Expert Systems With Applications 118 (2019) 231–245

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

T-shaped grouping: Expert finding models to agile software teams retrieval

Peyman Rostami*, Mahmood Neshati

Faculty of Computer Science and Engineering, Shahid Beheshti University, G.C, Iran

ARTICLE INFO

Article history:
Received 16 August 2018
Revised 6 October 2018
Accepted 9 October 2018
Available online 9 October 2018

Keywords:
Information retrieval
Expert finding
Agile methodologies
Team formation
T-shaped experts
StackOverflow

ABSTRACT

Agile team formation is an important requirement of software companies. Since the members of an agile team should be generalizing specialists (i.e. T-shaped experts), we need members who are specialist in a specific topic and have general knowledge in other topics of the team. Selecting such members results in an ideal team which is flexible, high-performing and low-cost. In this paper, we define the problem of *agile team formation* in which given a set of required skills of an agile team, the ideal output is a set of low-cost candidates who can collectively cover the required skills while they can effectively communicate with each other. We propose two retrieval models to address this problem and then we introduce three evaluation measures for assessment. These measures are coverage, communication and optimality. Our experiments on two test collections extracted from StackOverflow demonstrate the efficiency of our proposed models in comparison with several strong baselines.

1. Introduction

In recent years, agile methodologies have been used as a successful approach in software projects. These methodologies have many advantages in comparison with traditional life cycle models such as waterfall and spiral. For instance, agile methodologies primarily focus on the operational software instead of heavy documentations. Further, they are well known for flexibility against required changes in software life cycle (Stoica, Mircea, & Ghilic-Micu, 2013).

In addition to software development techniques, one of the most important characteristics of agile methodologies is their viewpoint to human resource management in software development teams. Specifically, in agile teams communication between members is managed to be in a flat structure (Danait, 2005). In other words, all members of an agile team are at the same level and collectively work with each other to perform the tasks in the backlog of the project. This approach is against of the traditional project management in which a team is consisted of some senior members and some junior ones as well. Considering the flat structure of agile teams, previous studies indicate that *generalizing specialist* members are more suitable to fill the required roles of agile teams (Ambler, 2012).

Generalizing specialists are team members with advanced knowledge in one skill-area (i.e. topic)[1] and general knowledge in several related skill-areas (Kulak & Li, 2017). For example, suppose a team with three main roles of a DBA (i.e. database administrator), a back-end developer and a front-end developer. We can assign the candidate C_1 (professional in DBA related skills and with general knowledge in two other mentioned fields), candidate C_2 (professional in back-end development skills and with general knowledge in two other mentioned fields) and candidate C_3 (professional in front-end development skills and with general knowledge in two other mentioned fields) to the required roles of the team. This team has the following characteristics:

1. Each role of the team is filled by a candidate who has advanced knowledge in the required professional skill-area of that role. In other words, each required skill-area is covered by exactly one team member (*coverage condition*).
2. Due to general knowledge of the members, they can support and cover each other and also communicate effectively in meetings (*communication condition*).
3. The cost of employment of such members is lower than the employment of full-stack developers[2] (*optimality condition*).

The mentioned team is a flexible, high-performing and low-cost team which is the ideal one recommended by agile methodologies.

* Corresponding author.
E-mail addresses: p.rostami@mail.sbu.ac.ir (P. Rostami), m_neshati@sbu.ac.ir (M. Neshati).

[1] Topic, skill-area and skill will be used exchangeably throughout the paper.
[2] By full-stack developer, we mean a candidate who has advanced knowledge in more than one skill-area.

https://doi.org/10.1016/j.eswa.2018.10.015

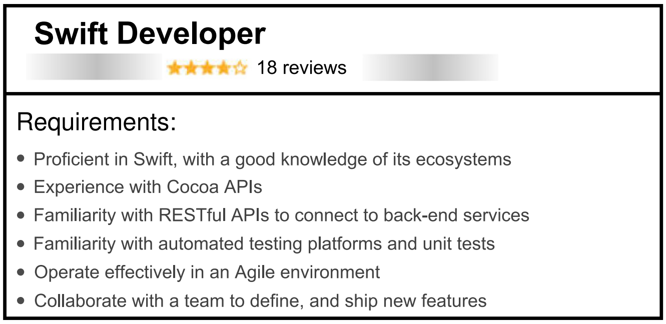

Fig. 1. A sample employment ad which is looking for a generalizing specialist for an agile team.

The job posting websites such as *Indeed*[3] are full of employment ads which are looking for generalizing specialists to form agile teams. For example, consider the ad displayed in Fig. 1. This ad indicates the requirement of a sample company for recruitment of an agile team member. This ad has the following important points:

1. It is looking for a candidate who is professional (i.e. specialist) in a single topic (i.e. Swift).
2. In addition, the candidate should have general knowledge (i.e. should be generalist) in several topics (i.e. Cocoa APIs, RESTful APIs and Software Testing).
3. The candidate should be able to communicate effectively with other team members.

Expert finding is a well studied topic in information retrieval (IR). In this area, the goal is to find people who are knowledgeable in a given topic. It can be formulated as to estimate the probability of $P(e|t)$ where e is a candidate and t is an expertise topic. In contrast, in our problem, we are going to find *generalizing specialist* members to form an agile team. Here, each member can be selected by the probability of $P\left(e \middle| t_{specialist}, \underbrace{t_1, t_2, \ldots, t_n}_{\text{generaling topics}}\right)$. The agile team formation is a novel problem with industrial motivation and to the best of our knowledge is not studied in previous research.

The main contributions of this paper are as follows:

1. We introduce the problem of agile team formation in which a set of skill-areas is given as input and the ideal output is a team of generalizing specialists.
2. We propose two models to solve the problem and compare these with several baselines on two real test collections extracted from StackOverflow[4].
3. We propose the evaluation measures of an agile team based on the coverage, communication and optimality conditions.

The rest of the paper is organized as follows. Shapes of expertise are explained in Section 2. Then, in Section 3, we discuss related work. In Section 4, we formally define the agile team formation problem and then introduce the baselines and propose two retrieval models for solving it. Next, in Section 5, we present three evaluation measures for the problem assessment. Experimental setup and results is described in Section 6. Finally, Section 7 concludes the paper and proposes future works.

2. Shapes of expertise

Previous research proposed several types of shapes of expertise for people in software development community (Donofrio, Sanchez, & Spohrer, 2010; Gharebagh, Rostami, & Neshati, 2018; Kumar & Pedanekar, 2016). Each of these shapes has a specific characteristic which is described as follows:

- *Non-expert* people: These people who do not have enough experience in any skill-area are named as Non-expert. It is necessary to educate these people to enhance their experience and knowledge in some skill-areas.
- *I-shaped* experts (i.e. specialists): These people are merely expert in a single skill-area and do not have any knowledge of other skill-areas. Fresh graduates from the universities are usually in this category (Donofrio, Spohrer, & Zadeh, 2010).
- *T-shaped* experts (i.e. generalizing specialists): The I-shaped experts can usually be converted to T-shaped experts by gaining experience in their working carriers. These people have advanced knowledge in a single skill-area (leg of T) while they have general knowledge in several related skill-areas (hat of T).
- *C-shaped*[5] experts (i.e. full-stack developers): A person with several years of experience in design, development and deployment of software project usually has advanced knowledge in several skill-areas. These people are called C-shaped.

Table 1 indicates the number and the level of skill-areas associated with each shape of expertise. Our definition of these shapes of expertise is borrowed from our previous work (Gharebagh et al., 2018). According to this definition, as an example, a T-shaped expert has advanced knowledge in one skill-area and intermediate knowledge (i.e. general knowledge) in one or more skill-areas.

3. Related work

Previous research related to our work can be divided into three parts: *Expert Finding, Shapes of Expertise* and *Team Formation* which are explained in the following sub-sections. Fig. 2 indicates these branches and their sub-topics. Specifically, one line of related research is about expert finding which is a well-known IR problem. The related work of this problem is divided into two sub-categories (i.e. CQA and non-CQA) according to the context of *expert finding*. Our problem is investigated on StackOverflow which is a question answering website. Therefore, our emphasis in this section will be on CQA branch. The second part of the related work is devoted to *shapes of expertise*. Finally, *team formation* is an important aspect of the related work which is reviewed in two categories (i.e. single aspect query and multi-aspect query). Here, we focus on multi-aspect query problem which is more relevant to our work.

3.1. Expert finding

Expert finding task is concerned with the retrieval of knowledgeable people in a given topic (Balog et al., 2012). In the past few years, expert finding problem attracted a lot of attention in the IR community. It has been studied in many environments such as organizations (Balog, Azzopardi, & de Rijke, 2009; Petkova & Croft, 2008), universities (Balog, Bogers, Azzopardi, De Rijke, & Van Den Bosch, 2007), bibliographic networks (Hashemi, Neshati, & Beigy, 2013; Neshati, Hashemi, & Beigy, 2014b), social networks (Neshati, Hiemstra, Asgari, & Beigy, 2014c; Smirnova, 2011) and CQAs (Dargahi Nobari, Sotudeh Gharebagh, & Neshati, 2017; Gharebagh et al., 2018).

Our work focuses on finding experts on StackOverflow, which is a well-known CQA. Several researchers have addressed the problem of expert finding in CQAs. These researchers focused on finding experts in order to: 1. Route newly asked questions with the objective of providing users with high-quality answers within a

[3] https://www.indeed.com/.
[4] https://www.stackoverflow.com.
[5] Comb-shaped.

Table 1
Comparing different shapes of expertise according to the number and the level of skill-areas.

