An Application of Text Mining and Association Rule Mining to Job and Skill Recommendations for IT Jobs

Napat Cheepmuangman[1] Puttimait Viwathara[2]
Pakkapond Pipattanasookmongkol[3] Rangsipan Marukatat[4*]

[1,2,3,4*]*Department of Computer Engineering, Faculty of Engineering, Mahidol University, Nakhon Pathom, Thailand*

*Corresponding Author. E-mail address: rangsipan.mar@mahidol.ac.th

Received: 17 July 2023; Revised: 22 October 2023; Accepted: 6 November 2023
Published online: 28 June 2024

Abstract

This research implemented a web application that employed text mining to extract skill requirements from online IT job announcements, and association rule mining to discover the co-occurrence of hard skills (technical abilities) and soft skills (personal competencies) specified by the jobs. The matching score of each job was calculated by comparing hard skills extracted from the job announcement with a user's current hard skills. Jobs were recommended to the user based on their matching scores. In addition, the discovered association rules were used to recommend new skills as follows: (1) based on the user's current hard skills as antecedents, new hard skills as consequences would be recommended; and (2) based on the user's current hard skills or soft skills as antecedents, new soft skills as consequences would be recommended. Online training courses to obtain such new skills were also recommended. The application was evaluated by 40 users, and received high satisfaction scores on both job recommendation and skill recommendation.

Keywords: Association rules, Hard skills, Jobs, Soft skills, Text mining

I. INTRODUCTION

It has been common for job seekers these days to search for job announcements on the Internet. But it is not easy to go through all search results to find jobs whose skill requirements match their skills. Moreover, if unable to identify sets of skills essential for the jobs, they may not know which skills they lack and should further acquire. For information technology (IT) jobs, sets of required technical skills often include specific technologies or tools, which can be different across different companies. For example, Data Analyst jobs in some companies require {Machine Learning, Python, TensorFlow}, whereas similar jobs in other companies require {Machine Learning, Cloud Computing, Microsoft Azure}. These requirements also change constantly in response to new technology development. In addition to technical or hard skills, different sets of soft skills are desirable for different IT jobs. For example, Critical Thinking and Presentation skills are advantageous for Data Analysts, whereas Conflict Management and Leadership skills are primarily desirable for Software Project Managers. Therefore, if we can extract all skills specified in numerous IT job announcements, we are likely to see frequent co-occurrence of certain hard and soft skills. By comparing skills that they already have with sets of skills frequently required together, the job seekers can pursue suitable skill training in order to prepare themselves for the job market.

The above can be formulated as an association rule mining problem. Itemsets in our context are skill sets. Our transactions or market baskets contain skills specified in the job announcements. This transactional dataset can be obtained from text mining. To realize the ideas, our research developed a web application *WorkWork* that retrieved online job announcements, performed text mining and association rule mining, and recommended jobs as well as skill training to users.

The rest of this paper is organized in the following sections. Section II reviews related research. Section III presents an overview of *WorkWork*, and elaborates on our job and skill recommendations. Section IV reports the results of association rule mining that were utilized by *WorkWork*, followed by the user evaluation of *WorkWork*. Finally, Section V concludes the paper.

II. LITERATURE REVIEW

This section reviews research on skill requirement analysis, followed by job and skill recommendations. First, we categorize skills into 2 main groups:

1) Hard skills: They are technical knowledge and abilities that are essential to perform specific jobs.

2) Soft skills: They are personal competencies that enable us to handle human and social aspects of the jobs, in order to accomplish the jobs [1], [2].

Wowczko [3] employed K-nearest neighbors (KNN) to classify IT job advertisements into 7 classes, which are: Administrator, Analyst, Developer, Engineer, Lead, Support, and Tester. Then, skills extracted from the advertisements in each class were simply displayed as word clouds.

From job advertisements in the Job section of StackOverflow.com, Papoutsoglou *et al.* [4] constructed 3 datasets of explicit hard skills (skills to use specific technologies such as Python), implicit hard skills (areas of expertise such as Machine Learning), and soft skills. After that, they applied exploratory factor analysis (EFA) to each dataset to find latent factors in the data. Each of these factors consisted of correlating skills or skills simultaneously needed in the job advertisements. But when all types of skills were analyzed together, they found significant correlation only between the explicit and implicit hard skills. Detecting soft skills in the advertisements was difficult because words describing them were general and vague. Hence, the soft skills were too sparse to yield any significant correlation.

Hiranrat and Harncharnchai [5] followed the skill categorization in [4] to analyze skill requirements for Software Development jobs posted on 2 popular job portals: Indeed.com and Jobsdb.com. Only English posts were collected and divided into 8 groups, which are: Business Analyst, System Analyst, Data Analyst, Software Architect, Designer, Developer, Tester, and Project Manager. They reported the top 5 of explicit hard skills, implicit hard skills, and soft skills needed for each group. They also used KH Coder [6] to build co-occurrence networks of skills found in the job posts.

