Design of Personal Job Recommendation Framework on Smartphone Platform

Chayaporn Kaensar

Abstract—Recently, Job Recommender Systems have gained much attention in industries since they solve the problem of information overload on the recruiting website. Therefore, we proposed Extended Personalized Job System that has the capability of providing the appropriate jobs for job seeker and recommending some suitable information for them using Data Mining Techniques and Dynamic User Profile. On the other hands, company can also interact to the system for publishing and updating job information. This system have emerged and supported various platforms such as web application and android mobile application. In this paper, User profiles, Implicit User Action, User Feedback, and Clustering Techniques in WEKA libraries were applied and implemented. In additions, open source tools like Yii Web Application Framework, Bootstrap Front End Framework and Android Mobile Technology were also applied.

Keywords—Recommendation, user profile, data mining, web technology, mobile technology.

I. INTRODUCTION

ONLINE Recruiting Systems are important components in the Human Resource Management System (HRMS) [1]-[3] because they can optimize the recruitment process and reduce the operational workload, reduce recruitment costs and also provide supplementary information of the job for the employees.

More and more websites tend to provide dozens of job opportunities; along with an increase in a job seekers' personal information and company's recruiting information. As a result, the information becomes overloaded which brings difficulty to web users of finding suitable jobs in the huge amount of online information and leads to low utilization rate as well. Moreover, most recommendation tools for recruitment systems are generally available in the form of web-based applications but do not exist in the form of mobile applications yet. Therefore, The Personalized Job Recommendation Framework, which is an online recruiting system, with personalized recommendations, has been proposed to handle the aforesaid issue for job seekers and companies through both web applications and android mobile platform.

We proposed a framework which can recommend job details according to user profiles using a Clustering Algorithm in WEKA tools (Waikato Environment for Knowledge Analysis) [4] and analyzing implicit user profiles as well. This system can be used for various platforms including web application and android mobile system. Furthermore, the job model can be

updated and it can affect the system directly when recruiters updated their job characteristics.

As a Personalized Job Recommendation Framework, we are mainly focusing on the capabilities of retrieving a list of job titles that satisfy a job seeker's desires, or a list of talented candidates that meet the user's requirements. With regard to profile matching, the recommendation technology and data mining are integrated according to representations of attributes of dynamic user profiles, job characteristics, user skill, working experiences, and companies. Another focus is to provide mobile tools for achieving a recommender system which users can view and operate on the information easily and conveniently.

The outline of this paper is organized as follows: In Section II, we presented a literature review about job recommendation system related works on recommendation methods. Section III introduces the structure of the proposed system and the detailed mechanisms of this system. Section IV shows the design of the proposed model by implementation. Finally, Section V contains some conclusions plus some ideas for future work.

II. RELATED WORK

Recommendation systems are the intelligent techniques to solve information overload problems and provide personalized recommendations e.g. recommending books, CDs, tourism, other products and services or even e-recruitment system.

We have found that the generation of recommendation systems, is mainly related to Information Retrieval [5], [6], Machine Learning [7], [8] and Data Mining Techniques.

Based on an approach for rating estimation, Job Recommender systems have been researched widely. To design more efficient and reliable recommendation methods, several researches are introduced as follows:

Reference [9] is a decision support that aids in the short listing of candidates for jobs by extracting various pieces of information from the unstructured resumes with the help of statistical data driven techniques.

Reference [10] is an intelligent online recruitment service. Initial data is composed of the job database and the log of the job website. To recommend information, they used collaboration filtering and personalized case retrieval. They provide user's feedback. However, no matter user profile or job description contains several features (e.g. salary, location, working experience and so on), they are all represented in the form of one case. It makes them inflexible models because of the increasing number of applied jobs. The extended features should be updated and changed automatically.

Chayaporn Kaensar is with the Faculty of Science, Ubon Ratchathani University, Thailand (e-mail: chayaporn.k@ubu.ac.th).

