

Received 25 November 2022, accepted 2 January 2023, date of publication 11 January 2023, date of current version 19 January 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3235953

RESEARCH ARTICLE

Learning Software Project Management From Analyzing Q&A's in the Stack Exchange

ALIREZA AHMADI¹, FATEMEH DELKHOSH², GOURI DESHPANDE³,
RAYMOND A. PATTERSON², AND GUENTHER RUHE⁴, (Senior Member, IEEE)

¹Department of Computer Science, University of Calgary, Calgary, AB T2N 1N4, Canada

²Haskayne School of Business, University of Calgary, Calgary, AB T2N 1N4, Canada

³Electrical and Software Engineering Department, University of Calgary, Calgary, AB T2N 1N4, Canada

⁴Department of Computer Science and Department of Electrical and Software Engineering, University of Calgary, Calgary, AB T2N 1N4, Canada

Corresponding author: Gouri Deshpande (gouri.deshpande@ucalgary.ca)

ABSTRACT Software Project Management (SPM) is considered the key driver for the success or failure of software projects. Project failure is caused by various factors, the most important of which is poor SPM. Thus, we investigated the needs of practitioners by focusing on Project Management Q&A communities. More precisely, we targeted Stack Exchange to identify the primary needs of software project managers. More than 5000 SPM questions were analyzed from the conceptual model given by the Project Management Body of Knowledge PMBOK. For pre-training of the Machine Learning classifiers, we implemented Bidirectional Encoder Representations from Transformers (BERT) and Doc2Vec text embedding and compared their performance. Our results showed that BERT outperforms Doc2Vec for pre-training in almost all scenarios. Schedule management, followed by resource management, are the main PMBOK knowledge areas of concern for project managers. Among the process groups, the emphasis of the questions is on planning. We compared the findings with the learning and training status quo in 11 top Canadian universities. We analyzed 46 SPM-related courses and found that the rank correlation of PMBOK knowledge areas is 0.23 between the key content of the analyzed courses and the focus of Q&A's knowledge areas analyzed from Stack Exchange.

INDEX TERMS Software project management, PMBOK, stack exchange, BERT, Doc2Vec, learning, industrial needs.

I. INTRODUCTION

Software Engineering is a knowledge-intensive discipline concerned with all the aspects of software development and evolution. Software Project Management deals with software projects and the challenges of human-based development (as opposed to the more deterministic processes in traditional projects). The higher flexibility in software development approaches puts new demands on the capabilities of software project management. Weaknesses in planning, organizing, staffing, directing, and controlling are hard to counter-balance by more efficiency in technical development work [1].

The dramatic shift in Information Technology in the recent past has resulted in new challenges in the software

engineering industry. Various factors, such as unrealistic project goals, inaccurate estimates, badly defined system requirements, inadequate reporting of the project's status, unmanaged risks, and poor communication among customers, developers, and users, could lead to project failure. Such failed project management can incur enormous financial costs for companies. The Consortium for Information and Software Quality determined that the total cost of poor software quality in the United States in 2020 was \$2.08 trillion [2]. Successful SPM demands human experts with a high level of knowledge [1].

Numerous resources are available for practitioners to help them expand their knowledge and learn new skills. Community Question and Answering (Q&A) is one of the well-known examples of effective knowledge-sharing sources in open online communities [3]. According to Garousi et al. [4], practitioners are more likely to use a source

The associate editor coordinating the review of this manuscript and approving it for publication was James Harland.

of information, such as Q&A communities, to ask questions and share their knowledge. Therefore, these communities can be considered rich data sources. Furthermore, the available textual data in these communities lead researchers to the actual needs of practitioners. In this study, we targeted Project Management Stack Exchange (PMSE),¹ a widely used project management Q&A community that is a well-known community for project managers. In PMSE, practitioners post questions that represent the concerns they face during a software project's life cycle, and other community members can view, up-vote or down-vote questions and respond to them. The Project Management Body of Knowledge (PMBOK) [5], a standard terminology and guidelines for project management, serves as the underlying conceptual model of analysis.

Mining SPM communities provide fundamental insights for further research and development for academia and industry. The implications of these insights for SPM education are investigated as an application. We compared the characteristics of software project managers' needs and concerns with the SPM-related topics covered by instructors in university courses. Even though work experience plays an irreplaceable role in the growth of a software project manager, and the role of education and learning in building up this career cannot be denied. 46 SPM-related courses in the eleven top Canadian universities were analyzed. Reviewing the courses' description and agenda has exposed the status quo of SPM education. A substantial gap between SPM education and practitioners' needs was found. Figure 1 depicts the workflow of the various phases of this study.

Overall, this research investigates three research questions (RQs):

- RQ1: Analyse the accuracy of classification results of the Q&A's from Stack exchange when combining Text Embedding and Machine Learning using (Doc2Vec, BERT) and (Random Forest, Support Vector Machine (SVM), Naive Bayes), respectively.
- RQ2: What are the most critical PMBOK knowledge areas and process groups, as analyzed in RQ1?
- RQ3: How well are SPM courses at the top 11 Canadian Universities aligned with the needs stated in RQ2?

The remainder of this paper is organized as follows: Section II provides the content of related work. Section III describes the data collection process and descriptive analysis of the data collected. Section IV presents the classification method. The results answering the stated research questions are the content of Section V. Finally, in Section VI, a discussion and future work are presented.

II. RELATED WORK

A. Q&A COMMUNITIES

Due to the rich sources of valuable data, numerous studies have explored different Q&A communities such as

Yahoo! Answers [6], Quora [7], Stack Overflow [8], [9], [10], and [11] for different purposes. Such purposes include the identification of important conversations [12], uncovering the crucial factors that contribute to unanswered questions [13], addressing duplicate questions and quality issues [14], and designing an interactive approach for searching questions [15].

Most Q&A community studies in Software Engineering are dedicated to Stack Overflow. The study [16] conducted on PMSE is a predecessor to this paper. The authors gathered data from popular Q&A sites such as Stack Overflow, Quora, and PMSE to understand the most challenging Requirements Engineering topics relevant to practitioners. They have used Latent Dirichlet Allocation (LDA) and statistical analysis to explore RE's main topics of interest from a practitioner's perspective. Our research focuses on Stack Exchange, the primary Q&A community for project managers, to identify the critical needs of project managers.

B. ANALYSIS OF TEXTUAL DATA IN SOFTWARE PROJECT MANAGEMENT

The second stream of the literature includes research that applied Artificial Intelligence (AI) tools for analyzing SPM. The tools included neural networks, deep learning [17], and ML methods [18]. Using three ML algorithms (SVM, neural networks, and generalized linear models) and cross-validation, Pospieszny et al. [19] built a decision support tool for effort and duration estimation for organizations that develop or implement software systems. In another study, Gobov and Huchenko [20] analyzed the current state of requirements elicitation techniques in different software project contexts and defined influencing technique selection based on the two classification models.

Other studies in this stream analyze textual data generated by users using different embedding techniques. Ahmadi [21] proposed the Deep QA-Miner method based on deep neural networks to analyze the textual data of Stack Exchange to extract the needs of SPM practitioners. The also examined the accuracy of the Deep – QA miner method compared with traditional ML methods and common simple structured deep neural networks in terms of accuracy. Results show that the Deep-QA miner method outperforms the other methods. In another study, de Araújo and Marcacini [22] presented the RE-BERT model to identify software requirements from app reviews. They implemented the RE-BERT model on eight different apps, and their results show that RE-BERT outperforms existing methods. In order to resolve the issue of low accuracy, Gao [23] developed a sentient classification model using pre-trained BERT that can extract the abstract text features of a single character based on the context semantic relationship with high accuracy. In our paper, we use two text embedding techniques, BERT and Doc2Vec, to use textual data for ML methods and then extract practitioners' needs.