Instead of detecting skills by comparing extracted words with lists of predefined skills (as in [4], [5]), Fareri *et al.* [2] adapted supervised named entity recognition (NER) to determine whether extracted entities referred to soft skills. Although their paper emphasized on soft skills, their published tool SkillNER was able to identify both soft and hard skills from text.

Regarding recommendation methods, there are 2 main approaches to recommending items to users. The content-based approach compares item attributes with the users' own profiles, while the collaborative filtering approach relies on ratings from other users who rate previous items similarly. In the context of job recommendation, individual users tend to have very few ratings on previous jobs and individual jobs are usually taken by single persons. So, the content-based approach would be better suited to our purpose. For example, Almalis *et al.* [7] built job profiles containing a number of skills for IT jobs. Each skill for each job was assigned a required numerical competency level. They also built job candidate profiles containing the candidates' skills and skill competency levels. Then, matching scores between the jobs and the candidates were calculated based on Manhattan and Euclidean distances. These scores could be used to recommend best matching candidates to jobs, or vice versa. Their method had one limitation that it needed numerical skill competency values, and the values determined by different job recruiters and candidates were rather subjective.

Upadhyay *et al.* [8] constructed a knowledge graph where users and jobs were graph nodes. Each node was associated with a document that was either user profile or job description. Edges between nodes were based on document similarity. From this graph, they recommended jobs to a user by retrieving jobs that were both connected to that user (via user-job edges) and to similar users (via user-user-job paths).

As for skill recommendation research, an example is Gugnani *et al.* [9]. After extracting hard skills from a user profile, they calculated the user's proficiency of each skill by considering the number of job roles in which the skill was used, along with the duration and the recency of each role. They retrieved related skills from IT skill ontology and created a skill graph for that user. From the skill graph, the user was recommended paths to acquire more advanced skills. Their method, however, required very detailed user profiles and a comprehensive skill ontology.

Some other works [10]–[12] employed collaborative filtering methods to recommend training courses to users, but their recommendations differed from ours. Our research aims to recommend new skills based on the job requirements in the job market, rather than on ratings of other users.

III. RESEARCH METHODOLOGY

Our web application *WorkWork* consisted of a front-end client, a back-end server, and a database. The front-end web client, interacting with users, was developed by using AngularJS framework (opensource) for JavaScript, HTML, and CSS. The back-end server, executing data collection, text mining, and association rule mining was developed by using Django framework (opensource) for Python. PostgreSQL (opensource) was used as the relational database engine.