Reference [11] focused on hybrid user profiling and there are multiple different data sources for the job as input e.g. user's personal information, their behavior, or actions. To do so, not only needs it to observe and record their action in real time, but also to provide different weighting for features. However, the difficult point is how to gain better system performance in the case of recording and processing all user responses, all the time. Furthermore, the aforementioned issues were limited to use in mobile applications for job seekers. Thus, we have found a few research papers is related to this field as follows: Reference [12] presented a SMS-based recommendation system for campus recruitment in China. Profile matching techniques are integrated according to attributes of student and companies. The system embedding SMS-based applications can rise the matching for users. However, lack of user feedback leads to recommendation problems because the system cannot adjust their recommendation results. Moreover, SMS was very limited to plain text presentation. Therefore, to overcome the aforementioned issue for job seekers and companies, we proposed an extended job recommender system framework which can support various platforms both web based application and android mobile technology. It has the capability of providing the appropriate jobs to job seekers and recommending some suitable candidates to the HR companies. To do so, we integrate Clustering Techniques in WEKA tool libraries package using Web Service and JSON message to allow interchange information and also extend the platform for mobile users to access recommendation details as well.
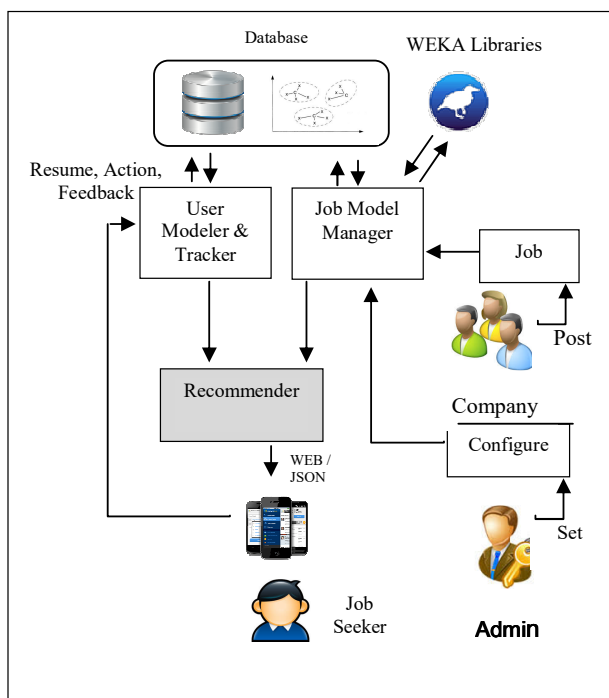

Fig. 1 Extended Job Recommendation System Architecture

III. System Framework

In this work, we proposed a job recommendation framework to support both web-based applications and mobile applications. Here the user first interacts to the system via UI Manager and Tracker (i.e. mobile, web browser). The process of recommending jobs will then be invoked. The job models were built using clustering algorithm from WEKA libraries in Model Manager Module. As a result model, the Job Recommender System used them in order to find a similarity calculation between job categories and user. Moreover, the implicit feedback and user feedback are also considered. The result list was sorted according to the ranking index of each job and shown on user interfaces. The job seeker can use it to filter the job or view more information. Moreover, we provide a subsystem for users to update their profile and job categories model. That is, the model can be updated automatically for keeping the information up to date. In additions, the administrator also configures the model and related parameter to update the model directly as shown in Fig. 1.

The overall design of the proposed system consists of 3 main components as follows:

A. User Interface Module

User Interface module provides both mobile based on web-based applications and Android platform. There are 2 subsystems as:

- Web Based Application. In this part, users can perform various tasks, for example, job seekers can create their resume on Registration Form, set preferences or update their profile. On the other hand, Companies can publish job qualifications and Administrator tasks were also supported. That is, Administrators can set configurations or any parameters to adjust the new model.
- Mobile Application. In this part, we build on Android platforms providing job recommendation information (e.g. job detail, company lists, salary and location)

B. User Modeler and Tracker

1. Gathering User Data

This system infers user interests and allows gathering the relevant implicit feedback at each user interacts to a system. This subsystem includes the rating functions:

a. **Explicit Rating**: This is a simple approach to identify the users' interest by asking the user to give directly a rating of relevant about job information they have browsed. The rating scale can be given on a graded relevance scale that allows the user to define different grade of relevance such as "highly relevant", "relevant", "maybe relevant" or "not relevant". This rating forces user to update their need.

b. **Implicit Feedback Approach**: Sometimes, not all users like to be involved in explicit rating because of data repetitive and frequency. Thus, we employ implicit feedback to learn the user's interest unobtrusively. Reference [11] explains that any interaction between the user and the system can be considered as an estimation of the user's interest. In order to infer the relevant

information from user's behavior, we use such factor (or indicator) because it's widely assumed that they are the most relevant indicator to observe the user behavior [13] as

- User Click or Specific Button Operation perform while reading the screen
- Saving or Bookmarking Action
- Visit the URL link or Printing.