Shape	The number of skill-areas		
	Advanced knowledge	Intermediate knowledge	Beginner knowledge
Non-expert	0	≥ 0	≥ 0
I-shaped	1	0	≥ 0
T-shaped	1	≥ 1	≥ 0
C-shaped	≥ 2	≥ 0	≥ 0

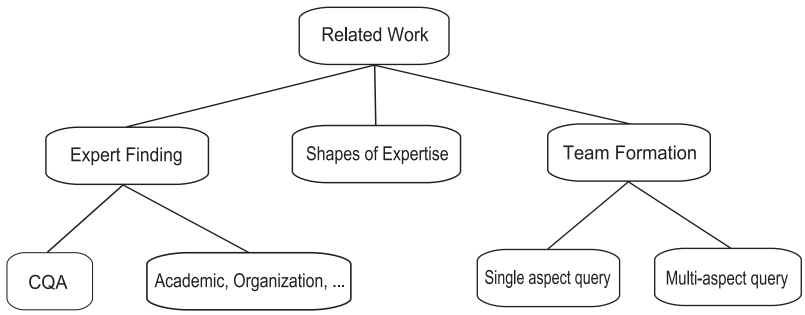

Fig. 2. Categorization of the related work.

reasonable time (Li, Jin, & Shudong, 2015; Riahi, Zolaktaf, Shafiei, & Milios, 2012), and 2. Help recruiters to hire suitable candidates for a job position (Gharebagh et al., 2018; Neshati, Fallahnejad, & Beigy, 2017). Riahi et al. (2012) focused on finding experts for a newly posted question. They investigated the suitability of two statistical topic models (Latent Dirichlet Allocation model and Segmented Topic model) for solving this issue and compare these models against more traditional information retrieval approaches (TF-IDF and language model). Li, Jin, et al. (2015) proposed a tag-LDA model to determine the user topic distribution and predicts the topic distribution of new questions. They considered user post contents, answer votes, ratio of best answers and user relations to find an appropriate user to answer a new question. Liu, Chen, Kao, and Wang (2013) proposed a hybrid approach to effectively find experts in the category of the target question in question answering websites. Their approach considered user subject relevance, user reputation, and authority of a category in finding experts. Zhu, Chen, Xiong, Cao, and Tian (2014) proposed an expert finding framework based on the authority information in the target category as well as the relevant categories. In addition, They developed a scalable method for measuring the relevancy between categories through topic models which takes consideration of both content and user interaction based category similarities. Also, they provided a topical link analysis approach, which is multiple category-sensitive, for ranking user authority by considering the information in both the target category and the relevant categories. Zhao, Zhang, He, and Ng (2015) considered the problem of expert finding from the viewpoint of missing value estimation. They presented a novel method called graph regularized matrix completion for inferring the user model. Furthermore, their approach integrated both the social relation of users and their past question-answering activities seamlessly into a common framework for this problem. Zhao, Yang, Cai, He, and Zhuang (2016) formulated the problem of expert finding from the viewpoint of learning ranking metric embedding. They proposed a novel ranking metric network learning framework for the expert finding by exploiting both usersâ;; relative quality rank to given questions and their social relations that tackle both the insufficiency of question representation and the sparsity of CQA data issue. Then they developed a random-walk based learning method with recurrent neural networks for ranking metric network embedding. Neshati et al. (2017) introduced the problem of future expert finding which focuses on the ranking of experts in future while the expertise evidence is observed at the current time. They proposed a supervised learning framework to predict such ranking on StackOverflow website. They examined the impact of four groups of features (i.e. topic similarity, emerging topics, user behavior and topic transition) to predict the probability of becoming an expert user in future time. Dargahi Nobari et al. (2017) proposed two translation models to augment a given query with relevant words for expert finding. The first model (Mutual Information-Based approach) is based on a statistical approach and the second one is a word embedding model. Gharebagh et al. (2018) proposed two models to find a specific type of experts called T-shaped users who were introduced in Section 2. They estimated the profile diversity of users in their models to detect those who have the feature of T-shaped people in CQAs.

In comparison with previous research in this area, our work has two main differences: First, in our work, we are looking for T-shaped experts while most of the previous works do not consider the shape of expertise in expert finding problem. It is worth mentioning that our work is an extension of our previously published work in ECIR 2018 (Gharebagh et al., 2018). Second, in contrast with previous research in this area, in our work, the input is a set of queries and the output is a group of experts, while in the previous works, the input is a given query and output is a ranked list of experts.

3.2. Shapes of expertise

Researchers introduced several types of shapes, namely I-shaped, π-shaped, T-shaped, Comb-shaped and Hyphen-shaped to demonstrate expertise and skills of a person. In this shapes, each leg (i.e. I (one leg), π (two legs), T (one leg), Comb (several legs) and Hyphen (no leg)) indicates the deep specialty knowledge of a candidate in one skill-area and breadth of each shape shows the general knowledge of a candidate in some skill-area (Demirkan & Spohrer, 2015). Kumar and Pedanekar (2016) introduced the idea of mining shapes of user expertise in a CQA. They found that expertise in CQA forums often involves making contribution in a variety of areas of expertise rather than a single area. They defined different expertise shapes including I-shaped, T-shaped, C-shaped, and Hyphen-shaped. Gharebagh et al. (2018) also labeled users in CQAs based on shapes of expertise. They first defined knowledge level of users in each skill-area, and then categorized users to Non-expert, T-shaped, and C-shaped based on their knowledge levels in all skill-areas.

As mentioned before, T-shaped experts are people who have deep specialty in one skill-area and general knowledge in many skill-areas. The earliest allusion to T-shaped people was coined by Guest (1991) in a 1991 London newspaper editorial. Iansiti (1999) further popularized the T-shaped concept. He suggests that the members of a R&D group should be T-shaped to successfully integrate with each other. Tim Brown, CEO of IDEO in Brown (2005) expressed that top organizations should look for people who are so inquisitive about the world that they are willing to try to do what you do. He called them 'T-shaped people' who have a principal skill that describes the vertical leg of the T, but they are so empathetic that can branch out into other skills. So, they should be able to explore insights from many different perspectives and recognize patterns of behavior that point to a universal human need.

More recently, with the advent of agile methodologies, a need has emerged for finding T-shaped people who can work interdisciplinarily in software project teams according to the manifesto of the agile software development (Rubin, 2012). Gharebagh et al. (2018) proposed two retrieval models to find and then rank T-shaped users for agile software teams. They estimated the profile diversity of users in their models to detect those who have the feature of T-shaped people in CQAs.

3.3. Team formation

The task of group finding or team formation has recently received increased attention. As a natural and challenging extension of expert finding, here, the goal is to find a set of candidates. The related works in this area can be divided into two categories:

1. Given a single aspect query, the problem is to find knowledgeable groups that have expertise on that query (Liang & De Rijke, 2013; Liang & de Rijke, 2016).
2. Given a multi-aspect query, the problem is to find knowledgeable groups that can collectively cover the required aspects of that query (Karimzadehgan, Zhai, & Belford, 2008; Neshati, Beigy, & Hiemstra, 2014a).

Our work is more related to the second category. So, in the following, we explained the related works in this category. Karimzadehgan et al. (2008) solved a multi-aspect expertise matching for review assignment problem which can be considered as a team formation problem. They matched reviewers to a paper with considering multiple aspects of papers and multiple expertise of reviewers so that the assigned reviewers would not only have the required expertise to review a paper but can also cover all the aspects of a paper in a complementary manner. They proposed three general strategies for solving this problem, including removing redundancy, modeling aspects based on reviewer expertise, and modeling aspects based on the paper to be reviewed. Karimzadehgan and Zhai (2012) tackled the problem of committee review assignment with multi-aspect expertise matching by casting it as an integer linear programming problem. Neshati et al. (2014a) proposed an optimization framework based on facility location analysis to retrieve an optimal group of experts to perform a multi-aspect task so that the group of assigned experts should be able to collectively cover all required skills of the group. Kargar and An (2011) studied the problem of discovering a team of experts from a social network. Given a project whose completion requires a set of skills, their goal was to find a team of experts with/without a leader that together have all of the required skills and also have the minimal communication cost among them. They proposed two communication cost functions designed for the two types of communication structures (with/without a leader). Li, Shan, and Lin (2015) studied the problem of finding teams of experts in social networks for a generalized task consisting of a set of required skills, so that the members should not only satisfy the requirements of the given task but also be able to communicate with one another in an effective manner. To compose an effective team of experts, they modified the Enhanced–Steiner algorithm to deal with generalized tasks. They proposed a grouping-based approach with the role composition algorithm to further boost the effectiveness of the constructed teams and boost the efficiency in the team formation process.

In contrast with related works in this area, our work is focused on finding T-shaped group which to the best of our knowledge is the first solution to find agile teams.

4. Agile team formation models

In this section, we first formally define the problem of agile team formation in Section 4.1 and then we propose several retrieval models to solve it. All of the models are based on the greedy selection approach which is described in Section 4.2. In Sections 4.3, 4.4 and 4.5, we introduce the baseline models and then we propose our solutions for the problem in Sections 4.6 and 4.7.

4.1. Problem specifications

The input of *agile team formation* problem is a set of n skill-areas ($S : \{sa_1, sa_2, \ldots, sa_n\}$) and the output is a team of n members in which each member ideally is a T-shaped expert who has advanced knowledge in one skill-area of S and has intermediate knowledge in other skill-areas of S.

- **Assumption 1.** The team members should be selected from a given pool of candidates. For each candidate e there is a set of associated documents which is represented by $D_e = D_{sa_1,e} \cup D_{sa_2,e} \cup \ldots \cup D_{sa_k,e}$, where k is the number of skill-areas in the whole collection. Here, $D_{sa_i,e}$ indicates the documents related to sa_i which are associated with candidate e. In other words, the documents associated with candidate e are categorized according to their subject. Here, we assume that the subject of each document is explicitly given.
- **Assumption 2.** Each selected member of the team is responsible to cover one single skill-area (associated to one specific role of the team). In other words, ith member of the team is responsible to cover the ith skill-area (i.e. ith member should have advanced knowledge in the ith skill-area.).
- **Assumption 3.** In an ideal team, a member who is responsible to cover the ith skill-area should have intermediate knowledge in all other skill-areas in the team. In this way, each member of the team can effectively communicate with other members. By this assumption, we prefer a T-shaped expert rather than an I-shaped expert for each position.
- **Assumption 4.** The cost of assignment of a C-shaped candidate to a role of the team is higher than the cost of assignment of a T-shaped one to that role. By this assumption, we prefer a T-shaped expert rather than a C-shaped expert for each position.

4.2. The greedy approach for member selection

As mentioned before in the problem specifications section (assumption 2), each team member is responsible to cover a role in the team. In all models described in this paper, we follow the greedy approach in which in ith iteration of the algorithm, we select a candidate to cover the ith role of the team. According to the assumptions of agile team formation problem, the best candidate to cover the ith role of the team is a candidate e who maximizes the following probability:

$$Best\,Candidate(i) = \operatorname*{Arg\,Max}_{e} P(e|sa_i, set_i), \tag{1}$$

in which sa_i is the skill-area which should be covered by the ith candidate, and set_i is the set of all skill-areas of the team except the ith skill-area. The candidate should have intermediate knowledge in all skill-areas of set_i.

