influence the quality of question and answers in CQA system, which is in general higher quality. Posts in discussion boards are more discussion based. By voting of the community, CQA system utilize the collective knowledge to filter undesirable posts while discussion boards have individuals in the role of moderators.

In the analysis of MOOC platforms in the section 3.2, it can be noticed that majority of MOOCs platforms use discussion boards. However, there are few courses which recently started to use CQA systems, e.g. CS50 course offered by Harvard on the EdX platform use StackExchange CQA system[15].

3.4.2 Existing CQA Systems in Educational Domain

Askalot

Askalot[16] proposed by (Srba 2015) is an open source CQA system that is successfully used in organization-wide domain, i.e. faculty domain in Slovak University of Technology. Askalot is a novel concept that fills the gap between open (access for everybody on a Web) and too restricted (e.g. access only within a specific course) class communities. The main idea of Askalot is to involve diverse students in a question answering, students from different classes and study degrees, with different grades and experience.

While creating a question, students are demanded to select a category of the question and corresponding tags. Askalot contains at most two-level hierarchy of the categories, at first level it is category for every course taught in university, and at the second level within courses it is the internal structure of the course (e.g. lectures, exercise sessions, assignments). Students can choose from predefined tags or create their own.

Because Askalot is used within a university domain, only students of the particular university can login and involve themselves in question answering process. Students even have an opportunity to ask question anonymously.

The next important concept to mention is the presence of professors and teaching staff. Teachers are part of the community as well as students and they can ask or answer questions. Their contribution is visually highlighted to indicate an expert answer.

To motivate users to contribute to the system, Askalot has built-in reputation system (Huna et al. 2016), which gives students points for being active and for the high-quality contribution. Based on reputation, Askalot has a gamification mechanism that allows users to collect badges. In addition to these motivations, reputation of the community and teachers' evaluation represent external motivational factors for knowledge sharing.

[15] https://cs50.stackexchange.com/
[16] https://github.com/AskalotCQA/askalot

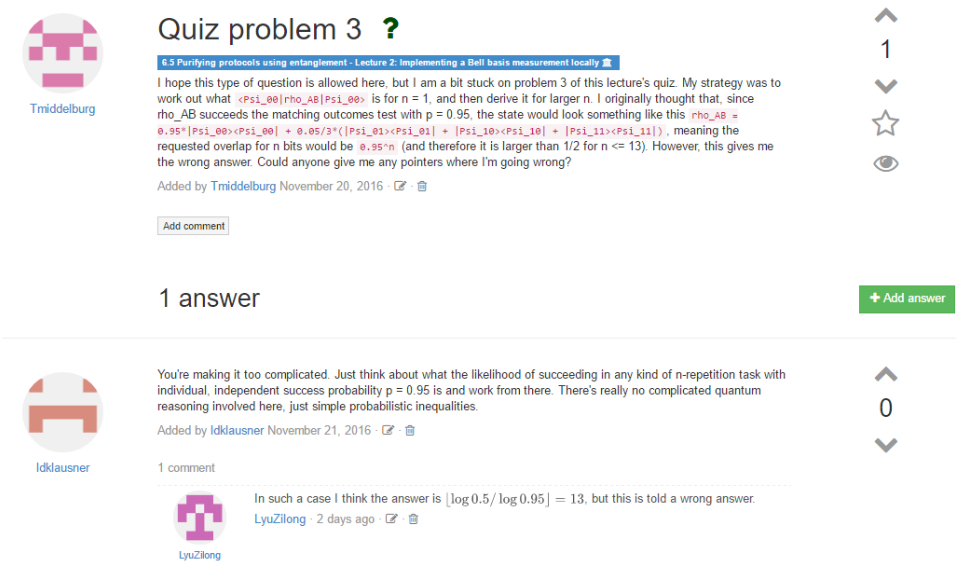


Figure 3-4: Question view in Askalot CQA system.

GreenDolphin

GreenDolphin proposed by (Aritajati & Narayanan 2013) is a CQA system for students learning programming. GreenDolphin focus on beginner programming courses and it is an example of a CQA course with restricted access where only enrolled students in these courses can interact. It has typical features of CQA systems but contains several different ideas as well.

Similar to other CQA systems, GreenDolphin has a reputation system. and utilizes the economy of points to encourage students' participation. On one hand, GreenDolphin awards students for collaboration with points, such as asking or answering a question. On the other hand, students are spending their points for direct questions to student experts or teaching staff.

Another important idea of GreenDolphin is that fast and high-quality answers can decrease collaboration. If these answers are from student experts of teaching staff, students may lose motivation to answer and opportunity to work on the problem by themselves. Therefore, system delayed these answers to provide more time to other students.

Piazza

Piazza[17] is one of the most popular educational question and answering forum. Piazza is an open system and highly used by many professors to support their courses. Every class has its own forum and course page for course information and course resources.

Principles are easy: students ask a question and receive an answer, one from teaching stuff and one from students. Piazza is based on wiki, meaning students collaboratively edit single student answer to a question and following with a discussion below. It has similar concepts as Quora.

Student can post to entire class or only to instructor. Not only a question can be asked, Piazza also supports creating a note or polls. Students can vote for a question and express their opinion by a phrase "thank you" for the answer. Moreover, teaching staff contribution is highlighted and they can endorse good content as well.

[17] https://piazza.com/

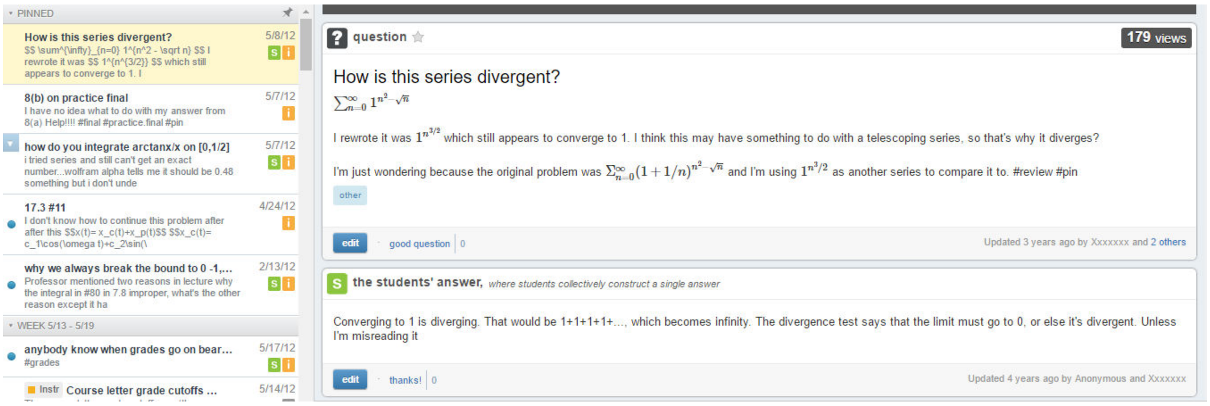


Figure 3-5: Question view of Piazza CQA system.

Open Study

Open Study[18] is an online social learning collaboration tool that help learners to connect to study together and engages them in interactions (Ram et al. 2011). Open Study is an open system, where everybody can join and learn and it is suitable for self-learners who are doing course at their own pace.

Students can choose from a variety of topics to learn. They can ask or answer questions, discuss about topics or chat with other learners. Community of learners can also collaborate on shared learning task formulated by a teacher. Open Study use also the concept of virtual currency and reputation. Reputation score is measured in areas of teamwork, problem solving and engagement.

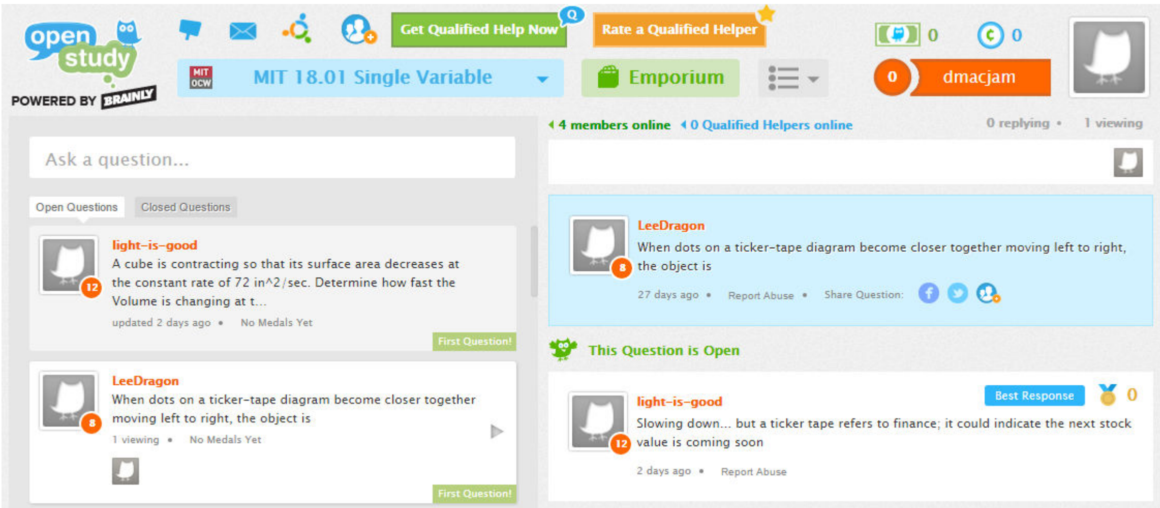


Figure 3-6: Open Study user interface.

3.5 Discussion

Based on the analysis, it is obvious that the activity in discussion forum or CQA system crucially improve probability of passing the online course. Therefore, proper design of educational CQA system for collaboration which increases the proportion of answered questions is essential. Existing collaboration support approaches mentioned in section 3.2.2 prove, that they are important in improving collaboration rate in online communities and decreasing dropouts in online courses.

---

[18] http://openstudy.com/

To summarize, we identified challenges for sustainable collaboration tool for every type of participants' behavior in MOOCs:

- Inactive. Participants, who do not use collaboration tools. The goal should be to involve them in collaboration.
- Dropouts. Participants willing to pass the course, but have difficulties with topic learned. The goal should be to motivate them to ask questions and be confident about using social tools for asking questions.
- Active. Participants that fully participates in collaboration tools. The goal is to preserve their activity.
- Lurkers. Participants consuming the content in collaboration tools without actively participating. The goal is to involve and motivate them in the question answering process.

4 Question Routing

Finding the right answerers who answer new questions in a reasonable time is essential in an educational domain, where the gap between completing or failing the course is very thin. Question routing in CQA systems is promising approach for finding suitable answerers for new questions. Based on the analysis so far, we decided to aim at question routing instead of question retrieval. The rationale is that utilizing community of students instead of CQA archives can tackle each new question without limiting to archive of questions that have been addressed in CQA system before. Moreover, community can also bring new and updated answers for the questions already asked. Another important reason is educational-specific advantage of question routing which consist of:

- Students can learn new skills and knowledge by contributing to CQA system.
- Greater part of the community can be involved into question answering.

4.1 Question Routing Process

One of the main goal of the CQA systems is to provide suitable answer to question in reasonably short time. Due to increasing number of questions and the problem of passive users in the CQA systems, many questions remain unanswered. Even when a user wants to help somebody and share his/her knowledge, in popular CQA systems it is difficult to find the right question to answer. Users are overwhelmed with the number of open questions and they have problems to find interesting questions or discussions suitable for them (Guo et al. 2008).

Question routing is solving the problem by filling the gap between questions without any or best answer (open questions) and potential answerers. Question routing recommend open questions to potential answerers who are most likely to provide a satisfying answer (Srba & Bieliková 2016a). The term question routing is relatively new in QA research; sometimes it is alternatively termed as answerer recommendation, expert finding or question recommendation.

From the seekers perspective, question routing can reduce time to answer their questions. It can increase satisfaction of askers and they might be more willing to contribute with their knowledge to the CQA system in the future (T. C. Zhou et al. 2012). From the answerers perspective, when question routing filters questions only interested for them, they would be more interested and have more expertise in providing answers to these questions. By recommending the right questions to the right users, the CQA system can fully leverage the knowledge of the community.

Question routing can be seen as a problem of given a new question to find ranking of the most suitable users to answer it. Term most suitable users is quite general, but in the following section we take an insight into different approaches done in the question routing field. Question routing process is usually composed of minimally three phases as was first defined by (Guo et al. 2008):

1. Construction of question profile, which aim is to capture topic(s) and information need.
2. Construction of user profile, which models users based on various features, e.g. user's expertise, activity or motivation.
3. Matching model for finding relevant user profiles for particular open question profile. Output of this model is usually an ordered list of users that are sorted by their probabilities in descending order.

4.1.1 Question Profile

Question is described by textual attributes – a title and a body of a question. These textual attributes are tokenized, stemmed or lemmatized, and stop words are removed.

Question is usually represented in vector space as a *bag-of-words* model. Bag-of-words model is built as a vector, which contains term frequencies (TF), or weighted terms frequencies by TF-IDF (short for term frequency–inverse document frequency). (Dror et al. 2010) adds filtering of N best terms and weights words by entropy.

Because texts with the same meaning can be written by different words (e.g. by using synonyms), more abstract representation is suitable to capture the semantics of the question. Therefore, texts of questions and answers can be represented also as probability distributions of belonging to the topics. These topics are called *latent*, because they are expressed only implicitly from words in the question or answer. Probability distributions of latent topics are used to compare questions or answer between each other. The current state-of-the-art probabilistic topic model is Latent Dirichlet Allocation (LDA) (Blei et al. 2003). Other approach is probabilistic Latent Semantic Analysis (pLSA) or Segmented Topic Model (STM) (used in (Riahi et al. 2012)) .

Other features that form a question representation include question metadata, such as a category, or hierarchy of categories, if available. (Szpektor et al. 2013) proposed unique approach and they represent questions as a combination of LDA topic vector, lexical bag of words model and category model. LDA model and category model captures high-level topics of the question while lexical model depicts fine-grained word level interests.

4.1.2 User Profile

For building a user profile, majority of studies use features derived from users' activity in CQA systems (we will be referring to them as QA data). It means that user profile is built mainly from users' asked questions and provided answers in CQA system. User profile is then created by an aggregation of particular question profiles or concatenation of question texts.

User's data from CQA system represent suitable features for question routing. However, not all features for recommendation are always available. For example, there are no QA data for newcomers or users with low level of activity. Consequently, several research papers utilize non-QA (data not extracted from the CQA system) data to improve question routing. (Luo et al. 2014) proposed a question routing in the CQA system in enterprise environment, which derives non-QA data from company's internal systems, e.g. personality tests, social network of employee and current work state. Similarly, (Srba et al. 2015) proposed question routing for CQA system StackOverflow and as a source of non-QA data they used users' "about me" texts and users' homepages. Their experimental results showed improvement in precision of question routing when using both QA and non-QA features.

It is important to mention, that users in the CQA systems usually have two roles, role of an asker and role of the answerer. (Xu et al. 2012) model both roles separately and underline that answerer role is more effective as user profile for question routing. We can conclude that answering of the question is an expression of expertise while asking a question is lack of expertise.

In the following sections, we are going to analyze different aspects that authors take into an account for question routing.

Topical expertise
Topical expertise of users measures the knowledge to answer the question.

(Liu et al. 2010) use for modeling user's expertise only the user's best answers within particular topic. (Riahi et al. 2012) use latent topics and build user profile based on all user's answering history. (Chen et al. 2014) combined user's provided tags with user's answers and user's browsed history of questions.

(Tian et al. 2014) compute user expertise based on data in StackOverflow by weighting positive votes and best answers positively, while negative votes negatively. They also model interest and expertise. Interest is tightly related to the expertise. It is represented as aggregation of all answered questions while expertise is computed as weighted aggregation of all answered question based on number of votes for each answer. The rationale behind interest is that users have a bigger tendency to answer questions that are related to their area of interest. They model user's interest as combination of latent topics from previous user's answers.

Other approaches that are tightly related to finding authorities in communities, use networks of question and answers. It is a *graph representation* of community, where nodes represent users and edges represents information flow. One of the early approaches proposed by (Jurczyk & Agichtein 2007) uses link analysis techniques based on HITS algorithm. (Zhou et al. 2009) use similar approach for re-ranking in question routing process. At first, they compute the expertise of users according to previous answered questions. Then, they re-rank the user expertise by adopting graph based algorithm PageRank for ranking users by their authority for a given question.

Activity

Activity can be reasonable feature to take into account when modeling user profile, because users can be active only at specific time periods, inactive for longer period, or completely lost interest in a topic. For question routing task, users with frequent and recurrent activity are more probable to answer new questions in reasonable time.

(Liu et al. 2010) models an activity as an exponential function which depends on the difference between last question time and last answer time. Other works, e.g. (Tian et al. 2014) and (Srba et al. 2015), followed this approach.

Motivation

Even if the users are able to answer a question, they may not be willing to answer it. It is important to model the motivation or willingness of the user. (Luo et al. 2014) utilized data from personality test to estimate motivation of the users. Different approach was proposed by (Chen et al. 2014), as they tried to estimate the right answering day and time for a user. Moreover, they kept track of number of answers user has provided in recent days in order to model user's question overload that is related to motivation. The last feature that contributes to motivation is unsocial tendency, i.e. click-through rate and answer rate of past routed question.

Combined approach

(Luo et al. 2014) combined three user profile aspects (expertise, activity and motivation) in an enterprise CQA system and add features measuring readiness. They model users' expertise based on their previous questions and answers. Moreover, they take into account employees' organization. For modeling users' activity, they used number of users' answers and for modeling users' motivation, they utilized data about their personalities, which was derived from the personality tests. By measuring readiness, i.e. users' work load, they use employees' work state and current number of routed questions.

4.1.3 Matching Model for Finding Potential Question Answerers

The first approaches, where question and user profile was represented as a bag of words, use *language models*. Language models are used to calculate the probability of user generating the question. (Liu et al. 2005) compares three language models in finding experts in the CQA systems task: query likelihood model, relevance model and cluster-based model. Query likelihood model slightly outperformed other methods and achieved best results in all datasets.

Even though *translation models* significantly outperform previous approaches as shown by (G. Zhou et al. 2012). These models can represent synonyms, but they still cannot reasonably capture semantic similarity between questions. However, *topic based models* solve this problem and LDA topic model is used in latest research works as state-of-the-art approach. Proof that LDA significantly outperform language models based on TF-IDF are in (Tian et al. 2014). Moreover, LDA also outperform language model based on query likelihood (Ji & Wang 2013).

(Szpektor et al. 2013) present unique matching model approach, which prevents well-known recommendation problem of filter-bubble. They proposed question routing that promotes diversity and freshness. Results were evaluated both offline and online on Yahoo! Answers, and algorithm promoting freshness and diversity show increased number of answers by 17%, increased daily session length by 10% and positive impact on associated CQA activities in comparison to previous user interface. The recommendation based only by relevance/interest underperformed previous user interface in number of answers.

Ranking model

In case of language, topic and translation models, two options for ranking question profile with user profile are used. Questions' and users' profiles can be either ranked by vectors similarity or query likelihood language model based on Bayesian inference.

