Security Professional Skills Representation in Bug Bounty Programs and Processes

Sara Mumtaz[1], Carlos Rodriguez[2], and Shayan Zamanirad[1]

[1] School of Computer Science and Engineering, UNSW Sydney, NSW 2052, Australia
{s.mumtaz, shayan.zamanirad}@unsw.edu.au
[2] Universidad Católica Nuestra Señora de la Asunción, Paraguay
carlos.rodriguez@uc.edu.py

Abstract. The ever-increasing amount of security vulnerabilities discovered and reported in recent years are significantly raising the concerns of organizations and businesses regarding the potential risks of data breaches and attacks that may affect their assets (e.g. the cases of Yahoo and Equifax). Consequently, organizations, particularly those suffering from these attacks are relying on the job of security professionals. Unfortunately, due to a wide range of cyber-attacks, the identification of such skilled security professional is a challenging task. One such reason is the "skill gap" problem, a mismatch between the security professionals' skills and the skills required for the job (vulnerability discovery in our case). In this work, we focus on platforms and processes for crowdsourced security vulnerability discovery (bug bounty programs) and present a framework for the representation of security professional skills. More specifically, we propose an embedding-based clustering approach that exploits multiple and rich information available across the web (e.g. job postings, vulnerability discovery reports) to translate the security professional skills into a set of relevant skills using clustering information in a semantic vector space. The effectiveness of this approach is demonstrated through experiments, and the results show that our approach works better than baseline solutions in selecting the appropriate security professionals.

Keywords: Bug Bounty Programs and Processes · Skills Representation · Embeddings Models · Ethical Hackers · Cyber Security.

1 Introduction

The advancement in the Web 2.0 technology and its widespread use in virtually all types of businesses has increasingly exposed us to security threats and cyber-attacks during the last years. These attacks result in several security breaches events targeting not only individuals but giant organizations including the US Department of Defence[3], JP Morgan[4] and many more. Perhaps, among these, the

[3] https://thehill.com/policy/cybersecurity/483853-defense-department-agency-suffers-potential-data-breach
[4] https://www.theguardian.com/business/2014/oct/02/jp-morgan-76m-households-affected-data-breach

S. Mumtaz, C. Rodriguez, S. Zamanirad, "Security Professional Skills Representation in Bug Bounty Programs and Processes," The 1st International Workshop on AI-enabled Process Automation, ICSOC 2020, December 2020, Dubai, UAE.

The final authenticated version will be available online at https://link.springer.com

most notable is the Equifax data breach, which exposed the sensitive information of 147 million people, with an estimated settlement of ∼650 million US dollars[5].

In response to these security breaches, organizations are increasingly relying on security professionals (SecPros) and investing in their services through security crowdsourcing platforms and processes (i.e. bug bounty programs) to find and address security vulnerabilities [2]. A bug bounty program offers rewards to external parties (through crowdsourcing) allowing them to perform a security assessment of their assets (e.g. software, hardware) [9].

These bug bounty programs are a useful complement to existing internal security programs and widely accepted by organizations [19]. Additionally, due to the nature of crowdsourcing, organizations are benefitting from its speed and the vast pool of available SecPros with diverse skills and expertise. For instance, one study [31] found that through these outsourced programs, a greater number of vulnerabilities can be found, and more quickly compared to the time required by in-house testers making the process more time- and cost-effective. However, despite the large pool of these SecPros, there is a lack of sufficiently skilled cybersecurity professionals [26]. For example, the MIT Technology Review[6] and Cybersecurity Venture[7] predicted that the demand for cybersecurity professionals is expected to increase by 350 per cent, from one million in 2013 to 3.5 million in 2021.

There could be many reasons for skills shortages, one of the main ones being the "skills gap" problem [7], that is, a mismatch between the skills of security professionals and the skills required for a particular job (vulnerability discovery in our context). Secondly, the different types of vulnerabilities require different levels of skills and expertise [2]. For example, Web application vulnerabilities require knowledge about the software itself, networking protocols, Web frameworks, and vulnerabilities that target Web technologies.

To address the aforementioned challenges, it is essential that SecPros get selected for tasks based on their skills. In turn, having the right SecPros assigned to tasks contributes to making bug bounty programs and processes successful. In this context, we propose an embedding-based clustering technique, which translates the *SecPro skills* into a set of relevant skills using clustering information in the semantic space. Firstly, the data related to SecPros skills is collected from heterogeneous, multiple sources and grouped them as semantically correlated clusters in an embedding space using clustering algorithms [15]. Then, when a vulnerability discovery task is presented, it is placed (vectorized) in the same embedding space as the skills. Lastly, the cosine distance between clusters of skills and a task vector is computed to either recommend a set of skills, or SecPros for the task. The core of our approach is the representation of a task and skills in the same embedding space, which helps to mitigate the "skill gap" problem.