According to the assumptions of the problem, the best candidate in ith iteration is the one who is T-shaped expert with advanced knowledge in the ith skill-area and the intermediate knowledge in other skill-areas of the team.

Consider the following events. Each event is represented by a random variable in parenthesis.

- **Event 1** (e): "The selection of candidate e in ith iteration of the algorithm".
- **Event 2** ($T = 1$): "The ith team member is a T-shaped expert".
- **Event 3** (sa_{adv}): "The ith member has advanced knowledge in sa_i".
- **Event 4** (set_{int}): "The ith member has intermediate knowledge in set_i".

The joint probability of above events means that "e is a T-shaped candidate who has advanced knowledge in sa_{adv} and has intermediate knowledge in set_{int} and is selected in the ith iteration" which is represented as follows:

$$P(T = 1, e, sa_{adv}, set_{int}). \tag{2}$$

In the following sub-sections, we describe models to estimate the probability of Eq. (2).

4.3. Single expertise model (SEM)

In this model, instead of a T-shaped user selection, we select a candidate who is expert in sa_{adv}. In other words, we assume the following simplification in our estimation:

$$P(T = 1, e, sa_{adv}, set_{int}) \approx P(e, sa_{adv}). \tag{3}$$

This model simply solves the problem by the following assumptions:

- The selected candidate should not be necessarily a T-shaped expert.
- The selected candidate should not necessarily have intermediate knowledge in set_{int}.

To estimate the probability of $P(e, sa_{adv})$, we have:

$$P(e, sa_{adv}) = P(sa_{adv}) \cdot P(e|sa_{adv}) \overset{rank}{=} P(e|sa_{adv}). \tag{4}$$

In above equation, $P(e|sa_{adv})$ indicates the expertise probability of candidate e on skill-area sa_{adv} which can be estimated as follows:

$$P(e|sa_{adv}) = \lambda_d \frac{|D_{sa_{adv},e}|}{|D_{sa_{adv}}|} + (1 - \lambda_d) \frac{|D_{sa_{adv}}|}{|D|}, \tag{5}$$

in which the $D_{sa_{adv},e}$ indicates the set of associated documents of candidate e which is related to skill-area sa_{adv}. $D_{sa_{adv}}$ indicates the whole set of documents related to skill-area sa_{adv}, and D is the set of whole documents in the collection. λ_d is the smoothing parameter which is between 0 and 1. If $\lambda_d \to 1$, it will promote users who have more relevant documents in a specific (i.e. rare) skill-area. On the other hands, if $\lambda_d \to 0$, it enhances the score of candidates who have documents in a general skill-area. To sum up, the above equation has an effect similar to IDF in ad hoc retrieval models.

4.4. Multi expertise model (MEM)

In this model, instead of a T-shaped user selection, we select a candidate who is expert in both sa_{adv} and all skill-areas in set_{int}. In other words, we assume the following simplification in our estimation:

$$P(T = 1, e, sa_{adv}, set_{int}) \approx P(e, sa_{adv}, set_{int}). \tag{6}$$

To estimate above probability, we have:

$$\begin{aligned} P(e, sa_{adv}, set_{int}) &= P(sa_{adv}) \cdot P(e|sa_{adv}) \cdot P(set_{int}|e, sa_{adv}) \\ &\approx P(sa_{adv}) \cdot P(e|sa_{adv}) \cdot P(set_{int}|e) \\ &\overset{rank}{=} P(e|sa_{adv}) \cdot P(set_{int}|e). \end{aligned} \tag{7}$$

In above equation, we assume the expertise of candidate e in set_{int} is independent of his expertise in sa_{adv}. The probability of $P(e|sa_{adv})$ can be estimated using Eq. (5). In order to estimate $P(set_{int}|e)$, we have:

$$\begin{aligned} P(set_{int}|e) &\approx \prod_{sa \in set_{int}} P(sa|e) \\ &= \prod_{sa \in set_{int}} \frac{P(sa)}{P(e)} \cdot P(e|sa) \\ &\overset{rank}{=} \prod_{sa \in set_{int}} P(e|sa). \end{aligned} \tag{8}$$

In above equation, we assume the expertise of candidate e in each skill-area $sa \in set_{int}$ is independent of other skill-areas in set_{int}. Furthermore, we assume $P(e)$ is uniform and does not affect ranking and $P(e|sa)$ can be estimated using Eq. (5).

To sum up, this model is looking for a candidate who has advanced knowledge in all required skill-areas in the team. Intuitively, we expect that this model performs better than the model proposed in Section 4.3 in terms of communication condition of agile teams, because it is more probable that all members have advanced or intermediate knowledge in all skill-areas of the team.

4.5. Entropy based model (EBM)

In this model, we are looking for a T-shaped candidate who is expert in sa_{adv}, but we ignore the set of skill-areas in which the candidate should have intermediate knowledge in them (i.e. set_{int}). Following the idea of T-shaped expert retrieval in Gharebagh et al. (2018), the simplification assumption is:

$$P(T = 1, e, sa_{adv}, set_{int}) \approx P(T = 1, e, sa_{adv}). \tag{9}$$

To estimate above probability, we have:

$$P(T = 1, e, sa_{adv}) = P(e) \cdot P(T = 1|e) \cdot P(sa_{adv}|T = 1, e), \tag{10}$$

where $P(e)$ is the prior selection probability of candidate e (assumed to be uniform), $P(T = 1|e)$ indicates the probability of candidate e being T-shaped and $P(sa_{adv}|T = 1, e)$ indicates that the candidate e has advanced knowledge in sa_{adv} assuming candidate e is a T-shaped expert. Probability of $P(T = 1|e)$ can be estimated by the following equation:

$$P(T = 1|e) \propto \frac{log|D_e|}{H(e)} \;\; ; \;\; |D_e| > 0, \tag{11}$$

in which D_e is the set of documents associated with candidate e and $H(e)$ indicates the entropy of candidate e. Intuitively, a high entropy candidate has high diversity in the number of his documents in several skill-areas and as a result, the chance of being T-shaped is decreased for him. We use Min-Max normalization for all candidates of the collection to adjust the $\frac{log|D_e|}{H(e)}$ between 0 and 1. In order to estimate $H(e)$, we have:

$$H(e) = -\sum_{i=1}^{k} P_{sa_i,e} \log P_{sa_i,e}. \tag{12}$$

In above equation, k is the number of skill-areas in the whole collection and $P_{sa_i,e}$ indicates the occurrence probability of skill-area sa_i in documents associated with candidate e. This probability is estimated as follows:

$$P_{sa_i,e} = \frac{|D_{sa_i,e}|}{|D_e|}. \quad (13)$$

The third part of Eq. (10) can be estimated by the following equation:

$$P(sa_{adv}|e, T = 1) \approx \frac{P_{sa_{adv},e}}{\underset{i}{Max}\,(P_{sa_i,e})}. \quad (14)$$

It is worth mentioning that $P(T = 1|e)$ in Eq. (10), estimates the probability of candidate e being a T-shaped expert, therefore, we expect that candidate e has advanced knowledge in a single skill-area sa. So, if sa equals to sa_{adv}, then the probability of Eq. (14) equals to one, otherwise this probability approaches to zero.

The model described in this section considers both the optimality and coverage conditions. In other words, this model promotes T-shaped users with advanced knowledge in sa_{adv}, but this model ignores the communication condition, because it does not consider the set_{int} in selection of candidates.

4.6. Extended entropy based model (XEBM)

The model proposed in Section 4.5, ignores the intermediate knowledge of the selected candidates. In this section, we extend the Entropy Based Model to consider set_{int} in candidate selection as follows:

$$P(T = 1, e, sa_{adv}, set_{int}) = P(T = 1, e, sa_{adv}) \cdot P(set_{int}|T = 1, e, sa_{adv}). \quad (15)$$

As indicated in above equation, the first part can be estimated by the Entropy Based Model (Eq. (10)). In order to estimate $P(set_{int}|T = 1, e, sa_{adv})$, we propose three methods in Sections 4.6.1–4.6.3.

4.6.1. Kernel based method

In this section, we propose a method which consider the knowledge value of candidates in skill-areas to estimate $P(set_{int}|T = 1, e, sa_{adv})$.

In order to estimate $P(set_{int}|T = 1, e, sa_{adv})$, we assume set_{int} is independent of sa_{adv} given $T = 1$ and e. So,

$$P(set_{int}|T = 1, e, sa_{adv}) \approx P(set_{int}|T = 1, e). \quad (16)$$

The above equation indicates the probability that candidate e has intermediate knowledge in set_{int} given "e is a T-shaped expert". Similar to the assumptions of Section 4.4, we assume having intermediate knowledge in $sa_{int} \in set_{int}$ is independent of other skill-areas in set_{int}. Therefore,

$$P(set_{int}|T = 1, e) = \prod_{sa_{int} \in set_{int}} P(sa_{int}|T = 1, e), \quad (17)$$

where $P(sa_{int}|T = 1, e)$ indicates the probability that candidate e has intermediate knowledge in skill-area sa_{int}. In order to estimate this probability, we consider the following ideas:

- **Idea 1.** For a given skill-area sa_{int}, if candidate e_1 has more documents than candidate e_2, then we expect that e_1 has more knowledge than e_2 on that skill-area sa_{int}.
- **Idea 2.** Let e_1, e_2 and e_3 be three candidates on skill-area sa_{int}. D_{sa_{int},e_1}, D_{sa_{int},e_2} and D_{sa_{int},e_3} indicate their associated documents on skill-area sa_{int}. If $D_{sa_{int},e_1} - D_{sa_{int},e_2} = 1$ and $D_{sa_{int},e_2} - D_{sa_{int},e_3} = 1$,then the difference of knowledge value of e_1 and e_2 is less than the difference of knowledge value of e_2 and e_3. Following the idea in Fang, Tao, and Zhai (2004), here, the intuition is that the changed in the knowledge value caused by increasing the associated documents of a candidate e from 1 to 2 should be larger than that caused by increasing the associated documents of a candidate e from 100 to 101.