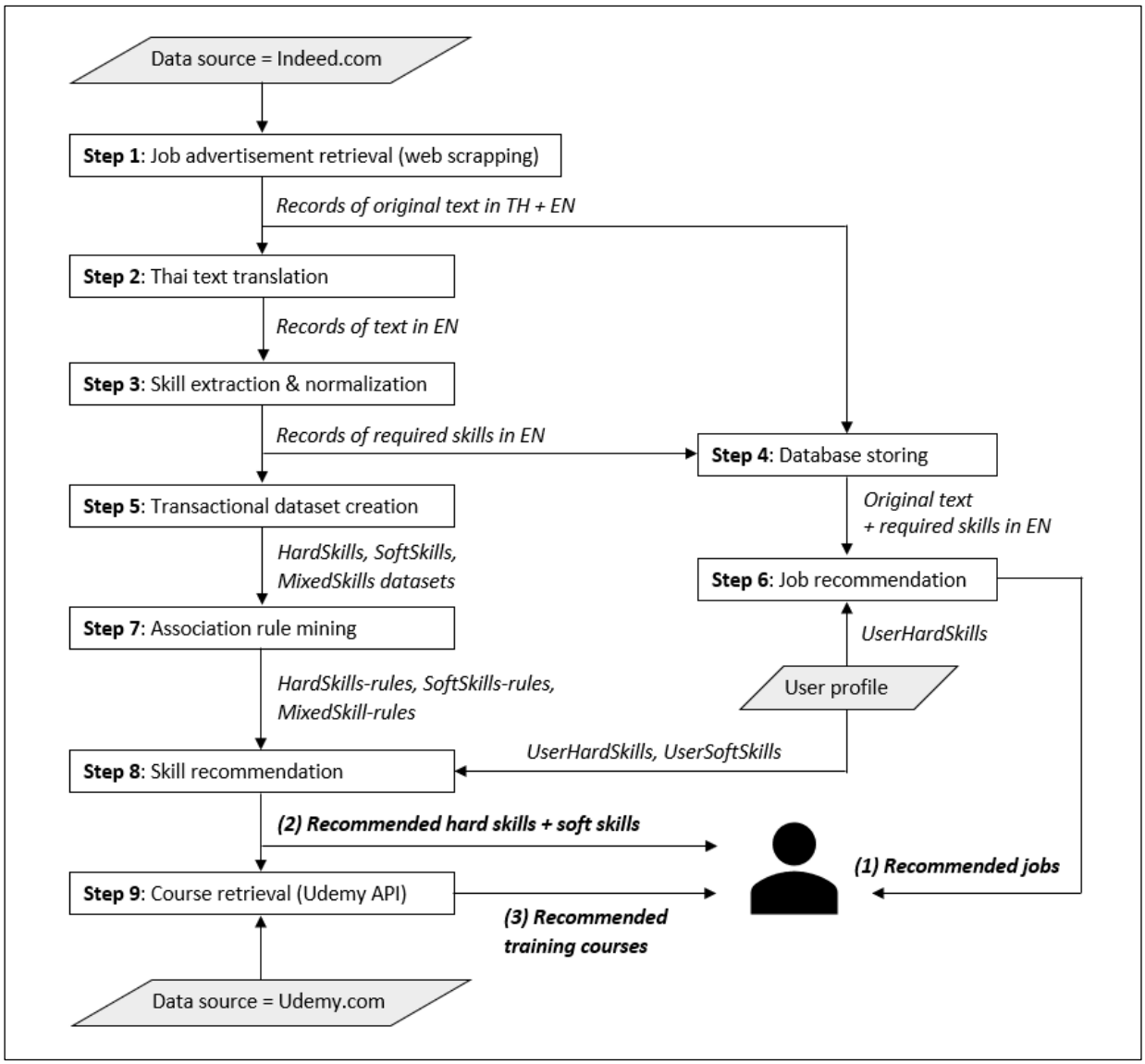

Figure 1: System diagram of *WorkWork*

Figure 1 illustrates the system diagram of *WorkWork*. After signing up, users could update their hard skills and soft skills in the User Profile Page. *WorkWork* offered 2 types of recommendations to the users.

1) Job recommendation. Job announcements, in both Thai and English, containing required hard skills that matched the users' hard skills were recommended together with their matching scores. This will be explained in Subsection B.

2) Skill recommendation. New skills and online courses on Udemy.com were recommended based on the users' skills and association rules discovered from skill datasets. This will be explained in Subsection C.

A. Data Preparation by Text Mining Process

Our research emphasized on recent IT jobs being offered in Bangkok, Thailand. First, we established lists of in-demand skills and constructed skill datasets from which association rules could be discovered. This was done by steps 1-5 in the system diagram:

1) IT jobs advertised on Indeed website during July 2019 and March 2023 were collected. We used 31 common IT job names from Skills Framework for the

Information Age (SFIA) [13] as keywords to search for the jobs. Scrapy and Beautiful Soup packages were used to scrape and clean the data, respectively. This step yielded a set of 1980 job announcements. Note that the web scrapping was employed because Indeed API we initially planned to use was unavailable during our project implementation.

2) Job announcements in Thai were translated into English by using Googletrans package, in order to consolidate the same skills written in different languages.

3) Hard skills and soft skills were extracted by using SkillNER package. SkillNER did stemming and lemmatization internally, in order to map the extracted skills to normalized skill words in all uppercase letters. For example, skills related to statistics that were written differently were all mapped to STATISTICAL skill. The outputs of this step were a list of 151 hard skills and a list of 80 soft skills. They would be displayed in the User Profile Page for the users to select skills they had.

4) For each job announcement, its original text (as retrieved in step 1), job title, company name, and the extracted hard skills and soft skills (in normalized skill words) were stored in the database.

5) As summarized in Table 1, three transactional skill datasets were created. Each transaction represented a job announcement, and items were skills found in that announcement. Note that SoftSkills and MixedSkills had only 1669 transactions as some job announcements did not specify any soft skill.

Table 1: Transactional skill datasets

Dataset	Types of Items in the Transaction	Number of Transactions
HardSkills	Only hard skills	1980
SoftSki lls	Only soft skills	1669
MixedSkills	Both hard and soft skills	1669