2. Computing User Data

The aim of our system is to recommend from content and behavioral actions of user, his/her degree of interest related to his/her current needs. Therefore, when the user starts apps and submit a query or select the job title from the result, we start to scan the user behavior actions. In the other hand,

To computer the feedback weight including explicit feedback and implicit feedback, the basic formula is represented as:

$$W_f = \alpha X + \mu M \tag{1}$$

where α and μ represents the predefine threshold value of explicit and implicit feedback value. We considered the explicit predefine weight for 0.6 and 0.4 for the other one because the explicit feedback is more accurate than information based on implicit feedback [13].

The procedure can be expressed as:

a) *Explicit Feedback*: To rate the score of "X", we asses a document giving a scale value of the interest from the less relevant, to the more relevant. We considered for the weight to each level, the following values: A maximum weight of 1 is given to "highly relevant", breaks down as follows: 0.75 for "relevant"; 0.5 for "average relevant" and 0.25 for "weakly relevant" and 0.01 for "Irrelevant".

b) *Implicit Feedback;* in particular, for each behavior (Reading (R), Saving (S), Visiting URL and Printing (V), we calculate the actual values of indicators. Thus, we compute the feedback degree based on combination of relevant implicit indicators and explicit feedback. The degree of implicit interest will be computed for all the job titles and the result X depend on the combination of the valued obtained for the three observed behavior. There are 3 steps as follows:

- **Step 1**: Compute the implicit values, for the user u, the degree of interest for any job title. We assume that if the job title is performed (read, saved or visited) by user, It must be consider as relevant and then the associated degree of interest will be equal to M_i, Where i is the number of user behavior for reading, saving and visiting respectively, 0 otherwise.
- **Step 2**: Normalize the weight by the normalization equation (2) and the result is used as the weight of each indication, where W_i is the user behavior such as reading, saving and visiting respectively

$$W_i = \frac{M_i - M_{\min}}{M_{\max} - M_{\min}} \tag{2}$$

- **Step 3**: Compute the final the user's degree of interest for the job and find average value of user u

$$W_u = Average(W_r, W_s, W_v) \tag{3}$$

C. Model Manager

In this module, we first prepared and preprocessed the data. Job details, which were from all the companies in database, were prepared and represented according to groups of job title. Then, we used K-Mean Algorithm in WEKA tools for clustering them. The similar features and closet values were grouped and represented in one model for each job title [14]. Moreover, we also updated and extended database dynamically, based on historical jobs details. That is, when HR insert or update their job profile into database, a new model can be rebuilt and updated according to aforementioned process.

D. Recommender System

The initial model of all job titles were then classified and generated in the Model Manager Module. To recommend job information for a job seeker, we describe the detailed procedure as follows:

- **Step 1**: Calculate the weighting of user's profile attributes which match each job title model using Euclidean Distance (EUD) for similarity computation between target job title model which is from Model Manager and each job seeker profile. The result of top 50 job titles will be sorted and selected as W_{Job} value

$$W_j = \{J_1, J_2, J_3, ..., J_n\} \tag{4}$$

- **Step 2:** Calculate the explicit weight and implicit weight regularly by using the procedure introduced in computing user data
- **Step 3**: Find Sum the total weighting scores and sort their values among W_{job} and W_u. The ranking score of each job was then sent to Ranker Module.

Then, the recommendation result will be sorted and ordered the ranking score of each job. Some brief views on the description of jobs including companies, location, working type will be shown in screen. Here the job seeker can perform on the job information such as viewing, applying and collecting job information.