Let $M_{sa_{int}}$ to be the average number of documents related to sa_{int} for each candidate. i.e. $M_{sa_{int}} = \underset{e'}{Avg}(|D_{sa_{int},e'}|)$.

- **Idea 3.** If $|D_{sa_{int},e}| >> M_{sa_{int}}$,then the probability of $P(sa_{int}|T = 1, e)$ should approach to zero. In this case, candidate e is likely to be an expert with advanced knowledge on sa_{int}. As a result, the probability of having intermediate knowledge should approach to zero.
- **Idea 4.** If $|D_{sa_{int},e}| << M_{sa_{int}}$,then the probability of $P(sa_{int}|T = 1, e)$ should approach to zero. In this case, candidate e is likely to be a beginner on sa_{int}. As a result, the probability of having intermediate knowledge should approach to zero.
- **Idea 5.** If $abs(|D_{sa_{int},e}| - M_{sa_{int}})$[6] $\to 0$, then the probability of $P(sa_{int}|T = 1, e)$ should approach to one. In this case, candidate e probably have intermediate knowledge on skill-area sa_{int}, because the number of his associated documents approach to the average.

According to ideas 1 and 2, we define a function which estimates the knowledge value of candidate e on skill-area sa_{int}. The function is defined as follows:

$$KV(sa_{int}, e) = \frac{\log(|D_{sa_{int},e}| + 1)}{2\log(M_{sa_{int}})}. \quad (18)$$

The above function is increasing in terms of $|D_{sa_{int},e}|$ (satisfies idea 1) and its derivative in terms of $|D_{sa_{int},e}|$ is decreasing (satisfies idea 2).

According to ideas 3 to 5, in order to transform the knowledge value of candidate e to probabilistic space, we need a function $0 \le F \le 1$ that satisfies the following conditions:

1. If $KV(sa_{int}, e) \le 1$ then
   (a) If $KV(sa_{int}, e) \to 0$ then $F(KV(sa_{int}, e)) \to 0$
   (b) If $KV(sa_{int}, e) \to 1$ then $F(KV(sa_{int}, e)) \to 0$
   (c) If $KV(sa_{int}, e) \to \frac{1}{2}$ then $F(KV(sa_{int}, e)) \to 1$
2. If $KV(sa_{int}, e) > 1$ then $F(KV(sa_{int}, e)) \to 0$

Fig. 3 indicates two functions which satisfy the above conditions. Function F_l is a linear and function F_q is a quadratic one. In these functions, we also consider the symmetry around $KV(sa_{int}, e) = 0.5$. In our experiments, we estimate the probability of having intermediate knowledge as follows:

$$P(sa_{int}|e, T = 1) \overset{linear}{=} F_l(KV(sa_{int}, e)) = \begin{cases} -|2KV(sa_{int}, e) - 1| + 1 & , \text{if } KV(sa_{int}, e) <= 1 \\ 0, & \text{otherwise} \end{cases} \quad (19)$$

$$P(sa_{int}|e, T = 1) \overset{quadratic}{=} F_q(KV(sa_{int}, e)) = \begin{cases} -(2KV(sa_{int}, e) - 1)^2 + 1, & \text{if } KV(sa_{int}, e) <= 1 \\ 0, & \text{otherwise} \end{cases} \quad (20)$$

To sum up, substituting Eq. (17) into Eq. (15), we can select a candidate according to XEBM-kernel based Model.

[6] The $abs(x)$ means the absolute value of x.

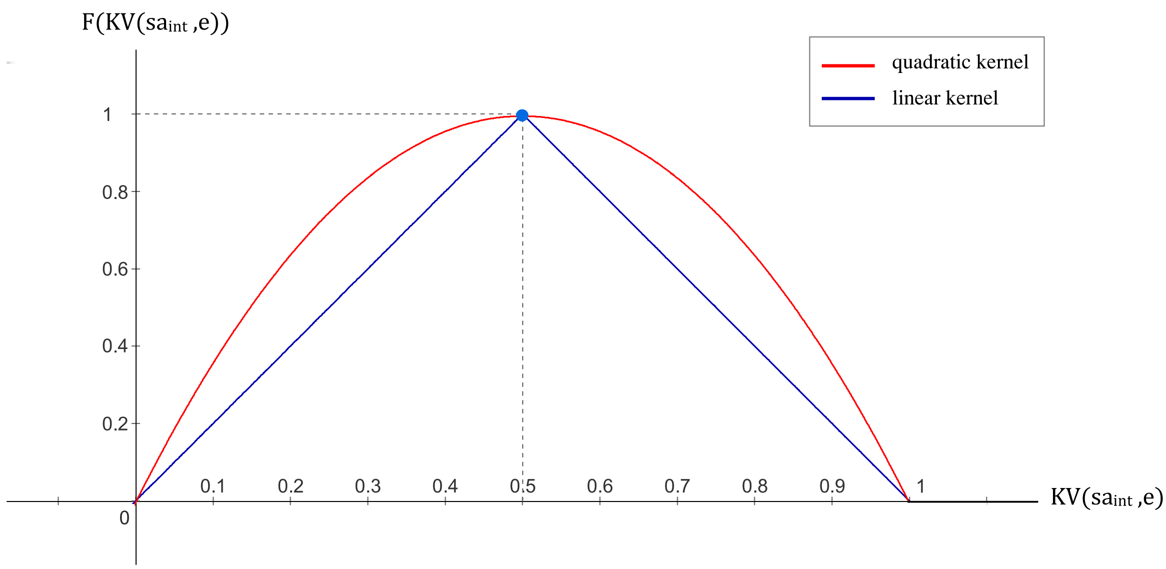

Fig. 3. Linear and quadratic kernels to estimate Intermediate knowledge score of candidates.

4.6.2. Overlap based method

Here, we propose another method to estimate the probability of $P(set_{int}|T=1,e,sa_{adv})$ using the following equation:

$$P(set_{int}|T=1,e,sa_{adv}) \approx \frac{|set_{int} \cap (set_e - \{sa_{adv}\})|}{|set_{int}|}, \tag{21}$$

in which set_e indicates the set of skill-areas which candidate e has at least one associated document on those skill-areas. In above equation, we assume that e is a T-shaped candidate and has advanced knowledge in sa_{adv}. As a result, we expect that there is no other skill-areas which candidate e has advanced knowledge in them. Therefore, we expect that candidate e has beginner or intermediate (not advanced) knowledge level in $(set_e - \{sa_{adv}\})$. The overlap of $(set_e - \{sa_{adv}\})$ and set_{int} is a good estimation of $P(set_{int}|T=1,e,sa_{adv})$.

To sum up, substituting Eq. (21) into Eq. (15), we can select a candidate according to XEBM-Overlap based Model.

4.6.3. Hybrid method

In Section 4.6.2, we explained that the set of $(set_e - \{sa_{adv}\})$ is expected to be the set of skill-areas which candidate e has beginner or intermediate knowledge in them. In order to precise the estimation of $P(set_{int}|T=1,e,sa_{adv})$, we use the combination of kernel and overlap based as follows:

$$P(set_{int}|T=1,e,sa_{adv}) \approx \frac{|set_{int} \cap (set_{Int(e)} - \{sa_{adv}\})|}{|set_{int}|}, \tag{22}$$

in which $set_{Int(e)}$ is defined as follows:

$$set_{Int(e)} = \{sa|sa \in set_e, F(KL(sa,e)) \geq \alpha\}. \tag{23}$$

By the definition of $set_{Int(e)}$, we only consider a skill-area sa as an intermediate one, if knowledge value of candidate e is around the central point of kernel function.

To sum up, substituting Eq. (22) into Eq. (15), we can select a candidate according to XEBM-Hybrid Model.

The model in this section considers the coverage, communication and optimality conditions. In other words, We expect that this model enhances the rank of T-shaped users with advanced knowledge in sa_{adv} and intermediate knowledge in set_{int}.

4.7. Relative deepness model (RDM)

In this section, we propose another model which considers all three conditions of agile teams. In order to estimate $P(T=1,e,sa_{adv},set_{int})$, we have:

$$P(T=1,e,sa_{adv},set_{int}) = P(T=1,e,sa_{adv}) \cdot P(set_{int}|T=1,e,sa_{adv}). \tag{24}$$

Probability of $P(set_{int}|T=1,e,sa_{adv})$ can be estimated using the methods proposed in Sections 4.6.1, 4.6.2 and 4.6.3. Here, we propose a new approach to estimate $P(T=1,e,sa_{adv})$. We have:

$$P(T=1,e,sa_{adv}) = P(e,sa_{adv}) \cdot P(T=1|e,sa_{adv}). \tag{25}$$

in which $P(e, sa_{adv})$ indicates whether candidate e has advanced knowledge in sa_{adv} or not:

$$P(e,sa_{adv}) \approx \begin{cases} 1 & \text{, if } sa_{adv} = \underset{sa}{ArgMax}(|D_{sa,e}|) \\ 0 & \text{, otherwise} \end{cases}. \tag{26}$$

The second part of Eq. (25) is $P(T=1|e,sa_{adv})$ which estimate the probability that candidate e is a T-shaped expert who has advanced knowledge in sa_{adv}. This probability is estimated as follows:

$$P(T=1|e,sa_{adv}) \approx \begin{cases} |set_{Int(e)}| \cdot RD(e,sa_{adv}) & \text{, if } P(e,sa_{adv}) = 1 \\ 0 & \text{, otherwise} \end{cases}, \tag{27}$$

where $set_{Int(e)}$ is calculated using Eq. (23) and indicates the probable set of intermediate level skill-areas of candidate e. $RD(e, sa_{adv})$ is a function which indicates the knowledge deepness of candidate e on skill-area sa_{adv} in comparison with all other his skill-areas,

$$RD(e,sa_{adv}) \approx \log|D_{sa_{adv},e}| - \max_{sa \in (set_e - \{sa_{adv}\})} (\log|D_{sa,e}|). \tag{28}$$

By the definition of Eq. (27), the $RD(e, sa_{adv}) \geq 0$. Fig. 4 indicates an example in which the $RD(e, sa_{adv})$ function is computed. Intuitively, the larger value of $RD(e, sa_{adv})$ indicates the more chance of having advanced knowledge in merely a single skill-area (i.e. sa_{adv}). It's worth mentioning that the left hand side of Eq. (27) is a probability, therefore, we normalized the quantity of $(|set_{Int(e)}| \cdot RD(e, sa_{adv}))$ in right-hand side using min-max normalization over all candidates.

