IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-5, May 2019 E-ISSN: 2347-2693

Integrated User Profiles for Effective Mining in Complex Online Systems

A. K. Shingarwade[1*], P. N. Mulkalwar[2]

[1]Department of Computer Science, College of Management & Comp. Sci., Yavatmal (M.S.), India
[2]Department of Computer Science, Amolakchand Mahavidyalaya, Yavatmal (M.S.), India

Corresponding Author: atul_shingarwade@rediffmail.com, Tel.: 9403015739

DOI: https://doi.org/10.26438/ijcse/v7i5.153159 | Available online at: www.ijcseonline.org

Accepted: 10/May/2019, Published: 31/May/2019

Abstract: User profiles include but are not limited to social media profile, professional online profile, e-commerce profile and search profile. Each individual user nowadays has multiple user profiles, due to the fact that these users are constantly using online and offline services. These profiles are not mutually exclusive as the search habits of a user directly showcase the user's shopping behaviour, and so on. Due to the presence of so many profiles of a single entity, there is a wide research area which has opened up in the recent years. Companies and researchers are harnessing this gap in order to provide better user experience via integrating multiple profiles and helping them to learn from one another. In this paper, we define a framework via which the user's social and e-commerce profiles can be combined in order to better recommend their buying patterns to companies based on the items purchased by the friends which the user's follow closely. Mining positive and negative rules (MOPNAR), firefly, top k rules and association rule mining is used in order to mine the usage patterns, and the results shows that an accuracy of more than 70% is observed when compared with the real time buying patterns.

Keywords: Profile, integrated, online, MOPNAR, firefly, e-commerce, social

I. INTRODUCTION

The sensational development of data on the WWW has accidentally prompted data over-burden and consequently finding a particular snippet of data has turned out to be troublesome and tedious. As techno-keen clients progress from the Social (Web 2.0) to the period of Semantic Web and Internet of Things (IoT), savvy proposal frameworks and client explicit administrations are essential. The prominence of these applications is somewhat founded on the reason of broad personalization to improve client experience. The idea of personalization alludes to the way toward modifying applications and administrations, for customized client encounters. This underscores the necessity for successful client profiling instruments that can catch both coarse-grained and fine-grained client inclinations after some time. Such client profiles may likewise be static or dynamic, where, the static profile never or seldom changes while dynamic profiles every now and again change after some time. Typically, clients who effectively devour application administrations can be displayed successfully by powerful profiling strategies. Some current methodologies [1] [2] [3] [4] [5] are of constrained use because of the way that their client profile age technique is subject to a solitary information source or is outfitted towards a specific area of administration. For instance, internet business is one such space where personalization is unavoidable. A client's profile and perusing/buy history encourages such sites to suggest increasingly important items trying to help deals and income [6][7][8]. In e-learning and u-learning (pervasive learning) conditions, the emphasis is on understanding client's experience, expertise level, capability and so on., with the goal that their learning achievement can be upgraded [9][10]. Such particular utilizations of client profiling are exceedingly engaged and subsequently are not versatile for different applications. Additionally, they frequently come up short on the ability to catch the fleeting and dynamic nature of client exercises.

A huge disadvantage of these frameworks is that they neglect to demonstrate client profiles that can catch their numerous personas, which may exist crosswise over various sorts of Web applications and administrations (like social, proficient, interest related, political perspectives, etc). At the point when client profiles are displayed and made out of different client explicit information sources, they are increasingly finished and consequently, progressively helpful for personalization based applications. An additional preferred standpoint is that a similar profile can be utilized crosswise over applications to give 'n' number of administrations that are customized to client's needs. All together that client profiles be accessible for utilization crosswise over applications, the essential prerequisite is to make any such information accessible in a standard and open configuration. The Semantic Web characterizes a few systems for explanation, distribution and utilization of open information dependent on metadata guidelines and Linked Open Data (LOD). LOD gives a strategy for distributing organized information with the goal that it very well may be interlinked and are accessible for dispersed questioning by semantic applications. The Resource Description Framework (RDF) is the Web standard which is utilized to express asset level metadata in machine justifiable way. Another prominent organization that has broad help of the Search Engine industry is JSON-LD (JavaScript Object Notation-Linked Data). JSON-LD empowers encoding of Linked Data utilizing JSON, along these lines enabling designers to utilize and serialize information in a straightforward, quick and effective way. The created client profile would then be able to be utilized for various applications and administrations like, question noting frameworks, web search tools, learning portrayal and thinking applications, and as a rule, man-made brainpower and Web Personalization applications.

The following area portrays different suggestion procedures dependent on client profiles; trailed by the proposed calculation. Lastly the outcomes and some fascinating perceptions about our work are exhibited in terms of delay and accuracy improvements. Here we define a framework via which the user's social and e-commerce profiles can be combined in order to better recommend their buying patterns to companies based on the items purchased by the friends which the user's follow closely. Mining positive and negative rules (MOPNAR), firefly, top k rules and association rule mining is used in order to mine the usage patterns, and the results shows that an accuracy of more than 70% is observed when compared with the real time buying patterns.

The organization of this paper is as follows. Section 1 contains the introduction of the topic. Section 2 discusses the literature available for writing this paper. Proposed hybrid recommendation engine is given in Section 3. Section 4 describes various results obtained by using this hybrid recommendation engine and analysis of the result. Conclusions are given in Section 6.