[5] https://www.nytimes.com/2019/07/22/business/equifax-settlement.html
[6] https://www.technologyreview.com/s/612309/a-cyber-skills-shortage-means-students-are-being-recruited-to-fight-off-hackers/
[7] https://cybersecurityventures.com/

The rest of the paper is structured as follows. Section 2 introduces our approach to representing security professionals' skills. The experiments and evaluations are presented in Section 3. Section 4 provides background information and related work on general and crowdsourced approaches for skills extraction and representation. Finally, Section 5 provides concluding remarks and future work.

2 Representing Security Professionals' Skills and Recommending to Vulnerability Discovery Task

This section presents our proposal for representing SecPros skills and recommending them to vulnerability discovery task. Fig. 1 presents an outline of our framework.

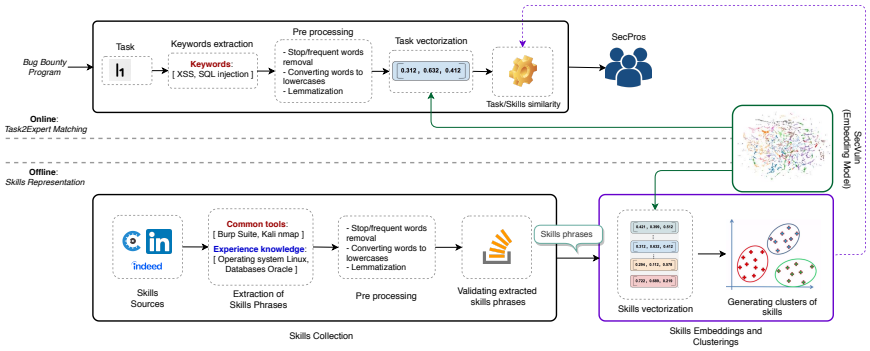

Fig. 1. Overall Framework

Our proposed framework exploits the heterogeneous information available across the Web such as job postings, resumes from job search portals, and complements these with other notable sources such as the skills declared on SecPros profiles across different platforms (e.g. Cobalt). More precisely, our approach consists of two phases (see Fig. 1):

- **Skills Representation.** This phase collects the SecPros' skills related information scattered across the Web. Then, by leveraging the property of word embeddings [20], we represent them in a semantic space via clustering. The generated skills clusters are then stored offline for further use by the following phase.
- **Task-SecPro Matching.** This phase represents the vulnerability discovery task in the same space as the skills (built in the previous phase). Then, this representation is matched with either expertise of SecPros or skills to find appropriate SecPros for the task.

In the following sections, we discuss in more details each of these two phases.

2.1 Skills Representation

Skills Collection. Skills collection consists of four steps: (i) Identification of skills sources, (ii) extraction of skills from identified sources, (iii) pre-processing, and (iv) normalization and validation of the extracted skills.

(i) Identification of skills sources. Skills can be extracted from multiple sources, including job postings (e.g. job descriptions and requirements section) from job portals (e.g. CareerBuilder.com), technical/skills sections in online resumes (e.g. indeed.com), and self-declared skills list from various platforms (e.g. LinkedIn, Cobalt). In this work, we prefer to use "skill phrase" (also called n-grams [3]), considering that skills are often made up of multiple-words (e.g. "penetration testing", "source code review"). After the identification of skills sources, the next step involves the extraction of skill-related phrases.

(ii) Extraction of skills phrases. The literature offers several techniques for the extraction of skills phrases. For example, [13] used Term-Frequency Inverse-Document-Frequencey (TF-IDF) [5] to extract relevant and essential keywords from job descriptions and resumes. Likewise, LinkedIn argued in [3] that users on LinkedIn use a comma-separation technique to provide a skills list in the "skills and expertise" section (e.g. "Java", "SQL", "Reinforcement Learning"). They utilized the comma-separated technique for identification and extraction of skills phrases. Similarly, we use TF-IDF and topic modelling [4] techniques to extract the important keywords representing the skills set of SecPros from notable sources (i.e. vulnerability discovery report, job descriptions). Furthermore, we also utilize the comma-separation technique and Web scrapping methods when necessary (e.g. in case of the self-declared skills list).