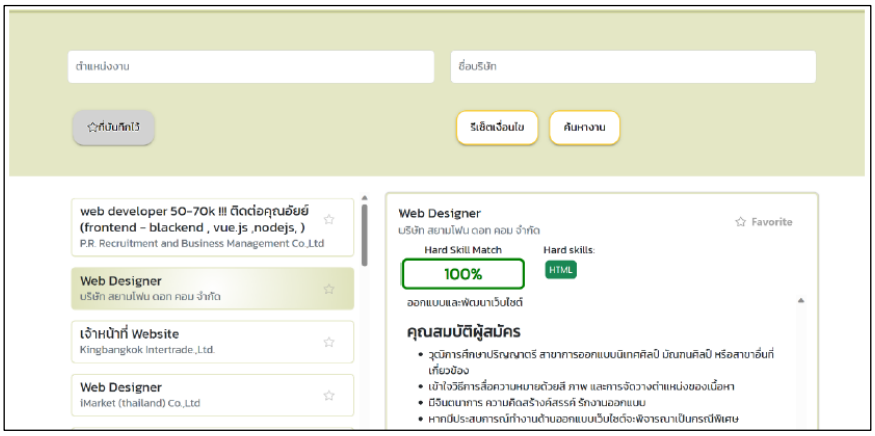

Figure 2: Jobs recommended to a user in Job Seeking Page

Figure 3: Matching score between a user and a job

B. Job Recommendation

Job recommendation was done by step 6 in the system diagram. An example of recommended jobs is shown in Figure 2. The left pane of the user's Job Seeking Page listed brief job announcements (with job titles and companies) whose hard skill requirements matched the user's. They were sorted decreasingly by matching scores. The user could also click a brief announcement to read its details which would be shown on the right pane. The top of the right pane listed all skills found in the announcement, with highlighted ones being the skills that the user had.

Figure 3 shows another job that required {AJAX, CSS, HTML, JAVA, XML} for hard skills and {COOPERATE, CREATIVITY, PATIENCE} for soft skills. Among them, {CSS, HTML, CREATIVITY} were highlighted as they matched the user's skills. We calculated the matching score as follows.

$$Matching\ score = \frac{Number\ of\ matched\ hard\ skills}{Number\ of\ required\ hard\ skills} \times 100 \qquad (1)$$

Only hard skills were considered because they were crucial for the job. Hence, the matching score for the job in Figure 3 was (2/5 x 100) = 40%.

Besides recommended jobs, the users could search others by using job title or company name, or both, as keywords. Skill matching and matching scores would also be reported for retrieved job announcements.

C. Skill Recommendation by Association Rule Mining

Skill recommendation was done by steps 7-9 in the system diagram. An example of recommended skills is shown in Figure 4. We used Apyori package for Apriori association rule mining. Let the rules obtained from their respective datasets be called HardSkills-rules, SoftSkills-rules, and MixedSkills-rules. In the case of MixedSkills-rules, we selected only the ones with hard skills as antecedents and soft skills as consequences.

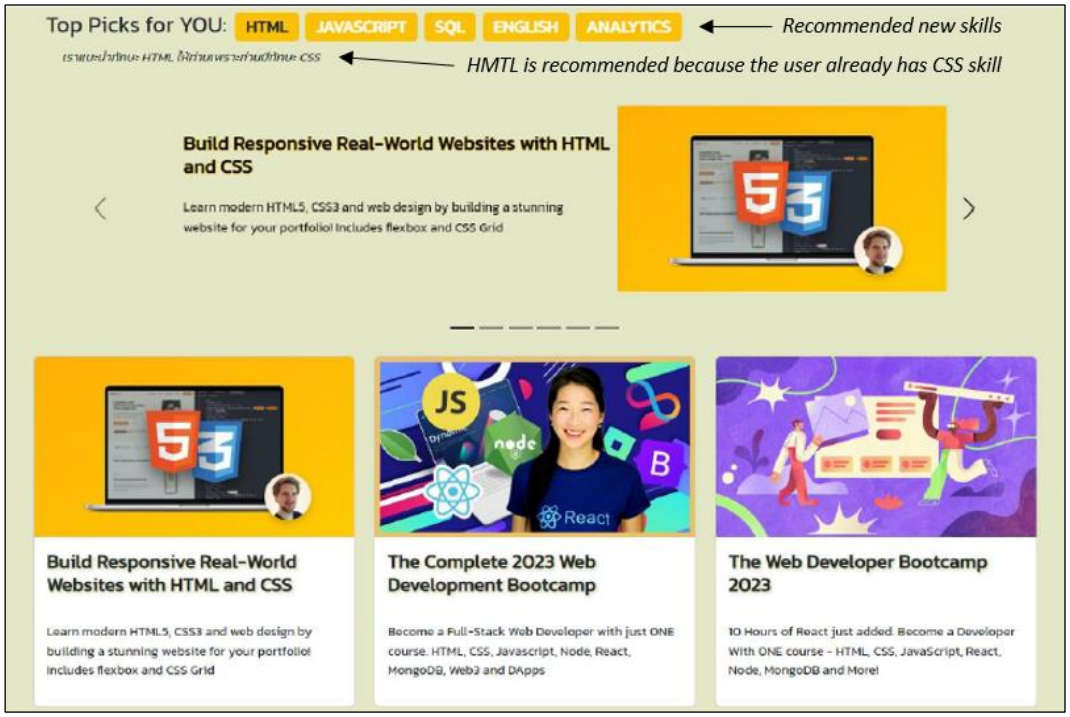

Figure 4: Skills and training courses recommended to a user in Training Courses Page

```
### Number of HardSkills-rules ###
df_hardskill_asso_pass.shape
```

```
(39, 5)
```

```
### HardSkills-rules listing ###
df_hardskill_asso_pass
```

