ISSN 2278-3091
Volume 8, No.6, November – December 2019
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse94862019.pdf
https://doi.org/10.30534/ijatcse/2019/94862019

Towards A Job Recommender Model: An Architectural-Based Approach

Adeagbo M. A. [1], Akhigbe B. I. [2], Afolabi B. S. [3]
[1]Computational Department, First Technical University, Ibadan, Nigeria, adedegy@gmail.com
[2]Computer Department, Obafemi Awolowo University, Ile-Ife, Nigeria, benplus1@gmail.com
[3] Computer Department, Obafemi Awolowo University, Ile-Ife, Nigeria, afolabib@gmail.com

ABSTRACT

The increase in the number of job Websites that have a multiplier effect on the number of online job information, especially in Nigeria is causing information overload in the area of information management for those seeking for a job. There is a need to fetch job information from different job Websites and suggest job information based on the profile of the user. In the architecture presented in this paper, the user profiles was extracted from social networks and job information from job Websites in Nigeria. Jaccard Similarity Coefficient was used to measure the similarity between terms in the profile of the user and job information to make a recommendation of the job for each user. The architecture complies with service-oriented architecture principles and drew on the Jaccard coefficient for similarity measurement, and content-filtering retrieval feature for the recommendation for its formulation. In determining the preset threshold, similarity coefficients of 275 sample of job information were measure against 10 user profile using classification accuracy. The threshold was set to 0.1 and varied up to 1 per job information to determine the point at which the level of correctness was the highest. The result showed that the lowest level of correctness was achieved at 0.1 thresholds while the highest level of correctness was achieved at 0.8 thresholds. The implementation of job Recommender application(s) from this architecture was able to reduce the problem of information overload and asymmetry. Users of the implemented application was also freed from the problem of subscribing to more than one job website.

Key words: Social media, service-oriented architecture, content-filtering, Jaccard coefficient

1. INTRODUCTION

Before the advent of the Internet, there were difficulties in getting to know which company needs employee(s). The only known avenues were through individual connection, television, radio, and newspaper advertisement. Overtime some software developers have come up with the idea of developing Websites for presenting job information using the Internet as a platform. Also, Organizations are embracing technology because of its importance in connecting people quickly and easily [1]. This has made the Internet an increasingly important source from where job information can be found. As a result, more and more companies post their job openings on the Web [2]. According to [3], there has been an increase in the number of job seekers visiting job Websites for job information, and the survey shows that 74.8% of people seeking for job search it online via different job website. The implication is that traditional means of searching for a job such as a poster and newspaper are gradually going on extinction. With the number of existing job Websites and the enormous volume of job information that is accessible on multiple Web pages, job seekers still find it difficult to decide which job website to subscribe to. In addition, locating the right job position within the right time frame could also be very difficult. Users sometimes spend a lot of time looking for a job of interest on Webpages, blogs, and social media platforms. Most times they are forced to visit multiple sources and scan through irrelevant content before finding useful job information.

The foregoing challenges underscore the need to propose a recommendation system, especially as the amount of information job seekers need to sift through grows. This motivation is consistent with the trend in this age of Information Technology, where the concept of personalization is the in-thing. With this concept, information that is tailored towards the need for an information user is easily brought to the user on the fly [4]. This means that the right information about a job vacancy or bidding opportunity will be delivered to match the right candidate [5]. The purpose of this work is to propose an architecture for Recommender systems that provide a reference model for the development of a job recommendation applications. This Architectural Recommendation Model (ARM) that is implementable leverages on the approach provided by Content-based Filtering Algorithms (CbFA). The belief in this paper is that on the implementation of the proposed ARM, RSs that suggest jobs' posting based on the profiles of users will be possible. The motivation for this paper comes from the fact that despite the existence of highly

useful CbFA, there is no systematic ARM to serve as a framework to implement them particularly into RS for a job recommendation. The remaining of the paper is organized with Sections 2, 3, and 4 containing a discourse of some basic concepts, literature review, and the methodology employed respectively. The paper's conclusion is presented in Section 5.

