International Journal of Advanced Smart Convergence Vol.9 No.3 145-152 (2020)
http://dx.doi.org/10.7236/IJASC.2020.9.3.145

IJASC 20-3-18

A Recommendation System for Health Screening Hospitals based on Client Preferences

Namyun Kim, Sung-Dong Kim

Professor, School of Computer Engineering, Hansung University, Korea
nykim@hansung.ac.kr, sdkim@hansung.ac.kr

Abstract

When conducting a health screening, it is important to select the most appropriate hospitals for the screening items. There are various packages in the screening hospitals, and the screening items and price are very different for each package. In this paper, we provide a method of recommending the screening packages in consideration of the customer's preferences such as screening items and minimum matching ratio. First, after collecting package information of hospitals, information such as basic items and optional items in the package are extracted. Then, we determine whether the client's screening items exist in the basic item or optional item of the package and calculate the matching rate of the package. Finally, we recommend screening packages with the lowest price while meeting the minimum matching rate suggested by the client. For performance analysis, we implement a prototype for recommending screening packages and provide the experimental results. The performance analysis shows that the proposed approach provides a real-time response time and recommends appropriate packages.

Keywords: *Health Screening, Screening Packages, Screening Items, Matching Ratio, Recommendation System*

1. Introduction

More than 15 million people annually conduct health screening in Korea, and the health screening industry is gradually expanding [1]. Health screenings are conducted to prevent diseases for asymptomatic or high-risk groups, and national health screening and private health screening programs are being conducted [2]. In order to efficiently operate health screenings, it is necessary to conduct personalized health screenings that take into account the individual's age, gender, and risk factors, etc. By doing so, it is possible to minimize the damage of false positives or false negatives and maximize the screening effect. However, the selection of screening items so far is simply based on age and gender and thus there is a problem that screening effect is poor [3, 4]. Meanwhile, there have been various attempts to apply machine learning or deep learning technology to health care service [5, 6]. In particular, we recently proposed a system for recommending personalized screening items in accordance with guidelines of government agencies [5]. As the next step in recommending screening items, a study is needed to select appropriate screening hospitals for screening items. In this paper, we describe

Manuscript Received: July. 14, 2020 / Revised: July. 21, 2020 / Accepted: July. 27, 2020
Corresponding Author: nykim@hansung.ac.kr
Tel: +82-2-760-4348, Fax: +82-2-760-4488
Professor, School of Computer Engineering, Hansung University, Korea

a technique for selecting screening hospitals by reflecting the client's preferences such as the hospitals of interest, screening items, and the minimum matching rate.

A number of attempts have been made to use data mining techniques to enhance the performance of optimization for solving problems in various fields [7, 8]. Optimization finds the best solution to a problem by scoring various solutions to determine their quality. In this paper, we propose a method of finding the minimum price of hospital packages that includes screening items. The consideration here is that since the screening items for each client are diverse, a package including all screening items may not exist. In this case, it is difficult to find an appropriate answer. In this paper, it is defined as the problem of finding hospital packages with the least price while satisfying the minimum matching ratio, which is the minimum ratio that should include screening items. If the clients present the minimum matching ratio as 100%, they can search for packages that include all screening items, and if the minimum matching ratio is 80%, they can search for packages containing more than 80% screening items. In this paper, the clients provide their preferences such as the hospitals of interest, screening items, and the minimum matching rate, and the server recommends packages that satisfy the minimum matching rate.

This paper has three steps to recommend screening packages. First, we collect information on hospital packages in Korea. Based on the collected packages, the basic item, optional item, and maximum selectable count in an optional item are extracted and saved in a file. In the second step, the matching rate of the package is calculated by determining whether screening items are included in the package. In the third step, the packages that satisfy the minimum matching ratio are sorted by price and delivered to the client. For performance analysis, we implement a prototype for recommending a hospitals and provide test results.

This paper is organized as follows. Section 2 presents the system architecture for recommending screening packages and describes the design method of system components. Section 3 presents the performance analysis in terms of response time and recommended package price. Finally, Section 4 discusses our conclusions and proposes future research topics.