Various *vector similarity* measures can be used for the ranking of relevant questions to the users. (Szpektor et al. 2013) implemented dot-product similarity. Other similarities that can be used are cosine similarity for vectors used by (Riahi et al. 2012) or Hellinger distance for probability vector distributions.

Query likelihood language model (QLLM) rank answerers based on the probability that their profile is about the same topic as a question. For computing probability $P(u|q)$ that question q is generated by user profile u uses Bayes' rule on equation (1). Equation (2) represents language model with smoothing parameter λ.

$$P(u|q) = \frac{P(q|u)P(u)}{P(q)} \tag{1}$$

$$P(q|u) = \prod_{i=1}^{|W|}[\lambda\, P(w_i|\theta_u) + (1-\lambda)P(w_i|\theta_C)] \tag{2}$$

$$P_{LDA}(w|\theta_u) = \sum_{k=1}^{K} P(w|z_k)\, P(z_k|\theta_u) \tag{3}$$

where θ_u represents user profile, θ_C represents whole corpus of questions and answers texts, $P(q)$ is the probability of question q, which is the same for all users. Probability $P(u)$ is a prior probability of a user u, that can be approximated by specific information known about the user from previous CQA information. $P(q|u)$ is a probability that question q is generated by user profile u, and it is usually computed by LDA as in equation (3) or by TF-IDF maximum likelihood.

Query likelihood language models are used in works (Tian et al. 2014), (Srba et al. 2015), (Riahi et al. 2012) and (Liu et al. 2010).

Classification models

Another category of matching models are *classification-based* approaches. Classification is the problem of categorizing observations into discrete classes. In other words, classification models are finding decision boundaries which divides the classes in the input space.

(T. C. Zhou et al. 2012) combine local features, that describe user and a question whereas global features describe users and questions in global perspective of CQA service (e.g. average question length). These features are used as an input for Support Vector Machine (SVM) classifier.

SVM is a classifier, that tries to find hyperplane decision boundary with maximum perpendicular distance (margin) between the closest points of different classes (James et al. 2014), as shown in Figure 4-1. Decision boundary can be expressed in terms of limited number of support vectors that lays on the margin of the decision boundary. To perform non-linear classification, SVM classifier uses kernel trick. Kernel trick maps input from input space (primal problem) to high-dimensional feature space (dual problem), where the problem can be linearly separated. However, kernel function must be manually specified. Most common kernel functions are linear, polynomial or radial basis function. SVM use penalty parameter C that regularize how misclassification of individual observations is tolerated. This parameter is usually fine-tuned to prevent overfitting.

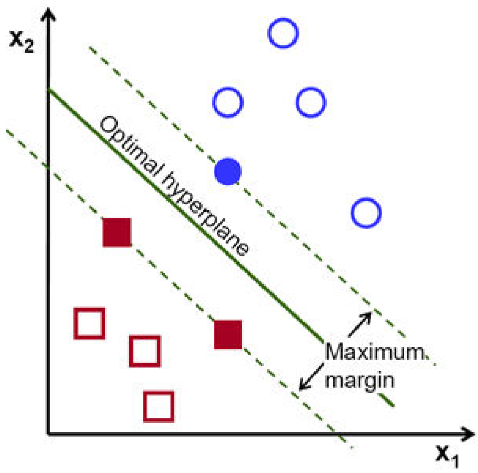


Figure 4-1: Hyperplane with maximum margin found by SVM.[19]

(Luo et al. 2014) predicts users' interest in answering a question by logistic regression and (Chen et al. 2014) predicts answerers by random forests algorithm.

Random forests classifier is based on the idea of ensemble learning, where independent predictions of multiple models are combined (James et al. 2014). Ensemble learning improves prediction accuracy because it reduces variance of final prediction. Random forests classifier is based on bagging, which is technique for majority voting or averaging predictions of many uncorrelated decision trees. To ensure that trees are not correlated, each individual decision tree consider only random subset of features for the split. Moreover, the decision tree is trained on the bootstrapped training samples. Decision tree is simple classifier, which builds binary tree and within each node it chooses one feature as a split criterion and threshold parameter for the split. The feature for split criterion is chosen by Gini impurity or information gain measured by entropy. The stopping

[19] http://docs.opencv.org/2.4/doc/tutorials/ml/introduction_to_svm/introduction_to_svm.html

criterion for building decision tree is maximum depth, node purity or number of data points in the node.

Logistic regression is classifier modelling probability that example belongs to a particular category (James et al. 2014). It is applying logistic sigmoid function y to a linear regression $h(x)$. The task is to estimate coefficients β_0 and β_1 which represent weights of features X by minimizing cost function J:

$$h(x) = \beta_0 + \beta_1 X \tag{4}$$

$$y = \frac{1}{1 + e^{-h(x)}} \tag{5}$$

$$J(\beta) = -\frac{1}{m}[\sum_{i=1}^{m} y \log h(x) + (1-y)\log(1-h(x))] \tag{6}$$

Common optimization algorithms for minimizing cost functions are gradient descent, stochastic gradient descent or conjugate gradient. Regularization weight is used to predict overfitting.

Other studies use techniques known from *recommender systems*. For example (Dror et al. 2010) combines recommendation based on collaborative filtering and classification. Authors proposed multi-channel recommendation model, which combines textual and interaction features and weigh them according to which of the seven channels (asked, best answered, answered, voted on question, voted on answer, traced) they belong to. Then they train binary decision tree classifier based on all the previous features to distinguish between question that meets user's preferences and skills, and questions that do not.

4.1.4 Evaluation of Related Works

Evaluation metrics

The most common metrics for question routing evaluation are success at N (S@N), precision at N (P@N), mean average precision (MAP@N), mean reciprocal rank (MRR) and normalized discounted cumulated gain (nDCG@N). These metrics are well-known from information retrieval field.

S@N equals to one if any predicted answerer is relevant in the top N users. It means whether a ground truth answerer is among the top N users ranked and it is computed as an average across all the queries.

P@N represents an overall number of predicted relevant answerers r for all queries Q in the top N users (or number of true relevant answerers R_i for a query i, if it is less than N):

$$P@N = \frac{1}{|Q|}\sum_{i=1}^{|Q|} \frac{r_i}{\min(R_i, N)} \tag{7}$$

MAP@N is computed as a mean of the average precisions for all queries:

$$AP@N = \frac{\sum_{k=1}^{N} P(k)}{\min(r, N)} \tag{8}$$

$$MAP@N = \frac{1}{|Q|}\sum_{i=1}^{|Q|} AP@N^{(i)} \tag{9}$$

where P(k) is precision at cut-off k, r is number of relevant answerers.

MRR is an average of reciprocal ranks of all routed questions Q:

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i} \tag{10}$$

where $rank_i$ refers to position at which first ground truth answerer was ranked.

The idea of *DCG* is that ground truth answerers appearing on lower positions should be penalized more. Because there might be various number of ground truth answerers for each question, all equations below are computed up to specified position k. nDCG is computed as average DCG across all queries Q normalized by ideal DCG (IDCG) as seen in equation (13).

$$DCG_k = \sum_{i=1}^{k}\frac{2^{rel_i}-1}{\log_2(i+1)} \tag{11}$$

$$IDCG_k = \sum_{i=1}^{|REL|}\frac{2^{rel_i}-1}{\log_2(i+1)} \tag{12}$$

$$nDCG@N = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{DCG_N^{(i)}}{IDCG_N^{(i)}} \tag{13}$$

where rel_i is relevance score for answerer on the position i.

Evaluation type

Majority of the previous question routing studies evaluates their results offline, e.g. (T. C. Zhou et al. 2012), (Tian et al. 2014), (Riahi et al. 2012). Offline evaluation is based on already answered question, where list of answerers or best answerer is considered as ground truth. The drawback of offline experiments is that they are biased, because by the time user will see a question, it may be already answered by high-quality answer. In that case, potential answerers lose motivation to answer such question. Classification based approaches are evaluated only offline and authors usually preprocess and filter data for question routing. That makes the recommendation easier, for example when not all users are taking into account as (Riahi et al. 2012) filtered only users that have at least 20 best answers. In spite of disadvantages, offline evaluation allows researchers to compare results that are tested on the same dataset.

Proposed approaches in question routing field are evaluated by online experiments rarely. However, these experiments are more realistic and provide more precise evaluation. (Szpektor et al. 2013) used offline experiment for comparison to other approaches, which was followed by online experiment realized by A/B test. (Chen et al. 2014) conducted an online experiment on big Chinese QA service Baidu Zhidao[20] where they measured click through rate (CTR), answer rate and answer latency. Unfortunately, features such as question views or voting are in majority of cases anonymous and not publicly available for offline experiments, therefore similar experiments can be usually conducted only by the owners of the CQA system.

4.1.5 Related Work Results

Due to diversity of CQA systems, differences in shared content and type of community, results of evaluations cannot be precisely compared. In the following sections, we will try to compare

[20] http://zhidao.baidu.com/

results based on several aspects. The most important analyzed papers in the next sections are side by side compared in the Table 1.

Question representation comparison
As we outlined in the section 4.1.1, topic-based models outperform language models. LDA outperforms language models based on TF-IDF by more than 18% in S@100 metric (Tian et al. 2014). As reported by (Ji & Wang 2013), LDA also outperforms language models based on query likelihood.

User profile comparison
It is possible to compare two users modeling approaches, (Riahi et al. 2012) models only users' expertise, while (Tian et al. 2014) tried to add to user expertise, user interest and activity. Both used LDA to model topics of questions and dataset from StackOverflow, which is one of the most popular experimental dataset in the CQA field. The first work used 123K questions and 1845 users with at least 20 best answers. On the other hand, the second work used 99K questions and considered only active users with at least five questions. Success at 5 is 8.56% for first mentioned approach in comparison with 5.48% for second approach.

Results of work of (Liu et al. 2010) indicates, that taking both user activity and authority into account produced better results when both of them alone. In general, both user authority and user activity are good features for question routing.

Majority of related works are modelling only user expertise in the user profile, e.g. (Riahi et al. 2012). However, the results obtained by (Luo et al. 2014) clearly indicates, that additional features beyond one's expertise, such as willingness and readiness to answer a question, help better predict suitable answerers of a question. Their results outperform baseline that is modelling only user expertise, by 13.8% in coverage rate in top 10 ranked users. (T. C. Zhou et al. 2012) investigated the most important user features of their trained classifier, and they are follows: question-user similarity with user's answered question, member since date and number of best answers the user provided.

Utilization of non-QA for question routing has gaining importance in recent years. As reported by (Srba et al. 2015), question routing performance based on non-QA outperforms question routing based on QA data in MRR and P@N. Different goals have (Luo et al. 2014) as they tried to engage inactive users in question answering process. In this paper, they used non-QA from enterprise environment and obtained promising results in an online experiment of increasing answering rate and asker satisfaction rate. Summing up the results, it can be concluded that non-QA data can be used to route questions to newcomers and lurkers, i.e. users that have low amount of QA-data available.

Table 1: Comparison of selected question routing approaches. (Q – question, U – user, BA – best answer, POS – part-of-speech, BoW – bag-of-words)

Reference	Question routing audience	Approach	Question profile	User profile	Matching model	Evaluation	Ground truth	Evaluation metrics	Dataset	
(Dror et al. 2010)	experts	Classification	Features (textual, category, user, bias)	Features (question-driven, relations, bias)	binary Gradient Boosted Decision Trees	Offline	BA answerer	A, AUC	Yahoo! Answers	1.3M Q
(Luo et al. 2014)	all	Classification	Features (Q type, BoW)	Features (expertise, motivation, availability)	Logistic regression	Offline + Experiment	Actual answerers	P@N	IBM Connect	24K Q
(Chen et al. 2014)	all	Classification	Features (keywords - TF-IDF + POS)	Features (expertise, motivation)	binary Random forest	Offline + Online	Clicks for recommended Q	P, R, A	Baidu Zhidao	4.6M clicks to recommended Q
(Riahi et al. 2012)	experts	Topic model - STM	BoW, LDA, STM	Expertise	Topic similarity	Offline	BA answerer	S@N	Stack Overflow	119K Q
(Tian et al. 2014)	experts	Topic model	BoW, LDA	Expertise (A Quality) + Interest + Activity	QLLM	Offline	BA answerer	S@N	Stack Overflow	99K Q
(Liu et al. 2010)	experts	Topic model - LDA	BoW, LDA	Expertise + Authority + Activity	QLLM	Offline	BA answerer	S@N	Iask	369K Q
(Srba et al. 2015)	all	Topic model - LDA	BoW, LDA	Expertise + Activity	QLLM	Offline	Answerers	MRR, P@N	Stack Overflow	33K Q
(Szpektor et al. 2013)	all	Topic model - LDA	BoW + LDA + Category information	Expertise	Vector similarity (dot-product) + Diversification	Offline + Online (A/B tests)	Answerers, overall community statistics	Activity level	Yahoo! Answers	119K U
(T. C. Zhou et al. 2012)	experts	Classification	Features (textual, Q-U relationship)	Features (activity, expertise, temporal)	binary SVM	Offline	Actual answerers	P, R, A, F1	Yahoo! Answers	1.4M Q

Matching models comparison
As we indicated in section 4.1.1, topic-based models can depict higher overview of the question, therefore they are more suitable for question representation. LDA is used as a state-of-the-art method in the majority of works in the question routing field. The LDA inference is usually based on Gibbs sampling and the number of topics is set empirically. For instance, (Liu et al. 2010) and (Tian et al. 2014) both have 100 topics, (Szpektor et al. 2013) have 200 topics and (Srba et al. 2015) have 20 topics.

Different topic-based model referred as STM by (Riahi et al. 2012) outperforms LDA. STM is based on LDA where the advantage of STM is its suitability for CQA profile structures. This means that instead of grouping all questions under a single topic distribution, it allows each question to have a different and separate distribution of topics. They compared LDA and STM on StackOverflow dataset (containing approximately 124K questions) and STM had on average 30% better results than LDA in S@N (success at N) evaluation metric. Experiment contains also language models, but they have significantly worse results than LDA and STM. However, STM is not used in any other research work in the field of question routing.

Question routing audience
From sustainability point of view to CQA system, routing questions preferably to users with high expertise or high activity is not suitable. We can classify majority of the previous approaches as question routing to the experts. On the other side, we are only aware of three works which have different aim. Their main goal is to engage inactive users in question answering process. These research works are (Luo et al. 2014), (Szpektor et al. 2013) and (Srba et al. 2015). We can refer to these approaches as a question routing to the whole community.

We must clearly differentiate between these two approaches as routing to experts is simpler task than question routing to all users. For example, (Zhou et al. 2009) routed questions only to users with high authority in the topic. Other approaches specified activity or answers constraints, e.g. that use classification (Tian et al. 2014) takes into account only users with number of answers greater than five. In case of (Riahi et al. 2012) it is even more as users with minimum 20 best answers are only considered.

4.2 Question Recommendation in Educational Domain

This section analyses question recommendation approach that is proposed for an educational domain. Question recommendation is not the same task as a question routing. Question recommendation is analogy to product recommendation where the input is a user and the task is to find relevant questions. However, in a question routing the input is a new question and the task is to find most suitable answerers. Moreover, question recommendation recommends any type of questions, mostly resolved ones to all kinds of users. Question recommendation is used to recommend questions beneficial for users and it is used for generating periodic recommendations (e.g. newsletters).

We are aware of only one research paper that studies question recommendation in MOOCs. It is a paper presented by (Yang et al. 2014) who proposed question recommendation specifically designed for discussion forum in MOOCs. Based on the analysis in section 3.2.1, we can see that every popular MOOCs platform is using integrated discussion forum. Discussion forums are related to the CQA systems and many of the concepts used in both systems are interrelated. In the following sections, we will concentrate on the research work by (Yang et al. 2014) in detail.

Authors identified the same issue as we can see in CQA systems: an increasing number of asked questions that makes it difficult to find interesting discussion opportunities. It leads to the problem, that nearly half of the posted questions are never resolved.

They utilize matrix factorization model, typically used as collaborative filtering approach in product recommendation. Uniqueness of their work is addition of specific constraints of MOOCs environment to the recommendation. These constraints include:

- Load balancing which considers students limited work capacity.
- Expertise matching which addresses level of question difficulty for a student.

To address constrained question recommendation problem, the researchers proposed two steps. In the first step, they design a context-aware matrix factorization model to predict students' preferences over questions. By context-aware authors consider student features, question features and implicit feedback. Students features contain answered question count, last week question count and the week in which student registered for the course. Question features are number of question replies and question length represented as total number of words. Implicit feedback represents whether similar users contributes to the question. Consequently, they used proposed features and trained context-aware prediction model for predicting relevance score of a question to the student.

In the second step, the task is to optimize predictions given the constraints. They build a max cost flow model for finding maximum flow in network, where the edges in the network represents constraints. Load balancing constraint represent minimum and maximum amount of questions recommended to a user. Furthermore, each question has specified minimum and maximum limits of participants. Expertise matching is represented as difference between question difficulty and student expertise over all students to which the question will be routed. This function should be minimized and at least one student has larger expertise than question requires. This overall optimization of the network model which maximizes flow function requires set of questions to optimally divide students to answer them. It is a problematic part in terms of real-time use and therefore it is designed for generating periodic recommendations rather than for online recommendation.

The researchers conducted an offline experiment on discussion forums from three courses offered by Coursera platform, where 70% of data were used for training. Their results for recommendation show that taking recommendation context into account is worthwhile. As there is no standard metric for constraint evaluation, they propose three metrics: student coverage, question coverage and overall community benefit. Student and question coverage measure how many questions/students are recommended to a student/question on average. Equation for overall benefit measures how well is the knowledge of the community utilized. In contrast to baseline methods based on top-k selection, their approach has improved overall benefit of the community.

To sum up the work by (Yang et al. 2014), this unique approach is focusing on optimizing community benefit. They try to involve whole community into question answering by effectively utilizing knowledge and time limits of the online student community. However, there are few weak points of this work, such as it is limited for real-time use. Moreover, question difficulty is represented as a count of question words. It is an important feature which is further used for computing the expertise of the student and such representation might be oversimplification.

4.3 Discussion

One of the open problems is to propose collaboration support mechanism for CQA system used in an educational domain. From existing collaboration supports in MOOCs analyzed in section 3.2.2, it can be concluded that collaboration support mechanism is productive for learning and it shows promising results in decreasing dropout rate in MOOCs. Question routing represents a recent type of collaboration support with potential to solve the issues present in MOOC courses.

One of the specifics of the MOOCs environment is the need to evaluate the question routing approach online in order to efficiently measure change in the community interactions. Work by (Szpektor et al. 2013) presents interesting approach for online usage and scalability. However, as indicated in the section 4.1.5, majority of question routing approaches are recommending new questions only to experts. These approaches do not utilize the full potential of the online community if they do not involve for example novice users or lurkers. These approaches are better from asker's perspective to get high-quality answer shortly, but they tend to overwhelm most active or expert users. This can cause a *long tail problem* – large number of popular questions can be routed to just a few experts. In the educational domain, there is little number of experts, as the majority are students concentrating on learning (teachers can be implicitly defined as experts).

(Szpektor et al. 2013) identified the same problem and showed that almost one third the answers on Yahoo! Answers are written by junior users, therefore their method focuses on an engagement of all users to maintain healthy community ecosystem. This might give an assumption that routing questions to whole online community in educational settings is even more essential as it gives students more chances to learn or motivates them to be more active. Moreover, by answering questions they can improve their skills which can later lead to becoming experts.