IV. IMPLEMENTATION AND USER INTERFACE DESIGN

In this section, we implemented some elements that will be used in the system. In addition, the component that discussed as:

A. User Interface Manager

To create web application, we employed PHP and Open Source Yii Framework which is a high-performance PHP framework and fit for developing Web 2.0 applications. It also provides a rich feature set including MVC, DAO/ActiveRecord, Caching, Authentication, and Testing Units. It can reduce your development time significantly [15]. Moreover, we employed Bootstrap for developing responsive,

mobile-first web sites. The application screens were shown in Fig. 2.

Fig. 2 Example of Recommendation Result Screen

As a part of mobile application development based on Android platform, we implemented application for user including access job recommendation, view job detail or find job position using Google MAP API.

B. Model Manager

To build this model, we employed WEKA tools for clustering group data. Firstly, when the data was prepared and uploaded to WEKA tools for data preprocessing according to Data Mining Techniques [14]. We can receive and sent data which is built from WEKA libraries to web application which is implemented by PHP. As a result, we show the obtained data to various screens such as cellphone or browser via web service and JSON message. The process detail was shown in Fig. 3.
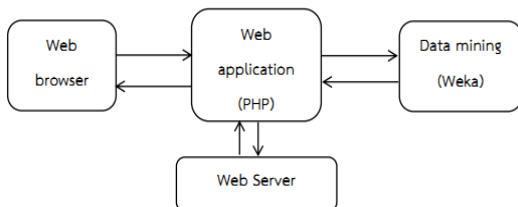

Fig. 3 Workflow of PHP Web Applications and WEKA libraries

To invoke the WEKA libraries, we implemented service modules in Model Manager using PHP. When the result model was done and sent them back to the module in the form of XML File. The example of PHP code for running uploaded WEKA classifiers was shown in Fig. 4. We also provide the configuration screen to update job model as shown Fig. 5. That is, admin, who participated in the survey and collect job data, could update and insert more job information from different organization and industries to the system. They can use the screen to rebuild model to improve further. This make the model can be expanded to train and build. Therefore, the accuracy the survey data will be better.

```php
<?php

function InvokeWEKA_LIB (Output) {

    $cmd = "start cmd /c java -cp weka.jar
            weka.clusterers.SimpleKMeans -N 2 -t
            data\namefile.arff –d
            model\modelname.model ^> result.txt";

    exec($cmd,$output);
    //process and show output…

} ?>
```

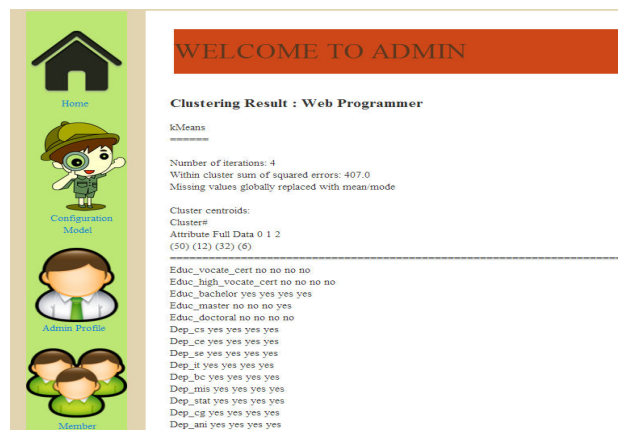Fig. 4 A PHP code for running uploaded Weka classifiers

Fig. 5 Configuration Screen for Administrator

In Fig. 6 (a), user can check job recommendation result, which was sorted according to their weight and their personal involvement in the job information through mobile device. In addition, when user clicked on any job item, the map screen will be displayed to locate job position which blue marker represented current user position and other red markers represented company location which published job and fit for job seekers as shown in Fig. 6 (b).
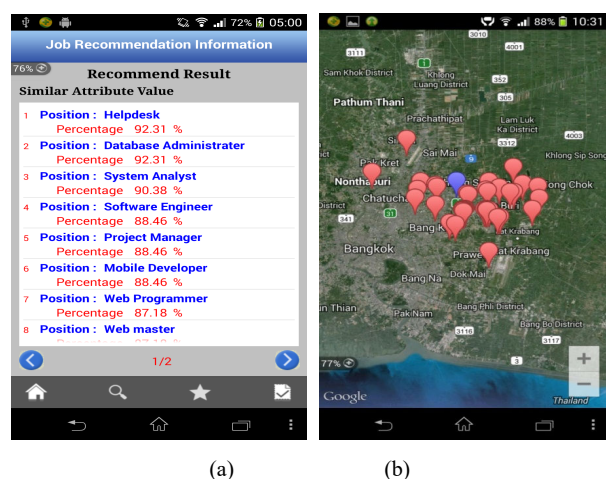