1.1 Basic Concepts

Recommender Systems (RS) is now the main tool for overcoming information overload. These days, they are depended upon to provide users with customized recommendations of personalized items, for example, news, films, music, books, and other relevant pages of Websites [6]. Each medium of information such as newspaper, books, television, notice board, radio, social media has different uniqueness in their techniques of presenting information [7]. The goal of an RS is to bring out useful suggestions for users a collection of things or items that may intrigue them [8]. The domain and pattern of data available determines recommendation engines of the RS to be designed which have a significant impact on the quality of relationship between users and items. For instance LinkedIn is a platform for professionals to display their profile. Also Twitter is a social media platform where user can express their thought, feeling and decision [9]. RSs vary in the manner they break down information to create ideas of relationship between users and items. This can be utilized to recognize well-matched pairs [10]. RS can be Demographic, Content-based, Collaborative or Hybrid [11]. The most two commonly utilized RS algorithms are Content-based and Collaborative [12]. In a real sense, collaborative filtering identifies the pattern of preference in a user network. Conversely, the approach of Content-based filtering, just think about the individual preferences of the user in the past and attempt to learn from its model dependent on features of the content of items representation being recommended [13]. In collaborative filtering systems, it chooses items based on the correlation between people with similar preferences [14]. This means that integrating the social network platform — since it guarantees the existence of the resource of people that are collaborating can ideally enhance the current recommendation system performance [12].

Before, the investigation of knowledge discovery in databases, and all the more explicitly of Recommender frameworks was restricted to the information accessible to researchers. With the advent of the Internet there a huge volume of user data that encompasses users' profiles and preferences than ever. Due to the current social network phenomenon online that has been bought into by users of the Internet [15]; there are lots and lots of social information, which can be used as a user profile to make a recommendation. There are two types of social information: Explicit and implicit social information [16]. Explicit social information is information that is explicitly defined by a user while implicit information is information that captures the behavioral pattern of a user on a system. Cognizance of these concepts the work reported in this paper drew on the theoretics of content-based filtering to propose a model to implement job recommender systems.

2. LITERATURE REVIEW

Motivated by the understanding that RS shows great potentials to assist users with discovering fascinating and significant Web service (WS) from an enormous database, [17] proposed a social trust-aware framework. The system was meant to recommend WS based on the qualities and trustworthiness of social networks. The use of social data and the concept of personalization for a content recommendation has become common for recommendations generally [17], [18]. This has been orchestrated by the rapid proliferation of online information, which has brought about the challenge of Information Overload and Information Asymmetry (IO&IA).

The increase of Web services indicates the subsequent development of the Internet. This makes the Web an avenue for online collaboration, thus exposing users to the provision of too much information or data, which is IO. This also encourages IA, which deals with the decisions made during transactions. In such transactions, one party has more or better information than the other [19], [8], [18]. In the context of our work, employers have all the information. Therefore, there is always an imbalance; that is, job seekers - for instance, do not have as much information about vacancies as employers. But, with the appropriate use of the concept of recommendation, it is believed that relevant jobs, experts, and projects can be recommended, and IO&IA can be managed. As a result, [18] proposed method that explore two-stage approach for recommending project opportunity in Research and Development (R&D). The work identifies suitable R&D project as a possible choice of set using method of information filtering. In another related work, [20] leveraged on the technique of collaborative tagging to propose an item recommendation system to manage the information overload that results from the increase in recommender services from users. The work also examines the possibility of using data fusion to enhance the validity of the recommendation of items. The work demonstrated that combining recommendation sets from various representations profile and functions used for measuring the similarity in user and item in collaborative filtering can enhance the certainty of suggestion.

Moreover, following (e.g. [21]), we argue that in recent years there has been the problem of IO&IA that results from the existence of the massive amount of job information and related data. In the work of [21], movie recommendation was highlighted and the researchers proposed a hybrid model-based movie recommendation system. The system on implementation utilized an improved K-means clustering

technique and genetic algorithms. But, this current work argues that the present level of IO&IA can be taking advantage of to develop a system that assists users to identify the exact job information they need. Many recommendation algorithms have been employed by researchers to implement job recommender. Based on this, clients' profiles on social networks are leveraged to gain the opportunity to create personalized offers and help to cope with IO&IA [19]. Based on the foregoing, [5] developed a job Recommender system that explores job information and profile of users, and the activity of users on the job website to suggest job information. Furthermore, [22] developed a job Recommender system that uses the user's career progression on a social network as a user profile.