In addition, existing question routing methods recommend questions to potential answerers within the similar topic based on their expertise. Other useful features for user profile modeling are willingness and activity. The non-QA data represents another promising source of data for the educational domain. To our knowledge, there exists only one paper for question recommendation in the MOOCs domain by (Yang et al. 2014). Therefore, further research into question routing in the educational domain is necessary which is the goal of our work.

5 Conceptual Design of Educational Question Routing Framework

Employing CQA systems in MOOCs is quite a recent research topic. CQA systems in MOOCs environment and education domain in general are different from general CQA systems. Educational CQA systems have less experts, because majority of students are learning about the topic for the first time. Furthermore, it is not expected from students to post perfect solutions to a problem; the goal is to learn by participating in a question answering. Our goal is to design question routing method for CQA systems in the educational settings.

5.1 Goals of Question Routing Framework

Based on the analysis, it can be concluded that educational question routing should be oriented to an answerer and it should involve greater part of the community in the question answering process.

As shown in Table 2, question routing in CQA systems on the open Web are oriented to askers as they aim at answering their questions in the shortest time possible with high quality answers. However, our approach focuses on answerer needs as it considers adequate students' expertise and their willingness to answer the question. The rationale for considering students' expertise is to support majority of students in learning by recommending open questions with reasonable difficulty suitable for them. Some students might have a suitable expertise to answer a question but not all of them are also motivated to answer. Therefore, willingness to answer is explicitly modelled which is derived from students' activity in the course and CQA system.

To involve majority of the community in question answering, recommendation of new questions should not overload students with many questions. It is necessary to balance recommended questions by students' working capacity and to involve more students without any QA activity in the recommendation. For this task, we are considering non-QA data from a MOOC course which are not present in the work by (Yang et al. 2014).

Our approach is using most of the QA features as (Yang et al. 2014) and we are including several more. The difference is that their approach is a question recommendation, which is recommendation of any type of question while we are using question routing. (Yang et al. 2014) estimate question difficulty as a length of a question text. Our approach considers information about the asker of the question and utilize knowledge gap phenomenon observed by (Lin et al. 2014). From observations, they implied a pattern where the question asked by expert have a high probability of being difficult. Therefore, non-experts do not have the needed expertise to answer the question as it is beyond their knowledge. On the other hand, easier questions are naturally asked by low-expert users and these questions are not very challenging for experts. This leads to lower motivation by experts to answer the question, so low-experts more often answer this type of question.

Hypothesis 1
Considering context of an educational domain in question routing, i.e. students' level of expertise, their willingness to answer and their answering capacity, increases the accuracy of answerers prediction.

Hypothesis 2
Educational question routing engages greater part of the community into question answering.

Table 2: Comparison of different recommendation approaches.

	Question routing in CQA systems used on the open Web	**Question recommendation in MOOCs** (Yang et al. 2014)	**Our proposed educational question routing**
High-level overview	• Asker-oriented	• Optimizing overall forum welfare • Involvement of whole community	• Answerer-oriented • Involvement of whole community
Features	• QA data • Non-QA data is present in a minority of papers	• QA data	• QA data • *Non-QA data from a MOOC course*
Answerer selection	• Maximizing expertise of answerers	• Suitable knowledge of answerers • Working capacity of students	• Suitable knowledge of answerers • *Willingness to answer* • Working capacity of students
Goals/Metrics	• Accuracy of answerers prediction • Question answering success rate • High quality answers in a short time	• Accuracy of answerers prediction • Question answering success rate • Involvement of greater part of the community	• Accuracy of answerers prediction • Question answering success rate • Involvement of greater part of the community

5.2 Educational Question Routing Framework

Figure 5-1 presents the overview of the question routing framework which routes new question to the most appropriate answerers. The input for the question routing framework is a new question and users' profiles which are extracted real-time and updated from the activity in a CQA system and MOOC course. The output for a new question is a list of recommended answerers sorted by their ranking of how likely they will answer the question. The framework is divided into four phases and first three phases are based on the analysis in section 4.1 while the last phase is added to fulfill the requirements of an educational domain:

- *Construction of a question profile.* When a new question is posted, the question textual content and asker information are processed.
- *Construction of a user profile.* Data from CQA system and MOOC course are extracted for modelling the user expertise and willingness to answer.
- *Matching of questions and users.* Compute ranking for each user as a probability of answering a new question.
- *Optimization.* Re-ranking and filtering of users by constraints.

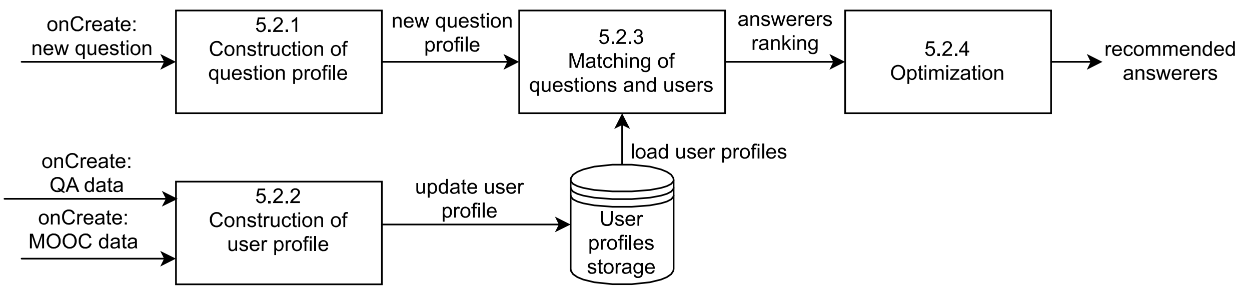


Figure 5-1: Educational question routing framework.

Question routing method is designed for a learning environment. Therefore, it is required to work real-time and route new questions in short time after the question is posted. As MOOC courses are short-term and intensive, the design needs to be scalable and adaptable to changes, i.e. considering new data in CQA system and MOOCs course throughout the period of the course.

5.2.1 Construction of a Question Profile

As shown in the analysis of existing MOOC platforms in sections 3.2.1 and 3.4.3, we consider following available textual information about a new question: title, body, hierarchy of categories and information about an asker. Question title and body are concatenated and preprocessed by tokenization, stop words removal and stemming. After preprocessing the question profile θ_q is created as a bag-of-words model. Latent topics are also inferred in this step. These two models are typically used in a question routing field as shown in the section 4.1.1. The answer profile $\theta_{a,q}$ is created in a similar way without the concatenation step because answers do not have a title. Hierarchy of categories and asker information are used in the matching of questions and users phase.

5.2.2 Construction of a User Profile

User profile depicts information about:

- topics of questions which users previously answered which is referred as a user text profile,
- qualitative, quantitative and temporal features extracted from previous user activities in MOOC course and CQA system.

As the base of our approach for user text profile modelling, we are going to use similar approach as proposed by (Szpektor et al. 2013) which is designed for online usage with respect to scalability. As mentioned in the previous section, question text profile is inferred from newly posted question immediately. A user text profile is then represented as an aggregation of answers and questions text profiles, to questions which the user provided an answer. When user answers another question, user's text profile θ_u is updated as a sum of an answer and question profile of question that user answered represented as a bag-of-words, leading to richer user profile with each additional answer:

$$\theta_u = \sum_{q \in Q_u} (\theta_q + \theta_{a,q}) \qquad (14)$$

where Q_u is a set of all questions which was answered by a user u.

Another QA features that measure user's expertise includes number of answers, comments and votes within each week and topic category. Besides QA data, we also use data from the MOOC course. It includes knowledge prerequisites as portion of seen lectures for each week of a course and student's assignment grades. Our rationale is that student who have already seen lectures for a given topic of new question or have good grades are more likely to have the suitable expertise.

To model user willingness to answer a question, we consider user activity in both CQA system and MOOC course. Activity in CQA includes total number of submitted answers, questions, comments and earned votes. To model latest activity as it can vary over period of the course, time related metrics such as last answer time and time of watching the lecture are important. Registration date for the course also influences the commitment as shown by (Yang et al. 2014).

We decided to use these QA related features for question routing:

- *Total answers count.* Total number of answers by a user.
- *Total comments count.* Total number of comments posted by a user.
- *Total questions count.* Total number of questions asked by a user.
- *Total votes earned.* Earned votes for all answers and questions the user posted.
- *Answers count in the recent period.* Number of answers in past few days.
- *Last answer time.* Computed as a difference between new question posting time t_q and t_u which is the most recent time the user posted an answer to a question. The difference is converted to number of seconds.

$$LastAnswerTime = t_q - t_u \tag{15}$$

- *Average CQA activity.* Ratio of days, that user was active in the CQA system, i.e. voted or posted a question, comment or answer, to total number of days the course has been running.
- *Seen questions within a category.* Ratio of questions in a category, that user has seen, to the total number of questions within a category where new question belongs.
- *Question-user text profile similarity.* Cosine similarity of vectors representing new question text profile and a user text profile. Text profiles vectors can be represented by bag-of-words model or LDA model.
- *Answers count within a category.* Number of user's answers in a category where new question belongs.
- *Earned votes count within a category.* Number of votes for user's answers in a category where new question belongs.
- *Total knowledge gap.* Knowledge gap is defined as a difference in knowledge of a potential answerer and asker. Knowledge is estimated as a sum of answers, votes and comments counts.

$$Knowledge(user) = Answers(user) + Votes(user) + Comments(user) \tag{16}$$

$$KnowledgeGap(answerer, asker) = Knowledge(answerer) - Knowledge(asker) \tag{17}$$

- *Knowledge gap within a category.* Same as equation (17), except the knowledge is estimated only within a category where new question belongs.
- *Average between CQA session activity.* Activities in CQA system are sorted for a user in an ascending order as an array $activities$. The difference is computed as number of days between two date types.

$$AvgBetweenActivity = \frac{\sum_{i=1}^{|activities|-1}[\ activities(i+1) - activities(\ i)\ ]}{|activities|} \quad (18)$$

Features extracted from the MOOC course (non-QA) are following:

- *Portion of seen lectures within a category.* Ratio of lectures in a category, that user has interacted with, to total number of lectures within a category where new question belongs.
- *Lecture freshness.* Computed as a difference between question posting time t_q and a time user has seen the related lecture for a topic of the question.
- *Average course activity.* Computed as a portion of days, where user was active in the MOOC course system, i.e. when user clicks on any lecture, to number of days the course is running.
- *Course registration date.* Computed as a difference between question posting time t_q and a registration date of a user. Same computation as in equation (15).
- *Average grade.* Grade is computed as an average of homework grades and lab grades.
- *Average between course session activity.* Same computation as in equation (18), but for the activities in the course.

Typical structure of educational course is that each week of the course consisting of several topics. Therefore, we utilize this structure and split each feature related to category into week and topic categories.

5.2.3 Matching of Questions and Users

We designed classification-based approach of matching questions and users. The QLLM was used as a base approach and it is mentioned in this section to represent our way of thinking.

QLLM
QLLM, which is analyzed in the section 4.1.3, can use as a language model either LDA or TF-IDF model. As a prior probability of user $P(u)$ all features from previous section can be used, an example is shown in the equation (19). However, the weights representing significance of the features w_i in prior probability can be set only empirically. In other words, this algorithm is not capable of adapting the weights of features in prior probability. Therefore, it is better to learn weights of features in a prior probability as a linear classification problem. However, best solution is to learn weights not only for the features in prior probability $P(u)$, but also for a question-user text profile similarity $P(q|u)$. It represents a classification problem and it is described in the next section.

$$P(u) = w_1\ AvgActivity(u) + \ w_2 KnowledgeGap(u, asker) \quad (19)$$

Classification
To computation of the ranking for each user given a new question is defined as a classification task. Using the profile of a new question and profiles of all potential answerers, we address the question routing as an ensemble of two explicit classification tasks:

1) Predicting whether user has sufficient expertise to answer a new question.
2) Predicting user's willingness to answer a new question.

The rationale for splitting the classification into two subtasks is to explicitly use both information in the last stage. Moreover, by using all features together in one classifier it is not possible to control which features are most significant for the classifier. In that case, classifier could learn to use only expertise features and the result could be asker-oriented approach which recommends only to experts. Another positive aspect is the possibility to create more positive and negative examples for each individual classifier than for one global classifier. In the Table 3 one can see how the positive and negative classes are generated. It is in the finer level of detail in comparison to only one global classifier, where positive examples would be only answers to a question.

Table 3: Definition of positive and negative classes for expertise and willingness classifiers.

	Positive class (y=1)	***Negative class (y=0)***
Expertise classifier	▪ Answer which gets positive votes difference. ▪ Answer which is marked as best answer. ▪ Answer with positive evaluation from teaching assistant.	▪ Answer which get negative votes difference ▪ Answer with zero votes and another answer was added later ▪ Answer with negative evaluation from teaching assistant
Willingness classifier	▪ Answer ▪ Comment	▪ Question view without interaction, i.e. no vote for question or answer, no answer

The design of two classifiers allow us to create the ensemble of these classifiers by custom integration of their predictions. It follows the idea discussed in section 4.1.3, when more diverse classifiers are stronger in prediction than one classifier. The final ensemble probability ranking is computed from individual classifiers predictions probabilities as the probability of both events occurring simultaneously:

$$P(y = 1) = P(expertise = 1) * P(willingness = 1) \qquad (20)$$

where $P(expertise = 1)$ is probability of expertise classifier prediction belongs to the positive class, $P(willingness = 1)$ is a probability of willingness classifier prediction belongs to the positive class. This final probability is assigned for each user and it is used to rank potential answerers for question routing.

For an online use the classifier should be able to learn online or it could be re-trained in a reasonable time. Furthermore, the classifier is required to predict the probability of sample belonging to a specific class. In general, it is possible to use any binary classification algorithm. However, based on the analyses in the section 4.1.3 and our requirements, following three classification algorithms achieved promising results in previous related works:

- SVM
- Random forest
- Logistic regression

Input features are divided into willingness and expertise features used by respective classifiers as shown the Table 4. Features are extracted either from the CQA system or from MOOC course (non-QA data).

Table 4: Expertise and willingness features divided into subgroups by their origin and type.

<table>
<tr><th></th><th></th><th>Educational</th><th>Non-educational</th></tr>
<tr><td rowspan="2">Expertise (11 features)</td><td>CQA</td><td>• total knowledge gap
• knowledge gap within a week category
• knowledge gap within a topic category</td><td>• question-user text profile similarity
• answers count within a week category
• answers count within a topic category
• earned votes count within a week category
• earned votes count within a topic category</td></tr>
<tr><td>non-QA (MOOC)</td><td>• average grade
• portion of seen lectures within a week category
• portion of seen lectures within a topic category</td><td></td></tr>
<tr><td rowspan="2">Willingness (16 features)</td><td>CQA</td><td>• portion of seen questions within a week category
• portion of seen questions within a topic category</td><td>• overall answers count
• overall comments count
• overall questions count
• answers count in the recent period
• last answer time
• average CQA activity</td></tr>
<tr><td>non-QA (MOOC)</td><td>• average MOOC activity
• lecture freshness
• portion of seen lectures within a week category
• portion of seen lectures within a topic category</td><td>• course registration date</td></tr>
</table>

5.2.4 Optimization

In the last step of question routing framework, the constraints are applied to the list of recommended answerers similar to (Yang et al. 2014) and (Luo et al. 2014). The goal of the optimization is to optimally utilize the knowledge of an online student community and to balance new questions to the members of the community. The constraint is maximum student workload, which is estimated as a number of question routed to the student in the recent time.

Teaching assistants in the course have a special role. It could be supposed that teaching assistants are implicitly experts in the course content. Therefore, teaching assistants can be considered in matching model normally. On the other hand, new question can be routed in this step to teaching assistants in case of all students ranking is below a threshold.

6 Implementation of Educational Question Routing Method

We implemented our proposed solution in the open-sourced Askalot CQA system deployed at the EdX platform. At first we explored what data are available in Askalot database structure. Based on that, we extracted and derived features from the raw data. Consequently, we defined steps required to process text of questions, comments and answers. Ensemble classifier is defined as a question-user matching algorithm. This ensemble classifier consists of two individual classifiers – the classifier predicting user's expertise and the classifier predicting willingness to answer a new question by a user. Finally, question routing parameters and forms of question routing are discussed.

6.1 Askalot CQA System

Open-source CQA system for university domain Askalot[21], described in section 3.4.2, is being actively used to support learning in Slovak University of Technology for a few years. After the success of Askalot at the home university, it has started to being used at the other universities as well. Furthermore, the creators of the Askalot port it to the EdX platform, so since Autumn of 2016 it has been used in one MOOCs course.

Askalot contains experimental infrastructure described in (Srba & Bielikova 2016). In this work the event dispatcher part of the experimental infrastructure is used. For offline evaluation, it allows us to reproduce events consequently by time they had happened. To implement our approach, we extended Askalot by defining listeners that are listening for multiple events, e.g. posting a question or answer, voting. With this pattern, experimental infrastructure allows us to use the same implementation for offline and online evaluation.

6.2 Available Data

Available data persisted by Askalot include: answers, questions, comments, votes, clicks on lectures and question views. All resources have user identifier associated with them.

EdX platform offers a grades report of the students. The grade report consists of homework and lab grades within each week of a course. Moreover, the grade report contains information about the participation in quizzes throughout the video lectures.

6.3 Software Technologies

Askalot is developed in the Ruby on Rails[22] web framework. We used this framework to implement modules responsible for showing the recommendations to the users. To implement the listeners responsible for listening for new events and updating the features in the database, we used Ruby[23] programming language. Askalot CQA system uses PostgreSQL[24] as a database system, which was used to persist and load features for each user which are necessary for the matching of questions and users.

[21] https://github.com/AskalotCQA/askalot
[22] http://rubyonrails.org/
[23] https://www.ruby-lang.org/
[24] https://www.postgresql.org/

For text processing and classification, the Python[25] programming language was used. The reason is that Python has libraries for text processing and machine learning which are high-quality, well documented and scalable. However, by choosing the Python language the communication between different programming languages becomes more complex.

Another positive aspect of using Python programming language is the reproducibility of the research by using Jupyter Notebook[26]. We used the Jupyter Notebook for visualizations and evaluation of the different classifiers.

The implementation was developed in 64-bit version of Ubuntu 16.04. Following libraries were used to implement the question routing method:

- Gensim[27] – Building words vocabulary and bag-of-words models, retrieving similar user profiles for a new question profile.
- NLTK[28] - Text processing by Snowball stemmer and removing of stop words.
- Scikit-learn[29] – Machine learning library for classification, hyper-parameter tuning, data normalization, feature selection and validation.
- Numpy[30] – Support for mathematical functions and efficient matrices representation.
- Psycopg2[31] –PostgreSQL database adapter for Python programming language.
- Imbalanced-learn[32] - Sampling techniques for preprocessing the data examples for classification.

6.4 Question Profile Construction

When new question or answer is created, vocabulary of words is updated. Each word in a vocabulary has its id and counter of occurrences. The vocabulary is always persisted to disk.