	Antecedent	Consequence	Support	Confidence	Lift
0	[CSS]	[HTML]	0.091414	0.766949	6.545514
1	[HTML]	[CSS]	0.091414	0.780172	6.545514
2	[CSS]	[JAVASCRIPT]	0.086364	0.724576	4.207217
3	[JAVASCRIPT]	[CSS]	0.086364	0.501466	4.207217
4	[HTML]	[JAVASCRIPT]	0.081818	0.698276	4.054505
5	[PYTHON]	[SQL]	0.079293	0.526846	2.173238
6	[CSS]	[JAVASCRIPT, HTML]	0.069697	0.584746	7.146893
7	[HTML]	[JAVASCRIPT, CSS]	0.069697	0.594828	6.887477
8	[CSS, HTML]	[JAVASCRIPT]	0.069697	0.762431	4.427018
9	[JAVASCRIPT, CSS]	[HTML]	0.069697	0.807018	6.887477
10	[JAVASCRIPT, HTML]	[CSS]	0.069697	0.851852	7.146893
11	[SCRUM]	[AGILE]	0.057071	0.849624	4.659988
12	[PHP]	[JAVASCRIPT]	0.056566	0.562814	3.267953
13	[JQUERY]	[JAVASCRIPT]	0.045455	0.750000	4.354839
14	[ANGULAR]	[JAVASCRIPT]	0.042424	0.636364	3.695015
15	[JQUERY]	[CSS]	0.039899	0.658333	5.523305

Figure 5: Top 15 HardSkills-rules by Support

Figure 5 shows some HardSkills-rules. They can be interpreted as follows.

The first rule (Rule 0) tells us that if a job required CSS skill (Antecedent), it would also require HTML skill (Consequence), with 0.766949 or 76.69% Confidence. The Support value indicates the rule coverage, which is about 9.14% of job announcements in the dataset. The Lift value greater than 1 confirms that both sides of the rule are positively correlated.

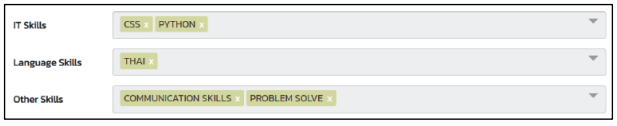

Figure 6: User's skills in User Profile Page

Skill recommendation can be illustrated as follows. Suppose that a user profile is as Figure 6.

1. Sets of the user's skills were identified as:

UserHardSkills = {CSS, PYTHON}

UserSoftSkills = {THAI, COMMUNICATION SKILLS, PROBLEM SOLVE}

2. HardSkills-rules with antecedents being subsets of UserHardSkills were selected. For example, Rules {0, 2, 5, 6} in Figure 5 would be selected for this user. According to each rule, consequence skills not yet earned by the user would be recommended, with the antecedent being the reason for such recommendation. As shown in Figure 4, {HTML, JAVASCRIPT, SQL} were recommended hard skills. The reason for recommending HTML (displayed upon clicking it) was because the user already had CSS skill according to Rules 0 and 6.

3. Soft skill recommendation was done in the same manner as the above. SoftSkills-rules with antecedents being subsets of UserSoftSkills and MixedSkills-rules with antecedents being subsets of UserHardSkills were selected. Soft skills appearing in the consequences of these rules, that were not yet earned by the user, would be recommended.

Furthermore, the recommended skills were used as keywords to search relevant courses on Udemy, a popular online skill training platform. Alternatively, the users could search other training courses by using skills or course names as keywords. Course searching and retrieval were done via Udemy API.

IV. RESULTS AND DISCUSSION

A. Results of Association Rule Mining

This Subsection reports the results of association rule mining. Rules that satisfied minimum Support, minimum Confidence, and minimum Lift thresholds were derived by Apriori. There were 3 conditions for setting Support and Confidence thresholds for each dataset, as summarized in Table 2, in order to obtain sufficient rules for skill recommendation. Rules with lower Support, representing rarer cases, were expected to have higher Confidence to ensure their usefulness. Lift threshold was set to 1 in all cases to ensure that only positively correlating rules were discovered.

Table 2: Apriori parameters

	Minimum Support (i.e., Transaction Coverage)	Minimum Confidence
HardSkills		
- Condition (1)	6.06% (120 transactions)	50%
- Condition (2)	4.04% (80 transactions)	60%
- Condition (3)	3.03% (60 transactions)	70%
SoftSkills		
- Condition (1)	6.00% (100 transactions)	50%
- Condition (2)	4.50% (75 transactions)	60%
- Condition (3)	3.00% (50 transactions)	70%
MixedSkills		
- Condition (1)	6.00% (100 transactions)	50%
- Condition (2)	4.50% (75 transactions)	60%
- Condition (3)	3.00% (50 transactions)	70%

There were 39 HardSkills-rules, 18 SoftSkills-rules, and 9 MixedSkills-rules being discovered in total. In this Subsection, we discuss top rules in each category.