The common observed goal that was pursued by the researchers whose work has been reviewed so far is the objective of implementing recommender systems that are adaptable to answer to the requests and needs of users. It is very easy to relate to the foregoing. However, much as the researchers were able to identify specific challenges with respect to the recommendation, such as IO&IA; the theoretic and evidential conceptualization in terms of a systematic approach to implementing the idea of recommendation are absent. There is no system of architecture or layered structure(s) that indicate in one piece what kind of process that is captured in specific modules in terms of the overall structure, logical components and interrelationships, and other conceptions. This type of architecture would project techniques, programs and so on that should be built and how they would interrelate. Its existence will help stakeholders to save a lot of time and other valuable resources when contemplating the development of RS. Furthermore, all the work reviewed so far lay credence to the fact that existing recommender systems demonstrate high performance in terms of accuracy. But, there is no known open source (or non-proprietary) philosophy in this respect. Therefore, the absence of a system architectural model for implementing RS underscores the novelty of the work reported in this paper.

3. METHODOLOGY

This section provides a description of the proposed system architecture, and the various resources it will drew on to be useful to implement a job recommendation information and, the philosophy underpinning its formulation.

3.1 The proposed system architecture

The system architectural presented in Figure 1, consists of three sub modules; the Job System (JS), Profile System (PS) and Recommendation System (RS). The JS is the module that will take care of the extraction of job information from Job Websites that are ubiquitous on the Internet. The job information will include job title, job description, job requirement, and so on. The job information is then processed and transformed from unstructured data to structured data before they are stored in the database. The PS module when implemented will be responsible for extracting user profile from social networks using the graph API - Oauth 2.0 technique. The module will simply prompt the graph API to request the extraction of user profile on social network using user's secret_Id, client_Id and access token. The social network then should verify and authenticate the request by sending data (in the form of user profile) back through the graph API to the PS. Furthermore, the data transformation module will take over and then transform the data into structured data before storing them in the database.

For the RS, it is the module that will recommends job information to users. It will be configured to retrieve job postings and user profiles from the database (see Figure 1. It will pass the outcome of the process to the Recommendation Module (RM). In the RM, the similarity between user's profile and job postings will be measured. The jobs information that will be recommended will be based on similarity measurements, with a threshold point that determines job posting.

3.2 The Job Information Extraction Module

In order to automatically extract relevant information from different job Website some assumption will be made. Firstly, it will be based on the assumption that highlighted and good structures are used to display important information on the web pages. This is because common practice for presenting important information on the website is to enhance it with excellent visual effects. Second is the assumption that the structure and pattern of web pages of a job website are similar. This is because the same or similar cascading style sheet (CSS) is basically used to design web pages. To test that these assumptions paid off; the use of keyword and similar pattern when done should solve the problem of crawling several Websites.

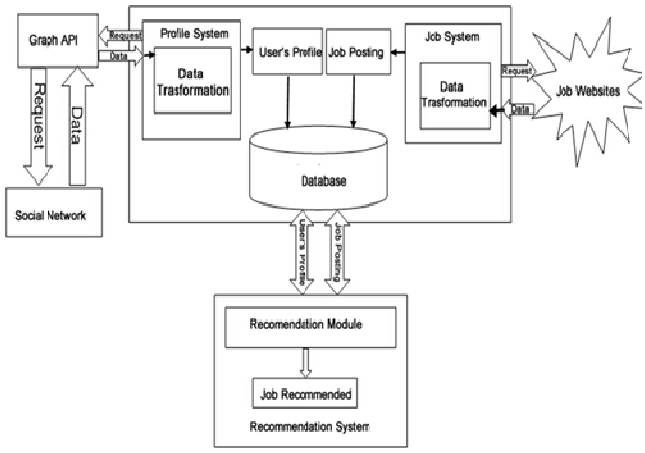

Figure 1: System Architecture

With the foregoing resolved, using the keyword approach for the mining technique is suggested for the extraction value that

has a closed relationship with keywords such as Job Title, Job Description, Qualification, or skill. It will first look for the specified keyword in the page, and when it is identified, the proposed goal of the RS will suggest the approach of heuristic rules based on the sample to be used. The sample-based heuristic rules was suggested here since job recommendation is the focus in this paper. This mines the target information automatically. This rule function using a sample-based mining technique will be needed to extract the information of the user based on a piece of information stated by the user. This is with the assumption that some sections of Web pages are likely to be structurally and style similar. For a typical Intranet of an Institution, all the Web pages of the Institution are layout by the same web designer. Therefore, users can get information (e.g. phone number), by first get the page where the information is located manually, and instruct the system for web pages that are similar automatically. Following best practices like the selector standard of OAuth 2.0 technique [23], users profile will be extracted. The use of OAuth 2.0 protocol that is suggested here is to certify and ensure that the integrity and security of users data are not compromised when access to user profile are demanded and granted. OAuth 2.0 is a framework that allows the application of the third-party to have to define access to the service of an HTTP [23]. The framework was employed to play the role of the owner of a resource, server that authorizes, client, and server. Its adoption will allow the granting of access authorization, the definition of access scope and duration, and the validation of access token to check for expiration. A synopsis of the flow of protocol of the OAuth 2.0 is as shown in Figure 2.