Question text profile is built by concatenating question title and text. It is further tokenized, stop words are removed and it is preprocessed by Snowball stemmer[33]. In the next step, each word is mapped to id and TF-IDF is computed for each word. Final textual profile of question as TF-IDF bag-of-words model is used in the matching model and it is also saved to the database to prevent later re-computation when next answer will be added to the question.

6.5 User Profile Construction

User profile features are updated in real-time on creation of: answer, comment, question, lecture view, question view and user registration. Three feature are updated once a day: average CQA activity, average course activity and recent answers count are computed once a day. Time-related features are converted to seconds. Answer count in the recent period is set to last 7 days.

[25] https://www.python.org/
[26] http://jupyter.org/
[27] https://radimrehurek.com/gensim/
[28] http://www.nltk.org/
[29] http://scikit-learn.org/stable/
[30] http://www.numpy.org/
[31] https://pypi.python.org/pypi/psycopg2
[32] http://contrib.scikit-learn.org/imbalanced-learn/
[33] http://www.nltk.org/_modules/nltk/stem/snowball.html

6.6 Question-User Matching

For each day, new data within a day are appended to the dataset according to rules defined in the Table 3. In the next step, both expertise and willingness classifiers are re-trained with steps defined in the Figure 6-1.

Features are scaled by z-score normalization for classification algorithm. For evaluation, we are using k-fold stratified cross validation. Probability threshold for predicting the class is found dynamically by maximizing AUC (Area Under the Receiver Operating Curve) metric.

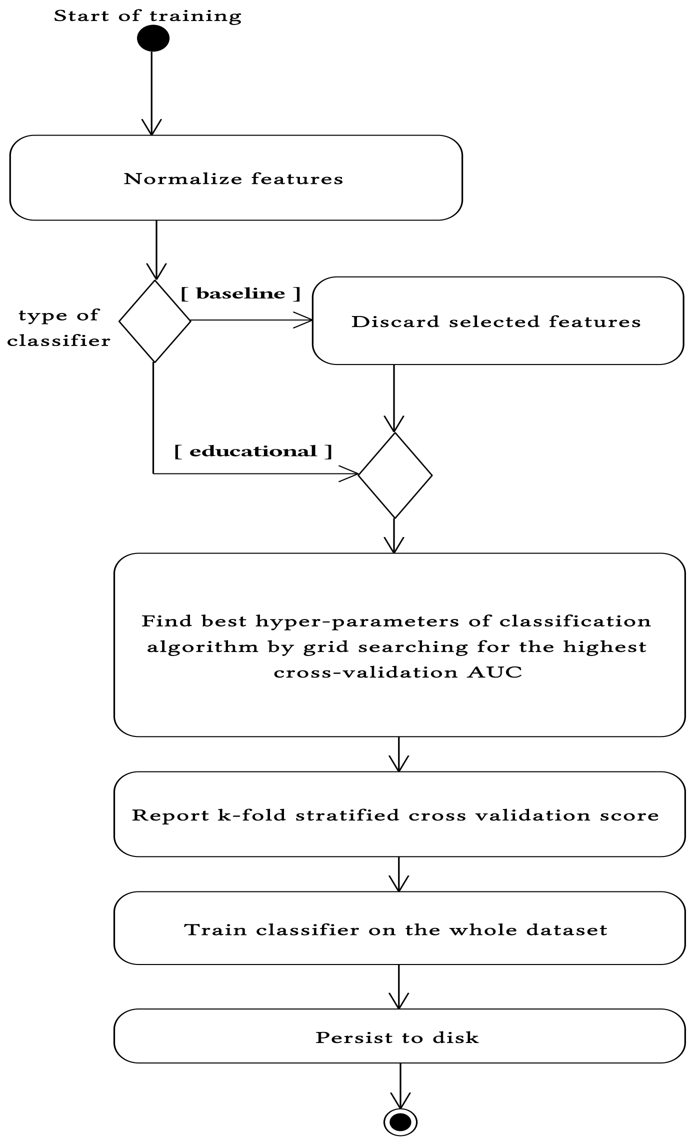


Figure 6-1: Activity diagram depicting training of expertise and willingness classifiers.

Hyper-parameters of classification algorithms are found by searching their best combination (highest cross-validation AUC score) from selected values or range. Hyper-parameters in the Table 5 are optimized in selected classification algorithms and they are used to prevent overfitting (analyzed in the section 4.1.3 for each classification algorithm). Classification is implemented in Python programming language and uses scikit-learn[34] machine learning library.

[34] http://scikit-learn.org

Table 5: Optimized hyper-parameters for classification algorithms.

Classification algorithm	**Hyper-parameters**
SVM	kernel function (sigmoid, linear function, radial basis function), penalty parameter
Random forest	number of trees, splitting criterion (Gini impurity, entropy), maximum tree depth, number of features considered for the split
Logistic regression	loss function, regularization term (L1, L2), number of iterations

To deal with unbalanced data problem, we assign the weight for each example inversely proportional to their classes frequencies. We experimented with random under-sampling and SMOTE (Synthetic Minority Over-Sampling Technique), but they did not overperform class weighting.

6.7 Forms of Recommendation

New questions are recommended by Askalot notification system and recent recommendations are listed on the Askalot dashboard as shown in Figure 6-2 and Figure 6-3. By using two elements for recommendation we are increasing the probability that user will see the routed question. Moreover, recommended questions are highlighted in the list of all questions.

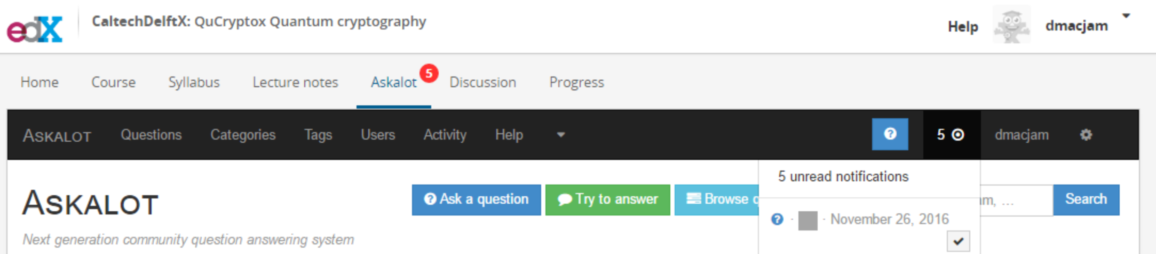


Figure 6-2: Recommendation is delivered as a notification. Number of unread notifications is shown in the Askalot and in the EdX menu.

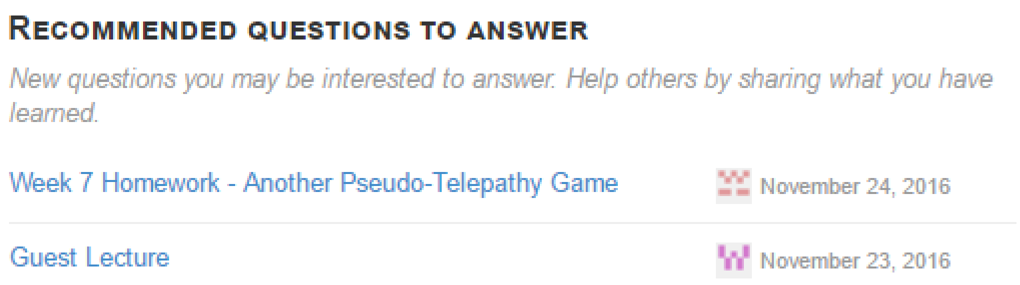


Figure 6-3: Example of recommended questions which are shown in the bottom left corner of the main Askalot page.

7 Evaluation of the Proposed Educational Question Routing Method

This section presents results of our approach in comparison with a baseline method by offline and online experiment conducted on the MOOC course at the EdX platform. The goal of both experiments is to evaluate the performance of our educational question routing method. Moreover, online experiment helped us to examine the real impact of educational question routing method to students' community. Source code for this section is accessible online[35].

7.1 Quantum Cryptography MOOC Course

Evaluation of our question routing approach is done in Askalot CQA system ported to the EdX platform. The MOOC course used for experiments is *QuCryptox Quantum cryptography*[36] offered by California Institute of Technology and Delft University of Technology.

The course is about quantum cryptography and requires advanced knowledge of linear algebra and probability. The course lasted 10 weeks from 10th October to 20th December 2016. Estimated workload for the course is 6 to 8 hours per week. Each week contains several video lectures which are usually followed by a quiz. Each video lecture within a week covers specific topic. Illustration of the course structure is shown in Figure 7-1. Furthermore, each week is pen and paper assignment and coding assignment.

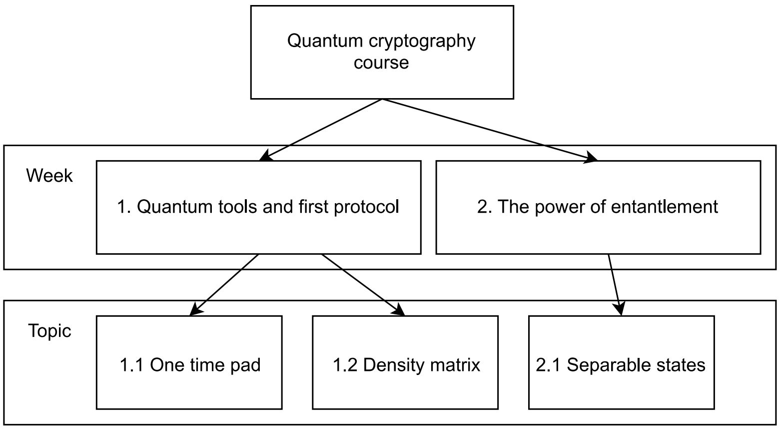


Figure 7-1: Sample of Quantum cryptography course structure.

The course content and CQA system was available for students before and after the official start and the end of the course. Therefore, we considered data two weeks before (from 26th September 2016) and two weeks after (2nd January 2017) the course. Summary statistics from this period is shown in the Table 6.

[35] https://github.com/dmacjam/dp-analysis-evaluation
[36] https://courses.edx.org/courses/course-v1:CaltechDelftX+QuCryptox+3T2016

Table 6: Summary statistics of QuCryptox Quantum cryptography course.

Metric	Quantity
Students enrolled in the course	8115
Students started the course	4618
Users participating in CQA (with any question view)	1098 (24%)
Users contributing in CQA	377 (8%)
Questions	281
Questions with answer	247 (88%)
Questions with best answer selected	51 (18%)
Answers	333
Comments	453
Teachers evaluations of answers	27

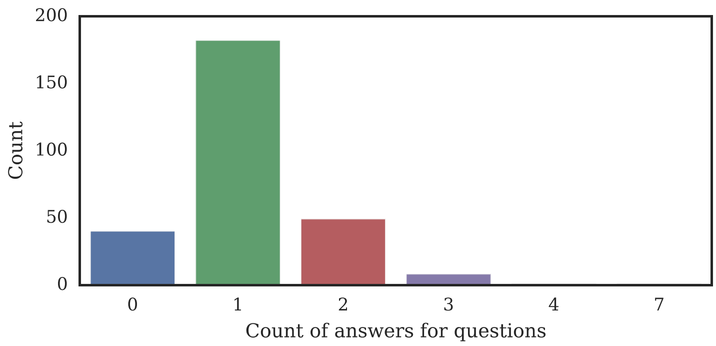


Figure 7-2: Distribution of answers frequencies for questions.

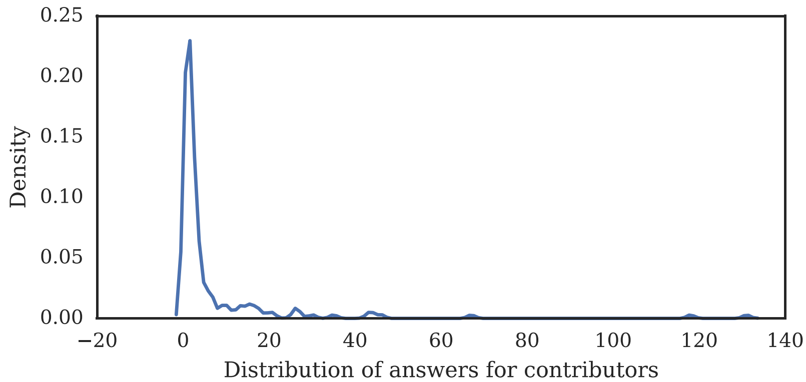


Figure 7-3: Distribution of answers or comments frequencies for users.

7.2 Baseline Question Routing Method

To the best of our knowledge, there is no other question routing method for the educational domain for direct comparison. Therefore, baseline question routing method is a variant of our proposed question routing method which does not consider educational features shown in left column of the Table 4. The selected baseline question routing method can be described as asker-oriented approach which is widely-used approach in the CQA systems on the open Web.

Moreover, most of the features in the baseline approach are used in question recommendation work by (Yang et al. 2014).

7.3 Offline Experiment

In the offline experiment, we filtered out redundant features and selected classification algorithms for expertise and willingness predictions. Moreover, we fine-tuned parameters of question routing framework for online experiment.

7.3.1 Experiment Setup

In the offline experiment, we consider data from the beginning of the course until 4th December, which covers eight weeks of the course length. As a first step, we performed feature selection of all features proposed in the section 5.2.2 by pairwise correlation of features and feature significance by ANOVA test.

In the next step, we generated positive and negative samples from the data as defined in the Table 3. These data samples are used for training of three classification algorithms for both expertise and willingness classification tasks and the best classification algorithm is selected by k-fold stratified cross validation.

Finally, the educational question routing and baseline question routing methods are compared by ground truth, i.e. users who answered the question. The offline approach is evaluated without the optimization step. By using Askalot experimental infrastructure we are simulating the events consequently as they happened. New question is recommended to users by both methods and these recommendations are evaluated in comparison with ground truth. Metrics used for offline evaluation (defined in the section 4.1.4) are: *Success (S@N)*, *Precision (P@N)*, *Mean Average Precision (MAP@N)*, *Normalized Discounted Cumulated Gain (nDCG@N)* and *Mean Reciprocal Rank (MRR)*.

7.3.2 Feature Selection

In this step, the most predictive subset of features is selected to prevent the curse of dimensionality problem. At first, we applied, correlation matrix between features for both expertise and willingness features. As shown in Figure 7-4, knowledge gap within topic is correlating with answers in a topic because answers in a topic is part of the knowledge gap computation shown in the equation (16). Another significant correlated features are answerer knowledge and knowledge gap, which is caused by the same problem. Therefore, we decided to remove asker knowledge and answerer knowledge to reduce the dimensionality of the problem because they are both captured in the knowledge gap feature.

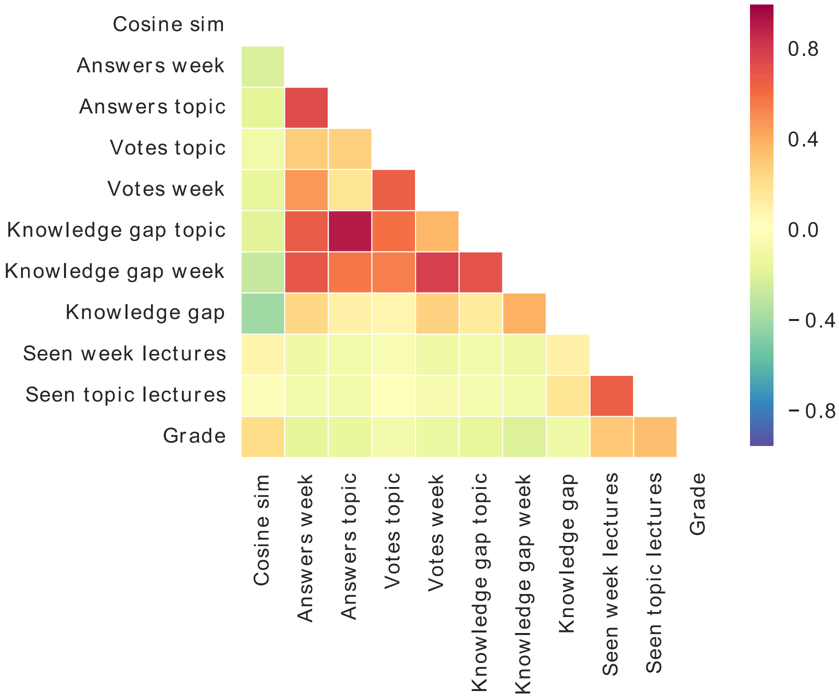


Figure 7-4: Correlation matrix for expertise features.

Correlation of willingness features is shown in Figure 7-5. Significant positive correlation is between answers count, comments count and votes count. However, this correlation is rational and can be explained with a fact, that more the users are contributing by answering or commenting, the more they are likely to get votes. Other significant correlations, e.g. seen topic questions and seen week questions seem natural.

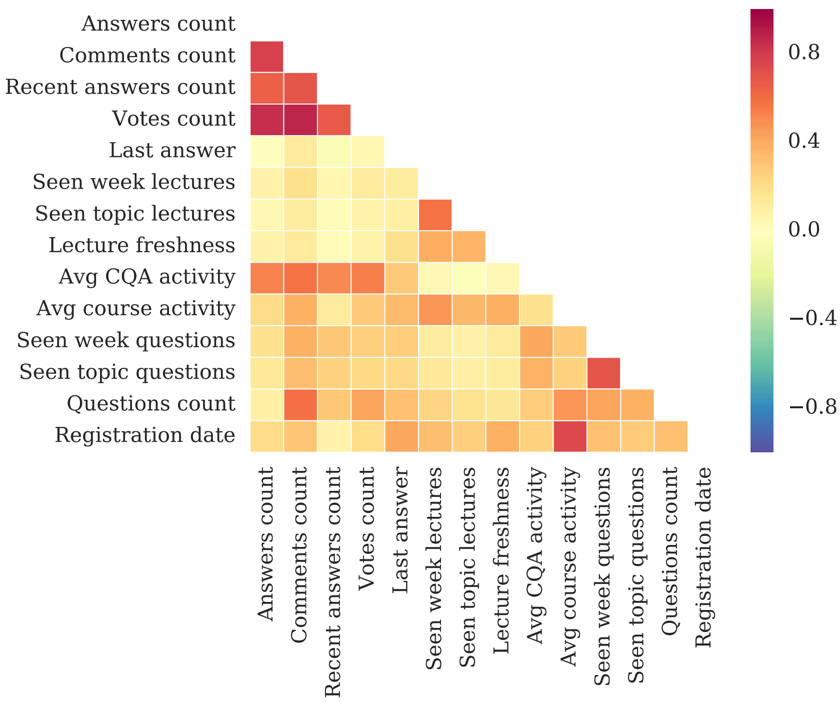


Figure 7-5: Correlation matrix for willingness features.

Secondly, we tried to find correlation between input features and target class by ANOVA statistical test. We found that only the cosine similarity has a significant impact ($F=4.60$, $p<0.05$) on the expertise predictions. For willingness features, majority of features are significant. The most significant features are: votes count ($F=603$, $p<0.01$), recent answers count ($F=579$, $p<0.01$), answers count ($F=509$, $p<0.01$), comments count ($F=491$, $p<0.01$) and seen questions within a topic ($F=221$, $p<0.01$). Furthermore, feature importance in models (see Figure 7-7 and Figure 7-8), forward selection or backward elimination could be used for feature selection.