¹<https://pm.stackexchange.com/>

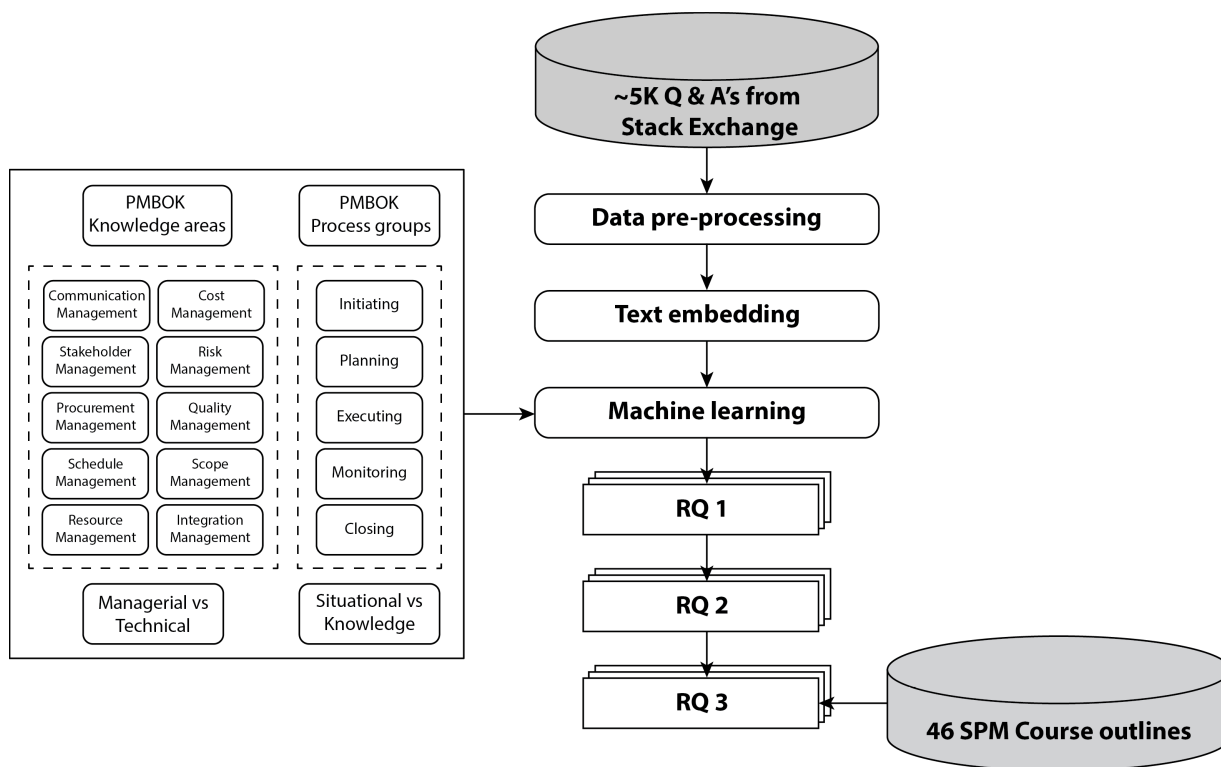


FIGURE 1. The conceptual model and workflow of the paper.

C. SPM EDUCATION

SPM education is the focus of papers in the third stream of the literature. Some studies aim to improve SPM learning value by using games and simulators [24], [25], and [26]. Other papers tried to reduce gaps between industry needs and university courses by taking advantage of literature [27] and [28]. Considering the major SPM activities recommended in the literature, [28] formulated an SPM graduate course. To identify the needed areas for further improvement, they compared their executed approach with the Portfolio, Program, and Project Management Maturity Model (P3M3) in SPM processes. Reviewing 33 papers from literature, [27] identified the most crucial industry-requested skills and revealed knowledge deficiencies in graduating Software Engineering students.

D. ANALYSIS OF THE NEEDS OF PRACTITIONERS

Some of the previous studies used surveys to determine the needs of practitioners. JS Makahaube [29] employed seven knowledge areas adapted from the PMBOK with thirty individual processes, including integration, scope, schedule, cost, quality, communication, and risk management, with five statements assigned to each project management process. Consistent with the Project Management Maturity Model (PMMM), five statements define the characteristics of each maturity level, with level one being the least mature and level five being the most mature. The survey results show risk management at level one, while scope, schedule,

communication, and quality management are at level two. Finally, integration and cost management are at the third level. According to the survey results, risk, scope, schedule, communication, quality, integration, and cost management are the areas that require more consideration and improvement. According to [30] the primary needs of practitioners in PMBOK area tasks are integration management, time management, risk management, scope management, communications management, resource management, procurement management, quality management, and cost management, respectively. [31] identified practitioners are more concerned with integration, time, scope, resource, cost, risk, quality, communication, and procurement management. Finally, [32] discovered the main needs of practitioners in PMBOK area tasks are time, scope, communication, integration, risk, quality, cost, procurement, and resource management respectively.

In terms of PMBOK process groups, Pereira et al. [5] demonstrated that the majority (87.8%) of participants' needs are in the planning phase, which causes the project's failure. [30] also discovered planning, monitoring, executing, initiating and closing respectively are the main needs of practitioners.

III. DATA COLLECTION AND ANALYSIS

To understand the primary needs of practitioners in this community, we mined 5335 questions and their attributes from PMSE across ten years (2011/01 to 2020/02) for our

TABLE 1. Data dictionary and descriptive statistics for our dataset.

Variable	Description	Mean	Standard deviation
<i>VotesCount</i>	Number of upvotes minus downvotes for each question	4.08	6.68
<i>AnswersCount</i>	Total number of answers to a question by registered users	3.02	2.31
<i>ViewsCount</i>	Total number of times that a question is viewed by all users	1994.39	7122.37
<i>QuestionerReputation</i>	Stack Exchange's measure of questioner's reputation	922.64	3116.21

analysis. Using the Delphi method [33] with four experts for creating our training set, a random sample of 1000 questions were classified from four different perspectives: PMBOK area, PMBOK process group, Managerial-level vs. Technical-level, and Situational-describing vs. Knowledge-seeking.

A. DATA COLLECTION

Registered users in the Stack Exchange community can publish and vote on questions. When a questioner posts a question, registered and unregistered users can view it, and the registered users can up-vote, down-vote, and answer it. For our analysis, we collected publicly available data regarding 5,335 questions containing information about questions and questioners from the PMSE section of the Stack Exchange community over the period spanning January 4, 2011, through February 10, 2020. For each question, we collected the identification number, body, title, assigned tags, the total number of up-votes minus downvotes, the total number of answers, the number of times a question is viewed, the date when the question was asked, and its actual status. The data crawled and used for this study can be accessed on our website.²

Each question's status can be *unanswered* (when there is no answer), *answered* (when there is at least one answer but it has not been marked as accepted by the questioner), or *answered-accepted* (when one of the answers has been marked as accepted by the questioner). We also garnered the reputation score for each questioner. Questioners' reputation score is a "simplified" measure of how much value they have brought to the community by asking and answering other users' questions. For data pre-processing, we used two well-known Python libraries, NLTK and SpaCy, to perform pre-processing steps to remove HTML Snippets, links, and punctuation.

B. ANALYSIS AND ANNOTATION PROCESS

Table 1 provides summary statistics for various metrics. Of the 5335 questions, 2676 questions were answered, 2596 were answered-accepted, and only 63 were unanswered. A randomly drawn sample of 1000 questions was created. Using the Delphi method involving a group of four experts, these 1000 questions were labeled from four different perspectives: the PMBOK

area, the PMBOK process group, managerial-level concerns versus technical-level concerns, and Situational-describing scenarios versus Knowledge-seeking questions.