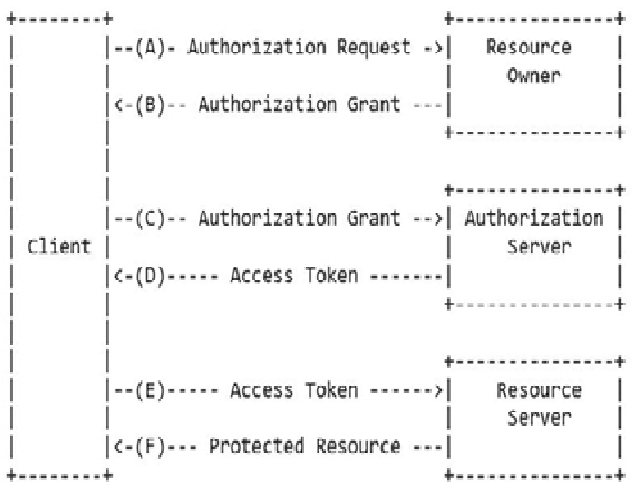

Figure 2: Abstraction protocol flow (Source: Blaine et al., 2012)

In the job recommendation module, the recommendation of job is done with baseline terms that rely on the measurement of similarity function. Considering the goal of this paper, the Jaccard Coefficient is suggested to compute the value that shows the comparison between the similarity and diversity of sample sets. Based on this method the similarity between the content of independent variable, (or user profile) and dependent variable, (or job information) are compared. This can be formally specified as shown in (1) as follows; in equation (1).

$$Jsc = \frac{|X \cap Y|}{|X \cup Y|} \quad (1)$$

Where

Jsc = Jaccard Similarity Coefficient;

X = set of characters of words that appears in the profile of user;

Y = set of characters of words that appears in the advertised job

Y = set of characters of words that appears in job advertisement

The J_{sc} is the score of the similarity ranking of the list of recommendations to be made. For the J_{sc} scores to be useful, all job advertisement is calculated using cosine similarity. For the purpose of this work, this will be achieved using the cosine similarity model, which should allow the formal conceptualization of all occurring key terms. For example, let ϕ be the set of keyterms that occurred in the job titles and descriptions crawled based on the position history in the profile of a user. This history position can formally presented as follows in (2):

$$H\mu \in U \quad (2)$$

Where

H = each user profile;

μ = the number of user profile; and

U = all user profile,

Moreover, (2) means that that is a list of a user's profile. Job advertisement is a sine qua none to job recommendation. This can be model as follows, with;

$$\square = T_j \cup D_j \quad (3)$$

Where

$\square$ = The combination of all the terms (title and description) in the job advertisement

T_j = a list of job title in the job advertisement for recommendation; and

D_j = a list of job description in the job advertisement for recommendation.

The implication of this is that all the terms (title and description) of a job advertisement can be conceptualized as $j \in J$. With the foregoing, the cosine recommendations of a job advertisement to a user score can be computed

as $\sigma_{\upsilon,j}$. The cosine recommendation score is those presented as follows.

$$\sigma_{\upsilon,j} = \frac{|\phi \cap \square|}{\sqrt{|\phi| X |\square|}} \quad (4)$$

The purpose of the formal models in (1), (2) (3) and (4) will be suggested considering the goal of this paper is to return a ranked list of recommendations from $R = [r_1, ..., r_k]$ the lowest to the highest. The recommendation of job is finally done, when a recommendation multiplier adds all the similarity scores and verify the outcome against a preset threshold - a benchmark for the job to be recommended to a particular user. However, any job posting that its score is above the benchmark is then pass to a user as a recommended job.