The $|set_{Int(e)}|$ of Eq. (27) gives more score to candidates with several intermediate level skill-areas (i.e. T-shaped and C-shaped). Moreover, the $RD(e, sa_{adv})$ of this equation gives more score to candidates who have advanced knowledge in solely one skill-area (i.e. I-shaped or T-shaped). As a result, we expect that T-shaped candidates get more score in total.

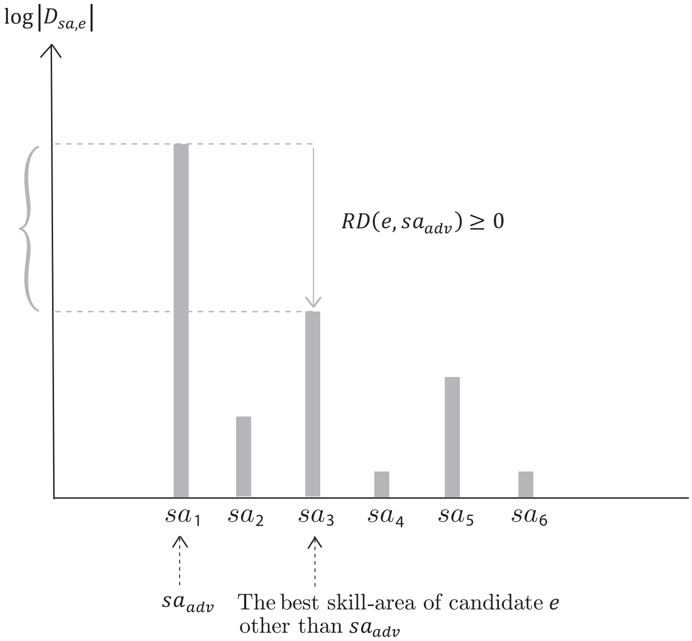

Fig. 4. An example of computation of $RD(e, sa_{adv})$ function.

The model proposed in this section -utilizing the profile of each candidate- tries to find the best T-shaped expert in each iteration and as a result, we expect that this model results in a team with good coverage, communication and optimality.

5. Evaluation measures

As explained before, three important criteria should be considered to evaluate an agile team. These measures (i.e. coverage, communication and optimality) cannot be evaluated using standard performance metrics of expert finding and team formation problem. Therefore, in this section, we propose three performance measures to evaluate an agile team.

As mentioned before, an ideal agile team should be:

- **High-performing:** All required skills of the project should be covered by the agile team members. According to this requirement, we define the *coverage* measure.
- **Flexible:** The members of agile team should be able to cover and support each other. According to this requirement, we define the *communication* measure.
- **Low-cost:** The members of agile team should be low-cost. According to this requirement, we define the *optimality* measure.

In addition to above measures, due to trade-off between coverage, communication and optimality, we define the *f-measure*. In order to show the trade-off, suppose a very expensive team with full coverage and communication. This team gets a low f-measure.

In the following sub-sections, we define the evaluation measures.

5.1. Coverage

This measure indicates how much a team can cover required skill-areas of the project. Suppose the required skill-areas are $sa_1, sa_2, \ldots, sa_n$.[7] Then the coverage measure is defined as follows:

[7] n is the number of required skill-areas. In this paper, we assume that each skill-area should be covered by a candidate. Therefore, the size of team is equal to n.

$$Coverage = \frac{\sum_{i=1}^{n} IsCovered(e_i, sa_i)}{n}, \quad (29)$$

where $IsCovered(e_i, sa_i)$ is a binary function which indicates whether the ith candidate (i.e. e_i) has advanced knowledge in skill-area sa_i in golden set or not. According to this definition, coverage measure is always between zero and one.

Impact of each shape of expertise on coverage measure is indicated in Table 2. Selection of Non-expert users who do not have advanced knowledge in any skill-areas, decreases the coverage, but selection of I-, T- or C-shaped experts for a role of team can improve the coverage, if the selected member has advanced knowledge in the main skill-area in that role.

5.2. Communication

This measure indicates how much members of a team can effectively communicate with each other. Intuitively, if the ith member of team has intermediate or advanced knowledge in main skill-area related to other members, he or she can support and communicate with them effectively. So, as indicated in Table 2, selection of T-shaped, C-shaped and Non-expert users can increase communication measure, but selection of I-shaped users for a role of the team will always decrease the communication measure, because they only have advanced knowledge in the main skill-area in that role and do not have intermediate knowledge in any skill-areas. The communication measure is defined as follows:

$$Communication = \frac{\sum_{i=1}^{n} Com(e_i)}{z}, \quad (30)$$

where $Com(e_i)$ is a function indicating the communication score of the ith member and $z = \binom{n}{2} = \frac{n(n-1)}{2}$ is the normalizing factor to ensure that the communication of a team is between zero and one. $Com(e_i)$ is calculated as follows:

$$Com(e_i) = \sum_{j=1, j\neq i}^{n} HasKnowledge(e_i, sa_j), \quad (31)$$

in which $HasKnowledge(e_i, sa_j)$ is a binary function and equals to one, if candidate e_i has intermediate or advanced knowledge in skill-area sa_j in golden set.

5.3. Optimality

This measure considers the cost of employment of team members. According to this measure, we prefer team members who only have advanced knowledge in the main skill-area of the respective role. In other words, we prefer team members who can cover the desired skill-area and does not have advanced knowledge in other skill-areas. So, as indicated in Table 2, selection of I-shaped or T-shaped experts will always increase optimality measure, but selection of C-shaped experts and Non-expert users will always decrease this measure, because C-shaped experts have advanced knowledge in more than one skill-area in the collection, and Non-expert users do not have advanced knowledge in any skill-areas. The optimality measure is defined as follows:

$$Optimality = \frac{\sum_{i=1}^{n} Opt(e_i)}{n}, \quad (32)$$

in which $Opt(e_i)$ is defined as follows:

$$Opt(e_i) = \begin{cases} \frac{1}{|G_{adv}(e_i)|^2} & \text{, if } e_i \text{ is I-, T- or C-shaped} \\ 0 & \text{, otherwise} \end{cases}, \quad (33)$$

where $G_{adv}(e_i)$ indicates the set of skill-areas who candidate e_i has advanced knowledge in them in golden set.

Table 2
Impact of each shape of expertise on coverage, communication and optimality measures.

Shape	Coverage	Communication	Optimality
Non-expert	↓	↕	↓
I-shaped	↕	↓	↑
T-shaped	↕	↕	↑
C-shaped	↕	↕	↓

Table 3
An instance of agile team formation problem.

	roles	sa_{adv}	set_{int}
	role 1	Spring	User Interface, Test
	role 2	User Interface	Spring, Test
	role 3	Test	Spring, User Interface

Table 4
General statistics of test collections.

Test collection	#Questions	#Answers	#Users
C#	763,717	1,453,649	84,095
Java	810,071	1,510,812	83,557

5.4. F-measure

As mentioned before, there is a trade-off between the coverage, communication and optimality measures. Therefore, in order to compare two teams, we use the harmonic average of mentioned measures, which are defined as follows:

$$F\text{-}measure = \frac{3}{\frac{1}{Coverage} + \frac{1}{Communication} + \frac{1}{Optimality}}. \quad (34)$$

As an example, consider a team formation problem which the goal is to find a team of three members for *User Interface, Spring* and *Test* skill-areas. The desired team has three roles which indicated in Table 3. Each role is shown by a different color. In an ideal team, each role should be filled by a T-shaped expert who has advanced knowledge in sa_{adv} and intermediate knowledge in set_{int}. For example, the first role (i.e. blue) should be filled by a candidate who has advanced knowledge in *Spring* and intermediate knowledge in *User Interface* and *Test* skill-areas. Fig. 5 indicates four recommended teams for this problem. The color of each candidate indicates the associated role of the team. Team A consists of three T-shaped experts. In this team, each member covers the main skill-area of her associated role and has intermediate knowledge in other required skill-areas of the team. As a result, team A indicates the best selection in terms of coverage, communication, optimality and f-measure. Team B consists of the C-shaped experts who in addition to cover the required skill-areas of the team, also cover some other skill-areas of the collection. This team is not optimal because each member has advanced knowledge in non-required main skill-area. As a result, the optimality and f-measure of team B is lower than one. Team C consists of three I-shaped experts. In this team, each member only covers the main skill-area of her associated role. So, this team is optimal in terms of employment cost of each member, but the team members cannot communicate with each other, therefore, the communication and f-measure scores are equal to zero. Finally, team D consists of the Non-expert users who have intermediate knowledge in the main skill-areas of the other roles, therefore, they can effectively communicate with each other, but the coverage of the team is zero. Also, according to Eq. (32) the optimality measure of this team is zero. So, the f-measure is also zero.

6. Experimental setup and results

In this section, we first introduce the test collections which are used in our experiments and then we explain the parameter settings. Finally, we describe our experiments and their results.

6.1. Test collection

Experiments of this paper were performed on a data collection extracted from Stackoverflow community question answering website. The data collection consists of posts (questions and answers) of StackOverflow from August 2008 up to march 2015. This collection has more than four million users and almost twenty four million posts. We filtered posts of StackOverflow by questions which have *C#* or *Java* as questions tags. Then, we partitioned the original data collection into two test collections of C# and Java. The Java collection includes all questions and their associated answers which have *Java* as one question tag. We did the same for C# collection. Table 4 indicates general statistics of extracted test collections.

We need a golden set which determine the following information for each user in the mentioned test collections.

- **The shape of expertise**: The shape of expertise (i.e. Non-expert, I-, T- and C-shaped) is determined for each user.
- **Advanced level skill-areas**: The set of skill-areas which the user has advanced knowledge in them.
- **Intermediate level skill-areas**: The set of skill-areas which the user has intermediate knowledge in them.
- **Beginner level skill-areas**: The set of skill-areas which the user has beginner knowledge in them.