A Delphi study aims to obtain consensus from a group of experts through repeated questionnaire responses and controlled feedback [34]. One significant advantage of this approach is that it avoids direct conflict between experts. The Delphi method works as follows. First, the group facilitator chooses a group of experts based on the topic under consideration. Following confirmation of all participants, each group member is sent a questionnaire with instructions to comment on each topic based on their personal opinion, experience, or previous research. The questionnaires are returned to the facilitator, who organizes the responses and creates copies of the data. Each participant receives a copy of the compiled comments and the opportunity to comment further. After each comment session, all questionnaires are returned to the facilitator, who determines whether another round is required or whether the results are ready for publication. The questionnaire rounds can be repeated as often as necessary to achieve a consensus.

In our study, each Delphi participant had at least two years of experience in SPM, and three of them were co-authors of this paper. After an independent assessment of each questions' label, Delphi participants attempted to reach a consensus on the overall label of the questions. A researcher facilitated a discussion of the evaluations, where conflicting views were negotiated and a consensus was reached. A conflict occurs when there is no complete unanimity. Two stages have been taken to resolve any conflicts. In the first stage, the labeling team members shared opinions and attempted to achieve unanimity in their classification. Any remaining conflicts from stage one are discussed in the second stage. As shown in Table 2, a complete agreement was achieved for all 1,000 questions.

For managerial-level vs technical-level, binary classification is used to determine whether a question is at the managerial or technical level. This viewpoint focuses on the level and type of information requested by the questioner. For example, if a question is asked about project management concepts, concerns, or situation handling, it is at the managerial level; otherwise, it is a technical-level question. Examples of technical level questions include finding/editing a feature in a PM tool such as Microsoft Project or finding the best resources to prepare for the PMP certification exam.

²<https://github.com/alirzahmadi/DeepQA-Miner>

TABLE 2. Frequency of each class in PMBOK area using Delphi method: The number of conflicts in two stages of labelling process show that by the third stage, a complete agreement was achieved.

Task	Number of questions to be labeled	Number of conflicts after labelling	Number of conflicts after stage 1	Average Number of conflicts after stage 2
PMBOK area	1000	166	52	0
PMBOK process groups	1000	58	2	0
Managerial vs Technical	1000	9	0	0
Situational vs Knowledge	1000	58	0	0

For the situational-describing vs knowledge-seeking viewpoint, we dive deep into the questions classified as managerial-level. This viewpoint focuses on the questioner's preferred method of asking the question about the context of the question. The questions are divided into two categories. First, *situational-describing* is when questioners describe a situation (scenario) to seek advice on how to deal with it (decision-making). Knowledge-seeking refers to the questions where questioners ask about a technique, method, definition, or concept related to project management. Note that there is a chance that questioners in the Knowledge-seeking class deal with a decision-making scenario, but they have been able to formulate their needs for a knowledge-seeking question. 820 questions are categorized as managerial-level questions, 503 of which are situational-descriptive, and 317 are knowledge-seeking questions.

IV. METHODOLOGY

It is not practical to manually classify thousands of questions and answers. Automation could help to accommodate a large data set. Thus, we explored how Machine Learning (ML) can be applied to classify the extended data set. Using the data collected in the Delphi study as the ground truth for the training/testing set, we utilize automated methods of generating category predictions for the four different labeling perspectives. We compare three ML algorithms: Random Forest, Support Vector Machine, and Naive Bayes. We compare the performance of these three algorithms when combined with two text embedding techniques, BERT and Doc2Vec. BERT and Doc2Vec convert the questions' textual content into numerical vectors used to train and test the ML algorithms.

We will show that classification results are substantially more accurate using BERT, especially for the multi-category classifications of the PMBOK area and the PMBOK process group, and this behaviour is regardless of the ML method used. The best ML methods are then used to classify the remaining 4335 questions.

A. BERT

BERT (Bidirectional Encoder Representations from Transformers) is a state-of-the-art language representation model developed at Google lab [35]. The use of transformers enables to capture of the contextual relationship with the words and sub-words, which makes it a state-of-the-art model. BERT has proven to be effective in a variety of NLP tasks, such

as question answering, language inference, and predicting relationships between sentences.

In this study, we use the BERT base model, which is trained on English Wikipedia. It exploits 12 layers or transformer blocks, 768 hidden units in each layer, and 110 million parameters. Although the basic BERT model can be further fine-tuned using task-specific data, we use the BERT model in the feature-based approach mainly for two reasons. First, task specific data is lacking for further fine-tuning. Secondly, pre-trained BERT models have proven to be approximately 0.3 units behind fine-tuned BERT model for most NLP tasks [35] in terms of the F1, a measure used to understand the performance of the ML model which ranges from 0 to 1.

B. Doc2Vec

Doc2Vec [36] is a context-independent word embedding method based on artificial neural networks. It generates a vector representing the document (in our context a sentence) to predict the target word.

V. ANSWERING RESEARCH QUESTIONS

In this section, the results are organized according to the three research questions formerly raised.

A. ANSWERING RQ1: ANALYSE THE ACCURACY OF CLASSIFICATION RESULTS OF THE Q&A'S FROM STACK EXCHANGE WHEN COMBINING EMBEDDING AND MACHINE LEARNING

We used two previously discussed text embedding methods (BERT and Doc2Vec) to answer this research question. We combined them with three ML classification methods. To determine the best combination of text embedding technique and ML algorithm, we first compared the results of the three ML methods (Random Forest, Naive Bayes, and SVM) after hyper-parameter tuning using BERT pre-training versus Doc2Vec pre-training. Hyper-parameter configuration for machine learning models has a direct impact on the model's performance. These parameters are configured to achieve optimal model performance [37]. Thus, model-specific parameters for the SVM and Random Forest methods were tuned, and the accuracy significantly improved.

As shown in Table 4, for the PMBOK area classification, BERT outperforms traditional Doc2Vec in terms of accuracy for all ML classifiers except Random Forest in the PMBOK area. Results using BERT pre-training are substantially better

for all other metrics (F1, precision, and recall) compared to results using Doc2Vec pre-training. Comparing the three ML methods using BERT pre-training for the PMBOK area classification, the Random Forest classifier slightly outperforms SVM on all four metrics (F1, Accuracy, Precision, and Recall). Furthermore, Random Forest and SVM substantially outperform the Naive Bayes method.

As shown in Table 4, the use of BERT pre-training generally enables better performance on all metrics with just a few exceptions. Overall, the use of BERT pre-training, compared to Doc2Vec pre-training, dramatically enhances the performance of both Random Forest and SVM. Using BERT pre-training, Random Forest slightly outperforms SVM.

Results for Managerial versus Technical classification are shown in Table 4. Again, Random Forest and SVM outperform the Naive Bayes method using both BERT pre-training and Doc2Vec pre-training. The results using BERT and Doc2Vec pre-training are similar for both the Random Forest and SVM methods. Finally, results for Knowledge versus Scenario classification are shown in Table 4 demonstrate the same results in terms of accuracy except for SVM.

In conclusion, we demonstrated that BERT pre-training achieves generally better results than the use of Doc2Vec pre-training, especially for the more complex multi-category classification tasks. Additionally, Random Forest and SVM substantially outperform Naive Bayes. We concluded that combining BERT pre-training with Random Forest and SVM generally produced the best classification results. Therefore, we used BERT pre-training with Random Forest and SVM for our remaining analysis.

B. ANSWERING RQ2: WHAT ARE THE MOST CRITICAL PMBOK KNOWLEDGE AREAS AND PROCESS GROUPS?

The results of classifications for the remaining 4,335 questions are presented in Figure 2. Schedule management generates the most questions at 46.21% for Random Forest and 42.02% for SVM. The next most prominent knowledge area is resource management, at 17.61% for Random Forest and 16.37% for SVM. A few questions were related to procurement and risk management, which were below 1.42%. This exceedingly low portion of risk management questions was opposite to our expectations.