4. RESULT AND DISCUSSION

The architectural model has been used to develop a recommender system for job information. The user profile was obtained from LinkedIn and membership mechanism, and job information was extracted from various job website. The system studies the job information collected, and applies a pattern and keyword mining approach with a view to transform it from unstructured to an appropriate format for proper recommendation of job information. Classification accuracy metrics was used to set the preset threshold used in the recommendation multiplexer. The study employed method in calculating the level of correctness to determining the threshold to set for the system is presented in equation (5).

$$ACCURACY = \frac{TP + TN}{TP + TN + FP + FN} \quad (5)$$

Where
TP = True Positives;
TN = True Negatives;
FP = False Positives; and
FN = False Negatives.

In determining the preset threshold, similarity coefficients of 275 sample of job information were measure against 10 user profile. The threshold was set to 0.1 and varied up to 1 per job information to determine the point at which the level of correctness was the highest. The result of this process as presented in Table 1 showed that the lowest level of correctness was achieved at 0.1 thresholds while the highest level of correctness was achieved at 0.8 thresholds. This implied that as the threshold increased, the number of true negative and false positive decreased and the level of correctness increased. Also, minimum level of correctness was achieved at the threshold where false positives began to become inferior to true negatives in value. In this study, 0.8 threshold which has 100 percent Level of Correctness (LoC) was therefore used in order to have the highest quality of job information being recommended to a user.

Table 1: The Threshold Using Classification Accuracy metric

Threshold	TP	TN	FP	FN	LoC %
0.1	18	3	254	0	8
0.2	18	48	209	0	24
0.3	18	64	193	0	30
0.4	18	146	111	0	60
0.5	18	192	65	0	76
0.6	18	231	26	0	91
0.7	18	252	5	0	98
0.8	18	257	0	0	100

TP = True Positive, TN = True Negative, FP = False Postive, FN = False Negative, LoC = Level of Correctness.

The sample of login prompt, the result of user profile extracted and the result of the recommendation is shown in Figure 3, Figure 4 and Figure 5 respectively.