(iii) Pre-Processing. We apply basic text pre-processing techniques to make our collected skill phrases available for further processing. These techniques include the removal of stop words, converting the whole dataset to lowercase and lemmatization. More importantly, frequently occurring words (e.g. knowledge, proficient, team-oriented in job requirements), are discarded as they can act as outliers and make the skills data noisy [14].

(iv) Normalizing and validating the extracted skills phrases. The goal of this step is to retain the valid skills phrases and discard any other keywords that are not valid skills phrases. As mentioned previously, the skills phrases are human-generated (job postings, resumes), and everyone has different ways of expressing them (i.e. different representations of the same concept/skill). For example, some may prefer to write a vulnerability type as "XSS", and others may write it as "Cross-Site Scripting". As a result, there could be a great deal of redundancy in the users' skills set.

We apply a normalization technique to express them in a standard (base) form. An example of a base form would be *penetration testing, pen testing, and pen test* into *penetration testing*. However, the lemmatization is usually done through Wordnet [21], which is a general-purpose database and, as expected, does not have specialized terminology. Likewise, there are several skills knowledge bases available to validate skill phrases, such as O*Net (used by US public

recruitment services) [6], and ESCO (a European skills taxonomy)[8]. Nevertheless, all these skills knowledge bases are for general purpose recruitment and do not necessarily contain terminology that is specific to cyber security domain. To tackle this problem, we utilize Wikipedia open search [14], and tags (Stack Overflow and Stack Exchange), and also rely on keywords from our previous work dataset [24] and other cybersecurity domain-specific sources (e.g. National Institute of Standard and Technology (NIST)[9]).

Skills Representation via Clustering. This step involves a semantic representation of skills phrases to reduce the skill gap problem. To do so, we present an embedding-based clustering method. Embedding models, more precisely, word embeddings, generate a dense, continuous, low-dimensional representation of words from the raw corpus in an unsupervised way [20]. The words (in our case, skills phrases) that have a similar context or semantics have close embeddings in the vector space. These vectors of skills phrases are further represented using clusters so that similar and semantically coherent skills should be in the same cluster. The assumption is that, since word embeddings span a semantic space, the clusters based on word embeddings would give a higher semantic space for the skills phrases [8].

Clustering. A cluster is a collection of items that are similar to each other and dissimilar to other clusters' items [15]. Clustering is essentially an unsupervised, machine learning method and is mainly used to classify unlabeled data. Examples of applications of clustering include text analysis, pattern recognition, segmentation (image processing) and collaborative filtering. Recently, it has been used successfully to represent taxonomies for topics based on academic papers [29] and experts finding [8].

Generating Clusters. Given m number of skill phrases $S = \{s_1, s_2, \ldots s_m\}$, we utilize our cyber security vulnerability word embedding (SecVuln) [24] to generate a vector representation for each skill phrase. Then, we apply a clustering algorithm, specifically hierarchical clustering [8] to group them into k clusters, that is, $C = \{C_1, C_2 ... C_k\}$ (e.g. $C_1 = \{burpsuite, kalilinux, nmap, metasploit\}$) such that semantically correlated skill phrases belong to the same cluster.

SecPro Expertise Representation. This step represents SecPro expertise to match it with a vulnerability discovery task. Using statistical language modeling [23], the expertise and skills of SecPros can be inferred from their relevant documents (e.g. email communications or answers in Q&A web sites). In our context, vulnerability discovery reports and self-declared skills of SecPros are an excellent illustration of their expertise. However, as previously mentioned, self-declared skills listed in profiles are human-generated and therefore prone to incompleteness or bias. Therefore, after the initial collection of skills phrases, we enriched it with the discovered vulnerabilities given in SecPros profiles.

Next, we leverage the clusters generated in the previous step to represent SecPros in a cluster form. The purpose of this step is to recognize the unspecified skills of SecPros. To do so, the skills phrases of each SecPro are matched with the

[8] https://ec.europa.eu/esco/portal/skill
[9] https://csrc.nist.gov/glossary

clusters of skills phrases using a simple keyword-matching algorithm. It is worth mentioning that the matching takes place at a certain threshold (e.g. if 50% of skills phrases are matched, then a cluster is chosen, otherwise discarded). The clusters are further aggregated using vector averaging technique [23] to represent the cluster as a vector. Unlike the result of skills phrases' vector averaging, the vector average of a clusters gives more accurate result as shown later in our experiments reported in Section 3, having the advantage of being semantically similar to each other.