The process of golden set generation is similar to our previous work (Gharebagh et al., 2018) and is publicly available[8]. In order to build the golden set, following steps are performed:

1. **Extractions of skill-areas**: Each question in Stackoverflow is associated with one or more tags. The set of related tags can demonstrate a skill-area. For example *Swing, JTable, User Interface, JPanel* and *JFrame* are related tags which demonstrate *Java User Interface* skill-area. Table 5 indicates the number of extracted skill-areas for each test collection.
2. **Segmentation of users in each skill-area based on their knowledge level**: In this step, we categorize the users according to their number of accepted answers on each skill-area into

[8] https://www.bit.ly/2LDKACi.

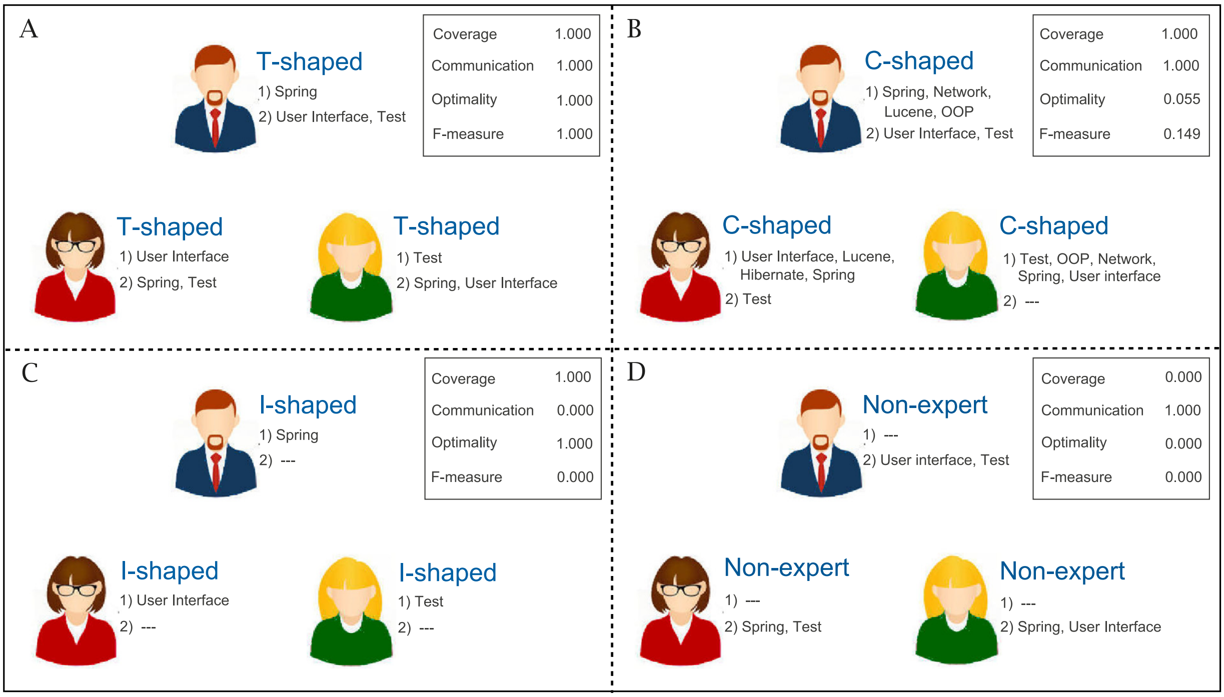

Fig. 5. Comparison of coverage, communication, optimality and f-measure calculated for 4 sample teams. For each candidate, number 1 and number 2 indicates the list of advanced and intermediate skill-areas, respectively. (For interpretation of the references to color in the text, the reader is referred to the web version of this article.)

Table 5
General statistics of C# and Java golden sets.

Test collection	#Skill-areas	#C-shaped	#T-shaped	#I-shaped	#Non-expert
C#	20	1,783	1,976	731	79,605
Java	26	1,673	1,744	721	79,419

three groups i.e. advanced, intermediate and beginner knowledge level.

3. **Categorization of users according to their shapes of expertise**: In this step, according to the number of advanced, intermediate and beginner skill-areas of each user (Table 1), we labeled him as Non-expert, I-shaped, T-shaped and C-shaped.

It is worth mentioning that the golden set is created using the number of accepted answers of each user (similar to Dargahi Nobari et al., 2017; Gharebagh et al., 2018; van Dijk, Tsagkias, & de Rijke, 2015 works) and this feature is not used in any proposed model. In our experiments, we use answers associated to each user as an evidence of his expertise. In other words, in this paper by associated documents of each user, we mean his or her answers.

Table 5 represents general statistics of golden sets including the number of extracted skill-areas and Non-expert, I-shaped, T-shaped and C-shaped users in test collections of C# and Java.

6.2. Parameter setup

In our experiments, we set the λ_d parameter of SEM and MEM models equal to 0.9 to put more emphasis on maximum likelihood estimation. We also set the α parameter of hybrid method (Section 4.6.3) equal to 0.5.

6.3. Results

In this section, we design several experiments to answer the following research questions:

- **RQ1**: How do SEM, MEM and EBM models perform compared to each other?
- **RQ2**: Both our proposed models (i.e. XEBM and RDM) are based on estimation of intermediate knowledge of users. Here, the question is which intermediate knowledge estimation method (i.e. kernel, overlap or hybrid) performs better in comparison with each other?
- **RQ3**: How do our models perform compared to the baselines and to each other?
- **RQ4**: How efficient are our proposed models in comparison with each baseline and to each other? Here, we compare the f-measure of models for each given instance of team formation problem.

We answer the above mentioned questions in Sections 6.3.1–6.3.4, respectively.

6.3.1. Comparison of baseline models

The radar diagrams in Fig. 6 clearly compare the baseline models in terms of coverage, communication, optimality and f-measure. For each measure, the average value is reported for all instances of team sizes of 2 to 7. In each test collection, in total, we have k skill-areas (Table 5), therefore, for a team size of t, the number of instances equals to $\binom{k}{t}$. The patterns observed on C# and Java test collections are quite similar. According to these figure, the *SEM* model has the best performance in terms of coverage, while the *MEM* model has the best communication performance. In contrast, the *EBM* model is the best model in terms of optimality. The important point here is that each of these mentioned models are able to optimize only two performance measures, and as a result, the harmonic average of performance measures (i.e. f-measure) is low for all of these models.

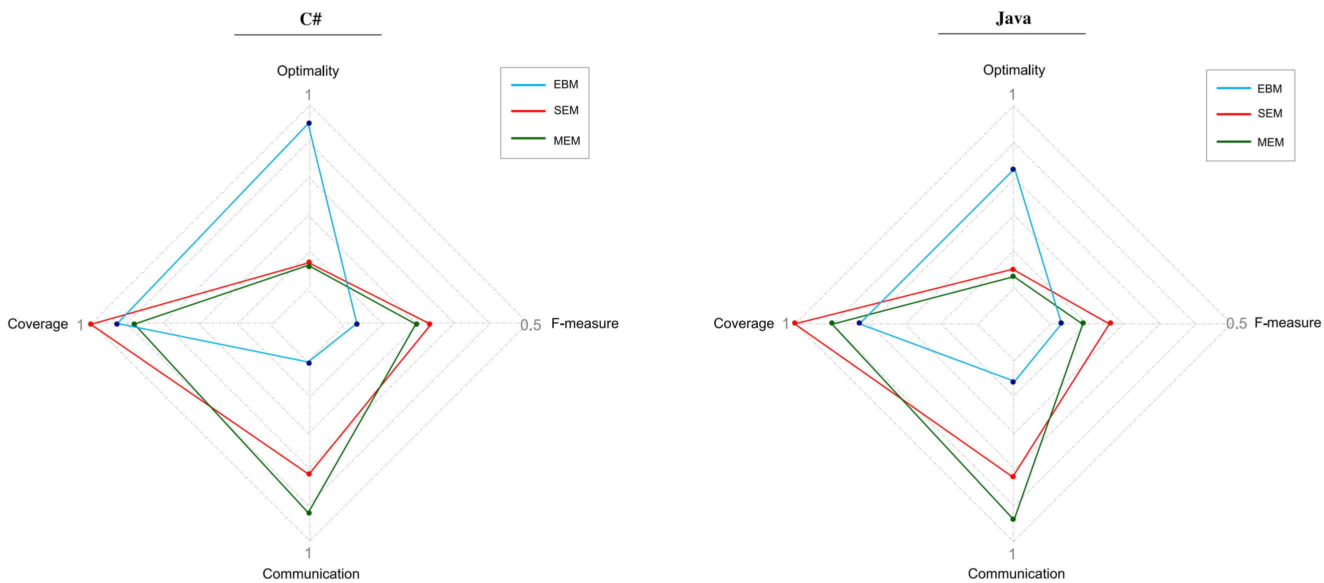

Fig. 6. Comparison of baseline models. For each measure, the average value is reported for all instances of team sizes of 2 to 7 in our collection.

Test Collection	Model	C-shaped	T-shaped	I-shaped	Non-expert
C#	SEM	92.4%	2.6%	5.0%	0%
	MEM	91.1%	4.0%	4.9%	0%
	EBM	0%	0%	90.0%	10.0%
Java	SEM	92.4%	7.5%	0.1%	0%
	MEM	95.0%	4.7%	0.3%	0%
	EBM	0%	0%	65.4%	34.6%

Fig. 7. The share of each shape of expertise in teams formed by SEM, MEM and EBM models. The average value is reported for team sizes of 2 to 7.

Fig. 7 indicates the share of each shape of expertise in the teams formed by *SEM, MEM* and *EBM* models for C# and Java test collections. For example, in C# test collection, the teams formed by *SEM* model are consisted of 92.4% C-shaped members, 2.6% T-shaped members, 5% I-shaped members and 0% Non-expert members. This figure can explain the behavior of each model and according to it, the following points can be observed.

- The *SEM* and *MEM* models tend to select the C-shaped candidates. As a result, the optimality and accordingly the f-measure of these models are low. The high share of C-shaped members result in high coverage and communication scores in these models.
- The *EBM* model tends to select I-shaped members. Therefore, the optimality of this model is high. On the other hand, the selected members by this model can usually cover the required skill-areas of the team. So, the coverage score is high, but the communication measure is very low and because of that the f-measure is also very low.