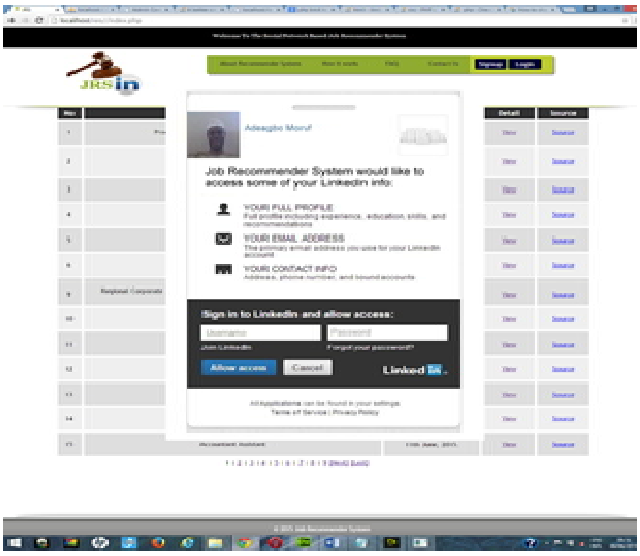

Figure 3: LinkedIn Login

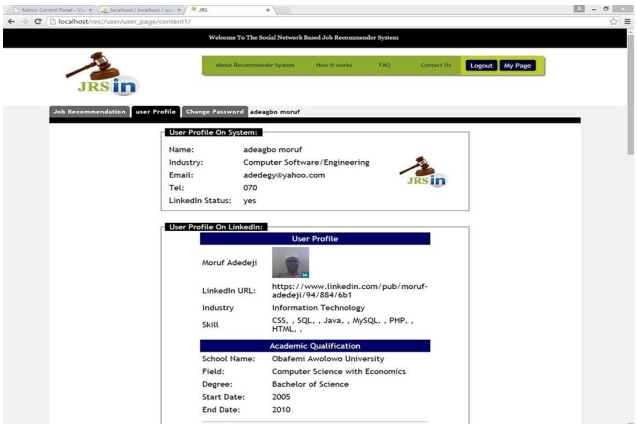

Figure 4: LinkedIn Profile Extracted

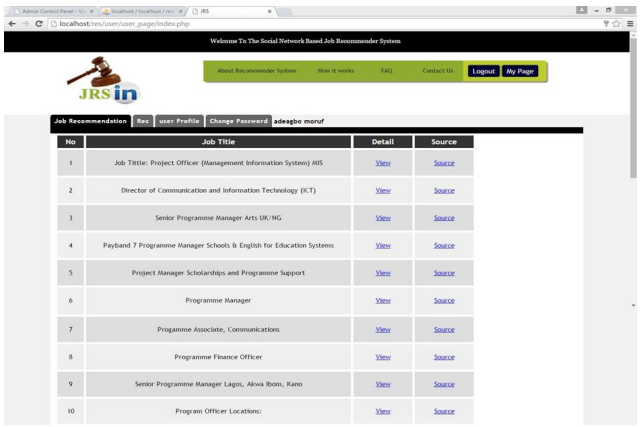

Figure 5: List of Job Information Recommended for a User

The result showed that the architectural model could be used in different approach to recommend information thereby address the problem of information overload.

5. CONCLUSION

A system architecture which is an architectural recommendation model is presented in this paper. As a recommender system architecture, it is meant to provide a reference archetype for the development of job recommendation application(s). The system architecture integrates data sources from social media networks and the Web generally (see Figure 1), drawing on content-filtering technique to bring about the easy use of user profile information. A unique feature of the archetype is that it is flexible. It is not built around retrieval technique of content-filtering alone. It is robust in that it can be adapted to use relevant techniques that will meet the need of software engineers. The usefulness of the proposed system architecture therefore lies in the architecture's provisions with regards to handling the challenge of IO&IA.

Furthermore, the architecture satisfies service oriented principles [24]. This is why it integrates well with social network media and the Web. Thus, it will be easily adopted to support interoperability among heterogeneous components and systems [24]. All these strengths make it plausible to highlight the fact that the job recommender system that will be implemented drawing on the proposed architecture would assists users to be able to receive job information that meets their profile. Based on the aforestated baseline properties of the system architecture software engineers will find the archetype useful to develop job recommender application(s) that will help users get personalized job information. A major limitation of the system is that there is need to improve on the architecture to include rating(s), feedback(s) and other features aside the use of keywords. In order to be consistent with best practices as highlighted in [24], we plan to further improve the architecture specification and develop a framework around it to support the design of applications for job recommendation.

ACKNOWLEDGEMENT

We like to acknowledge all those on top of which shoulders we looked from in order to do this work as we perceive it. We say that without their work this one would not have been able to find it rightful place in the body of knowledge. We also appreciate Obafemi Awolowo University for the space and support provided to do the work from where this paper originated. To all ISRG team members, we say thank you.

REFERENCES

1. Q. C. Kainde, and J. R. Batmetan. **Digital Business Model for Digital Startup in Industrial Era 4.0**, *International Journal of Advanced Trends in Computer Science and Engineering*, Vol. 8, No. 1.5, pp. 177-181, 2019. https://doi.org/10.30534/ijatcse/2019/3481.52019
2. L. Danielle, and B. Peter. **Proactive Comprehensive Access to Jobs**, *Journal of Information Processing Systems*, Vol. 8, pp. 4, 2012. https://doi.org/10.3745/JIPS.2012.8.4.721
3. MyJobMag, **MyJobMag Job Search Survey 2019**: *How Job-Seekers Find Jobs in Nigeria*, March 20, 2019. Retrieved November, 2019 from https://www.myjobmag.com/blog/668/myjobmag-job-search-survey-2019-how-job-seekers-find-jobs-in-nigeria.
4. P.G. Rossi, and L. Fedeli. **Personalization, Adaptivity, Attunement**, *Journal of e-Learning and Knowledge Society*, Vol. 11, pp. 11-24, 2015.
5. L. Yao, E. Sandy, and G. Denis. ***A Recommender System for Job Seeking and Recruiting Website***, React Group, EPFL, Lausanne, Switzerland, 2013.
6. M. Chuanmin, S. Xiaofei, and M. Jing. ***Improved Network-Based Recommendation Algorithm***, A Report Submitted to the College of Economics and Management, Nanjing University of Aeronautics and Astronautics, Nanjing, China, 2014.
7. A. Baharum, N. H. Nazlan, N. A. Mat Noor, R. Hanapi, I. Ismail, and N. A. Mohamed Noh. **Information Visualization for Food Mobile Application: Hangri 2.0**, *International Journal of Advanced Trends in Computer Science and Engineering*, Vol. 8, No.1.3, pp. 306-314, 2019. https://doi.org/10.30534/ijatcse/2019/5781.32019
8. M. Prem, and S. Vikas. ***Recommender Systems***. A Report Submitted to IBM T.J. Watson Research Center, Yorktown Heights, New York, 2009.
9. I. Ferwana, D. Alhenaki, A. Alfayez, and S. L. Marie-Sainte. **Clustering Arabic Tweets for Saudi**