For instance, skills phrases extracted from a vulnerability report would consist of keywords with different semantics (e.g."Persistent XSS via filename in projects", a title of a vulnerability discovery report on HackerOne[10]). However, the skills phrases within a cluster are already related to each other, and hence would be more useful in accurately matching SecPros with tasks. The representations of the selected cluster vectors $\overrightarrow{C} = \left\{\overrightarrow{C_1}, \overrightarrow{C_2}...\overrightarrow{C_k}\right\}$ are then stored offline as a distribution over skills.

2.2 Task-SecPro Matching

The purpose of this phase is to recommend either SecPros to the given task or skill phrases to the given task. Upon the arrival of a task to the crowdsourcing platform (e.g. HackerOne[11]), we perform similar pre-processing and keywords extraction (from the description of task) as in the previous phase to obtain a list of keywords T. Then, we leverage the word embedding model to generate a vector representation $\overrightarrow{T}$ for the task based on the extracted keywords T.

After obtaining the vector $\overrightarrow{T}$, the task matching between $\overrightarrow{T}$ and the clusters of skills phrases $\overrightarrow{C}$ takes place using cosine similarity [5], which is defined as follows:

$$sim(T, C) = \frac{\overrightarrow{T}.\overrightarrow{C}}{|\overrightarrow{T}||\overrightarrow{C}|}$$

The similarity score ranges from [-1, 1], where the closer the value to 1 the more relevant to the task is to the expertise of a SecPro.

3 Experiments

In this section, we present the experimental results of our approach using the following evaluation techniques.

- *Validation of cluster quality:* To examine how closely the skills phrases are related to each other within the cluster.
- *Validation based on information retrieval:* To determine the effectiveness of our approach in selecting the appropriate SecPros for a given task.

[10] https://hackerone.com/reports/662204
[11] https://www.hackerone.com/

3.1 Dataset

In this work, we collected data from popular job search portals such as *indeed.com*[12] and *monster.com*[13] with cyber security jobs related query (e.g. "penetration testing", "code reviews") [26]. The collected data is further enriched with vulnerability discovery reports from HackerOne. Specifically, we focused on the section where the required skills are listed. Moreover, we utilized SecPros profiles on Cobalt for collecting self-declared skills along with the vulnerabilities they had discovered. The intuition is, if a set of skills and discovered vulnerabilities appear in the same profile (co-occurred), then they are important for each other.

Test Data. For test purposes, we select the vulnerability discovery tasks (e.g. Sony Vulnerability Discovery Program[14]) that are available on the HackerOne platforms. It is worth mentioning here, that during cluster generation we did not consider these tasks as a source, so that test data and training do not overlap.

Ground Truth. To examine how well our technique can determine the right SecPro for a given task, we need to have a ground truth for comparison (between the actual SecPros and the SecPros returned by our technique). To do so, we collected the profiles of top 100 SecPros from Cobalt[15]. Cobalt rank these SecPros according to the vulnerabilities that have discovered along with the quality of reports they submitted to the platform.

3.2 Embedding Model

We utilized the embedding model (SecVuln) built for the cybersecurity domain in our previous work [24]. However, to cope with the new terminologies in the job advertisements, we enriched our previous model with information extracted from job descriptions and resumes. We followed the same parameter settings as reported in [24].

3.3 Evaluation

Comparison Method. In order to demonstrate the effectiveness of our proposed approach, we compared it with the a baseline approach, that is, the vectors averaging technique [23].

Evaluation Metrics. To determine the effectiveness of our proposed approach in terms of quality of clusters and retrieving the appropriate SecPro, we used the (i) silhouette index [15], and (ii) information retrieval measure such as Precision at N (P@N) [5].

(i) ***Cluster Quality.*** Embedding-based clustering is expected to learn coherent and semantically correlated skills phrases within the clusters to facilitate the

[12] https://au.indeed.com/
[13] https://www.monster.com/
[14] https://hackerone.com/sony
[15] https://app.cobalt.io/pentesters

semantic understanding of these phrases. Hence, we evaluate the coherence of clusters using a silhouette index [15]. The silhouette index indicates the compactness and separation of clusters. For example, a set of skills clusters represented by $C = C_1, C_2 \ldots C_k$, consists of n number of vectors; then, the silhouette index is given below:

$$S(C) = \frac{\frac{1}{n}\sum_{i=1}^{n}(b_i - a_i)}{max(a_i, b_i)}$$

where a_i denotes the average distance of skill i to other skills in the cluster, whereas b_i is the minimum of average distance of a skill b_i to other skills of clusters. The value of the silhouette index ranges from -1 to 1. A higher value represents a better quality of clusters. In our case, the result amounts to approximately 0.75, which indicates the quality of our clusters.
(ii) ***Precision.*** Precision is one of the widely used information retrieval measures for expert finding [23], which measures the percentage of correct results (relevant SecPro found) out of total results (total number of SecPros returned) from the system. Formally, let R_c and R_w represent correct (true positives) and wrong results (false positives) respectively. Then, precision is defined as $P = \frac{R_c}{R_c+R_w}$. Instead, Precision at N (P@N) is the percentage of relevant SecPros found at the top N retrieved, ranked results (e.g. P@5 shows the total relevant SecPros until 5).

Table 1 shows that the proposed clustering-based technique perform better compared to the baseline technique. The clustering technique has an advantage over the keywords' vector-averaging technique. For instance, vector averaging technique, combines all available keywords extracted from multiple sources (which may consist of skill phrases and other words)

Table 1. Task-to-SecPros Matching

Technique	**P@5**	**P@10**
Vector Averaging	0.55	0.45
Clustering (our proposal)	0.60	0.57

3.4 Discussions and Limitations

The use of keywords other than skill phrases may add noise and lead to an inaccurate vector representation. The clustering-based approach presented in this paper groups the semantically related skills phrases, which helps in overcoming this problem. Furthermore, our proposed approach offers the following advantages:
Skills Representation. Skills representation can help educational institutions to address the skills gap between industry and current curriculum offerings (as

these skills come from the 'hands-on expertise' of SecPros (ethical hackers)). For instance, organizations like NIST have already initiated a program called NICE (National Initiative for Cybersecurity Education)[16] to fill the gap; they can further leverage our work for improvement. Secondly, the organization can also benefit from this pool of skills; for example, they can train their internal security (testers) on a specific type of vulnerability like Web API vulnerability.

SecPros Expertise Representation. Moreover, the representation of SecPros' expertise can help crowdsourcing platforms, after launching bug bounty programs, to directly contact SecPros (mapping between task and SecPro expertise) and invite them to participate.

Limitations. Despite its advantages, our approach has limitations. For instance, to represent SecPros expertise, we rely on textual contents only, and moreover, only one source (i.e. self-declared profiles on Cobalt) is taken into account. This approach can be further improved by incorporating SecPros' social activities and their interactions on social networks (e.g. Twitter) [25] through network embedding.

Regarding the computing of SecPro ranking in terms of their expertise, we consider only one expertise signal (i.e. report quality on Cobalt). However, as mentioned in [23], "expertise" is an umbrella term and comprises many signals (e.g. SecPros certifications, platforms ranking, badges, hall of fame). Moreover, [2] conducted a comprehensive study and found different indicators such as certifications and number of the vulnerabilities discovered as signals of SecPro expertise. Our work can leverage that study and add more signals for computing the expertise.

SecPros data is scattered across the Web and different platforms provide different information (expertise signals). For instance, HackerOne discloses the reports submitted to their platform following their bug bounty policy (not every organization discloses its reports). BugCrowd, on the other hand, provides information about the type and severity of vulnerabilities discovered by SecPros. The key challenge here is to combine all those signals and information about a specific SecPro from different platforms. However, the prevalence of social platforms (LinkedIn and Twitter) and the presence of SecPros on these platforms can mitigate this problem by using SecPros' social identifiers to recognize them on different platforms.

Moreover, we observed from experiments that the proposed clustering technique is prone to the problem of over-representation of users' skills and expertise. On the one hand, clustering helps in identifying any unspecified skills. However, some clusters list skills which are not necessarily a substitution of skills. For instance, the cluster defining the skills phrases indicates that there are different techniques for finding vulnerabilities; they do not need to have knowledge of all of them. As mentioned in [18], sometimes they prefer low hanging fruit and finding vulnerabilities and utilize the tools they already have.

[16] https://www.nist.gov/itl/applied-cybersecurity/nice

4 Related Work

Our work in this direction inherits a rich ecosystem of commercial job search platforms and general skills modeling techniques and draws on the insights offered by previous works in regard to the selection of workers in security crowdsourced platforms (bug bounty).

4.1 General Approaches for Skills Extraction and Representation

One of the most challenging tasks for any employer is the hiring of new people from a large pool of job applications. [16] developed a system, Elisit (Expertise Localization from Informal Sources and Information Technologies), that peruses data from Wikipedia and LinkedIn to extract skills from text documents. The authors claim that their approach could be easily integrated with any skills search engine or HR automation in any automatic meta-data extraction systems.