Among the top 15 HardSkills-rules in Figure 5, we found that most of them contained skills related to web applications such as {CSS, HTML, JAVASCRIPT, PHP, JQUERY, ANGULAR}. They were explicit hard skills to use specific languages or frameworks. This enabled us to recommend precise skill training to the users. There was also a rule indicating the requirement for SCRUM and AGILE project management skills.

The obtained HardSkills-rules were dominated by only a few obvious skills. To see some other rules, we lowered Support threshold to 1.05% (20 transactions) and Confidence threshold to 50%. Rules containing {CSS, HTML, JAVASCRIPT, PHP, WEB} were filtered out. Figure 7 shows additional rules being discovered. We found rules about software deployment that required skills to use DOCKER and KUBERNETES. Although both of them could run on any operating system, a few jobs preferred LINUX. We also found a rule that required JENKINS for continuous software integration along with GIT for version control. Unfortunately, these rules had too low Support and Confidence, and thus were not used for skill recommendation. In the future, if more data are collected, these rules may be frequent enough to pass the thresholds and be utilized.

```
### HardSkills-rules with too low supports (not passing thresholds) ###
pd.set_option('display.max_colwidth', None)
df_hardskill_asso_lowsupport.head(15)
```

	Antecedent	Consequence	Support	Confidence	Lift
76	[MACHINE LEARNING]	[STATISTICAL]	0.027273	0.556701	8.478985
99	[R]	[STATISTICAL]	0.025253	0.561798	8.556612
119	[JENKINS]	[GIT]	0.024242	0.631579	8.173375
163	[PYTHON, MACHINE LEARNING]	[STATISTICAL]	0.022727	0.671642	10.229621
167	[R, PYTHON]	[STATISTICAL]	0.022727	0.633803	9.653304
168	[STATISTICAL, PYTHON]	[R]	0.022727	0.642857	14.301766
174	[KUBERNETES]	[LINUX]	0.022222	0.511628	6.708763
214	[ARTIFICIAL INTELLIGENCE]	[PROFESSIONAL SERVICE]	0.021212	0.636364	9.473684
225	[ASP NET]	[C #]	0.020707	0.719298	8.092105
226	[BOOTSTRAP]	[JQUERY]	0.020707	0.577465	9.528169
239	[JENKINS]	[DEVOPS]	0.020202	0.526316	7.776905
242	[AGILE, DOCKER]	[KUBERNETES]	0.020202	0.625000	14.389535
318	[KUBERNETES, DOCKER]	[LINUX]	0.018687	0.544118	7.134788
319	[LINUX, DOCKER]	[KUBERNETES]	0.018687	0.637931	14.687249
358	[PYTHON, MACHINE LEARNING]	[R]	0.018182	0.537313	11.953715

Figure 7: HardSkills-rules with too low Support

```
### Number of SoftSkills-rules ###
df_softskill_asso_pass.shape
```

```
(18, 5)
```

```
### SoftSkills-rules listing ###
df_softskill_asso_pass
```

	Antecedent	Consequence	Support	Confidence	Lift
0	[THAI]	[ENGLISH]	0.112626	0.638968	1.833562
1	[PROBLEM SOLVE]	[ANALYTICS]	0.097475	0.634868	2.479368
2	[COMMUNICATION SKILLS]	[ENGLISH]	0.090909	0.584416	1.677019
3	[ANALYTICAL SKILL]	[ANALYTICS]	0.050000	1.000000	3.905325
4	[PROBLEM SOLVE, ENGLISH]	[ANALYTICS]	0.046970	0.624161	2.437552
5	[THAI, ANALYTICS]	[ENGLISH]	0.042929	0.833333	2.391304
6	[COMMUNICATION SKILLS, ANALYTICS]	[ENGLISH]	0.039899	0.642276	1.843054
7	[CONSULTING, THAI]	[ENGLISH]	0.038384	0.873563	2.506747
8	[PROBLEM SOLVE, CONSULTING]	[ANALYTICS]	0.033838	0.705263	2.754282
9	[PROBLEM SOLVE, INTEGRITY]	[ANALYTICS]	0.031818	0.724138	2.827994

Figure 8: Top 10 SoftSkills-rules by Support

```
### Number of MixedSkills-rules ###
df_mixedskill_asso_pass.shape
```

```
(9, 5)
```

```
### MixedSkills-rules listing ###
df_mixedskill_asso_pass
```

	Antecedent	Consequence	Support	Confidence	Lift
0	[PROJECT MANAGEMENT]	[ENGLISH]	0.070202	0.538760	1.546006
1	[SUSTAINABLE]	[INNOVATIVE]	0.061616	0.659459	3.248084
2	[PROFESSIONAL SERVICE]	[CONSULTING]	0.061111	0.909774	4.393545
3	[PYTHON, SQL]	[ANALYTICS]	0.047475	0.598726	2.338220
4	[STATISTICAL]	[ANALYTICS]	0.046970	0.715385	2.793810
5	[MACHINE LEARNING]	[ANALYTICS]	0.036869	0.752577	2.939059
6	[R]	[ANALYTICS]	0.034343	0.764045	2.983844
7	[PYTHON, STATISTICAL]	[ANALYTICS]	0.031313	0.885714	3.459003
8	[PYTHON, R]	[ANALYTICS]	0.030303	0.845070	3.300275