Classifications for the PMBOK process group perspective are shown in Figure 2. Planning is the most frequent group classification in the set of questions, with estimates of 46.56% for Random Forest and 42.01% for SVM. Executing is next at 22% and 25.34% for Random Forest and SVM, respectively. Initiating, monitoring and closing are all at or below 20%. It is worth mentioning that the least frequent group is *closing*, with 1.50% for Random Forest and 1.52% for SVM. This phase is mostly overlooked during SPM.

Our results also find that more than 95% of the community topics target managerial concepts instead of technical topics. Among these 95% percent managerial questions, in more than 68% of the cases, questioners describe a specific situation. These cases can be mapped to challenging

decision-making scenarios. We also examined changes in PMBOK knowledge areas related to questions and PMBOK process group needs over time, using all (5335) questions. As shown in Figure 4, Schedule management is consistently dominant compared to other areas. Resource and scope management are growing in relative importance, whereas integration management is declining in prominence. According to the results of Figure 5, planning dominates all other process groups every year. It increases over time while the questions related to executing appear to be declining. For the other three process groups (initiating, monitoring and closing), there have been no significant changes over time.

Consistent with our expectations, our findings indicate that schedule, scope and integration management require more attention and improvement. However, opposite to our expectations, risk management is not a major concern. Our findings also show that resource management is one of the main concerns of practitioners. Furthermore, our findings in the PMBOK process group revealed that the planning phase is the primary concern of most of the PMSE questioners.

We also analyzed the PMBOK knowledge area questions in their occurrence across the process group timeline for the entire data set. Table 3 shows the questions for each primary PMBOK knowledge area for each PMBOK process group. A large number of questions belong to the schedule management and planning group. Interestingly, our results show that the executing phase of projects brings considerable concerns in dealing with resources and stakeholders. It also shows that planning the scope is a challenging step for practitioners.

C. ANSWERING RQ3: HOW WELL ARE SPM COURSES AT THE TOP 11 CANADIAN UNIVERSITIES ALIGNED WITH THE NEEDS STATED IN RQ2?

Mining SPM communities provide a fundamental insight that can be leveraged in different possible directions. These insights may result in further research and development for academia and industry. In this section, we answer the third research question by comparing the characteristics of software project managers' needs and concerns with the SPM-related topics covered by courses in 11 Canadian universities.

First, to come up with a comprehensible picture of the SPM-related topics, we studied the SPM-related courses at top-ranked [38] universities in Canada:

- 1) University of Toronto
- 2) McGill University
- 3) University of British Columbia
- 4) Université de Montréal
- 5) University of Alberta
- 6) McMaster University
- 7) University of Waterloo
- 8) Western University
- 9) University of Calgary
- 10) Queen's University
- 11) University of Ottawa

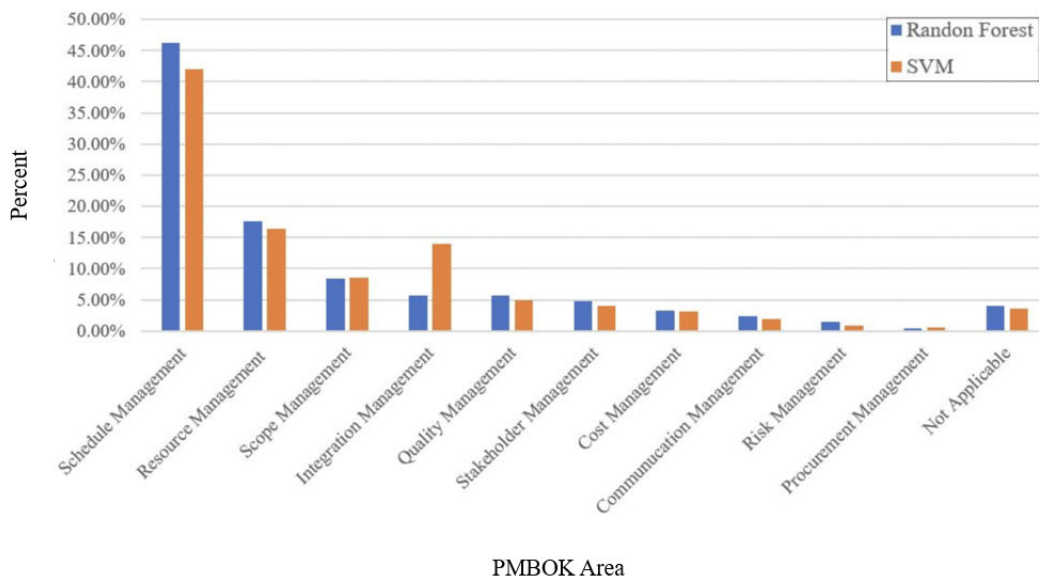


FIGURE 2. Ranking of PMBOK knowledge areas based on frequency of questions.

TABLE 3. Questions of PMBOK knowledge area per process group.

Class Name	Initiating	Planning	Executing	Monitoring	Closing	Not Applicable	Total
Schedule Management	134	1542	163	601	9	16	2465
Resource Management	44	272	481	120	12	10	939
Scope Management	16	289	33	106	5	2	451
Integration Management	99	36	75	64	30	0	304
Quality Management	4	115	120	50	8	8	305
Stakeholder Management	20	70	137	17	3	6	253
Cost Management	6	120	40	6	2	3	177
Communication Management	18	16	80	12	0	0	126
Risk Management	4	17	29	17	6	2	76
Procurement Management	1	4	17	1	0	2	25
Not Applicable	0	3	0	1	0	210	214
Total	346	2484	1175	995	75	259	5335

Apart from a direct Google search, the department websites were accessed to ensure no course was missed. Both Computer Science and Electrical & Computer Engineering departments were targeted. In other words, the courses considered are the ones that are offered specifically for students with a software-related major. At the same time, most universities have business/management departments or project management programs offering general PM courses. Although there are specialized courses for SPM, as long as a part of the course agenda is linked to SPM-related concepts, the course is selected for investigation. This process has resulted in 46 courses. The level of available information varied between the courses. There were courses with a dedicated web page showing the course’s goals, the topics covered per week, the assignments, and the course’s outcome for students. Conversely, courses are explained in a short paragraph on the course catalogue web page, mentioning its goal and overall concepts. All the available information is used to structure the topics covered during the semester. The list of courses is brought in Section VI as an appendix. This list includes

the university name, department name, course name/code, graduate/undergraduate level, and access link.

The amount and type of material dedicated to SPM are different across the courses. Therefore, the courses are categorized into four categories based on covering SPM concepts. The categories, along with their main attribute, are listed below:

- 1) **Full:** The whole agenda is dedicated to SPM.
- 2) **Partial:** Only a part of the topics is about SPM.
- 3) **Specialization:** All the agenda is about a specific area in the SPM.
- 4) **SA:** The course is dedicated to Software Analytics for SPM.

Figure 6 shows the number of courses in each category separated based on the course level.

The majority of the undergraduate courses are offered for one specific SPM area. The second largest category includes courses that partially cover SPM concepts. These courses are usually offered under titles similar to “Introduction to Software Engineering” or “Software Engineering Project.”

TABLE 4. BERT and Doc2Vec results after hyperparameter tuning for different perspectives.