However, the self-declared skills (e.g. those explicitly given in the LinkedIn profile) may be incomplete or biased. To address this problem, [27] introduced approaches to analyze individuals' communication data (e.g. emails, discussion forums) to infer their skills. [28] also utilized personal skill information derived from social media platforms (e.g. Twitter) for skills inferences. They proposed a joint prediction factor graph model to infer user skills automatically from their connections on social networks.

Commercial Based Approaches. Several works address the skills representation in commercial job search portals for talent search using their built-in systems [14][11]. Some of the works from notable job search portals (e.g. CareerBuilder and LinkedIn) are described below.

CareerBuilder. To overcome the "skill gap" in the labor market, CareerBuilder (US most prominent human capital solution) [30] [14] presented an in-house skill terms extraction system, SKILL, for the extraction of keywords (aka skills) from both job descriptions and users' resumes. More specifically, in this work [14], the authors assumed the contents of individuals' resumes (technical section) and job ads (descriptions) as indicative of specific skills. They utilized a well-known algorithm, Word2vec [20] with the assumption that related skills are likely to appear in the same documents (resumes and job ads). For instance, "Python" would always be a *programming language* in their system instead of a *snake*. This work has achieved almost 91% accuracy and 76% recall, and the system is successfully deployed in multiple business intelligence projects.

As an improvement on their previous work, the authors [32] quantified the relevance of the skills to the job titles. To do so, they used a simple yet effective technique, TF-IDF (term frequency and inverse document frequency) [5], to measure the skills-job title relationship, assuming that a particular skill is important if it constitutes part of the job title.

In further work, they proposed a representation learning [7] to jointly represent job titles and skills in the same vector space for skills to skill similarity via three networks/graphs (i.e. job skill graph, job transition graph, and skills co-occurrences graph). These graph are constructed using skills (nodes) from the

same resume. For example, an edge is formed between skills (e.g. Data mining and Machine learning) if they both appear in the same resume. Likewise, they extended this work and proposed [17] a tripartite vector representation of job posting (i.e. job, skills and location) for a better job recommendation. The vector representation of job title and the skills required for that job are added to a personalized vector for a specific position in one vector representation. Then, this vector is further concatenated with the location vector, and is currently being used within CareerBuilder.

LinkedIn. LinkedIn is the world's largest professional online social network with 500 million users profiles, indicating their professional identity. Their talent search system is widely used by job seekers and employers and generates approximately 65% of company revenue [11]. LinkedIn presented [3] "Skills and Expertise" feature as a part of their current system. They built a folksonomy (often used for categorization of contents) using a data-driven approach.

To further improve their in-house system, LinkedIn introduced another technique [11] to address the problem of personalized expertise search. More specifically, this work utilized collaborative filtering and matrix factorization techniques to infer the member' skills and expertise from the existing set of skills. Next, they combined these skills with other personalized (e.g. location, social connections) and non-personalized (e.g. textual contents) features to rank members accordingly against the query.

4.2 Workers (SecPro) Selection in Bug Bounty

As previously mentioned, bug bounty programs inherit all the properties of crowdsourced platforms [19]. Hence, they have implemented the same strategies for crowd/SecPro selection as those used by general crowdsourced platforms, such as qualification tests [1] The qualification test is a pre-selection criterion used to screen potential workers. It is used to assess the level of expertise of SecPros before recruiting them for the real task of vulnerability discovery. Like general crowdsourced platforms (e.g. Amazon Mechanical Turk), the bug bounty platforms also ask SecPros to correctly answer the questions with already-known solutions. For instance, [10] developed a conceptual expertise tool that relies on a set of questions to distinguish a novice from an expert SecPros. However, it relies on the self-declared skills and assessment of the expertise of the SecPro. Similarly, Synack[17], a crowdsourced vulnerability discovery platform, evaluates the SecPros through written and practical tests to ensure that candidates are eligible to join the platform. Likewise, Upwork[18], an online freelancer market, assesses the competency of the freelancer using prior knowledge like certification and then determines the skills via online testing. Apart from the preliminary tests, some organizations may also impose specific predefined criteria (e.g. eligibility) for participation in the task. For instance, Mozilla bug bounty[19] do not allow their own employees to participate in any of their bug bounty programs.