7.3.3 Selection of a Classification Algorithm

We considering three classifiers described in the section 4.1.3:

- SVM
- Random forest
- Logistic regression with stochastic gradient descent (SGD) learning

They are trained on the dataset of positive and negative samples which is summarized in the Table 7.

Table 7: Quantities of generated data samples.

Data	Positive class (y=1)	Negative class (y=0)
Expertise dataset	134	50
Willingness dataset	891	13 475

As one can see in Figure 7-6, the features are overlapping and there is lack of discriminative power between these conditions. Therefore, we suppose that the decision boundary for the question routing problem is non-linear. Therefore, logistic regression might not be suitable for the problem. On the other hand, logistic regression is simpler model than other two, therefore by following Occam's razor principle it is less prone to overfitting.

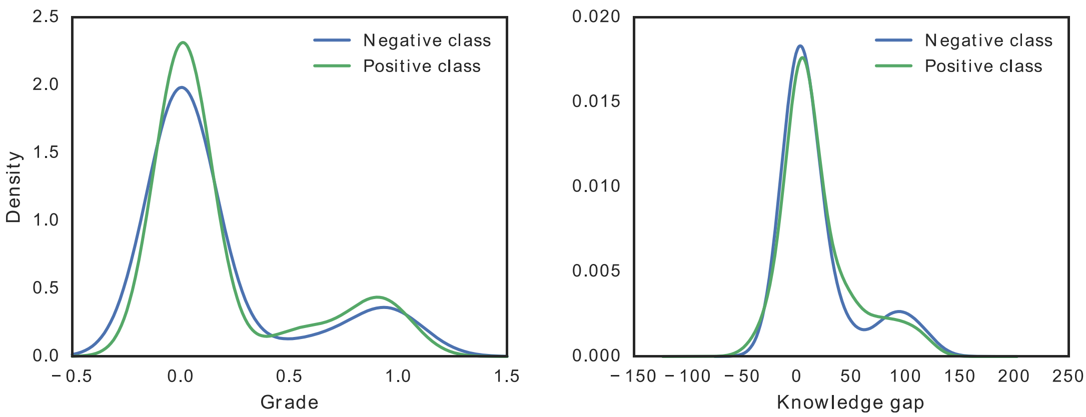


Figure 7-6: Density comparison of chosen expertise features.

Table 8: Classification algorithm comparison for expertise features based on 6-fold stratified cross validation.

Metric	SVM	Random forest	Logistic regression
AUC	0.60 (+/- 0.08)	**0.67 (+/- 0.06)**	0.66 (+/- 0.08)
F1	0.67 (+/- 0.06)	0.69 (+/- 0.18)	**0.70 (+/- 0.04)**

Table 9: Classification algorithm comparison for willingness features based on 10-fold stratified cross validation.

Metric	SVM	Random forest	Logistic regression
AUC	0.69 (+/- 0.06)	**0.73 (+/- 0.06)**	0.72 (+/- 0.05)
F1	0.72 (+/- 0.10)	**0.76 (+/- 0.08)**	0.75 (+/- 0.08)

The SVM training was very slow comparing to other two approaches (tens of minutes compared to tens of seconds) and the performance is the worst for both cases. Logistic regression and random forest results are comparable. To choose the final classifier, it is about the trade-off between interpretability of logistic regression and non-linearity of decision boundary in case of random forest. We decided to use random forest classifier as the final solution for both classification problems.

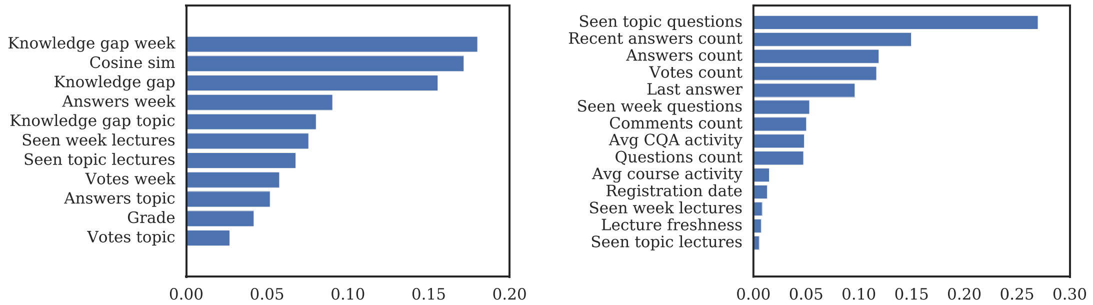


Figure 7-7: Features significance for random forest expertise (left) and willingness (right) classifiers.

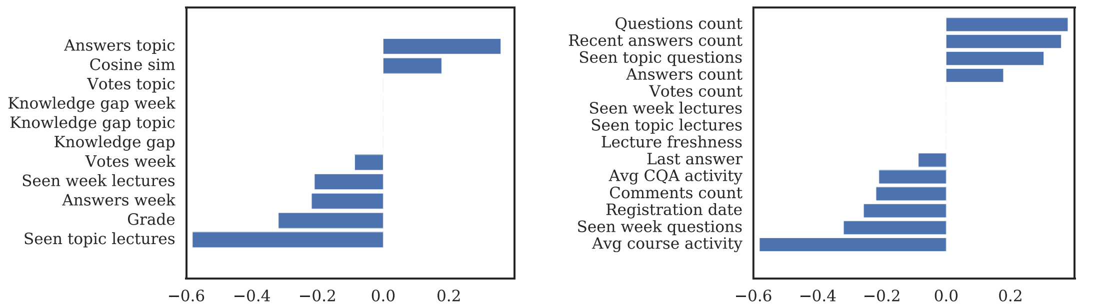


Figure 7-8: Features significance for logistic regression expertise (left) and willingness (right) classifiers.

7.3.4 Question Routing Results

Classifier for expertise classification and willingness classification in both cases is random forest (maximum depth = 4, number of trees = 100, split criterion = Gini impurity). For final classification, the probabilities of positive class for each classifier are combined by multiplication as shown in the equation (20).

Table 10: Results for educational and baseline question routing approaches on selected metrics.

Metric	Educational			Baseline		
	N=5	*N=10*	*N=100*	*N=5*	*N=10*	*N=100*
S@N	0.418	0.601	0.924	0.369	0.548	0.894
P@N	0.378	0.533	0.884	0.323	0.489	0.858
MAP@N	0.221	0.242	0.260	0.193	0.216	0.234
NDCG@N	0.288	0.345	0.413	0.254	0.312	0.384
MRR	0.284			0.259		

As shown in the Table 10 and in the Figure 7-9, our approach outperformed the baseline approach in all metrics. Thus, we can conclude that features specific to learning environments help in predictions of new question answerers. As an example, if we route question to 10 most suitable answerers, we would hit any true answerer in 60.1% compared to 54.8% of baseline method.

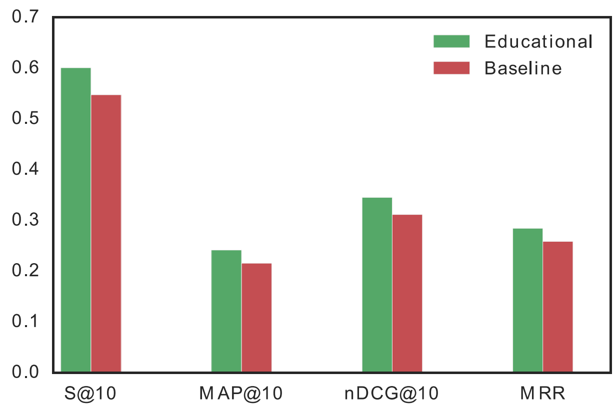


Figure 7-9: Educational and baseline question routing performance on selected metrics.

7.4 Online Experiment

As we pointed out in the section 4.1.4, research works in question routing are evaluated in majority of cases by offline experiments, while online experiments are conducted very rarely. One of the biggest limitation of offline experiment is that we would not know how users would behave when they get the recommendation. Offline evaluations can only consider users who answered a question as a positive example. However, they do not consider cases when a user does not choose to answer a question because it is already answered by a high-quality answer. Moreover, information about question view or votes are usually not available in the datasets for our domain.

Online experiment addresses these limitations by supplementing the offline evaluation with online evaluation of our method measuring performance and total impact on the student community.

7.4.1 Experiment Setup

We conducted an online experiment by A/B testing in the *QuCryptox Quantum cryptography* started from 14^{th} November 2016 (week 6 of the course). Start of the online experiment split the evaluated data in a half, where both periods before and during online experiment contains 7 weeks of the course (as we take into account two weeks before/after the start/end of the course). At the beginning of week 6 of the course, all users in MOOC course were randomized into three groups of n users. Randomized assignment was stratified by user's answer counts to reduce variability of users. Result of randomization is shown on the left chart in the Figure 7-12. Students who signed up for the course during online experiment were not considered. The three user groups are:

- *Educational/Edu group (n = 1306).* Users in this group had questions routed by our proposed educational question routing method.
- *Baseline group (n = 1306).* Users in the baseline group had questions routed by the baseline method.
- *Control group (n = 1306).* Users in the control group did not have any question routing and thus did not receive any recommendation.

Each new question is routed to 10 users in educational group and to 10 users in the baseline group. As an optimization step considering student workload, student could get maximum 4 recommendations per 7 days.

We were collecting explicit feedback throughout the online experiment, i.e. clicks on recommendation and source of the click (dashboard or notification). In addition to implicit feedback, the explicit feedback was collected by a questionnaire which is shown in the Figure 7-10. Users could express whether they are able to answer a question and whether they have willingness to answer a question. The questionnaire is suitable to use in case when user clicked on the recommendation and the question is already answered with a reasonable answer or when user has not enough expertise or willingness to answer a question.

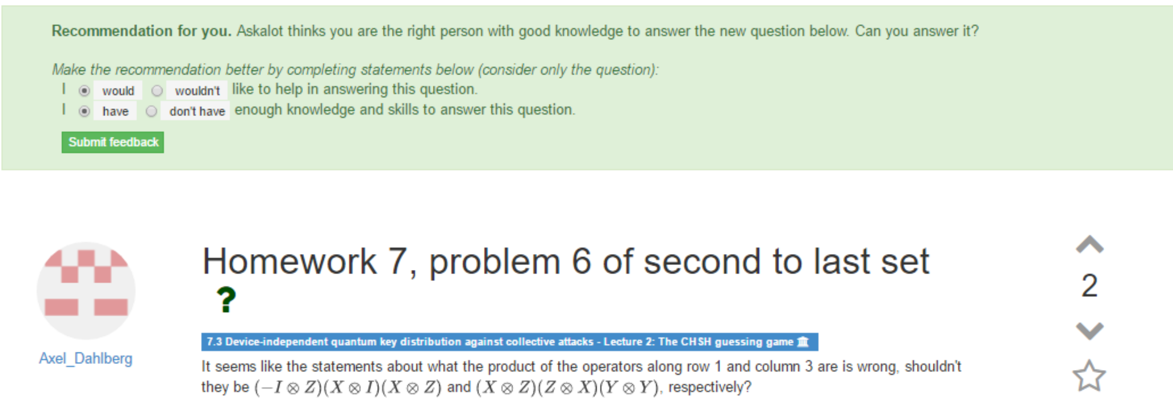


Figure 7-10: Question routing feedback questionnaire which shows above the recommended question. User can check the right words in two sentences which describes whether they have suitable expertise and willingness to answer the question.

7.4.2 Metrics

Question routing methods are evaluated in online experiment by following metrics:

- *Click-through rate (CTR)* – Proportion of clicked question recommendations out of total number of question recommendations.
- *Question routing S@N* – Proportion of routed questions for which one or more predicted answerers (out of N predicted answerers) contributed.
- *User's coverage rate* – Proportion of users who received a new question recommendation out of total number of users in an experimental group.
- *Time to answer* – Time in hours for a newly posted question to receive an answer.
- *Answer quality* – Difference between positive and negative votes for an answer.
- *Dropout rate* – Proportion of users who dropout from the course out of all users in an experimental group.

7.4.3 Results

During online experiment, 132 new questions were routed to potential answerers resulting in total 2640 recommendations.

As our goal is to decrease the burden on answerers, we evaluated CTR which measures relevance of recommendation. The CTR was 23.25% for users in the educational group, but only 18.29% for the baseline group which was significantly different as shown in the Table 11. This difference in CTR by 4.96% means that our method for educational question routing increased the chances of users clicking through by 27%.

A second metric for question routing quality is the proportion of routed questions answered or commented by their recipients, or the question routing S@10. The question routing S@10 was 15.91% for the educational group, but only 10.61% for the baseline group. While this difference did not reach statistical significance as shown in the Table 11, this is likely a reflection of the small sample size of routed questions providing insufficient statistical power to detect differences, as the difference of 5.30% represents an increase of 50% beyond baseline group.

Based on the comparison of results for CTR and question routing S@10 we deduce that additional accuracy were achieved by incorporating MOOC data that improved predictions of expertise and willingness to answer a question.

Table 11: Accuracy of question routing.

Metric	**Educational**	**Baseline**	**Statistical significance**
CTR	23.25%	18.29%	$\chi^2(1, N = 2640) = 10.03, p < 0.01$
S@10	15.91%	10.61%	$\chi^2(1, N = 264) = 1.61, p = 0.20$

Our second goal of our is to involve more students into question answering. Educational group has highest proportion of participating users as shown in the Table 12 which is by 3.81% and 4.44% more as baseline and control group during online experiment. However, it did not reach statistical significance, $\chi^2(2, N = 903) = 3.77, p = 0.15$. On the other hand, educational question routing method increased significantly ($\chi^2(1, N = 1120) = 8.27, p < 0.01$) proportion of involved students out of all active MOOC users in a group compared to period before online experiment. However, this finding is not clear as there might be other relevant factors which influences contributions to the CQA system, e.g. during second half of the course mostly motivated students stayed in the course compared to the first half.

As we could not afford to overload students in the online experiment by many requests to answer new questions, we applied optimization step with workload restriction also on the baseline method. Therefore, we cannot accurately compare question routing method with and without optimization step in terms of user's coverage rate. The user's coverage rate was 10.72% for educational group and 10.03% for the baseline group.

Table 12: Proportion of contributing users to CQA system out of all users active in a MOOC for each group.

Period	**Educational**	**Baseline**	**Control**
Before experiment	62 (7.60%)	73 (8.99%)	74 (9.12%)
During experiment	40 (13.16%)	26 (9.35%)	28 (8.72%)

In addition to direct impact of question routing, introducing these methods can also increase overall activity in the CQA system. We report data for each of the groups both before and during the online experiment. The active part of students in MOOC course and CQA system in both periods is shown in the Figure 7-11.

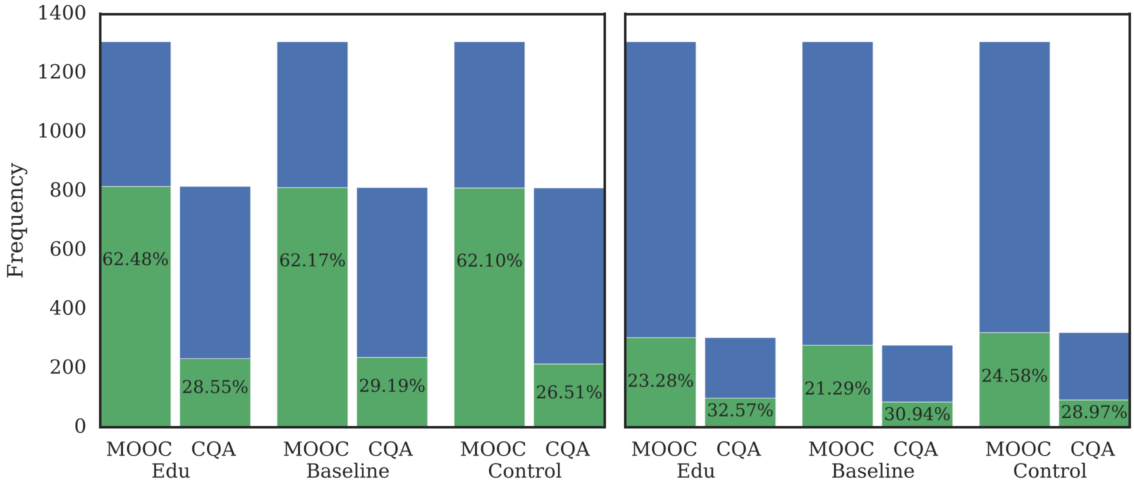


Figure 7-11: Number of students in MOOC course and CQA system for each group before (left) and during (right) the online experiment. For MOOC course, the green part represents active users in a MOOC course out of all users in a group during a period. For the CQA system, the green part represents portion of students using CQA system out of all active students in the MOOC course during a period.

CQA usage behavior is shown in the Figure 7-12, in terms of posting questions, answers, comments, and viewing and voting on posts. The introduction of question routing appears to have led to greater activity in the use of the CQA system in both the educational and baseline group, relative to the control during the online experiment. Only average count of answers is higher in control group in comparison to the baseline group. This finding is quite surprising as it is possible to see that baseline group started to ask more questions and posting more comments than answering questions. However, in comparison by sum of answers and comments, which are considered as a contribution to CQA system, baseline group outperformed control group.

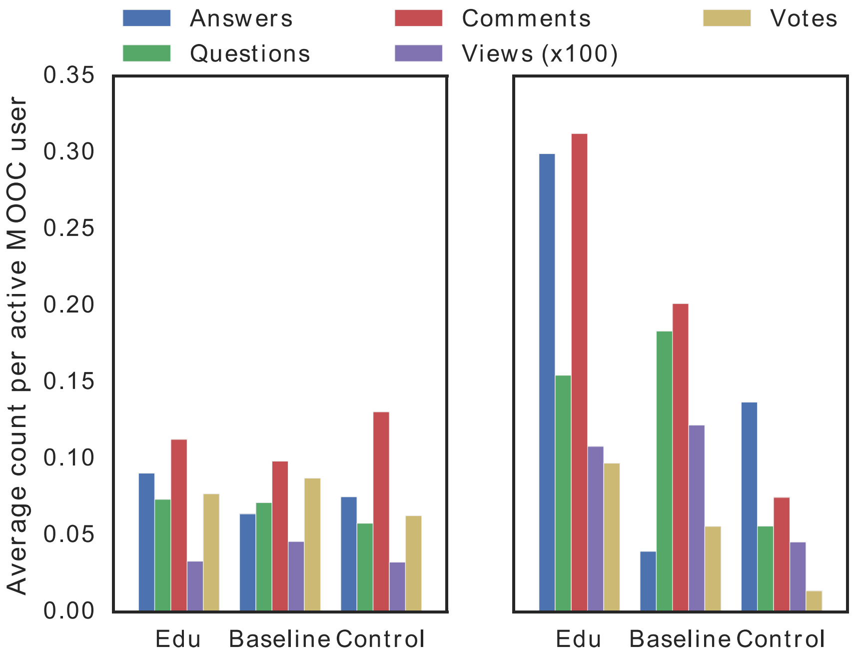


Figure 7-12: Average amount of activity per active MOOC student in a group before (left) and during (right) the online experiment. Student is considered active in MOOC if he/she had any course interaction in the period before or during the online experiment.