		BERT				Doc2Vec			
		F1	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall
PMBOK area	Random Forest	0.52	0.44	0.74	0.44	0.40	0.45	0.47	0.27
	SVM	0.49	0.42	0.62	0.42	0.40	0.41	0.42	0.35
	Naive Bayes	0.39	0.39	0.41	0.39	0.30	0.35	0.43	0.29
PMBOK Process	Random Forest	0.64	0.54	0.80	0.54	0.43	0.50	0.53	0.35
	SVM	0.62	0.53	0.70	0.53	0.46	0.47	0.45	0.34
	Naive Bayes	0.37	0.40	0.37	0.40	0.37	0.37	0.47	0.43
Managerial vs. Technical	Random Forest	0.91	0.88	0.97	0.88	0.92	0.86	0.87	0.97
	SVM	0.90	0.89	0.92	0.89	0.92	0.87	0.91	0.93
	Naive Bayes	0.80	0.82	0.81	0.82	0.70	0.60	0.94	0.55
Knowledge vs. Scenario	Random Forest	0.78	0.77	0.78	0.77	0.77	0.71	0.67	0.92
	SVM	0.76	0.76	0.77	0.76	0.82	0.79	0.83	0.82
	Naive Bayes	0.75	0.76	0.76	0.76	0.68	0.66	0.85	0.57

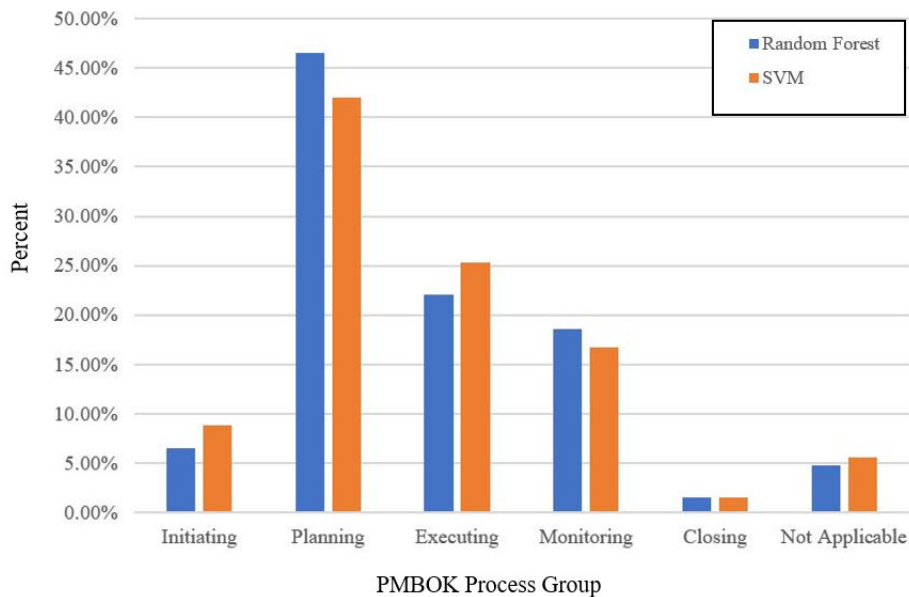


FIGURE 3. Main characteristics of practitioners' needs in PMBOK process group.

Surprisingly, there are only ten courses entirely dedicated to SPM. Out of these, only one of them is a graduate course.

The 21 courses, categorized as specializations, cover only four different areas of SPM. The majority of the courses are concerned with scope management. The main activities covered in these courses are requirements elicitation, modeling, validating, and tracking. There is only one course dedicated to schedule and communication management, where the covered material of the latter is about technical communication during a software project and the primary communication skills needed in the industry.

Thirteen courses have devoted a *part* of their agenda to SPM concepts. The detailed assessment of related concepts revealed the following findings:

- 1) In four courses, by introducing SPM, an overview of different areas is provided. The main effort in these courses has been to cover the main areas of software engineering involved in the development process.
- 2) Two courses have targeted agile methodologies. Both courses have a final project, and the students are asked to apply one of the agile methodologies in their course projects.

TABLE 5. The percentage of each class in PMBOK area perspective.

Class Name	Random Forest	SVM
Schedule Management	46.21%	42.02%
Resource Management	17.61%	16.37%
Scope Management	8.46%	8.58%
Integration Management	5.69%	14.02%
Quality Management	5.69%	4.98%
Stakeholder Management	4.74%	4.00%
Cost Management	3.32%	3.12%
Communication Management	2.37%	1.88%
Risk Management	1.42%	0.90%
Procurement Management	0.47%	0.50%
Not Applicable	4.02%	3.63%

TABLE 6. The percentage of each class in the PMBOK process group.

Class Name	Random Forest	SVM
Initiating	6.48%	8.83%
Planning	46.56%	42.01%
Executing	22%	25.34%
Monitoring	18.60%	16.75%
Closing	1.50%	1.52%
Not Applicable	4.80%	5.56%

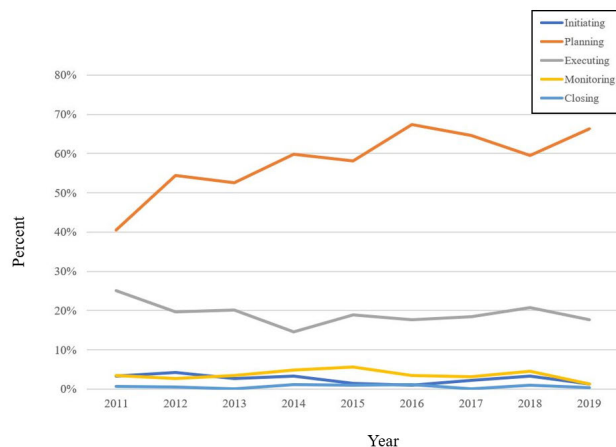


FIGURE 5. 9 year trend lines for PMBOK process groups using Random Forest results: Percentages of annual questions .

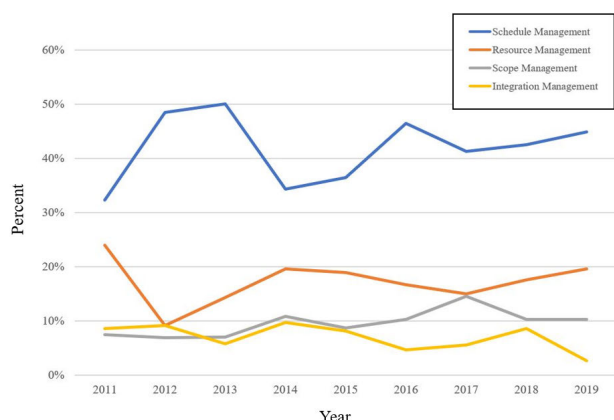


FIGURE 4. 9-year trend lines for PMBOK knowledge areas using Random Forest: Percentages of annual questions .

- 3) One of the courses has looked into SPM from a business viewpoint. By looking at the software market's challenges, procurement, quality, and cost management shape the main SPM-related concepts of this course.
- 4) Six courses have included concepts linked to three specific areas of SPM. Similar to specialization courses, the three covered areas are scope, quality, and communication management.

We only found ten courses fully dedicated to SPM. Although all these courses try to provide an overview of all components of SPM, some areas are being emphasized according to their agenda. Schedule management is an inseparable aspect of these courses. Also, five courses have highlighted risk management techniques. Only three (two) courses have explicitly mentioned resource (cost) management, respectively. One course also allocates a part of the agenda to SPM tools.

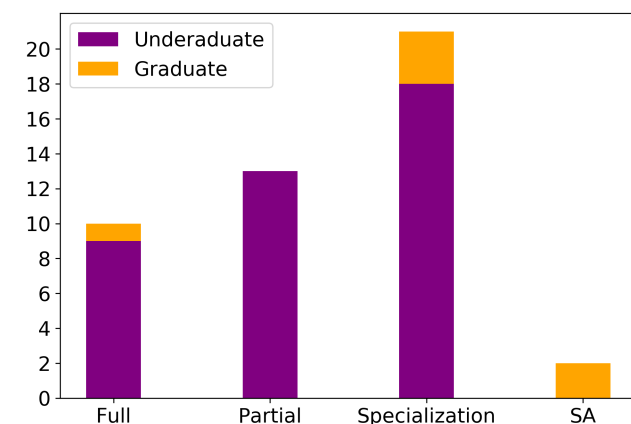


FIGURE 6. The counts of courses in different levels based on the amount of SPM-related teaching material.