2. System Architecture

In order to recommend the packages for the screening items, we propose a system architecture as shown in Figure 1. When offline, the package information extractor processes the packages in the hospital and stores information in an appropriate format. That is, basic items, optional items and the maximum selectable count of optional items for each package are extracted and stored in a file or database. When online, the client transmits information including hospitals of interest, screening items, and minimum matching rate to the server. The minimum matching ratio refers to the ratio at which screening items must be included in the package. The package recommendation server passes this information to the matching ratio calculator, which calculates the matching rate of a package and recommends packages that satisfy the minimum matching ratio suggested by the client. Finally, the package recommendation server sends the recommended packages to the client.

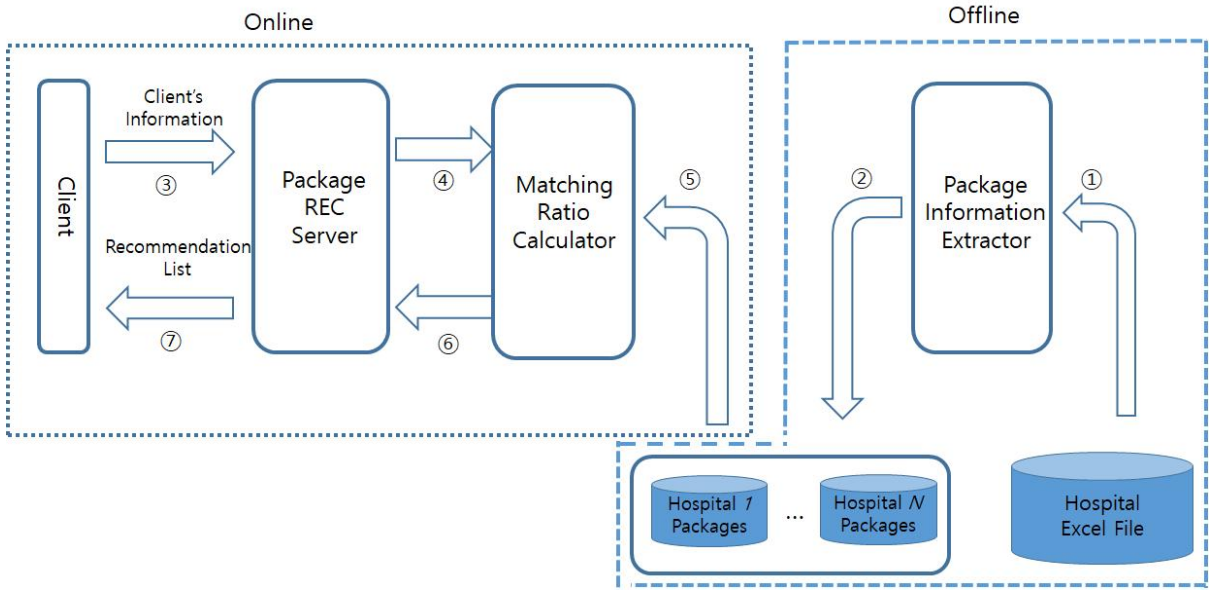

Figure 1. System Architecture for Recommending Screening Packages

2.1 Package Information Extractor

After reading information about the packages of the hospital, a preprocessing step is performed. Table 1 shows an example of a hospital package. It consists of the screening items, code, inclusion (basic item, optional item, not included), and the maximum selectable count in the optional item. A unique code is assigned to the screening item in the package, which is compared with the screening item code of the client for inclusion test. And the inclusion field indicates whether the screening items are in basic items, optional items, or non-included. For example, Gastroscopy (Non-Sleep) belongs to option A in which up to two items can be selected. After reading the package information, the package information extractor extracts the basic item, the optional item and maximum selectable count, and stores them in a file as shown in Table 2. In this paper, Python saves a file in Pickle format.

Table 1. Example of a Package in Screening Hospital

Screening Item	Code	Inclusion	Max Selectable Count
Gastroscopy(Non-Sleep)	2720276	Option A	2
Gastroscopy(Sleep)	2720001	Option A	2
UGI	2720278	Option A	2
Colonoscopy(Non-Sleep)	1390142	No	-
Colonoscopy(Sleep)	1390001	Option C	1
Mammography	1250135	Basic	-
BreastSono	1250266	Option B	1
…	…	…	…

Table 2. Example of Processed Package Information

Name	Value		
Basic Items		{1250135}	
Optional Items	{2720276, 2720001, 2720278}	{1250266}	{1390001}
Max Selectable Count	2	1	1

2.2 Matching Ratio Calculator

In order to identify the most suitable package for the screening items, the clients present their preferences such as hospitals of interest, the screening items, and the minimum matching rate as shown in Table 3. The matching ratio calculator calculates the percentage of the client's screening items included in the hospital's package, and recommends packages whose ratio exceeds the minimum matching rate.