National Vision 2030, *International Journal of Advanced Trends in Computer Science and Engineering*, Vol. 8, No. 1.4, pp. 243-248, 2019. https://doi.org/10.30534/ijatcse/2019/3781.42019

10. M. Madhavan, M. T. Tamil, and M.K. Nandha. **An Intelligent Recommender System for Web User of Technical Community**, *International Journal of Science, Technology and Management*, Vol. 3, pp. 12, 2014.
11. B. T. Poonam, R.M. Goudar, and B. Sunita. **Survey on Collaborative Filtering, Content-based Filtering and Hybrid Recommendation System**, *International Journal of Computer Applications*, Vol. 110, pp. 4, 2015. https://doi.org/10.5120/19308-0760
12. H. Jianming, and W.C. Wesley. ***Social Network-Based Recommender System (SNRS),*** Computer Science Department, University of California, Los Angeles, 2008.
13. L. Pasquale, d. Marco, and S. Giovanni. **Content-Based Recommender Systems: State of the Art and Trends**, *Recommender Systems Handbook*, pp. 73–105, 2011. https://doi.org/10.1007/978-0-387-85820-3_3
14. V.M. Robin, and V.S. Maarten. **Using Content-Based Filtering for Recommendation**, *NetlinQ Group*, Gerard Brandtstraat, Amsterdam, The Netherlands, pp. 26-28. 1999.
15. V.B Cleomar, and A.O. Marcius. **Recommender systems in social networks**, *Journal of Information Systems and Technology Management*, Vol. 8, pp. 3, 2011.
16. A. Joshua, K. Irena, Y. Kalina, P. Luiz, K. Judy, and R. Tomasz. **Explicit and Implicit User Preferences in Online Dating**, *In Proceedings of the 15th international conference on New Frontiers in Applied Data Mining*, pp. 15¬27, 2011.
17. N. Abderrahim, and S. M. Benslimane. **Towards Improving Recommender System: A Social Trust-Aware Approach**, *International Journal of Modern Education and Computer Science*, Vol. 7, pp. 8-15, 2015. https://doi.org/10.5815/ijmecs.2015.02.02
18. W. Xu, J. Sun, J. Ma, and W. Du. **A Personalized Information Recommendation System for R&D Project Opportunity finding in Big Data Contexts**, *Journal of Network and Computer Applications*, Vol. 59, pp. 362-369, 2015. https://doi.org/10.1016/j.jnca.2015.01.003
19. F. Damian, and Z. Radosław. **An Architecture of a Web Recommender System Using Social Network User Profiles for E-Commerce**. *Proceedings of the Federated Conference on Computer Science and Information Systems*, pp. 287–290, 2012.
20. H. Wu, Y. Pei, B. Li, Z. Kang, X. Liu, and H. Li. **Item Recommendation in Collaborative Tagging Systems via Heuristic Data Fusion**, *Knowledge-Based Systems*, Vol. 75, pp. 124-140, (2015). https://doi.org/10.1016/j.knosys.2014.11.026
21. Z. Wang, X. Yu, N. Feng, and Z. Wang. **An Improved Collaborative Movie Recommendation System Using Computational Intelligence**. *Journal of Visual Languages & Computing*, Vol. 25, pp. 667-675, 2014. https://doi.org/10.1016/j.jvlc.2014.09.011
22. H. Bradford, K. Alfred, W. Wayne, B. Mike, and C. Paul. ***Combining Career Progression and Profile Matching in a Job Recommender System***. School of Computer Science and Engineering University of New South Wales, Australia, 2014.
23. C. Blaine, S. Peter, T. Hannes, L. Barry, A. Derek, D. Lisa, and F. Stephen. ***The OAu-th 2.0 Authorization Framework***. Internet Engineering Task Force (IETF), Microsoft Incorporation, 2012.
24. J.F.P. Queiroz, I.R. Guilherme, and R.L. Cagnin. ***Towards a Multi-Agent Architecture for Process Supervision and Control System***. In Proceedings on the International Conference on Artificial Intelligence (ICAI), July 21-24, Las Vegas, USA, 2014.