Figure 9: MixedSkills-rules by Support

Among the top 10 SoftSkills-rules in Figure 8, we found that many jobs required ANALYTICS and ENGLISH skills together. There was some redundancy such as on both sides of Rule 3, as SkillNER did not consolidate ANALYTICAL SKILL and ANALYTICS into the same skill. Finally, among MixedSkills-rules in Figure 9, we found that hard skills related to data analysis (antecedents) were often required along with ANALYTICS soft skill.

We reckon that while the users could easily claim and prove that they had certain hard skills, they may struggle to do so for soft skills. Recommending new soft skills based on both the current soft skills (according to SoftSkills-rules) and current hard skills (according to MixedSkills-rules) helped alleviate a problem of lack of soft skills in the user profiles.

B. User Evaluation of WorkWork

We deployed *WorkWork* on Heroku.com cloud platform during April and May 2023. Third-year and fourth-year students in IT-related fields including: Computer Engineering, Computer Science, Information Technology, Industrial Design, and Robotics Engineering were invited to test and evaluate the application. Forty of them submitted their evaluations via Google Form. Two aspects of the application were evaluated: the user interface design and the usability of various modules. This Subsection reports the user evaluations on Job Recommendation (Job Seeking Page) and Skill Recommendation (Training Courses Page). Evaluation criteria and rating scores in five-point Likert scale are displayed in Table 3. Let the rating score 5 be strong satisfaction and 1 be weak satisfaction, the average score for each criterion was calculated as follows.

$$Average\ rating\ score\ = \frac{\sum_{rating=1}^{5}(rating \times number\ of\ raters)}{Total\ raters} \quad (2)$$

Table 3: User evaluation results

Evaluation Criterion	Average Score
Job Seeking Page	
1. Jobs are recommended according to your skills.	4.50
2. Jobs retrieved by keyword searching match your expectation.	4.65
3. Job announcements shown in the page have sufficient details and are organized appropriately.	4.40
4. Checking the skill requirements of each job is easy.	4.65
5. The overall satisfaction of this page	4.60
Training Courses Page	
6. Recommended new skills are appropriate and related to your current skills.	4.525
7. You agree with recommended courses.	4.15
8. You are interested in recommended courses.	3.75
9. Courses retrieved by keyword searching match your expectation.	4.525
10. Courses shown in the page have sufficient details.	4.10
11. The overall satisfaction of this page	4.425

In summary, the users were very satisfied with the Job Seeking Page, and slightly less satisfied with the Training Courses Page. Job recommendation (Criterion 1) and skill requirement analysis (Criterion 4) functions received high rating scores. Although the users agreed with recommended new skills (Criterion 6), they were

not much interested in recommended training courses (Criterion 8). Their criticism was that the courses were only in English. In fact, there were a lot of Udemy courses in Thai. But when using skills in English words as keywords, the API returned only courses in English.

It should be noted that the above survey results were from nearly-graduated students who had not yet entered the job market. Hence, their expectation and satisfaction on job and skill recommendations may differ from those of mid-career workers.

C. Comparison with Other Works

This Subsection compares our work with the others that were reviewed in Section 2. First, the others [3]–[9] processed only English text, but we processed both English and Thai text. By translating Thai text into English, we could extract skills in normalized English words and create consolidated datasets for further analysis. But as reported previously, skills in English words could retrieve only Udemy courses in English.

Our job recommendation was based on the matching between users' skills and skills required by jobs, as in Almalis *et al.* [7] and Upadhyay *et al.* [8]. But unlike [7], we did not take skill competency into account because most job announcements did not explicitly specify them, and the users' self-evaluated skill competency would be subjective. Our similarity calculation was applied to skills already extracted from the user profiles and job announcements, not on full text documents as in [8]. Hence, we believe that our matching would be more precise and incur less computation overhead in real time. Nevertheless, our method still had a limitation that it was based on the exact matching of skill words. If the user had {PYTHON, WEB PROGRAMMING} skills and a job required DJANGO or FLASK, the application could not infer that PYTHON and WEB PROGRAMMING would also be required for this job. To enable this ability, we need to know the hierarchy and relationship between skills. Skill ontology proposed by Gugnani *et al.* [9] can be helpful for this task.

Our skill recommendation was based on the association rules discovered from skill datasets. The other works also discovered skills frequently required together, but in other forms such as word clouds [3], factors [4], clusters [2], [5], and co-occurrence networks [2], [5], [6]. Our association rules were well-organized and easily interpretable. We could use antecedent items (i.e., the users' current skills) to predict or recommend consequence items (i.e., new skills). But without the skill ontology, we could recommend only individual skills, not skill acquisition paths as in [9].