To further analyze the impact of involving more students in question answering, we verified whether it did not have a negative influence on quality of answers and time to the first answer for a question. The results of statistical tests comparing educational group and control group showed that the question routing preserves answer quality and time to first answer for a question.

Course instructors spent a significant amount of time by collaboration and discussions with students. The high instructors' involvement is reflected in the proportion of contributions by instructors and teaching assistants. It was 37.28% before online experiment and 31.25% during online experiment. The drop of 6.03% indicates decreased instructors' workload with question routing. So far, significance of this finding is not clear as the drop of instructor's load could be influenced by other variables.

We evaluated dropout rate for each user group in both MOOC course and CQA system. We found a difference in dropout rate of contributors from the CQA system. While in the control group, 81.08% of users who contributed before online experiment stopped contributing during online experiment, it was 79.45% in the baseline group and 64.50% in the educational group. This results indicates a positive influence of question routing on keeping users motivated and devoted to question answering.

Feedback questionnaire was added to the CQA system in later stage of the online experiment and we received only few feedback results. The feedback shown in the Table 13 it is possible to see that students were mostly willing to answer but they lack expertise. However, as it contains only 13 samples, we cannot conclude anything from the feedback.

Table 13: Feedback statistics. Character + means positive feedback, character – means negative feedback.

Expertise	Willingness	Quantity	
		Educational	Baseline
+	+	1	1
+	-	0	1
-	+	4	5
-	-	1	0

8 Conclusions

This work proposed a new question routing approach for MOOCs. Based on the analysis, we proposed two innovations making our question routing method suitable for educational domain. At first, the question routing approach is answerer-oriented rather than oriented to askers. It models user's willingness to answer the question and combines it with the expertise of a user. Secondly, we utilized non-QA data from MOOC course such as students' grades, activity in the course and knowledge prerequisites to successfully answer a question.

In the proposed educational question routing framework, the task of finding the answers for a new question is split into two subtasks, predicting user's expertise for a question and user's willingness to answer a question. Such design helps us to create more accurate data samples and it allows us to easily combine these two predictions with even more constraints which needs to be considered for a user or within a whole community. Moreover, constraint on student work load is applied to decrease the information overload of a student and to balance new question in the online student community.

Further research in identifying type of a question is needed. Currently, we tackle all types of questions and users equally. However, questions in MOOCs are also about organization of the course and these questions can be answered only by instructors. If we are able to identify them, we can route them only to course instructors. Another promising future direction is better optimization of students' knowledge for computing knowledge gap. More research is still necessary to compare TF-IDF bag-of-words model with topic modelling such as LDA. Finally, applying question routing in MOOC courses with thousands of students brings scalability issues which need to be addressed.

The proposed question routing approach was evaluated by an offline experiment, to fine tune the models and evaluate accuracy of recommendation, and the online experiment, which is very seldom in this domain, to measure a total impact of question routing on the online student community. Online experiment was conducted on a MOOC course about quantum cryptography with more than 4600 students at the EdX platform.

Summing up the results based on the comparison with the baseline, it can be concluded that the proposed educational question routing framework achieved higher accuracy of potential answerers predictions. This resulted in a higher interest of students in the routed question and engaged more students in contribution to the CQA system. The existence of these effects led to increase of average contributions and activities per active MOOC student.

Literature

1. Alario-Hoyos, C. et al., 2014. Delving into Participants' Profiles and Use of Social Tools in MOOCs. *IEEE Transactions on Learning Technologies*, 7(3), pp.260–266.
2. Aritajati, C. & Narayanan, N.H., 2013. Facilitating Students ' Collaboration and Learning in a Question and Answer System. *Proceedings of the 2013 conference on Computer supported cooperative work companion*, pp.101–105.
3. Blei, D.M., Ng, A.Y. & Jordan, M.I., 2003. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3(1), pp.993–1022.
4. Breslow, L. et al., 2013. Studying learning in the worldwide classroom: Research into edX's first MOOC. *Research & Practice in Assessment*, 8(March 2012), pp.13–25.
5. Cai, L. et al., 2011. Learning the Latent Topics for Question Retrieval in Community QA. *Ijcnlp*, pp.273–281.
6. Cao, X. et al., 2010. A Generalized Framework of Exploring Category Information for Question Retrieval in Community Question Answer Archives. *the International World Wide Web Conference 2010*, (December 2005), pp.201–210.
7. Dror, G. et al., 2010. I Want to Answer, Who Has a Question? Yahoo! Answers Recommender System. *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp.1109–1117.
8. Ekstrand, M.D., 2011. Collaborative Filtering Recommender Systems. *Foundations and Trends® in Human–Computer Interaction*, 4(2), pp.81–173.
9. Ferschke, O. et al., 2015. Positive impact of collaborative chat participation in an edX MOOC. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. pp. 115–124.
10. Grunewald, F. et al., 2013. OpenHPI - A case-study on the emergence of two learning communities. *IEEE Global Engineering Education Conference, EDUCON*, pp.1323–1331.
11. Guo, J. et al., 2008. Tapping on the potential of q&a community by recommending answer providers. *Proceeding of the 17th ACM conference on Information and knowledge mining - CIKM '08*, pp.921–930.
12. Huna, A., Srba, I. & Bielikova, M., 2016. Exploiting Content Quality and Question Difficulty in CQA Reputation Systems. In A. Wierzbicki et al., eds. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Lecture Notes in Computer Science. Cham: Springer International Publishing, pp. 68–81.
13. Chen, T. et al., 2014. Instant Expert Hunting : Building an Answerer Recommender System for a Large Scale Q & A Website. *Sac*, pp.260–265.
14. James, G. et al., 2014. *An Introduction to Statistical Learning: With Applications in R*, Springer Publishing Company, Incorporated.
15. Ji, Z. et al., 2012. Question-answer topic model for question retrieval in community question answering. *Cikm*, p.2471.
16. Ji, Z. & Wang, B., 2013. Learning to rank for question routing in community question answering. *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management - CIKM '13*, (August), pp.2363–2368.
17. Jordan, K., 2014. Initial trends in enrolment and completion of massive open online courses Massive Open Online Courses. *International Review of Research in Open and Distance Learning*, 15(1), pp.133–160.
18. Jurczyk, P. & Agichtein, E., 2007. Discovering authorities in question answer communities by using link analysis. *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management - CIKM '07*, pp.919–922.
19. Klusener, M. & Fortenbacher, A., 2015. Predicting students' success based on forum activities in MOOCs. *2015 IEEE 8th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, 2(September), pp.925–928.
20. Lin, C., Chen, Y.-L. & Kao, H., 2014. Question difficulty evaluation by knowledge gap analysis in Question Answer communities. *2014 IEEE/ACM International Conference on*

Advances in Social Networks Analysis and Mining (ASONAM 2014), (Asonam), pp.336–339.
21. Liu, M., Liu, Y. & Yang, Q., 2010. Predicting Best Answerers for New Questions in Community Question Answering. In L. Chen et al., eds. *Proceedings of the 11th international conference on Web-age information management - WAIM '10*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 127–138.
22. Liu, Q., Agichtein, E. & Dror, G., 2012. When web search fails, searchers become askers: understanding the transition. *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pp.801–810.
23. Liu, X., Croft, W.B. & Koll, M., 2005. Finding Experts in Community-based Question-answering Services. *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, pp.315–316.
24. Liu, Y., Bian, J. & Agichtein, E., 2008. Predicting information seeker satisfaction in community question answering. *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR '08*, (Section 2), p.483.
25. Luo, L. et al., 2014. Who Have Got Answers? Growing the Pool of Answerers in a Smart Enterprise Social QA System. *Proceedings of the 19th international conference on Intelligent User Interfaces*, pp.7–16.
26. Onah, D.F.., Sinclair, J. & Boyatt, 2014. DROPOUT RATES OF MASSIVE OPEN ONLINE COURSES : BEHAVIOURAL PATTERNS MOOC Dropout and Completion : Existing Evaluations. *Proceedings of the 6th International Conference on Education and New Learning Technologies (EDULEARN14)*, pp.1–10.
27. Ram, A. et al., 2011. Open Social Learning Communities. In *Proceedings of the International Conference on Web Intelligence, Mining and Semantics - WIMS'11*. New York, New York, USA: ACM Press.
28. Riahi, F. et al., 2012. Finding Expert Users in Community Question Answering. In *Proceedings of the 21st international conference companion on World Wide Web - WWW '12 Companion*. New York, New York, USA: ACM Press, pp. 791–798.
29. Rosmalen, P. Van, Brouns, F. & Sloep, P., 2007. *Question-answering through selecting and connecting peer-students*,
30. Shah, C., Kitzie, V. & Choi, E., 2014. Modalities, motivations, and materials - investigating traditional and social online Q&A services. *Journal of Information Science*, 40(5), pp.669–687.
31. Srba, I., 2015. Askalot : Community Question Answer- ing as a Means for Knowledge Sharing in an Educational Organization. *Cscw*, pp.179–182.
32. Srba, I. & Bielikova, M., 2016. Design of CQA systems for flexible and scalable deployment and evaluation. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. pp. 439–447.
33. Srba, I. & Bieliková, M., 2016a. A Comprehensive Survey and Classification of Approaches for Community Question Answering. *ACM Transactions on the Web*, p.Accepted.
34. Srba, I. & Bieliková, M., 2016b. Why Stack Overflow Fails? Preservation of Sustainability in Community Question Answering. *IEEE Software*, p.Preprint.
35. Srba, I., Grznar, M. & Bielikova, M., 2015. Utilizing Non-QA Data to Improve Questions Routing for Users with Low QA Activity in CQA. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015 - ASONAM '15*. New York, New York, USA: ACM Press, pp. 129–136.
36. Szpektor, I., Maarek, Y. & Pelleg, D., 2013. When Relevance is not Enough : Promoting Diversity and Freshness in Personalized Question Recommendation. *Proceedings of the 22nd international conference on World Wide Web*, pp.1445–1456.
37. Tian, Y. et al., 2014. Predicting Best Answerers for New Questions: An Approach Leveraging Topic Modeling and Collaborative Voting. In *Proceedings of Social Informatics 2013 International Workshops - SocInfo '13*. Springer Berlin Heidelberg, pp. 55–68.

38. Xu, F., Ji, Z. & Wang, B., 2012. Dual role model for question recommendation in community question answering. *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval - SIGIR '12*, pp.771–779.
39. Yang, D., Adamson, D. & Rosé, C.P., 2014. Question recommendation with constraints for massive open online courses. *Proceedings of the 8th ACM Conference on Recommender systems - RecSys '14*, pp.49–56.
40. Zhou, G., Liu, K. & Zhao, J., 2012. Joint relevance and answer quality learning for question routing in community QA. *Proceedings of the 21st ACM international conference on Information and knowledge management - CIKM '12*, pp.1492–1496.
41. Zhou, T.C., Lyu, M.R. & King, I., 2012. A classification-based approach to question routing in community question answering. *Proceedings of the 21st international conference companion on World Wide Web WWW 12 Companion*, p.783.
42. Zhou, Y. et al., 2009. Routing questions to the right users in online communities. *Proceedings - International Conference on Data Engineering*, pp.700–711.

Resumé in Slovak Language

1. Úvod a motivácia

Rastúca popularita masívnych otvorených online kurzov (MOOCs) umožnila komukoľvek s internetovým pripojením prístup k množstvu zdrojov vzdelávania. Týmto spôsobom vznikajú mnohopočetné a rôznorodé online komunity študentov.

Jedným z najväčších problémov MOOCs kurzov je vysoké percento študentov, ktorý kurz nedokončia. Podľa (Onah et al. 2014) dokončí kurz len 13% študentov. Na výsledky študenta v kurze pozitívne vplýva aj aktivita v diskusnom nástroji (Klusener & Fortenbacher 2015). Množstvom študentov v kurze však vzniká veľké množstvo nových otázok a to vedie do stavu spôsobujúceho informačné preťaženie študentov. To znamená že študenti majú problém si nájsť zaujímavú otázku na odpovedanie alebo diskutovanie a taktiež nie je v silách inštruktorov odpovedať na všetky otázky. Podľa (Yang et al. 2014) až polovica otázok v diskusných fórach MOOCs kurzov zostáva nezodpovedaných.

MOOC platformy ako napríklad EdX[37] alebo Coursera[38] ponúkajú zabudované diskusné fóra. Niektoré kurzy však používajú aj iné diskusné nástroje ako sociálne siete, online chat alebo systémy pre odpovedanie na otázky v komunitách (CQA). CQA systémy, ako napríklad StackOverflow[39], Quora[40] a Yahoo Answers![41] sú rozšírené na otvorenom Webe, vo firemnom prostredí (Luo et al. 2014) a začínajú sa používať aj v MOOC kurzoch[42]. CQA systémy poskytujú alternatívu k tradičným diskusným fóram a poskytujú väčšie možnosti kolaborácie (hlasovanie, gamifikácia, výber najlepšej odpovede) a sú viac orientované na komunitu (profily používateľov, sledovanie).

Problémom, ktorý vplýva na veľké množstvo nezodpovedaných otázok je malá časť študentov, ktorí aktívne prispievajú v diskusnom nástroji. Podľa štúdie (Breslow et al. 2013) na jednom z prvotných kurzov na EdX platforme, len 3% z celkovo 155 000 študentov participovalo v diskusnom fóre. Ostatná časť komunity konzumuje vytváraný obsah a aktívne neprispievajú pri odpovedaní na otázky.

Otvoreným problémom v MOOCs doméne je nedostatok vzdelávacej podpory zo strany učiteľov a aj členov online študentskej komunity. Našim cieľom vyriešiť spomenuté problémy návrhom novej metódy pre odporúčanie nových otázok vhodným študentom na odpovedanie.

2. Súčasné riešenia

V doméne CQA systémov na otvorenom Webe sa používajú dva hlavné prístupy na podporu kolaborácie:

- vyhľadávanie otázok v archívoch CQA systému (angl. question retrieval),
- smerovanie/odporúčanie nových otázok najvhodnejším potenciálnym odpovedajúcim (angl. question routing).

[37] https://www.edx.org/
[38] https://www.coursera.org/
[39] http://stackoverflow.com/
[40] https://www.quora.com/
[41] https://answers.yahoo.com/
[42] https://cs50.stackexchange.com/

V našej práci sa zameriavame na smerovanie nových otázok, pretože tento prístup má väčší potenciál vo vzdelávacom prostredí s pohľadu podpory kolaborácie študentov. Odporúčanie nových otázok naviac umožňuje zapojiť väčšiu časť komunity do odpovedania na otázky a tak im pomôcť vo vzdelávaní.

Odporúčanie nových otázok je problém, kedy pre novú otázku hľadáme najvhodnejších používateľov na jej odpovedanie. Zvyčajne sa úloha smerovania otázok skladá z troch fáz:

1. vytvorenie profilu otázky, kde najpoužívanejšími metódami je model tém Latent Dirichlet Allocation (LDA) (Blei et al. 2003) a model vrece slov vypočítané pomocou TF-IDF (Chen et al. 2014),
2. vytvorenie profilu používateľa, ktorý sa využíva na modelovanie znalostí (Szpektor et al. 2013), aktivity (Tian et al. 2014), motivácie (Luo et al. 2014) alebo správneho času pre odpovedanie (Chen et al. 2014),
3. hľadanie relevantných používateľov k novej otázke, kde výstupom je zvyčajne usporiadaný list používateľov zoradených podľa pravdepodobnosti alebo iného spôsobu zoradenia pre odpovedanie na danú otázku.

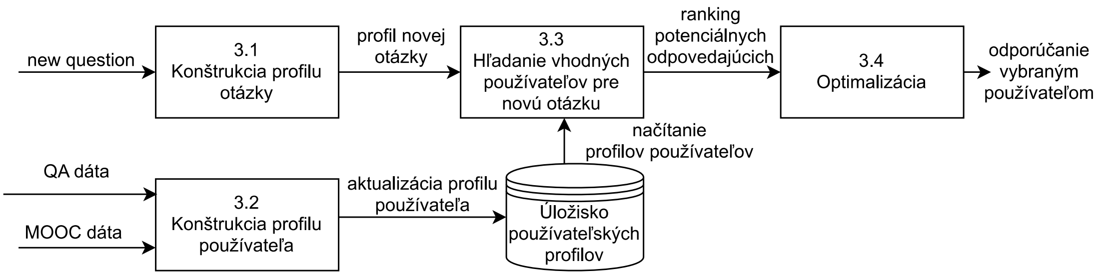
Väčšina existujúcich prístupov smerovania nových otázok na otvorenom Webe je založená na uspokojení potrieb pýtajúceho sa a preto sú tieto otázky odporúčanie len nízkemu počtu používateľov s vysokou úrovňou znalostí. Tento prístup čo však v doméne vzdelávania nie je vhodný. Sme si vedomý len troch prác, ktoré mali odlišný cieľ a tým je zapojenie aj neaktívnych používateľov do prispievania v diskusnom nástroji. Práce (Luo et al. 2014) a (Srba et al. 2015) na túto úlohu využili mapovanie na iné zdroje dát pri modelovaní používateľa a práca (Szpektor et al. 2013) sa zameriava na diverzifikáciu otázok a s tým spojené predchádzanie uzavretia používateľa do odporúčacej bubliny.

V doméne MOOCs sme identifikovali len jeden relevantný článok. Ide o odporúčanie otázok (angl. question recommendation) v diskusnom fóre v rámci MOOCs kurzov (Yang et al. 2014). V prvom rade je nutné spomenúť, že sa jedná o odlišný typ úlohy ako smerovanie nových otázok, pretože pri odporúčaní otázok sa odporúčajú akékoľvek otázky, aj vyriešené. Pri úlohe odporúčania otázok je naviac vymenené poradie vstupov a výstupov ako pri smerovaní otázok, teda vstupom je používateľ a na výstupe je zoznam otázok, ktoré sú pre neho relevantné. (Yang et al. 2014) navrhla odporúčanie ako kolaboratívne odporúčanie s kontextom. Následne aplikuje obmedzenia, ktorými sú obmedzený čas prispievania do diskusie a vhodná náročnosť otázky pre odpovedajúceho, a ich cieľom je optimalizácia celkovej spokojnosti komunity. Tieto obmedzenie sú aplikované ako optimalizácia naprieč celou komunitou a preto nie je takéto odporúčanie možné použiť v reálnom čase, ale skôr na generovanie pravidelných letákov.

V doméne vzdelávania je malý počet študentov s výbornými znalosťami o danej téme kurzu, pretože väčšina študentov sa učí o tejto téme a preto existujúce prístupy smerovania otázok tu nie sú vhodné. Cieľom v doméne vzdelávania je zvýšiť množstvo naučených vedomostí študentov prispievaním v diskusnom nástroji. V našej práci navrhujeme metódu smerovanie nových otázok v doméne MOOC kurzov. Našim prínosom je využitie dát z MOOC kurzu pre presnejšie modelovanie študentov a explicitné modelovanie ochoty študenta odpovedať na danú novú otázku.