6.3.2. Comparison of kernel-based, overlap-based and hybrid methods

In Sections 4.6.1, 4.6.2 and 4.6.3 we proposed three methods to estimate the probability of $P(set_{int}|T=1, e, sa_{adv})$. In this section, we want to answer which of these proposed methods perform better. Fig. 8 indicates the comparison of overlap-based, kernel-based (linear and quadratic) and hybrid methods. According to this figure, following points are observable[9]:

- Overally, the performance of overlap-based method is better than the kernel-based.
- The quadratic kernel in kernel-based method generally performs better than the linear kernel, but the difference is not significant.
- The performance of the hybrid method is better than the kernel and overlap-based methods in all of our experiments.

Since the hybrid method dominates the kernel and overlap-based methods, in experiments of the rest of paper we use the hybrid method to estimate probability of $P(set_{int}|T=1, e, sa_{adv})$ in the proposed models.

6.3.3. Comparison of proposed models with baselines

Fig. 9 compares the proposed models with baselines for the C# and Java- test collections. According to this figure, the following points are observable:

- The *XEBM* model is an extension of *EBM* model which considers the intermediate knowledge level of team members. As a result, the communication measure of this model is significantly better than the communication of *EBM* model. This improvement comes with the cost of decreasing the optimality measure in

[9] The same patterns are observed for Java test collection, but for the sake of brevity we only show the C# collection.

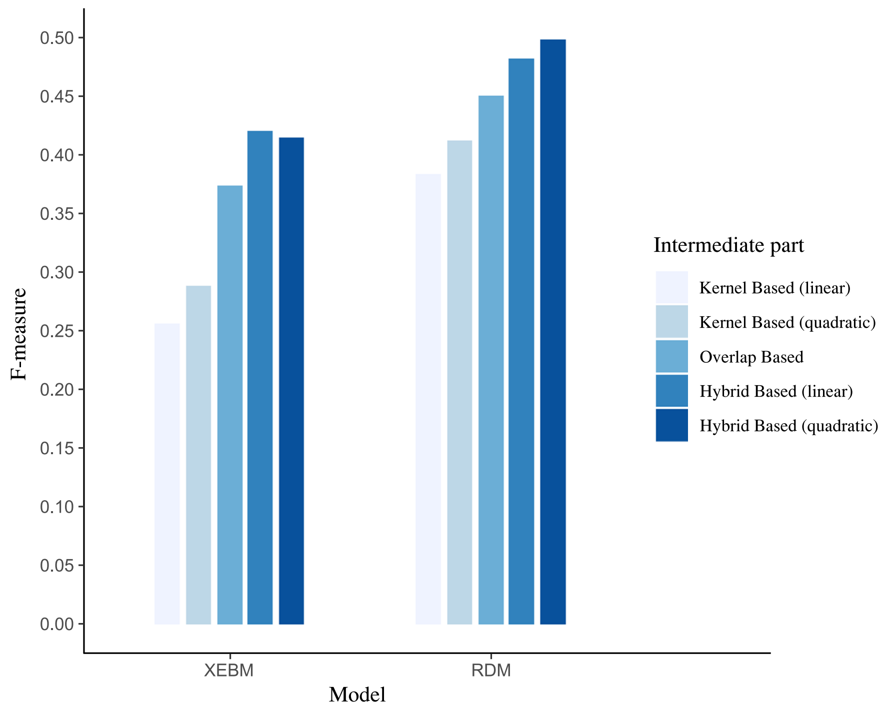

Fig. 8. Comparison of overlap-based, kernel-based and hybrid methods on C# data collection. In each method, the average of f-measure is reported for all instances of team sizes of 2 to 7 in our collection.

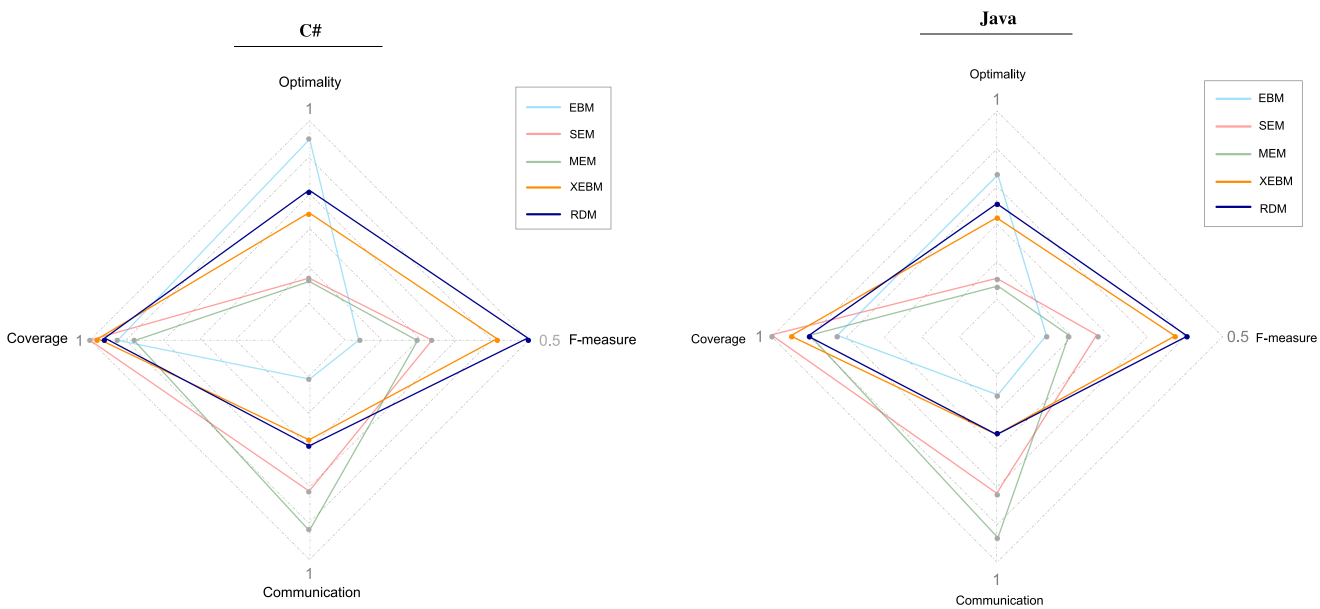

Fig. 9. Comparison of proposed models with baselines and to each other. For each measure, the average value is reported for all instances of team sizes of 2 to 7 in our collection.

comparison with *EBM* model. However, the *XEBM* model creates teams which balance coverage, communication and optimality measures. Therefore, the f-measure of *XEBM* model is significantly better than the *EBM* model.

- The *XEBM* model in comparison with *SEM* and *MEM* models has almost the same performance in terms of coverage measure. However, It performs significantly better than those baselines in terms of optimality measure. As a result, although the *XEBM* has lower communication, the f-measure is significantly higher than the baselines.
- While the *RDM* model slightly decreases the coverage measure, it can significantly improve the optimality in comparison with *XEBM*. As a result, the *RDM* model is the best performing model in terms of f-measure.

6.3.4. Pairwise comparison of team formation models

In order to compare the team formation models better, in Fig. 10 we demonstrate the difference of f-measure between each pair of models. In each sub-figure, the x-axis indicates the instances of team formation problem for team size of 5. For each instance, model 1 and model 2 suggest team 1 and team 2, respectively. The y-axis indicates the difference of f-measure for team 1 and team 2. Specifically, if the team proposed by model 1 is better than team 2 in terms of f-measure, the difference of f-measure is

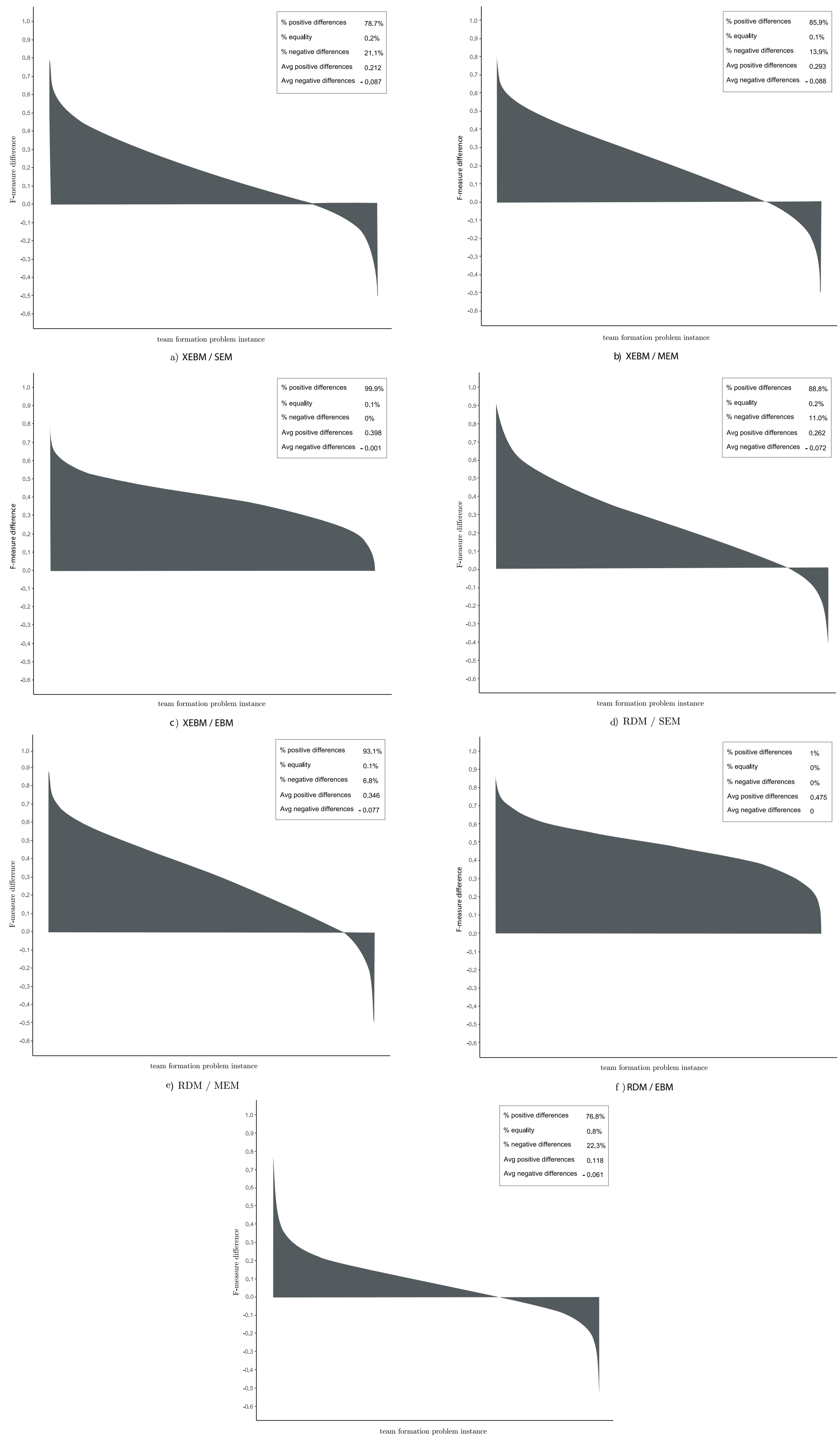

Fig. 10. Pairwise comparison of proposed models with baselines and to each other.