(a)   (b)

Fig. 6 (a) Job Recommendation Result Screen, (b) Job MAP Detail Screen

V. CONCLUSION

This paper demonstrates an integrated and innovative framework solution for supporting web-based application and mobile technology using dynamic user profiles and classification techniques in WEKA libraries, web application technology, web services, and JSON format. This system provides the recommendation of jobs that satisfy the changeable preferences of job seekers. In particular, we allow both user and companies to update their profiles which feature that obtained from a resume or job profile to be changed dynamically.

Due to the time constraint, this paper only focused on IT Software profession which was very limited to be applied in the real world. In addition, to address this challenge, the scope of the IT domain should be broadening to cover infrastructure and hardware part of IT professional. Therefore, application of the model can be used by more users in future. Moreover, we will focus on the more precise matching methods for various attributes representations and extending the application domains such as course management systems, college-student matching, and recommendation. Moreover, we also integrated the system in social networks.

REFERENCES

[1] Lee D. H. and Brusilovsky P.: Fighting information overflow with personalized comprehensive information access: a proactive job recommender. In: Proceedings of the Third International Conference on Autonomic and Autonomous Systems, IEEE Press, USA (2007).

[2] Paparrizos T., Cambazoglu B. B. and Gionis A.: Machine learned job recommendation, In: Proceedings of the fifth ACM Conference on Recommender Systems, pp. 325--328. IEEE Press, Chicago, USA (2011).

[3] Rafter R. , Bradley K. , and Smyth B.: Automated collaborative filtering applications for online recruitment services: Adaptive Hypermedia and Adaptive Web-Based Systems, pp. 363--368, (2000)

[4] Machine Learning Group: Download and Installing WEKA, (updated 2015May, 16), http://www.cs.waikato.ac.nz/ml/weka/.

[5] Baeza-Yates R. A. and Berthier R.-N.: Modern Information Retrieval. Addison-Wesley Longman Publishing Co., Inc., Boston, USA (1999)

[6] Salton G., Wang A. and Yang C.: A vector space model for information retrieval. In: Journal of the American Society for Information Science, pp. 613--620, (2005).

[7] Campos L. M., Fern andez-Luna J. M., Huete J. F. and Rueda-Morales M. A.:, Combining content-based and collaborative recommendations: A hybrid approach based on bayesian networks. In: International Approx. Reasoning, pp. 785--799, (2010).

[8] Salakhutdinov R. and Mnih A.: Bayesian probabilistic matrix factorization using Markov chain Monte Carlo, (2008).

[9] Singh A. , Rose C., Visweswariah K. , Chenthamarakshan V., and Kambhatla N. : PROSPECT: a system for screening candidates for recruitment. In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, pp. 659--668, Toronto, Canada, (2010).

[10] Rafter R., Bradley K., and Smyth B.: Personalised retrieval for online recruitment services. In: Proceedings of the 22nd Annual Colloquium on IR Research, (2000).

[11] Hutterer M.: Enhancing a job recommender with implicit user feedback, In Fakultät für Informatik, Technischen Universität Wien. pp. 107-112 (2011).

[12] Xiangpei Hu et.al,: SMS-based Mobile Recommendation System for Campus Recruitment in China. In: The 10th International Conference on Mobile Business. pp152-157 (2011).

[13] F. Ricci, L. Rokach and B. Shapira, Introduction to Recommendation Systems Handbook, pp. 1–35. Springer, 2011.

[14] Han J.: Data Mining: Concepts and Techniques. 3rd Edition. Elsevier Science & Technology Publisher. (2011).

[15] Yii Web Application Framework, (updated 2015 Jul, 11) http://www.yiiframework.com/.

[16] Bootstrap Front Ed Web Framework, (updated 2015 Jul, 26) http://getbootstrap.com/.