3. Rámec pre smerovanie nových otázok vo vzdelávacom prostredí

Obrázok 1 zobrazuje schému rámca pre smerovanie nových otázok. Cieľom smerovanie tohto rámca je zníženie zaťaženia študentov vhodnými odporúčaniami, zapojenie väčšej časti komunity do odpovedania a zvýšiť aktivitu študentov v kurze. Vstupom pre rámec je nová otázka a výstupom je zoznam potenciálnych odpovedajúcich zoradených podľa ich pravdepodobnosti odpovedať. Na základe aktivity používateľov v CQA systéme a MOOC kurze sú v reálnom čase upravované črty potrebné v kroku hľadania odpovedajúcich pre novú otázku. Rámec je rozdelený na 4 časti, ktoré sú opísané podrobnejšie v nasledujúcich častiach textu.



Obrázok 1: Schéma rámca pre smerovanie nových otázok vo vzdelávacom prostredí.

3.1. Konštrukcia profilu otázky

Profil otázky je vytvorený ihneď po jej pridaný do systému používateľom. Profil otázky zachytáva obsah otázky (na základe názvu a textu otázky), polohu v hierarchii kategórií a informácie o pýtajúcom sa. Názov a text otázky sú spojené a predspracované tokenizáciou, odstránením stop slov a extrakciou koreňu slova. Profil otázky je reprezentovaný pomocou modelu vrece slov s TF-IDF váhami. Na vytvorenie profilu odpovede sa používa rovnaký postup. Informácie o pýtajúcom sa a kategórii sa ďalej používajú v 3. kroku.

3.2. Konštrukcia profilu používateľa

Relevantný odpovedajúci by mal mať vhodné znalosti odpovedať a aj ochotu odpovedať na novú otázku. Preto profil používateľa zachytáva tieto charakteristiky a obsahuje informácie o predchádzajúcich príspevkoch používateľa (textový profil používateľa) a tiež aj kvantitatívnych, kvalitatívnych a časových čŕt vytvorených na základe predchádzajúcich aktivít používateľa v CQA systéme a MOOC kurze. Jednotlivé črty sa priebežne v reálnom čase prepočítavajú a ukladajú oddelene pre každý týždeň a tému kurzu. Týmto spôsobom nasledujeme štruktúru kurzu, ktorá sa skladá z jednotlivých týždňov kurzu a v rámci každého týždňa kurzu je niekoľko lekcií.

Textový profil používateľa je vytvorený ako suma jeho profilov odpovedí a príslušných profilov otázok, na ktoré používateľ odpovedal. Profil používateľa ďalej modeluje aktivitu používateľa v CQA systéme, ako napríklad počet pridaných otázok, odpovedí alebo komentárov, a tiež v MOOC kurze, ako napríklad časť videných lekcií. Naviac profil používateľa zachytáva aj kvalitu jeho aktivity pomocou získaných hlasov v CQA systéme a známkami v kurze. Taktiež sa modelujú aj predpoklady používateľa odpovedať na novú otázku, teda či študent prešiel relevantné časti kurzu týkajúce sa novej otázky. Keďže sa aktivita používateľa môže meniť v čase, modelujeme aj črty týkajúce sa času, ako napríklad čas poslednej odpovede, počet odpovedí za nedávnu dobu a čas registrácie do kurzu.

3.3 Hľadanie vhodných používateľov pre novú otázku

Zoradenie používateľov podľa relevancie odpovedať na novú otázku je navrhnuté ako klasifikačná úloha. Vstupom do klasifikačného algoritmu je nová otázka a profily používateľov. Klasifikačná úloha je navrhnutá ako súbor dvoch klasifikačných úloh:

1. Predikcia dostatočných znalostí používateľa odpovedať, kde je vstupom 11 čŕt z ktorých je 6 špecifických pre vzdelávanie.
2. Predikcia ochoty používateľa odpovedať na novú otázku, kde je vstupom 14 čŕt z ktorých je 6 špecifických pre vzdelávanie.

Logickým dôvodom na rozdelenie klasifikačnej úlohy na dve podčasti je explicitné použitie týchto výsledkov v ďalšej fáze. Naviac, týmto spôsobom vieme kontrolovať zastúpenie aj používateľov s vysokými znalosťami a aj používateľov s ochotou odpovedať na novú otázku. V prípade jedného globálneho klasifikátora by to mohlo skĺznuť do naučenia sa predikovať len používateľov s vysokými znalosťami a to nie je našim cieľom. Naviac, návrhom súboru klasifikátorov vieme presnejšie vytvárať pozitívne a negatívne príklady na trénovanie oboch klasifikátorov.

Finálne zoradenie používateľov je zoradené podľa pravdepodobnosti, ktorá je vypočítaná ako pravdepodobnosť, že používateľ má aj znalosti odpovedať ale aj ochotu odpovedať na novú otázku:

$$P(y = 1) = P(znalosti = 1) * P(ochota = 1)$$

kde $P(znalosti = 1)$ je pravdepodobnosť, že používateľ ma znalosti odpovedať (výsledok predikcie prvého klasifikátora patrí do pozitívnej triedy) a $P(ochota = 1)$ je pravdepodobnosť, že používateľ má ochotu odpovedať (výsledok predikcie druhého klasifikátora patrí do negatívnej triedy).

3.4 Optimalizácia

V poslednom kroku sú aplikované obmedzenia danej domény podobne ako v prácach (Yang et al. 2014) a (Luo et al. 2014) . Obmedzením je to, že nemôžeme presiahnuť aktuálnu pracovnú kapacitu používateľa, ktorá je odhadovaná ako počet odporúčaných otázok v nedávnej dobe. Cieľom tejto fázy je zapojenie väčšej časti komunity do odpovedania na otázky.

4. Experimentálne overenie

Naše riešenie sme overili pomocou offline a aj online experimentu. Cieľom offline experimentu bolo natrénovať a vyladiť parametre navrhnutého rámca. Pri online experimente sme skúmali reálny dopad smerovania otázok na komunitu. Naše riešenie sme porovnali so základnou metódou smerovania otázok, ktorá neobsahuje črty špecifické pre vzdelávanie, teda jedná sa o smerovanie otázok používané v CQA systémoch na otvorenom Webe.

4.1. CQA systém a MOOC kurz

Rámec pre smerovanie nových otázok vo vzdelávacom prostredí je overený v CQA systéme Askalot[43] použitý na MOOCs platforme EdX[44]. MOOC kurz na ktorom prebehlo overenie je QuCryptox Quantum Cryptography[45] ponúkaný univerzitami California Institute of Technology a Delft University of Technology. Kurz obsahoval základy kvantovej kryptografie a vyžadoval pokročilé znalosti algebry a pravdepodobnosti. Trvanie kurzu bolo od 10. októbra 2016 do 20. decembra 2016. Keďže kurz bol k dispozícií pár týždňov pred a po začiatku a konci kurzu, analyzované dáta sú z obdobia dvoch týždňov pred a po kurze, teda od 26. septembra 2016 do 2. januára 2017. Tabuľka 1 zobrazuje sumárnu štatistiku tohto kurzu.

[43] https://github.com/AskalotCQA/askalot
[44] https://www.edx.org/
[45] https://courses.edx.org/courses/course-v1:CaltechDelftX+QuCryptox+3T2016/

Tabuľka 1: Sumárna štatistika MOOC kurzu o kvantovej kryptografii.

Metrika	**Počet**
Zapísaní študenti v kurze	8115
Študenti ktorí začali kurz	4618
Aktívni používatelia CQA systému (aspoň jedno zobrazenie otázky)	1098 (24%)
Prispievatelia v CQA systéme	377 (8%)
Otázky	281
Otázky s odpoveďami	247 (88%)
Otázky s vybranými najlepšími odpoveďami	51 (18%)
Odpovede	333
Komentáre	453
Hodnotenia odpovedí učiteľmi	27

4.2 Realizácia

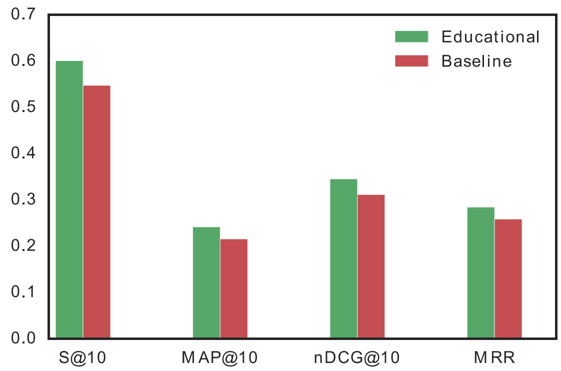
Na realizáciu offline experimentu je použitá experimentálna infraštruktúra CQA systému Askalot, ktorá umožňuje simulovať udalosti v systéme v poradí ako nastali v čase. Následne bolo možné rovnakú implementáciu použiť aj pri online experimente.

Črty v profiloch používateľov sú aktualizované v reálnom čase. Naviac, pre každý deň kurzu sa aktualizuje trénovacia dátová sada a oba klasifikátory sú pretrénované. Pre oba klasfikátory bol použitý klasifikačný algoritmus náhodné lesy (angl. random forest). Klasikačné algoritmy sme optimalizovali pre dosiahnutie najvyššej hodnoty pre metriku plocha pod krivkou (AUC). Riešenie sme implementovali v programovacom jazyku Ruby on Rails. Programovací jazyk Python bol použitý na spracovanie textu a strojové učenie pri ktorom sme využili predovšetkým knižnice Gensim[46] a Scikit-learn[47].

4.2 Offline experiment

Obrázok 2 zobrazuje výsledky smerovania otázok v porovnaní so základným prístupom. Merané metriky sú štandardne používané v tejto doméne. Navrhnuté smerovanie nových otázok je lepšie ako základné smerovanie otázok vo všetkých meraných metrikách. V prípade odporúčania nových otázok 10 najvhodnejším odpovedajúcim predikujeme aspoň jedného skutočného odpovedajúceho v 60,1% prípadov v porovnaní s 54,8% dosiahnutého základnou metódou.

[46] http://radimrehurek.com/gensim/
[47] http://scikit-learn.org/



Obrázok 2: Výsledky vybraných metrík v rámci offline overenie.

4.3 Online experiment

Online experiment sme nasadili v od 14. novembra 2016 (6. týždeň) MOOC kurzu. Na začiatku tohto experimentu boli používatelia rozdelený do troch skupín:

1. *Vzdelávacia skupina* (n=1306) s navrhnutým smerovaním nových otázok.
2. *Základná skupina* (n=1306) so základným smerovaním nových otázok.
3. *Kontrolná skupina* (n=1306) bez smerovania otázok.

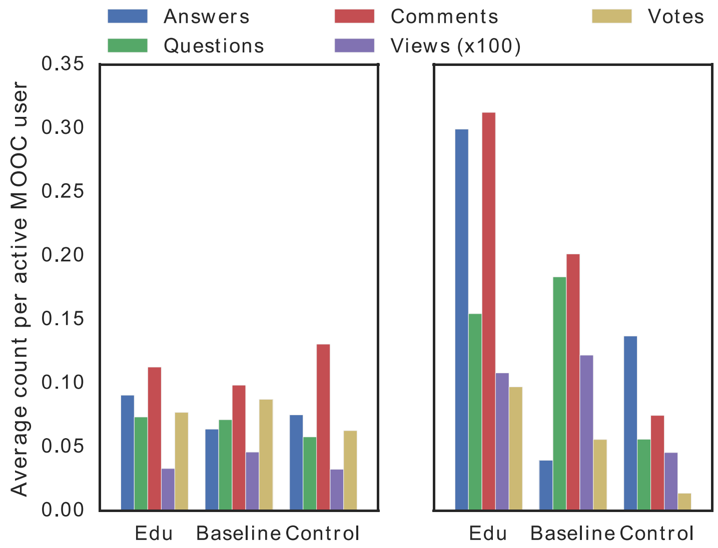
V online experimente odporúčame novú otázky 10 používateľom v prvej skupine a 10 používateľom v druhej skupine. Obmedzením je smerovanie maximálne štyroch otázok v priebehu siedmych dní.

Počas online experimentu bolo smerovaných 132 nových otázok potenciálnym odpovedajúcim, teda bolo vygenerovaných spoločne 2640 odporúčaní. Navrhnutá metóda smerovania otázok dosiahla štatisticky významne vyššiu mieru prekliknutia ako základná metóda. V prípade metriky S@10, ktorá určuje časť zo smerovaných otázok odpovedaná odporúčaným používateľom, navrhnutá metóda prekonala základnú metódu, ale nie štatisticky významne. Lepšie výsledky navrhnutej metódy potvrdzujú že použitie non-QA dát z MOOC kurzu zvýšilo presnosť predikcie.

Tabuľka 2: Presnosť smerovania otázok v rámci online experimentu.

Metrika	**Navrhnuté smerovanie otázok**	**Základné smerovanie otázok**	**Štatistická významnosť**
Miera prekliknutia (CTR)	23.25%	18.29%	$\chi^2(1, N = 2640) = 10.03, p < 0.01$
S@10	15.91%	10.61%	$\chi^2(1, N = 264) = 1.61, p = 0.20$

Obrázok 3 zobrazuje priemernú aktivitu v kurze na jedného aktívneho študenta v MOOC kurze. Je možné vidieť, že zavedením smerovania otázok sa priemerná aktivita zvýšila v porovnaní s kontrolnou skupinou. Skupina s navrhnutým odporúčaním zvýšila aktivitu prispievania do kurzu (odpovedanie, komentovanie) v porovnaní s ostatnými skupinami.



Obrázok 3: Priemerná aktivita na jedného aktívneho študenta v MOOC kurze pred (vľavo) a počas (vpravo) online experimentu.

5. Zhodnotenie

Navrhli sme novú metódu pre smerovanie nových otázok v MOOCs prostredí. Navrhli sme dve inovácie vhodné pre vzdelávaciu doménu. Prvou je explicitné modelovanie ochoty používateľa odpovedať na novú otázku, ktoré je skombinované so znalosťami používateľov. Druhou inováciou je využitie non-QA dát z MOOC kurzu na modelovanie študentov, ako napríklad známky v kurze, aktivita v MOOC kurze alebo prejdenie relevantných lekcií v kurze k novej otázke.

Ďalšími možnosťami na zlepšenie navrhnutej metódy je rozlišovanie inštruktorov kurzu a typov otázok, ktoré sa týkajú organizácie kurzu. Zaujímavým smerovaním je aj použitie modelovania tém (napríklad LDA) na spracovanie textu otázok a odpovedí a škálovateľnosť riešenia pre väčšie MOOC kurzy.

Navrhnutá metóda bola overená offline a aj online experimentom. Experimenty boli uskutočnené na MOOCs kurze v rámci EdX domény s viac ako 4600 študentami. Online experimenty sú v tejto doméne veľmi zriedkavé, ale poskytujú presnejšiu formu overenia a celkový dopad na komunity ako offline experimenty. V porovnaní so základným smerovaním otázok dosiahla navrhnutá metóda vyššiu presnosť predikcie relevantných odpovedajúcich pre novú otázku. To viedlo k zvýšeniu záujmu študentov o smerované otázky, zapojeniu väčšej časti komunity do prispievania a zvýšeniu priemernej aktivity študentov na aktívneho používateľa MOOC kurzu.

Appendices

A. Technical realization

This section describes implementation details of the proposed question routing approach.

A1. Application modules

Module *Yeast* contains classes for offline evaluation used for listening for events in the system and calling appropriate methods. Module *Services* contains listeners for events necessary for question routing in the online system. Both modules are dependent on *Question routing* module implemented in Python programming language as shown in the Figure A-1. *Question routing* module contains updating user profiles, ensemble training, text processing and matching of new questions and users.

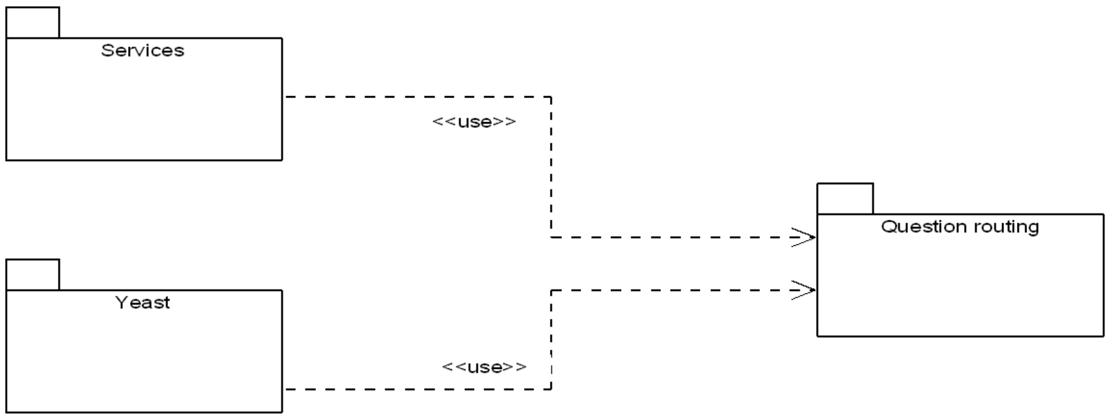


Figure A-1: Related application modules for question routing approach.

A2. Database model

In Figure A-2, part of the Askalot database model is shown which is related to the question routing method. One can see the main entities representing *Users*, *Questions* and *Answers*. Entity *UserProfiles* are used to store all features that are used for classification. On the other hand, entity *QuestionProfiles* contains TF-IDF question profile.

Entity *Views* represents views of question by a user. Entity *Lists* contains views of lecture for a category in the course. Entity *Votes* contains positive and negative votes for a question or an answer by a user. Entity *Evaluations* is used for storing evaluations by teachers.