Table 3. Information provided by client

Name	Value
Hospital List	hospital A, hospital B, … , hospital N
Screening Items	1250135, 1580001, 1560164, 1250266, 2720278, 1560164, 2720276, 1390001
Minimum Matching Ratio	80

Figure 2 shows the process of calculating the matching rate of a package. The screening items are {1, 2, 3, 4, 5, 6}, and there are semi-synthesis, synthesis, and VIP packages in hospitals A and B. Each hospital's package is tested for inclusion of the client's screening items. In the semi-synthesis package of hospital A, there are screening items {1,2} in the basic item, screening items {3} in the optional item 1, and screening items {4,5} in the optional item 2. If the maximum selectable count of optional item 2 is 1, only one of item 4 or 5 can be selected. Therefore, the matching rate is 4/6= 66.7%. If the minimum matching rate suggested by the client is 80%, this package is excluded from recommendation. Similarly, by repeating the above process for the remaining packages, the matching rate for each package can be calculated.

In order to determine whether screening items are included in the package's basic or optional items, the basic and optional items in the package are sorted according to code values, and then searched through binary search [9]. Therefore, if the number of items in the package is N, it is possible to search in $O(\log_2 N)$.

After calculating the matching rate of the package, it is necessary to sort the hospitals to preferentially recommend the package with the least price. Since there are multiple packages in a hospital, the price of the package with the minimum value is determined as the price of the hospital. In Figure 2, the price of Hospital A is determined as 240,000, which is the minimum price of the Synthesis and VIP package, and the price of Hospital B is determined as 180,000, which is the minimum price of the Semi-Synthesis, Synthesis, and VIP package. Finally, by providing the order of Hospital B (Semi-Synthesis, Synthesis, VIP) and Hospital A (Synthesis, VIP), the package with the minimum price can be identified first. The pseudocode for calculating matching ratio of packages is shown in Figure 3.