V. CONCLUSION

This research employed text mining to extract skill requirements from online IT job announcements, and association rule mining to discover the co-occurrence of hard and soft skills. Results from these methods were used to recommend jobs, new skills, and training courses to the users. This allows the users to not only find their best-matched jobs easily, but also to pursue suitable skill training to increase the chance of being employed.

One improvement to our application is to collect online job announcements via APIs instead of web scrapping. But to avoid high overhead of real-time text processing, we propose that this should be done by the back-end server periodically. The processed text and extracted skills can be stored in our database and transactional skill datasets as before. With continuously updated datasets, association rules with more diverse skills or even emerging skills may be discovered.

Other potential extensions are as follows. First, we can use user profiles on social networks or technical platforms to determine skill competency of the users. For example, skills of LinkedIn users can be endorsed

by their colleagues and experts. StackOverflow user profiles contain metrics such as Badges, Upvotes, and Downvotes for their questions and answers about programming. GitHub user profiles contain software contribution metrics such as Followers, Contributions, Repositories, and Stars. From these technical platforms, we can also analyze trends about hard skills that are commonly used in IT tasks, and recommend them to the users.

REFERENCES

[1] A. T. V. Pham and H. T. T. Dao, "The importance of soft skills for university students in the 21st century," in *Proc. 4th Int. Conf. Adv. Artif. Intell. (ICCAI)*, London, U.K., Oct. 2020, pp. 97–102.

[2] S. Fareri, N. Melluso, F. Chiarello, and G. Fantoni, "SkillNER: Mining and mapping soft skills from any text," *Expert Syst. Appl.,* vol. 184, Dec. 2021, Art. no. 115544.

[3] I. A. Wowczko, "Skills and vacancy analysis with data mining techniques," *Informatics*, vol. 2, no. 4, pp. 31–49. 2015.

[4] M. Papoutsoglou, N. Mittas, and L. Angelis, "Mining people analytics from StackOverflow job advertisements," in *Proc. Euromicro Conf. Softw. Eng. and Adv. Appl. (SEAA)*, Vienna, Australia, Aug. 2017, pp. 108–115.

[5] C. Hiranrat and A. Harncharnchai, "Using text mining to discover skills demanded in software development jobs in Thailand," in *Proc. 2nd Int. Conf. Educ. and Multimedia. Technol. (ICEMT)*, Okinawa, Japan, Jul. 2018, pp. 112–116.

[6] Y. Kino, H. Kuroki, T. Machida, N. Furuya, and K. Takano, "Text analysis for job matching quality improvement," *Procedia Comput. Sci.*, vol. 112, pp. 1523–1530, 2017.

[7] N. D. Almalis, G. A. Tsihrintzis, N. Karagiannis, and A. D. Strati, "FoDRA — A new content-based job recommendation algorithm for job seeking and recruiting," in *Proc. 6th Int. Conf. Inf. Intell. Syst. and Appl. (IISA)*, Corfu, Greece, Jul. 2015, pp. 1–7.

[8] C. Upadhyay, H. Abu-Rasheed, C. Weber, and M. Fathi, "Explainable job-posting recommendations using knowledge graphs and named entity recognition," in *Proc. IEEE Int. Conf. Syst., Man, and Cybern. (SMC)*, Melbourne, Australia, Oct. 2021, pp. 3291–3296.

[9] A. Gugnani, V. K. R. Kasireddy, and K. Ponnalagu, "Generating unified candidate skill graph for career path recommendation," in *Proc. IEEE Int. Conf. Data Mining Workshops (ICDMW)*, Singapore, Singapore, Nov. 2018, pp. 328–333.

[10] C. Wang, H. Zhu, C. Zhu, X. Zhang, E. Chen, and H. Xiong, "Personalized employee training course recommendation with career development awareness," in *Proc. Int. World Wide Web Conf. (WWW)*, Taipei, Taiwan, Apr. 2020, pp. 1648–1659.

[11] Z. Pan, L. Zhao, X. Zhong, and Z. Xia, "Application of collaborative filtering recommendation algorithm in internet online courses," in *Proc. 6th Int. Conf. Big Data and Comput. (ICBDC)*, Shenzhen, China, May 2021, pp. 142–147.

[12] Y. Liu, "Research on MOOC course design for college students based on collaborative filtering algorithm," in *Proc. 5th Int. Conf. E-Bus. Inf. Manage. and Comput. Sci. (EBIMCS)*, Hong Kong, Hong Kong, Dec. 2022, pp. 80–84.

[13] I. Seward, "SFIA 8: The global skills and competency framework for the digital world," SFIA Foundation, London, U.K., 2021. Accessed: Jun. 5, 2024. [Online]. Available: https://sfia-online.org/en/sfia-8