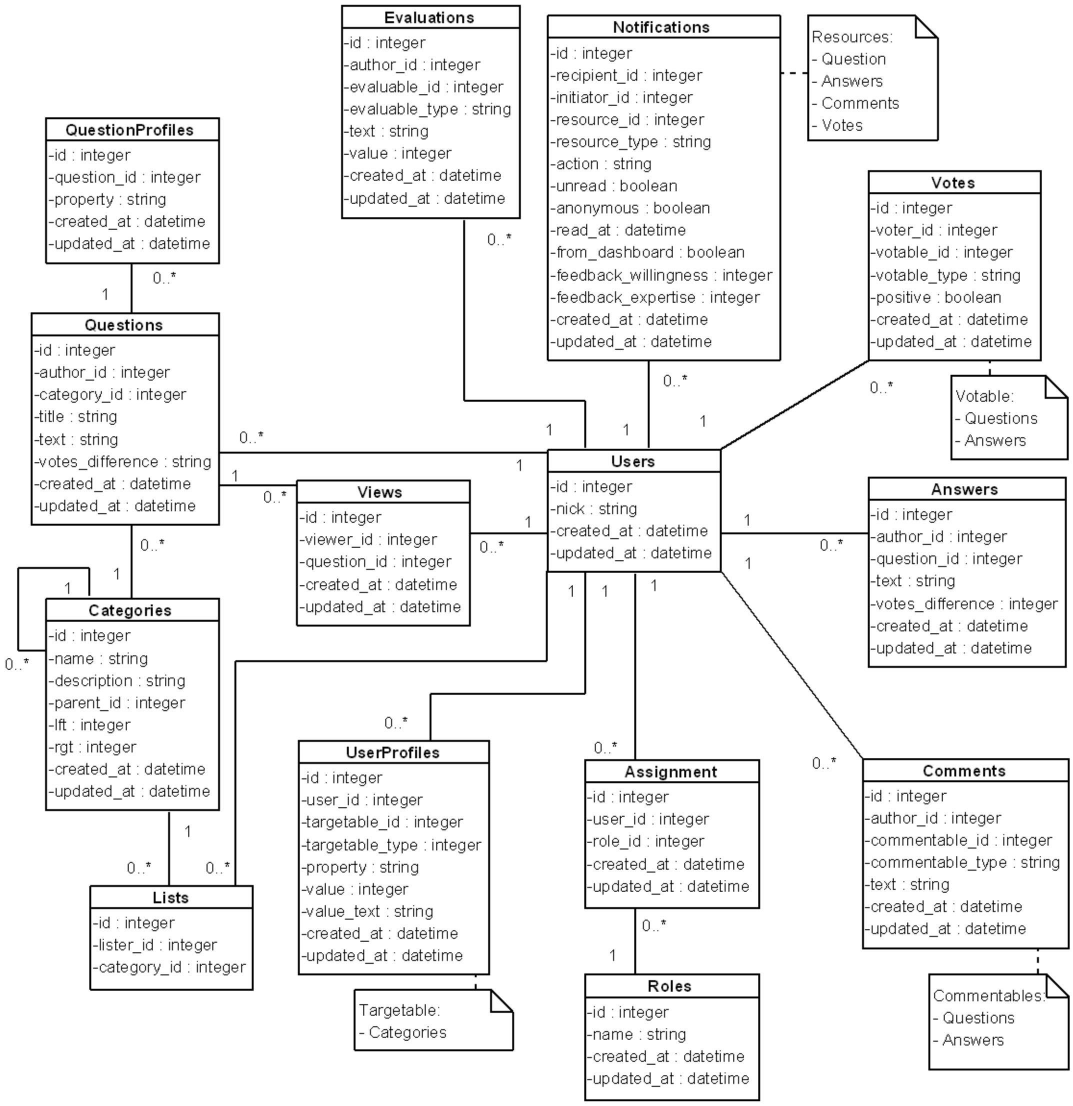


Figure A-2: Part of the Askalot database model used by question routing method.

A3. Updating features and question routing

In Figure A-3 and Figure A-4, sequential diagrams of updating features used for recommendation and routing new question is shown. Both figures contain *Shared::Yeast* module which is part of Askalot experimental infrastructure. This module dispatches all events by the time the they had happened. The question routing method for online deployment is the same, except the *Shared::Yeast* module. The module responsible for publishing events is *Shared::Events::Dispatcher* which publish events for create and update action for the following resources: *Shared::Answer, Shared::Question, Shared::Vote, Shared::List, Shared::View* and *Shared::Comment.*

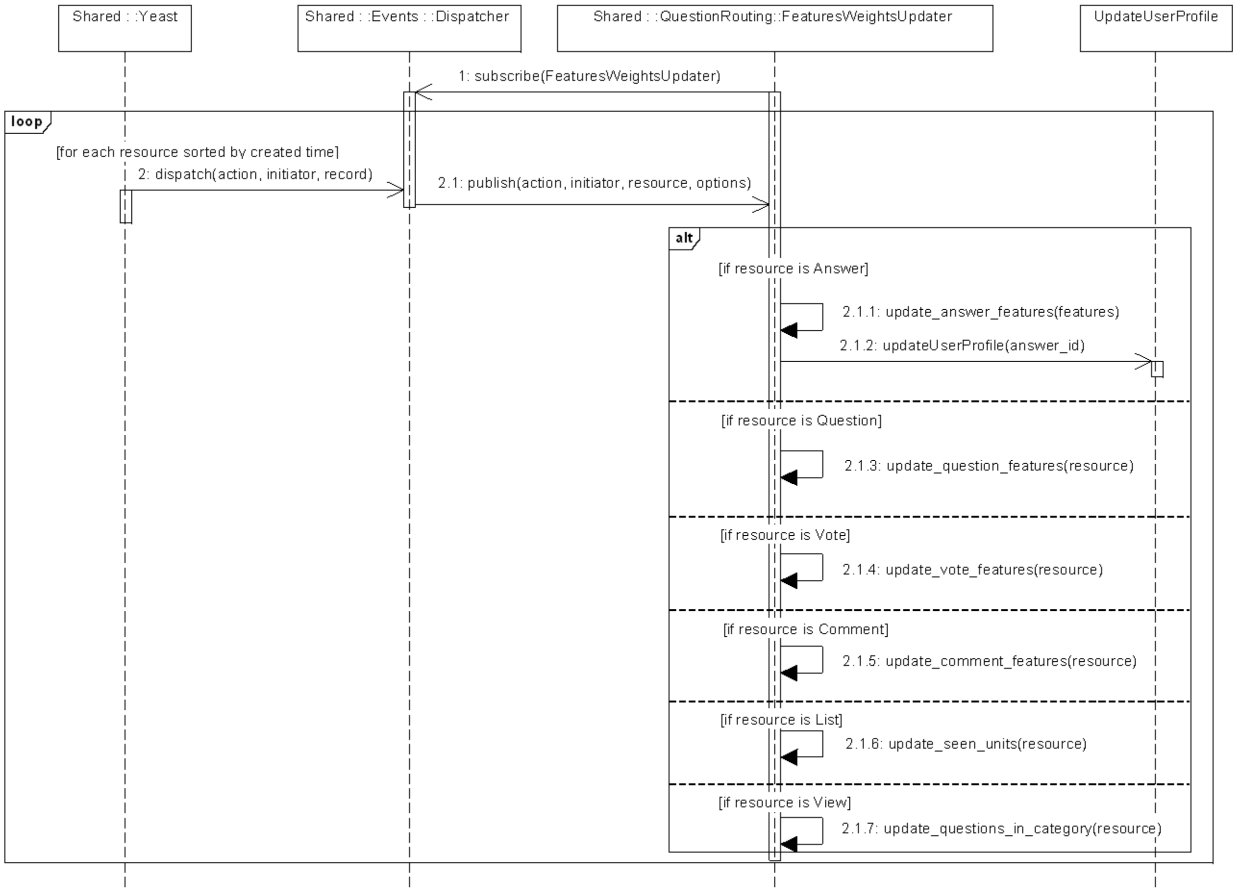


Figure A-3: Updating features used in UserProfiles table.

In case of quantitative features, they are incremented or decremented (for negative votes). For time-related features, the time is changed in updated_at column. Example code for updating user feature *CommentsCount*:

```
def update_comment_features(comment)
  update_feature(comment.author, 'CommentsCount')
end

def update_feature(user, property)
  if Shared::User::Profile.exists? ({user: user, targetable_id: -1,
                        targetable_type: property, property: property})
    Shared::User::Profile.where(user: user, targetable_id: -1,
                                targetable_type: property,
                                property: property)
                         .first.increment!(:value)
  else
    Shared::User::Profile.create(user: user, targetable_id: -1,
                                 targetable_type: property,
                                 property: property, value: 1)
  end
end
```

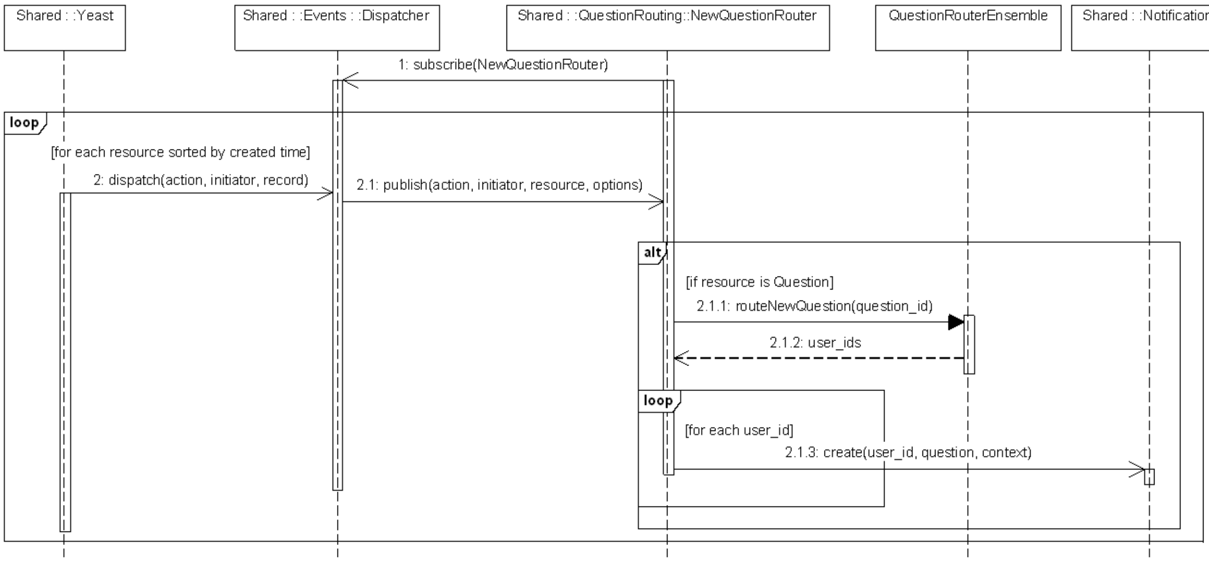


Figure A-4: Sequential diagram of new question routing.

A4. Text processing

Text preprocessing is done by method *preprocess_document,* which is part of a *TextualDictionary* class:

```
def preprocess_document(self, text):
    words = [self.preprocess_word(word) for word in
             utils.tokenize(text, lowercase=True, deacc= True)]
    return [word for word in words if self.is_valid_word(word)]

def preprocess_word(self, word):
    return self.stemmer.stem(word)

def is_valid_word(self, word):
    if len(word) < MIN_WORD_LENGTH or word in self.stop:
        return False
    return True
```

When user post an answer, his/her profile will be updated by a question and answer text profiles:

```python
def update_user_profile(textual_dictionary, answer):
    if hasattr(answer, "question_id"):
        text = process_answer(answer, textual_dictionary)
    else:
        text = process_comment(answer, textual_dictionary)
      # Update dictionary with answer/comment text
    textual_dictionary.vocabulary.add_documents([textual_dictionary \
                                    .preprocess_document(answer.text)])
    bow = textual_dictionary.vocabulary \
              .doc2bow(text, allow_update=False)
    if len(bow) == 0:
        return
    # Load and update user profile if exist
    user_profile = DataManager.get_user_profile_property(answer.author_id,
                                                         'BoW')
    if user_profile:
        user_bow = DataManager.load_bow_json(user_profile.text_value)
        bow = Utils.sum_bows(user_bow, bow)
        DataManager.update_user_profile(answer.author_id, 'BoW',
                                        json.dumps(dict(bow)))
    else:
        DataManager.insert_user_profile(answer.author_id, 'BoW',
                                        json.dumps(dict(bow)))


def sum_bows(bow1, bow2):
    bow_dict = dict(bow1)
    for word in bow2:
        tokenid = word[0]
        count = word[1]
        if bow_dict.get(tokenid) is not None:
            bow_dict[tokenid] = bow_dict[tokenid] + count
        else:
            bow_dict[tokenid] = count
            return [(k, v) for k, v in bow_dict.iteritems()]
```

A5. Matching of questions and users

In Figure A-5, the class diagram for classification task of matching questions and users is shown. *Ensemble* contains one instance of an expertise classifier and one instance of willingness classifier. In the method *predict* of an *Ensemble* instance, prediction probabilities are combined. *Ensemble* class is used by *Training* module for fitting classifiers to the data. Consequently, it is used by *QuestionRouterEnsemble* to predict answerers for a new question.

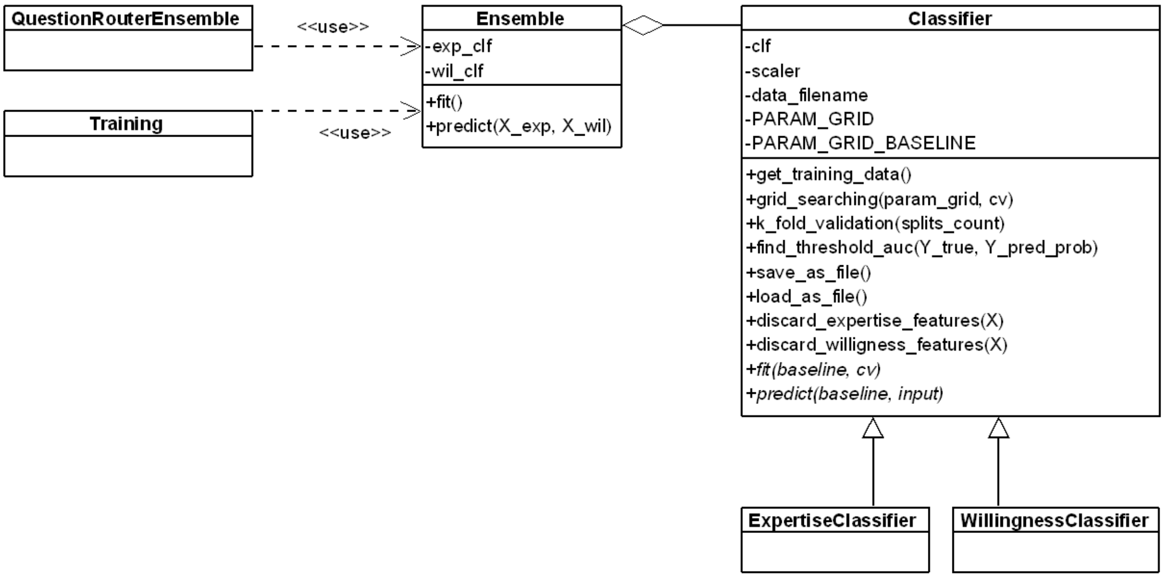


Figure A-5: Class diagram for ensemble classification.

Combination of expertise and willingness predictions is done in the *Ensemble* method predict:

```
def predict(self, X_exp, X_will):
    exp_predictions = self.exp_clf.predict(self.baseline, X_exp)
    will_predictions = self.will_clf.predict(self.baseline, X_will)
    indices = [ind for ind, (i, j) in enumerate(zip(exp_predictions,
                                                    will_predictions))]
    probabilities = exp_predictions[indices] * will_predictions[indices]
    # Sort descending based on probabilies array
    i = np.array(probabilities).argsort()[::-1]
    indices = np.array(indices)[i]
    return indices, exp_predictions, will_predictions
```

B. User guide

In order to install the proposed question routing method, it is necessary to install Askalot[48] CQA system at first.

B1. Askalot installation

Install the requirements for the Askalot CQA system:

- Ruby 2.3 or higher
- Ruby on Rails 4.2
- PostgreSQL 9.3 or higher
- Elasticsearch 1.7

Copy the source code from the attached media to any folder and run following command from that folder:

```
bundle install
```

Copy the following configuration files:

```
cp config/configuration.{yml.example,yml}
cp config/database.{yml.example,yml}
cp config/newrelic.{yml.example,yml}
```

In the *database.yml*, configure the connection to the database. Start the Elasticsearch by running the following command from the folder where Elasticsearch is installed:

```
./bin/elasticsearch
```

Next step is to create database and load schema of the database:

```
RAILS_ENV=edx_development rake db:create db:structure:load
DB_STRUCTURE=components/mooc/db/structure.sql
```

B2. Question routing method

Requirements for the question routing method are:

- Python 2.7
- Gensim
- Scikit-learn
- NLTK
- Numpy
- Psycopg2

[48] https://github.com/AskalotCQA/askalot

It is possible to download the requirements using pip package manager. If you are using Conda[49] package manager, the environment with all dependencies can be imported by following command, where CD_PATH is path to the media attached:

```
conda env create -f CD_PATH/askalot-conda-env.env
```

Import the database data from the attached media:

```
pg_restore -d askalot_edx_development < CD_PATH/data/askalot-
anonymized.backup
```

Open Python console and download NLTK stop words:

```
nltk.download("stopwords")
```

Finally, to start the offline evaluation of the proposed question routing method, truncate *UserProfiles* table in the database and run following command:

```
rake yeast:feed FEEDERS=NewQuestionRouter,FeaturesWeightsUpdater
RAILS_ENV=edx_development
```

Output is saved to the directory *recommendation* and contains:

- Metrics, where N in filename is used for metrics computation:
  - *full-evaluation-N.dat* – evaluation of educational question routing
  - *baseline-evaluation-N.dat* – evaluation of baseline question routing
- Datasets for training:
  - *expertise-train.dat* – dataset for expertise classifier
  - *willingness-train.dat* – dataset for willingness classifier
- Logs files used for logging debug and error outputs:
  - *traning.log*
  - *update-profile.log*
  - *qrouting-error.log*
- Trained classifiers:
  - *expertise-classifier.pkl, expertise-baseline-classifier.pkl*
  - *willingness-classifier.pkl, willingness-baseline-classifier.pkl*
- Vocabulary of words: *vocabulary.dat*
- Grades export from EdX instructors panel: *grades.csv*

[49] http://conda.pydata.org/docs/intro.html

C. Paper submitted for RecSys 2017

This full paper was submitted for the 11th ACM Recommender Systems Conference RecSys 2017 conference (acceptance rate is 20-24%). Preliminary version of this paper was accepted for student research conference IIT.SRC 2017.

D. Plan review

During the last three semesters, we had meetings with my supervisor each week. Usually, we discussed about new ideas or implementations. I was writing summary of my work done at home and a summary of every meeting with my supervisor.

We did an extensive research in a topic of CQA systems, question routing and MOOCs in the first semester. The plan for the second semester is shown in the Table D-1. We built the first prototype of question routing method in the fourth week of autumn semester, which was sooner than planned. Then we iteratively improved the method by proposing features for user modelling and by tuning of classification algorithms. We postponed LDA training to the spring semester and starts with TF-IDF bag-of-words text representation to finish the method as quickly as possible before the end of the EdX course used for evaluation. We deployed our question routing framework in the eight week and add questionnaire for feedback in the CQA system two weeks after.

Table D-1: Plan for the autumn semester.

Task	**Duration (in weeks, 12 total)**
Setting up environments, tools	1-2
Revision of proposed method	2-3
Data preprocessing	3-4
Proposed method implementation, LDA training	4-6
Deployment to EdX	7-9
Writing evaluation report	9-11

In the final semester, which was planned as shown in the Table D-2, we spent most of the time by evaluation of online experiment. We skip LDA training and put our efforts on writing a paper for the RecSys conference, which we submitted in eight week. In the remaining weeks, we improved computation of TF-IDF bag-of-words similarities and finalized the thesis.

Table D-2: Plan for the spring semester.

Task	**Duration (in weeks, 10 total)**
LDA training	1-2
Online experiment evaluation (comparison within groups, comparison within individuals, computing metrics, coverage rate by educational-specific method applied to baseline question routing)	2-6
Revision of proposed method in terms of scalability	6-7
Finalizing the thesis	7-10

To sum up, the most important tasks from our plans were fulfilled according to a plan. However, we adapted our plan in spring semester for writing a research paper and skipped LDA training.

E. Content of attached media

Content of the attached media:

Directory	Content
/src	Source code of Askalot CQA system with educational question routing.
/data	Database backup.
/evaluation	Source code of offline and online experiment evaluation.;