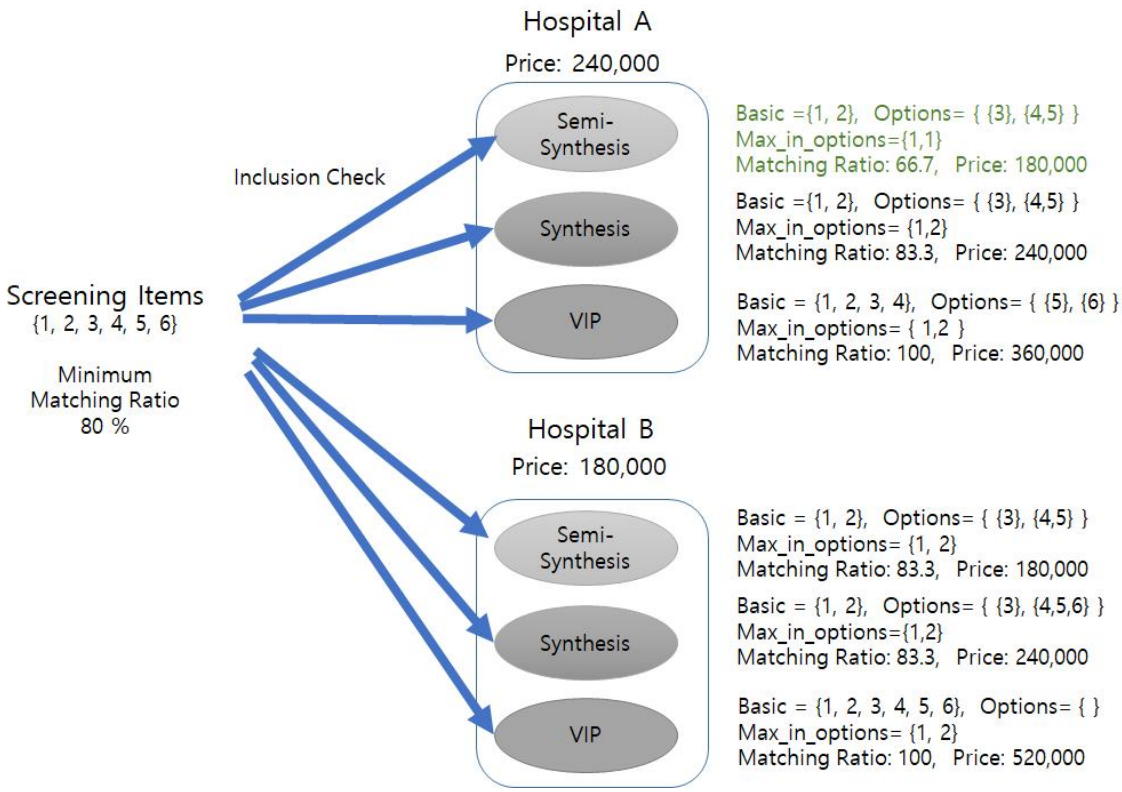

Figure 2. Example showing how Matching Ratio Calculator is performed

```
Input: hospital_list, screening items, min_matching_ratio
for each hospital in hospital_list do
    open a file containing information about the screening hospital
    read number of packages from a file
    while package id < number of packages do
        read basic_items, optional_items, max_in_options, and package_price from a file
        for each item in a list of screening items do
            run binary search to check if an item is included in package (basic or optional)
        end for
        calculate matching ratio for a package
        if matching ratio >= min_matching_ratio then
            store matching result for package in the response message
            (package name, price, matching ratio, included item, not-included items)
        end if
    end while
    determine the minimum price of packages as hospital price
end for
```

Figure 3. Pseudocode for Calculating Matching Ratio of Packages

2.3 Package Recommendation Server

After receiving the list of hospitals of interest, the minimum matching rate, and screening items from the client, the server performs a role of delivering a recommendation list for the hospitals. To do this, it selects packages that satisfy the minimum matching rate after calculating the hospital's package matching rate through the matching ratio calculator. Then the list of hospitals is sorted according to the hospital price. Finally, a server delivers recommended hospitals to the client. The hospital list includes information such as package name, items included in the basic items or in the optional items, items not included, package price, and the matching ratio. The client can select the appropriate hospitals based on this information.

3. Experiment

In this paper, we collected information on 64 packages from 18 hospitals. After extracting information about the package as shown in Table 2, files for each hospital were created. For this experiment, we built a prototype API using Python and the Flask web framework [10] and ran 10 experiments on local machine (MacBook Pro 2.6GHz Quad-core Intel Core i7).

3.1 Server Response Time

The package's matching rate calculation relies on the matching ratio calculator algorithm in Figure 3. If the binary search technique is used to determine whether each screening item is included in the basic item or optional item for package *i*, it takes $(\log_2 b_i + \log_2 o_i)$ time in the worst case. Here, $\mathrm{b_i}$ is the number of basic items of package *i*, and $\mathrm{o_i}$ is the number of optional items of package *i*. Therefore, the time to calculate the matching ratio for *n* items is equal to $\mathrm{n} \times (\log_2 b_i + \log_2 o_i)$ In the end, the matching ratio calculation time for *p* packages can be calculated as $\mathrm{n} \times \sum_{i=1}^{p} (\log_2 b_i + \log_2 o_i)$. For simplicity here, assuming that the number of items in all packages is *m* and the number of basic items (m/2) and the number of optional items (m/2) are the same, the time to calculate the matching ratio of packages is

$$\begin{aligned} \mathrm{n} \times \sum_{i=1}^{p} (\log_2 \frac{m}{2} + \log_2 \frac{m}{2}) &= \mathrm{n} \times \sum_{i=1}^{p} \log_2 \frac{m^2}{4} \\ = \mathrm{n} \times \mathrm{p} \times (2 \log_2 m - 2) &= \mathrm{n} \times \mathrm{p} \times 2(\log_2 m - 1) \end{aligned} \quad (1)$$

After all, the complexity of the algorithm is $\mathrm{O}(\mathrm{n} \times \mathrm{p} \times \log_2 m)$. If the number of screening items *n*, the number of packages *p*, and the number of items in the package *m* are in the range of tens to thousands, it can be said to be a sufficiently computational range. Table 4 shows the results of measuring server response time for 64 packages in 18 hospitals. The response time of the server refers to the time from the request of the client to the delivery of the recommendation list after the server calculates the matching rate. Test cases were classified according to the number of screening items, and the actual response time was about 12 *ms*, indicating that real-time processing is possible.