[17] https://www.synack.com/red-team/
[18] https://www.upwork.com/
[19] https://www.mozilla.org/en-US/security/bug-bounty/

Furthermore, some of the bug bounty platforms (e.g. HackerOne, BugCrowd) maintain the SecPro' profiles utilizing their details (e.g. certifications) and ongoing activities (e.g. number of vulnerabilities they have discovered, relative ranking, and any reward they received) on the platforms. After launching bug bounty programs, organizations may invite the top SecPros (the top 100, for example) to participate.

Several studies have been conducted for worker/people selection in general crowdsourcing platforms. However, to the best of our knowledge, we did not come across any such work for security crowdsourced platforms (bug bounty) other than empirical studies. For example, [31] performed an empirical study to determine the characteristics of SecPros. Their study focuses on the tools and methods used by SecPros for discovering vulnerabilities and the type of vulnerability is common in the community. They determined how SecPros approach vulnerability discovery task. However, their study did not explore the criteria for SecPros' expertise indicators to accomplish the task. On the other hand, [12] investigated the heterogeneity among the SecPros participating in crowdsourced vulnerability discovery tasks. The authors discovered that there are two different types of SecPros participating in crowdsourced vulnerability discovery. Most SecPros are non-project-specific (i.e. submit vulnerabilities to multiple tasks) and are different from traditional SecPros who work on specific projects (i.e. submit vulnerabilities only to tasks that they are interested in making the software secure). However, unlike the previous approaches, [1] conducted a comprehensive empirical study to determine SecPros expertise indicators to improve the quality of software vulnerability discovery.

Keeping the limitations of previous works in mind, our study aimed to propose computational techniques for skills representation and task matching for crowdsourced vulnerability discovery platforms and processes (bug bounty programs).

5 Conclusion

In this paper, we addressed the skills gap problem in the context of platforms and processes for crowdsourced vulnerability discovery by proposing a word embedding-based clustering method for skill representation. The key to our approach is the representation of skills phrases and task keywords in the same semantic space to minimize any differences and offer the best mapping between them. To this end, by combining different and multiple skills-related information, we create an embedding space that incorporates the syntactic and semantic relationship between skills, SecPros expertise and vulnerability discovery tasks. The clustering algorithm further grouped them in semantically correlated groups. Furthermore, we have conducted experiments that demonstrate the effectiveness of our approach in finding the promising SecPros for vulnerability discovery tasks. These encouraging results open up opportunities for improving people-to-task assignment in crowdsourced vulnerability discovery processes and programs. Directions for future work include the use of additional sources that can help

improve our skills representation model as well as as the integration of other indicators as identified in [1].

Acknolwedgement. This research was done in the context of the first author's Ph.D. thesis [22]. We thank Scientia Prof. Boualem Benatallah for the useful feedbacks provided on this work.

References

1. Al-Banna, M., Benatallah, B., Barukh, M.C.: Software security professionals: Expertise indicators. In: 2016 IEEE 2nd International Conference on Collaboration and Internet Computing (CIC). pp. 139–148 (2016)
2. Al-Banna, M., Benatallah, B., Schlagwein, D., Bertino, E., Barukh, M.C.: Friendly hackers to the rescue: How organizations perceive crowdsourced vulnerability discovery. In: PACIS. p. 230 (2018)
3. Bastian, M., Hayes, M., Vaughan, W., Shah, S., Skomoroch, P., Kim, H., Uryasev, S., Lloyd, C.: Linkedin skills: large-scale topic extraction and inference. In: Proceedings of the 8th ACM Conference on Recommender systems. pp. 1–8 (2014)
4. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of Machine Learning Research **3**(Jan), 993–1022 (2003)
5. Christopher, D.M., Prabhakar, R., Hinrich, S.: Introduction to information retrieval. An Introduction To Information Retrieval **151**(177), 5 (2008)
6. Council, N.R., et al.: A database for a changing economy: Review of the Occupational Information Network (O* NET). National Academies Press (2010)
7. Dave, V.S., Zhang, B., Al Hasan, M., AlJadda, K., Korayem, M.: A combined representation learning approach for better job and skill recommendation. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management. pp. 1997–2005. ACM (2018)
8. Dehghan, M., Abin, A.A.: Translations diversification for expert finding: A novel clustering-based approach. ACM Transactions on Knowledge Discovery from Data (TKDD) **13**(3), 1–20 (2019)
9. Finifter, M., Akhawe, D., Wagner, D.: An empirical study of vulnerability rewards programs. In: Proceedings of the 22nd USENIX conference on Security. pp. 273–288 (2013)
10. Giboney, J.S., Proudfoot, J.G., Goel, S., Valacich, J.S.: The security expertise assessment measure (seam): Developing a scale for hacker expertise. Computers & Security **60**, 37–51 (2016)
11. Ha-Thuc, V., Xu, Y., Kanduri, S.P., Wu, X., Dialani, V., Yan, Y., Gupta, A., Sinha, S.: Search by ideal candidates: Next generation of talent search at linkedin. In: Proceedings of the 25th International Conference Companion on World Wide Web. pp. 195–198 (2016)
12. Hata, H., Guo, M., Babar, M.A.: Understanding the heterogeneity of contributors in bug bounty programs. In: 2017 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM). pp. 223–228. IEEE (2017)
13. Hughes, S.: How we data-mine related tech skills (2015), `https://insights.dice.com/2015/03/16/how-we-data-mine-related-tech-skills/?ads_kw=idf`
14. Javed, F., Hoang, P., Mahoney, T., McNair, M.: Large-scale occupational skills normalization for online recruitment. In: Twenty-Ninth IAAI Conference (2017)