We also listed the concerns in PMBOK areas for comparison (Table 7). The second column resulted from SPM QA community analysis and is considered a proxy for the status quo in the industry. The analysis of the course described above is taken as the status quo for teaching SPM in the third column. The results of a systematic mapping study by Nayebi et al. [39] were used as another proxy for reflecting the focus of research in SPM. The ranking of areas is given as the fourth column of the table.

There are differences between the rankings of the three perspectives. For example, as the most favourite research area, cost management was found as one of the practitioners' lowest concerning area. Although there are only ten areas to be ranked, as an attempt to formulate the alignment of viewpoints, the ranking correlation between each pair of two was calculated. The resulting values are shown in Table 8.

While the correlation between community rankings and education is low, the correlation between community and research is even negative. The highest correlation has resulted from areas' ranking (order) in SPM research and education. The reason may be that researchers and instructors share

TABLE 7. The ranking of PMBOK areas from different viewpoints.

Rank	SPM QA Community	SPM Education	SPM Analytics Research
1	Schedule Management	Scope Management	Cost Management
2	Resource Management	Quality Management	Risk Management
3	Scope Management	Schedule Management	Integration Management
4	Integration Management	Risk Management	Scope Management
5	Stakeholder Management	Resource Management	Quality Management
6	Communication Management	Cost Management	Schedule Management
7	Risk Management	Communication Management	Resource Management
8	Quality Management	Procurement Management	Stakeholder Management
9	Cost Management	Integration Management	Procurement Management
10	Procurement Management	Stakeholder Management	Communication Management

TABLE 8. Rank correlation among PMBOK knowledge areas.

Ranking Correlation	SPM QA Community	SPM Education	SPM Analytics Research
SPM QA Community	1	0.23	-0.06
SPM Education		1	0.33
SPM Analytics Research			1

the same communities and likely have common interests. Although the trend during the last five years is not available, a general consideration for SA researchers would be shifting the focus to match software project managers' real needs. This may also result in the tools, dashboards, and techniques to support them during the project.

VI. DISCUSSION OF LIMITATIONS AND FUTURE WORK

The dynamic changes and dramatic complexity increase in our world are mapped to software project knowledge that needs to be up-to-date. Practitioners tend to access and participate in Q&A sites like Stack Exchange, which can be a great source of information. Practitioners use these channels to ask inquiries and express their concerns. As a result, the accessible textual data can guide researchers to the actual needs of practitioners. In this study, we used a variety of ML methods as well as descriptive statistics to identify the primary needs of practitioners. According to our findings, schedule management is the most concerning area for software project managers, and planning is the most challenging phase for them.

There are several limitations to our study. We are limited to Canadian universities, which does not allow generalization to other countries. The analysis reveals a strong gap between what is taught in these courses and what is discussed in Stack Exchange. Similar investigations based on programs outside Canada would be needed to obtain a complete picture. Most

former work is limited to finding the areas of main concern. While SPM is of pivotal importance for software project success, there is almost no analysis in the literature that relates the teaching of SPM to its industrial needs. An additional limitation comes from the restricted time frame. Even though ten years is substantial, the crawling was finished in 2020 and might not cover the most recent trends.

The Q&A's from Stack Exchange is considered a proxy for industrial needs, which is a simplification of reality. Using the established framework of PMBOK helps at least to cover the whole range of responsibilities of SPM. Online blog posts, technical reports, and Q&A communities are the trusted sources for practitioners to access data [4]. Practitioners tend to share their concerns through Q&A communities and seek any advice and help. Therefore, these communities are good venues for accessing their actual daily needs. However, the results need to be synthesized with the findings from possible other communities and the application of knowledge elicitation techniques such as interviews, observations, and document retrieval. With all the given limitations, the interpretation of all the results should be more qualitative than quantitative.

The analysis of Q&A sites like Stack Exchange was based on formulated questions. However, not all difficulties faced by real-world SPM can be easily formulated as a question. One tentative example of that is Risk Management. Appearance and reasons for challenges in Risk Management are more intangible and more difficult to get an answer for. The survey conducted by Makahaube [29] has Risk Management as the highest priority, which contradicts the results shown in Figure 4.

Another avenue of future work would be to differentiate between different types of projects and practitioners. "One size does not fit all" also applies to SPM methods and techniques. The project domain, the size of the project, and its criticality might trigger individual knowledge needs. Our dataset has additional attributes for each question, such as the number of votes, the number of views, the author's reputation score, etc. One possible research question can be finding the factors contributing to a question getting answered in a more specific context.

Software engineering practices such as agile frameworks and open-source software development have been largely influential in the area of Software Project Management. Exploring their implications in this study would be part of future work.

APPENDIX A ANALYZED SPM-RELATED UNIVERSITY COURSES

In this appendix, the 11 courses evaluated for RQ3 are listed. For each course, the name of the university, department/program, and course along with the course level and the accessed link ([↗](#) icon) are provided.

1- *University of Toronto, Computer Science, CSC 301: Introduction to Software Engineering* - Undergraduate level [↗](#)

2- *University of Toronto, Electrical and Computer Engineering, CSC 444: Software Engineering* - Undergraduate level [↗](#)

3- *McGill University, Computer Science, COMP 361D1: Software Engineering Project* - Undergraduate level [↗](#)

4- *McGill University, Electrical and Computer Engineering, ECSE 326: Software Requirements Engineering* - Undergraduate level [↗](#)

5- *McGill University, Electrical and Computer Engineering, ECSE 428: Software Engineering Practice* - Undergraduate level [↗](#)

6- *McGill University, Electrical and Computer Engineering, ECSE 611: Software Analytics* - Graduate level [↗](#)

7- *University of British Columbia, Electrical and Computer Engineering, CPEN 421: Software Project Management* - Undergraduate level [↗](#)

8- *University of British Columbia, Electrical and Computer Engineering, CPEN 523: Software Project Management* - Graduate level [↗](#)

9- *Université de Montréal, Polytechnique Engineering, LOG 3000: Software Engineering Process* - Undergraduate level [↗](#)

10- *Université de Montréal, Polytechnique Engineering, LOG 1000: Software Engineering* - Undergraduate level [↗](#)

11- *Université de Montréal, Polytechnique Engineering, LOG 8371: Software Quality Engineering* - Graduate level [↗](#)

12- *University of Alberta, Computer Science, CMPUT 401: Software Process and Product Management* - Undergraduate level [↗](#)

13- *University of Alberta, Computer Science, CMPUT 402: Software Quality* - Undergraduate level [↗](#)

14- *University of Alberta, Electrical and Computer Engineering, ECE 321: Software Requirements Engineering* - Undergraduate level [↗](#)

15- *University of Alberta, Electrical and Computer Engineering, ECE 322: Software Testing and Maintenance Engineering* - Undergraduate level [↗](#)

16- *McMaster University, Software Engineering, SFWRENG 3I03: Communication Skills* - Undergraduate level [↗](#)

17- *McMaster University, Software Engineering, SFWRENG 3K04: Software Development* - Undergraduate level [↗](#)

18- *McMaster University, Software Engineering, SFWRENG 3RA3: Software Requirements And Security Considerations* - Undergraduate level [↗](#)

19- *McMaster University, Software Engineering, SFWRENG 3XA3: Software Engineering Practice & Experience: Software Project Management* - Undergraduate level [↗](#)

20- *McMaster University, Computer Science, COMPSCI 3EA3: Software Specifications and Correctness* - Undergraduate level [↗](#)