Table 6
Comparison of agile team formation models in terms of f-measure.

		F-measure	
		C#	Java
Based Model	EBM	0.034	0.029
	MEM	0.196	0.089
	SEM	0.236	0.166
Proposed Model	XEBM	0.414	0.375
	Δ SEM	75.4%[a]	125.9%[a]
	RDM	**0.498**	**0.407**
	Δ SEM	111.0%[a]	145.4%[a]
	Δ XEBM	20.3%[b]	8.6%[b]

[a] Indicates significant improvement over SEM model.
[b] Indicates significant improvement over XEBM model.

a positive value. This value indicates how much the first method performs better than the second model. According to this figure following points are observable.

- According to sub-figures c and f, the f-measure of our proposed models are always better than *EBM* for all instances of team formation problem. However, the average of f-measure difference for *RDM* model is larger than *XEBM*.
- According to sub-figures a and b, the f-measure of *XEBM* is better than *SEM* and *MEM* in 78.7% and 85.9% of problem instances, respectively. This observation indicates that *XEBM* significantly improve the f-measure in comparison with baselines.
- According to sub-figures d and e, the f-measure of *RDM* is better than *SEM* and *MEM* in 88.8% and 93.1% of problem instances, respectively. This observation indicates that *XEBM* significantly improves the f-measure in comparison with baselines.
- According to sub-figure g, in 76.8% of problem instances, the f-measure of *RDM* model is higher than the f-measure of *XEBM*. This observation indicates the *RDM* model performs significantly better than *XEBM*.

Finally, Table 6 indicates comparison of all models in terms of f-measure. In this experiment, agile teams of size 2 to 7 are formed by each model. The measures reported in Table 6 are the average of f-measure for each team size. According to this table, The *SEM* model is the best performing baseline in terms of f-measure. The *XEBM* model can significantly improve the f-measure on both test collections in comparison with *SEM* model, and *RDM* model performs significantly better than *SEM* and *XEBM* model on both test collections.

7. Conclusion and future work

In this paper, we introduced the problem of agile team formation. We precisely investigated the requirements of a flexible, high-performing and low-cost agile team. We proposed that the best candidate for an agile team is T-shaped expert. In order to solve the agile team formation problem, we proposed two efficient models (i.e. XEBM and RDM). The XEBM model is based on the entropy of profile of each candidate and the RDM model is based on the relative deepness of skill-areas of candidates. We showed that agile team formation problem is a well motivated industrial problem and we used two real test collections extracted from Stack-Overflow to compare our proposed models with three baselines. Our experiments indicate that the performance (i.e. f-measure) of proposed models is significantly better than the baselines. It can be explained by the fact that our proposed models simultaneously improve all the coverage, communication and optimality measures but each baseline can only improve two measures simultaneously.

For the future work, we plan to use machine learning approaches to optimally solve the agile team formation problem. In addition, complex network aspects of the problem (e.g. closeness and centrality of candidates) can be considered to define new problems and solve them.

References

Ambler, S. (2012). *Agile database techniques: Effective strategies for the agile software developer*. John Wiley & Sons.
Balog, K., Azzopardi, L., & de Rijke, M. (2009). A language modeling framework for expert finding. *Information Processing & Management, 45*(1), 1–19.
Balog, K., Bogers, T., Azzopardi, L., De Rijke, M., & Van Den Bosch, A. (2007). Broad expertise retrieval in sparse data environments. In *Proceedings of the 30th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 551–558). ACM.
Balog, K., Fang, Y., de Rijke, M., Serdyukov, P., Si, L., et al. (2012). Expertise retrieval. *Foundations and Trends® in Information Retrieval, 6*(2–3), 127–256.
Brown, T. (2005). Strategy by design. *Fast Company, 95*, 52–54.
Danait, A. (2005). Agile offshore techniques-a case study. In *Agile conference, 2005. Proceedings* (pp. 214–217). IEEE.
Dargahi Nobari, A., Sotudeh Gharebagh, S., & Neshati, M. (2017). Skill translation models in expert finding. In *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval* (pp. 1057–1060). ACM.
Demirkan, H., & Spohrer, J. (2015). T-shaped innovators: Identifying the right talent to support service innovation. *Research-Technology Management, 58*(5), 12–15.
Donofrio, N., Sanchez, C., & Spohrer, J. (2010a). Collaborative innovation and service systems. In *Holistic engineering education* (pp. 243–269). Springer
Donofrio, N., Spohrer, J., & Zadeh, H. S. (2010b). Driven medical education and practice: A case for t-shaped professionals.
Fang, H., Tao, T., & Zhai, C. (2004). A formal study of information retrieval heuristics. In *Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 49–56). ACM.
Gharebagh, S. S., Rostami, P., & Neshati, M. (2018). T-shaped mining: A novel approach to talent finding for agile software teams. In *European conference on information retrieval* (pp. 411–423). Springer.
Guest, D. (1991). The hunt is on for the renaissance man of computing. *The Independent (London), 17*.
Hashemi, S. H., Neshati, M., & Beigy, H. (2013). Expertise retrieval in bibliographic network: A topic dominance learning approach. In *Proceedings of the 22nd ACM international conference on conference on information & knowledge management* (pp. 1117–1126). ACM.
Iansiti, M. (1999). *Real-world R&D: Jumping the product generation gap*. Harvard Business School Press.
Kargar, M., & An, A. (2011). Discovering top-k teams of experts with/without a leader in social networks. In *Proceedings of the 20th ACM international conference on information and knowledge management* (pp. 985–994). ACM.
Karimzadehgan, M., & Zhai, C. (2012). Integer linear programming for constrained multi-aspect committee review assignment. *Information Processing & Management, 48*(4), 725–740.
Karimzadehgan, M., Zhai, C., & Belford, G. (2008). Multi-aspect expertise matching for review assignment. In *Proceedings of the 17th ACM conference on information and knowledge management* (pp. 1113–1122). ACM.
Kulak, D., & Li, H. (2017). *The journey to enterprise agility: Systems thinking and organizational legacy*. Springer.
Kumar, V., & Pedanekar, N. (2016). Mining shapes of expertise in online social Q&A communities. In *Proceedings of the 19th ACM conference on computer supported cooperative work and social computing companion* (pp. 317–320). ACM.
Li, C.-T., Shan, M.-K., & Lin, S.-D. (2015a). On team formation with expertise query in collaborative social networks. *Knowledge and Information Systems*, 42 (2), 441–463.
Li, H., Jin, S., & Shudong, L. (2015b). A hybrid model for experts finding in community question answering. In *Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2015 International Conference on* (pp. 176–185). IEEE.
Liang, S., & De Rijke, M. (2013). Finding knowledgeable groups in enterprise corpora. In *Proceedings of the 36th international ACM SIGIR conference on research and development in information retrieval* (pp. 1005–1008). ACM.
Liang, S., & de Rijke, M. (2016). Formal language models for finding groups of experts. *Information Processing & Management, 52*(4), 529–549.
Liu, D.-R., Chen, Y.-H., Kao, W.-C., & Wang, H.-W. (2013). Integrating expert profile, reputation and link analysis for expert finding in question-answering websites. *Information processing & management, 49*(1), 312–329.
Neshati, M., Beigy, H., & Hiemstra, D. (2014a). Expert group formation using facility location analysis. *Information Processing & Management, 50*(2), 361–383.
Neshati, M., Fallahnejad, Z., & Beigy, H. (2017). On dynamicity of expert finding in community question answering. *Information Processing & Management, 53*(5), 1026–1042.
Neshati, M., Hashemi, S. H., & Beigy, H. (2014b). Expertise finding in bibliographic network: Topic dominance learning approach. *IEEE Transactions on Cybernetics, 44*(12), 2646–2657.
Neshati, M., Hiemstra, D., Asgari, E., & Beigy, H. (2014c). Integration of scientific and social networks. *World Wide Web, 17*(5), 1051–1079.

Petkova, D., & Croft, W. B. (2008). Hierarchical language models for expert finding in enterprise corpora. *International Journal on Artificial Intelligence Tools, 17*(01), 5–18.
Riahi, F., Zolaktaf, Z., Shafiei, M., & Milios, E. (2012). Finding expert users in community question answering. In *Proceedings of the 21st international conference on world wide web* (pp. 791–798). ACM.
Rubin, K. S. (2012). *Essential Scrum: A practical guide to the most popular Agile process*. Addison-Wesley.
Smirnova, E. (2011). A model for expert finding in social networks. In *Proceedings of the 34th international ACM SIGIR conference on research and development in information retrieval* (pp. 1191–1192). ACM.
Stoica, M., Mircea, M., & Ghilic-Micu, B. (2013). Software development: Agile vs. traditional. *Informatica Economica, 17*(4), 64.
van Dijk, D., Tsagkias, M., & de Rijke, M. (2015). Early detection of topical expertise in community question answering. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval* (pp. 995–998). ACM.
Zhao, Z., Yang, Q., Cai, D., He, X., & Zhuang, Y. (2016). Expert finding for community-based question answering via ranking metric network learning. In *Ijcai* (pp. 3000–3006).
Zhao, Z., Zhang, L., He, X., & Ng, W. (2015). Expert finding for question answering via graph regularized matrix completion. *IEEE Transactions on Knowledge and Data Engineering, 27*(4), 993–1004.
Zhu, H., Chen, E., Xiong, H., Cao, H., & Tian, J. (2014). Ranking user authority with relevant knowledge categories for expert finding. *World Wide Web, 17*(5), 1081–1107.