Table 4. Response Time of Recommendation Server

Test Cases	Number of Screening Items	Response Time (second)
1	5	0.01191
2	10	0.01251
3	15	0.01264

3.2 Package Price

When the minimum matching rate is given, the price of the package recommended by the algorithm and the average price of the randomly selected packages are shown in Table 5. Looking at the table, there is a price reduction effect of more than 250%. In conclusion, it can be seen that the appropriate package is recommended while satisfying the minimum matching rate. In addition, in this paper, there is an advantage that can help clients by providing detailed information on whether or not screening items are included in hospital-specific packages.

Table 5. Package Price: Recommended vs Random Packages (min_matching_ratio=80%)

Test Cases	Price of Recommended Package	Average Price of Random packages
1	180,000	508,333
2	180,000	503,809
3	180,000	601,428

4. Conclusions

In this paper, we presented a technique for recommending hospitals in consideration of the client's preferences such as screening items and minimum matching ratio. We determine whether the client's screening items exist in the basic item or optional item of the package and calculate the matching rate of the package. Then, we recommend screening packages with the lowest price while meeting the minimum matching rate. In addition, by providing screening items included in the basic and optional items in the package, client can review the entire packages and can select the appropriate package. For this experiment, we proposed a system architecture and built a prototype. As a result of the experiment based on the 64 packages in Korea, it was confirmed that real-time response processing is possible and a package with a minimum price can be provided. In the future, based on past screening results, we plan to study the automatic recommendation of screening items and hospitals.

Acknowledgement

This research was financially supported by Hansung University.

References

[1] Hyun Su Kim, et al., "National Screening Program for Transitional Ages in Korea: A New Screening for Strengthening Primary Prevention and Follow-up Care," *Journal of Korean Medical Science*, 2012.
DOI: https://doi.org/10.3346/jkms.2012.27.S.S70

[2] Be-long Cho and Eun-Mi Ahn, "Present Status and Problems of Health Screening Program in Korea", *Health and welfare policy forum*, pp. 48-54, 2013.

[3] The American Cancer Society medical and editorial content team, *https://www.cancer.org/healthy/find-cancer-early/cancer-screening-guidelines/american-cancer-society-guidelines-for-the-early-detection-of-cancer.html*.

[4] Consumer Reports, *https://www.consumerreports.org/men-s-health/mens-health-checklist-for-every-age*.

[5] Namyun Kim and Sung-Dong Kim, "Adaptive Recommendation System for Health Screening based on Machine Learning," *International Journal of Advanced Smart Convergence(IJASC)*, Vol.9, No.2, pp. 1-7, 2020.
DOI: http://dx.doi.org/10.7236/IJASC.2020.9.2.1

[6] Seokmin Han, Suchul Lee, and Jun-Rak Lee, "A Practical Implementation of Deep Learning Method for Supporting the Classification of Breast Lesions in Ultrasound Images," *International Journal of Advanced Smart Convergence(IJASC)*, Vol.8, No.1, pp. 24-34, 2019.
DOI: https://doi.org/10.7236/IJASC.2019.8.1.24

[7] Tomar, N and Manjhvar, A. K. "A survey on data mining optimization techniques," *International Journal of Science Technology & Engineering*, Vol. 2(6), pp. 130-133, 2015.

[8] A. Lim, B. Rodrigues and X. Zhang, "A simulated annealing and hill-climbing algorithm for the traveling tournament problem," *European Journal of Operational Research*, Elsevier, Vol. 174(3), pages 1459-1478, 2006.

[9] Robert Sedgewick and Kevin Wayne, *Algorithms, 4th Edition*, Addison-Wesley Professional, 2011.

[10] M. Grinberg, *Flask Web Development: Developing Web Applications with Python*, O'Reilly Media, 2018.