21- *University of Waterloo, Computer Science, CS 430: Applications Software Engineering* - Undergraduate level [↗](#)

22- *University of Waterloo, Computer Science & Computer Engineering & Software Engineering, CS 445 - ECE 451 - SE 463: Software Requirements Specification and Analysis* - Undergraduate level [↗](#)

23- *University of Waterloo, Computer Science & Computer Engineering & Software Engineering, CS 447 - ECE 453 - SE 465: Software Testing, Quality Assurance and Maintenance* - Undergraduate level [↗](#)

24- *University of Waterloo, Computer Science, CS 645: Software Requirement Specification & Analysis* - Graduate level [↗](#)

25- *University of Waterloo, Computer Science & Computer Engineering, CS 647 - ECE 653: Software Testing, Quality Assurance and Maintenance* - Graduate level [↗](#)

26- *University of Waterloo, Software Engineering, SE 101: Introduction to Methods of Software Engineering* - Undergraduate level [↗](#)

27- *University of Waterloo, Software Engineering, SE 390: Design Project Planning* - Undergraduate level [↗](#)

28- *Western University, Computer Science, CS 3377: Software Project Management* - Undergraduate level [↗](#)

29- *Western University, Computer Science, CS 4472A: Specification, Testing and Quality Assurance* - Undergraduate level [↗](#)

30- *Western University, Software Engineering, SE 3351A/B: Software Project & Process Management* - Undergraduate level [↗](#)

31- *Western University, Software Engineering, SE 3352A/B: Software Requirement and Analysis* - Undergraduate level [↗](#)

32- *Queen's University, Computer Science & Computer Engineering, CISC-SOFT 423: Software Requirements* - Undergraduate level [↗](#)

33- *Queen's University, Computer Science & Computer Engineering, CISC-CMPE 327: Software Quality Assurance* - Undergraduate level [↗](#)

34- *Queen's University, Computer Science & Computer Engineering, CISC-CMPE 320: Fundamentals of Software Development* - Undergraduate level [↗](#)

35- *University of Calgary, Computer Science & Computer Engineering, SENG 511: Software Process and Project Management* - Undergraduate level [↗](#)

36- *University of Calgary, Computer Science & Computer Engineering, SENG 471: Software Requirements Engineering* - Undergraduate level [↗](#)

37- *University of Calgary, Computer Science & Computer Engineering, SENG 300: Introduction to Software Engineering* - Undergraduate level [↗](#)

38- *University of Calgary, Computer Science & Computer Engineering, SENG 607.8: Special Topics in Software Engineering (Analytical Software Project Management)* - Graduate level [↗](#)

39- University of Ottawa, Computer Engineering, **GNG 2101: Introduction to Product Development and Management for Engineers and Computer Scientists** - Undergraduate level [↗](#)

40- University of Ottawa, Software Engineering, **SEG 2105: Introduction to Software Engineering** - Undergraduate level [↗](#)

41- University of Ottawa, Software Engineering, **SEG 3103: Software Quality Assurance** - Undergraduate level [↗](#)

42- University of Ottawa, Software Engineering, **SEG 3101: Software Requirements Analysis** - Undergraduate level [↗](#)

43- University of Ottawa, Software Engineering, **SEG 4105: Software Project Management** - Undergraduate level [↗](#)

44- University of Ottawa, Computer Engineering, **ELG 5100: Software Engineering Project Management** - Undergraduate level [↗](#)

45- University of Ottawa, Computer Science, **CSI 5111 Software Quality Engineering** - Undergraduate level [↗](#)

46- University of Ottawa, Computer Science, **CSI 5112: Software Engineering** - Undergraduate level [↗](#)

REFERENCES

- [1] G. Ruhe and C. Wohlin, *Software Project Management in a Changing World*, vol. 21. Berlin, Germany: Springer, 2014.
- [2] H. Krasner, "The cost of poor software quality in the US: A 2020 report," in *Proc. Consortium Inf. Softw. Quality (CISQTM)*, 2021, pp. 1–46.
- [3] I. Srba and M. Bielikova, "Askalot: Community question answering as a means for knowledge sharing in an educational organization," in *Proc. 18th ACM Conf. Companion Comput. Supported Cooperat. Work Social Comput.*, Feb. 2015, pp. 179–182.
- [4] V. Garousi, M. Felderer, M. V. Mäntylä, and A. Rainer, "Benefitting from the grey literature in software engineering research," in *Contemporary Empirical Methods in Software Engineering*. Cham, Switzerland: Springer, 2020, pp. 385–413.
- [5] J. Pereira, J. Varajão, and N. Takagi, "Evaluation of information systems project success—Insights from practitioners," *Inf. Syst. Manage.*, vol. 39, no. 2, pp. 138–155, 2021.
- [6] E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne, "Finding high-quality content in social media," in *Proc. Int. Conf. Web Search Web Data Mining*, 2008, pp. 183–194.
- [7] Z. Zhao, L. Zhang, X. He, and W. Ng, "Expert finding for question answering via graph regularized matrix completion," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 4, pp. 993–1004, Apr. 2015.
- [8] P. Berger, P. Hennig, T. Bocklisch, T. Herold, and C. Meinel, "A journey of bounty hunters: Analyzing the influence of reward systems on StackOverflow question response times," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell. (WI)*, Oct. 2016, pp. 644–649.
- [9] A. Baltadzhieva and G. Chrupala, "Predicting the quality of questions on stackoverflow," in *Proc. Int. Conf. Recent Adv. Natural Lang. Process.*, 2015, pp. 32–40.
- [10] V. Bhat, A. Gokhale, R. Jadhav, J. Pudipeddi, and L. Akoglu, "Min(e)d your tags: Analysis of question response time in stackoverflow," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2014, pp. 328–335.
- [11] M. Albonico, I. Malavolta, G. Pinto, E. Guzman, K. Chinnappan, and P. Lago, "Mining energy-related practices in robotics software," 2021, *arXiv:2103.13762*.
- [12] G. A. Wang, H. J. Wang, J. Li, A. S. Abrahams, and W. Fan, "An analytical framework for understanding knowledge-sharing processes in online Q&A communities," *ACM Trans. Manage. Inf. Syst.*, vol. 5, no. 4, pp. 1–31, 2014.
- [13] M. Asaduzzaman, A. S. Mashiyat, C. K. Roy, and K. A. Schneider, "Answering questions about unanswered questions of stack overflow," in *Proc. 10th Work. Conf. Mining Softw. Repositories (MSR)*, May 2013, pp. 97–100.
- [14] L. Wang, B. Wu, J. Yang, and S. Peng, "Personalized recommendation for new questions in community question answering," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2016, pp. 901–908.
- [15] M. Liu, X. Peng, Q. Jiang, A. Marcus, J. Yang, and W. Zhao, "Searching stackoverflow questions with multi-faceted categorization," in *Proc. 10th Asia-Pacific Symp. Internetware*, Sep. 2018, pp. 1–10.
- [16] Z. S. H. Abad, A. Shymka, S. Pant, A. Currie, and G. Ruhe, "What are practitioners asking about requirements engineering? An exploratory analysis of social Q&A sites," in *Proc. IEEE 24th Int. Requirements Eng. Conf. Workshops (REW)*, Sep. 2016, pp. 334–343.
- [17] P. S. Kumar, H. S. Behera, A. Kumari, J. Nayak, and B. Naik, "Advancement from neural networks to deep learning in software effort estimation: Perspective of two decades," *Comput. Sci. Rev.*, vol. 38, Nov. 2020, Art. no. 100288.
- [18] J. Wen, S. Li, Z. Lin, Y. Hu, and C. Huang, "Systematic literature review of machine learning based software development effort estimation models," *Inf. Softw. Technol.*, vol. 54, no. 1, pp. 41–59, 2012.
- [19] P. Pospieszny, B. Czarnacka-Chrobot, and A. Kobylinski, "An effective approach for software project effort and duration estimation with machine learning algorithms," *J. Syst. Softw.*, vol. 137, pp. 184–196, Mar. 2018.
- [20] D. Gobov and I. Huchenko, "Software requirements elicitation techniques selection method for the project scope management," in *Proc. ITPM*, 2021, pp. 1–10.
- [21] A. Ahmadi, "Question-and-answer community mining in software project management—A deep learning approach," M.S. thesis, Dept. Comput. Sci., Univ. Calgary, Calgary, AB, Canada, 2020.
- [22] A. F. de Araújo and R. M. Marcacini, "RE-BERT: Automatic extraction of software requirements from app reviews using BERT language model," in *Proc. 36th Annu. ACM Symp. Appl. Comput.*, Mar. 2021, pp. 1321–1327.
- [23] J. Gao, "Chinese sentiment classification model based on pre-trained BERT," in *Proc. 2nd Int. Conf. Comput., Inf. Process. Adv. Educ.*, 2021, pp. 1296–1300.
- [24] A. Bollin, E. Hochmüller, and R. T. Mittermeir, "Teaching software project management using simulations," in *Proc. 24th IEEE-CS Conf. Softw. Eng. Educ. Training (CSEET)*, May 2011, pp. 81–90.
- [25] C. M. Zapata J. and G. Awad-Aubad, "Requirements game: Teaching software project management," *CLEI Electron. J.*, vol. 10, no. 1, pp. 3–14, Jun. 2007.
- [26] P. Mandl-Striegnitz, A. Drappa, and H. Lichter, "Simulating software projects—An approach for teaching project management," in *Proc. INSPIRE III, Process Improvement Through Training Educ.*, 1998, pp. 87–98.
- [27] V. Garousi, G. Giray, E. Tuzun, C. Catal, and M. Felderer, "Closing the gap between software engineering education and industrial needs," *IEEE Softw.*, vol. 37, no. 2, pp. 68–77, Mar. 2020.
- [28] M. E. Maqsood and T. Javed, "Practicum in software project management: An endeavor to effective and pragmatic software project management education," in *Proc. the 6th Joint Meeting Eur. Softw. Eng. Conf. ACM SIGSOFT Symp. Found. Softw. Eng.*, Sep. 2007, pp. 471–480.
- [29] J. S. Makahaube, "Performance measure methodology of project management practices in local government agencies," Ph.D. dissertation, Univ. Texas El Paso, El Paso, TX, USA, 2020.
- [30] P. Mustaro and R. Rossi, "Project management principles applied in academic research projects," in *Proc. Informing Sci. Inf. Technol. Educ. Conf. Informing Science Institute*, 2013.
- [31] O. Zwikael, "The relative importance of the PMBOK guide's nine knowledge areas during project planning," *Project Manage. J.*, vol. 40, no. 4, pp. 94–103, Dec. 2009.
- [32] J. Eastham, D. J. Tucker, S. Varma, and S. M. Sutton, "PLM software selection model for project management using hierarchical decision modeling with criteria from PMBOK knowledge areas," *Eng. Manage. J.*, vol. 26, no. 3, pp. 13–24, Sep. 2014.
- [33] V. Mahajan, "The Delphi method: Techniques and applications," *J. Marketing Res.*, vol. 13, no. 000003, p. 317, 1976.
- [34] D. Nevo and Y. E. Chan, "A Delphi study of knowledge management systems: Scope and requirements," *Inf. Manage.*, vol. 44, no. 6, pp. 583–597, Sep. 2007.