15. Kaufman, L., Rousseeuw, P.J.: Finding groups in data: an introduction to cluster analysis, vol. 344. John Wiley & Sons (2009)
16. Kivimäki, I., Panchenko, A., Dessy, A., Verdegem, D., Francq, P., Bersini, H., Saerens, M.: A graph-based approach to skill extraction from text. In: Proceedings of TextGraphs-8 graph-based methods for natural language processing. pp. 79–87 (2013)
17. Liu, M., Wang, J., Abdelfatah, K., Korayem, M.: Tripartite vector representations for better job recommendation. arXiv preprint arXiv:1907.12379 (2019)
18. Maillart, T., Zhao, M., Grossklags, J., Chuang, J.: Given enough eyeballs, all bugs are shallow? revisiting eric raymond with bug bounty programs. Journal of Cybersecurity **3**(2), 81–90 (2017)
19. Malladi, S.S., Subramanian, H.C.: Bug bounty programs for cybersecurity: Practices, issues, and recommendations. IEEE Software **37**(1), 31–39 (2019)
20. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
21. Miller, G.A.: Wordnet: a lexical database for english. Communications of the ACM **38**(11), 39–41 (1995)
22. Mumtaz, S.: People Selection for Crowdsourcing Tasks: Representational Abstractions and Matching Techniques. Ph.D. Thesis, School of Computer Science and Engineering, Faculty of Engineering, UNSW Sydney (2020)
23. Mumtaz, S., Rodriguez, C., Benatallah, B.: Expert2vec: Experts representation in community question answering for question routing. In: International Conference on Advanced Information Systems Engineering. pp. 213–229 (2019)
24. Mumtaz, S., Rodriguez, C., Benatallah, B., Al-Banna, M., Zamanirad, S.: Learning word representation for the cyber security vulnerability domain. In: 2020 International Joint Conference on Neural Networks (IJCNN). pp. 1–8. IEEE (2020)
25. Mumtaz, S., Wang, X.: Identifying top-k influential nodes in networks. In: the 26th ACM International Conference on Information and Knowledge Management. pp. 2219–2222 (2017)
26. Potter, L.E., Vickers, G.: What skills do you need to work in cyber security?: A look at the australian market. In: Proceedings of the 2015 ACM SIGMIS Conference on Computers and People Research. pp. 67–72 (2015)
27. Shankaralingappa, D.M., De Fransicsi Morales, G., Gionis, A.: Extracting skill endorsements from personal communication data. In: Proceedings of the 25th ACM International on Conference on Information and Knowledge Management. pp. 1961–1964 (2016)
28. Wang, Z., Li, S., Shi, H., Zhou, G.: Skill inference with personal and skill connections. In: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers. pp. 520–529 (2014)
29. Zhang, C., Tao, F., Chen, X., Shen, J., Jiang, M., Sadler, B., Vanni, M., Han, J.: Taxogen: Unsupervised topic taxonomy construction by adaptive term embedding and clustering. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 2701–2709 (2018)
30. Zhao, M., Javed, F., Jacob, F., McNair, M.: Skill: A system for skill identification and normalization. In: Twenty-Seventh IAAI Conference (2015)
31. Zhao, M., Grossklags, J., Liu, P.: An empirical study of web vulnerability discovery ecosystems. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. pp. 1105–1117 (2015)
32. Zhou, W., Zhu, Y., Javed, F., Rahman, M., Balaji, J., McNair, M.: Quantifying skill relevance to job titles. In: 2016 IEEE International Conference on Big Data (Big Data). pp. 1532–1541. IEEE (2016)