- [35] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
- [36] J. H. Lau and T. Baldwin, "An empirical evaluation of doc2vec with practical insights into document embedding generation," 2016, *arXiv:1607.05368*.
- [37] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, Nov. 2020.
- [38] *QS World University Rankings*. Accessed: Sep. 3, 2020. [Online]. Available: <https://www.topuniversities.com/university-rankings/world-university-rankings/2021>
- [39] M. Nayebi, G. Ruhe, R. C. Mota, and M. Mufti, "Analytics for software project management—Where are we and where do we go?" in *Proc. 30th IEEE/ACM Int. Conf. Automated Softw. Eng. Workshop (ASEW)*, Nov. 2015, pp. 18–21.



ALIREZA AHMADI received the master's degree from the University of Calgary, with a focus on data analysis in project management area, in 2020. He is a Project Manager in industry. His research interests include multiple areas of project management and specifying real need of industry practitioners.



FATEMEH DELKHOSH received the B.Sc. and M.Sc. degrees in industrial engineering from the Iran University of Science and Technology. She is currently pursuing the Ph.D. degree in business technology management with the Haskayne School of Business, University of Calgary, Canada. Her research interests include economics of blockchain, with a particular interest in how to manage and design blockchain-based platforms, economics of information systems using analytical modeling, and econometric methods.



GOURI DESHPANDE received the Ph.D. degree in computer science from the University of Calgary. She worked as a Software Engineer in IT industry for over a decade before switching to academia. She is currently an Assistant Professor with the Department of Electrical and Software Engineering, University of Calgary.



RAYMOND A. PATTERSON received the B.S.B.A. degree (Hons.) (magna cum laude) in accounting and management information systems (MIS) and the MBA degree in industrial marketing and finance from Bowling Green State University, Bowling Green, OH, USA, and the Ph.D. degree in accounting and MIS in cognitive psychology and microeconomics from The Ohio State University, Columbus, OH. He is currently a Professor of business technology management (BTM), holds the Haskayne Research Professorship in BTM, and works as the BTM Area Chair at the Haskayne School of Business, University of Calgary, Canada. He formerly held the Alberta School of Business Fellow and the Canada Research Chair in management information systems (MIS) at the University of Alberta. He has taught at the University of Calgary, University of Alberta, University of Texas at Dallas, The Ohio State University, and Bowling Green State University at the undergraduate, master's, doctoral, and executive education levels. He has over 25 years of experience collaborating with organizations in several industries, including accounting, healthcare, transportation, telecommunications, resource management, and government. He has worked Full-time as a Staff-Auditor at Ernst and Young at Toledo, OH, and as an Accounting and a IS Supervisor in the construction industry. His experience includes a wide range of management areas. He has published extensively in premier journals such as, *MIS Quarterly*, *Information Systems Research*, *Journal of Management Information Systems*, *Operations Research*, *Production and Operations Management*, *Decision Sciences Journal*, *Decision Support Systems*, *European Journal of Operational Research*, *International Journal of Production Economics*, *ACM-Transactions on Management Information Systems*, and *Geographic Information Science*. His research interests include information systems, analytics, and quantitative decision, and artificial intelligence technologies. He served as the Co-Editor-in-Chief for the *JOURNAL INFORMATION TECHNOLOGY and MANAGEMENT* from 2013 to 2019, and serves as an Associate Editor for *Decision Sciences Journal*.



GUENTHER RUHE (Senior Member, IEEE) is the Industrial Research Chair in software engineering with the University of Calgary, this is a joint position between the Department of Computer Science and the Department of Electrical and Software Engineering at the University of Calgary. Before coming to Calgary, he was the Deputy Director at the Fraunhofer Institute for Experimental Software Engineering Fh IESE, Germany. His main research interests include product release planning, software project management, decision support, open innovation, data analytics, empirical software engineering, and search-based software engineering. He is a member of the ACM. He is the Founder and a CEO of Expert Decisions Inc., which is a University of Calgary spin-off company. He won various awards such as, the Alberta iCORE (Informatics Center of Research Excellence) Award, the IBM Research Award, and the Microsoft Research Software Engineering Innovation Award. Since 2016, he has served as an Editor-in-Chief for the *journal of Information and Software Technology*, published by Elsevier.