II. LITERATURE REVIEW

As the volume and assortment of Web information keeps on expanding exponentially, the assignment of finding applicable data quick is progressively troublesome, antagonistically influencing client experience. This can thwart seek encounters on internet business sites, Web look, e-learning and so on. Proposal frameworks have turned out to be progressively omnipresent in light of these issues. Client profiling is the focal reason of recommender frameworks, and a great part of the work around there is this specific situation. Tao et al [1] proposed a customized cosmology show for clients with the end goal of learning portrayal and thinking over client profiles. The capacity of clients to peruse a record and choose whether it is important to that person is reproduced with the assistance of ontologies alluded to as ontological client profiles or customized ontologies [1]. Tao et al's work likewise suggests that client foundation information can be found in a superior way if both all inclusive and privately investigated learning can be incorporated. The worldwide learning or the world information can likewise be said as the presence of mind learning which incorporates all the essential data and actualities. The neighbourhood learning incorporates client explicit data which is private to the client. The customized metaphysics made gives promising outcomes. The upside of this framework is that the learning investigation isn't confined to a specific nearby or worldwide space however incorporates both. The drawback is that the work expects that all the nearby client vaults will have reference to the subjects which are available in the worldwide information base.

Skillen et al [2] proposed a novel methodology for personalization of Help-On-Demand benefits in unavoidable conditions utilizing Ontological client demonstrating and semantic rule based thinking. The personalization is accomplished through the execution of Ontological client profile displaying. Attributes of a client are broken into lower granularity and displayed into the client profile information structure. The framework effectively gives a model which can be connected onto Context-mindful applications [2]. The constraint of this methodology is that it neglects the particular idea of client ideas in the ontological client profiling. At present, the client profile manages just those few highlights indicated in the nonexclusive profile. Zhao et al [3] proposed a strategy for incorporating ontologies in Linked Open Data. The issues in Linked Open Data, principally the heterogeneity of ontologies, are tackled by the proposed structure framework called Framework for Integrating Ontologies (FITO) [3]. FITO utilizes the ideas of Graph Theory to incorporate comparable ontologies so it is anything but difficult to bring all the datasets inside this metaphysics for semantic web designers. In any case, this framework fizzles when another cosmology relationship becomes possibly the most important factor in the Linked information, the change won't be reflected consequently in the coordinated philosophy as metaphysics coordinating is utilized for incorporation.

Hawalah and Fasli [4] proposed a few strategies to keep up the dynamic idea of client profiles for Web personalization. They followed client interests by grouping it into present moment and long haul interests. The client's perusing information is gathered from which interests are gotten. The creators propose 2 calculations in particular: Gradual Extra Weight (GEW) Algorithm and Contextual Concept Clustering (CCC) Algorithm. GEW calculation chooses how much weightage must be given while including terms in classes as client interests. CCC calculation guarantees that the client intrigues fall in the correct classification or setting after the utilization of GEW calculation. The upside of the framework is that the dynamic idea of client profiles is kept up to an exceptionally decent measure. The downside of this work is that ontological client profiles are excluded. Existing ontologies are utilized to show client profiles, which might need in successfully demonstrating the large number of clients. Phuoc et al [5] set forward a novel methodology towards aggregating a live learning chart of associated things named as Graph of Things (GoT). It utilizes other information sources from Linked Open Data too alongside information from the 'Things'. The SSN Ontology is utilized for the classification of IoT sensors information and makes important relationship mapping from IoT information to the interlinked LOD datasets [5]. Notwithstanding, the absence of client profiles and setting mindfulness inside the framework doesn't make it client explicit or smart. Another vital downside is the absence of security of the ordered information. There are no entrance confinements determined, which implies that anybody can get to delicate physical things information or client information.

Grcar et al [11] planned a framework that models client profiles dependent on his or her perusing history dependent on semantic Web information by gathering the pages visited by client into points utilizing k-implies grouping. The client can see themes and pages related with every subject. The most as of late visited pages are bunched into one single gathering called the present intrigue. The favourable position here is that interests of the client is assembled well, yet the weakness is that the client profile is totally founded on the interests and no other data is accessible in the client profile. Also, no other information source or exercises of the client are considered. In view of our perceptions, there are a few roads that can be investigated for further improving client profiling methods for recommenders and other personalization based applications. Different restrictions of different works like, absence of regard for catching the dynamic and fleeting nature of clients, crosswise over different heterogeneous information sources, use of non-standard/constrained ontologies and no help for information reuse are noteworthy focuses that can influence the client profiling process. Our proposed methodology attempted to relieve these issues by coordinating existing ontologies to show the profile. The client profile created utilizing the proposed framework can go about as an exceptionally decent learning base for personalization and suggestion frameworks or applications as it tends to be distributed and expended by means of an assortment of arrangements including a light-weight, open information design like JSON-LD to the LOD cloud.

III. HYBRID RECOMMENDATION ENGINE

The proposed hybrid recommendation engine is based on the concept of multi-profile integration along with multi-algorithm recommendation. In the proposed recommendation model, the input dataset from both E-Commerce and Social Media is given to the novel clustering method which is already described [12]. The results from this clustering algorithm are processed as follows,

- Results of ecommerce dataset and social media dataset are stored separately
- The results which have same user ID and are in the same cluster set are intersected from both the datasets and stored for further processing
- Rest of the results are discarded, as they are floating entries

The stored results are then given to a MOPNAR algorithm (Mining of Positive and Negative Association Rules), and initial recommendations are observed. The algorithm works as follows [13],

Input: Datasets, minconf and Number of rules M.
Output: Classified Association rules.
1: Perform data pre-processing.

2: Initialize the weight vectors. Generate the initial population with N chromosomes. Initialize the reference point z and the EP.

3: Update: For all N

a) Generate two offsprings by crossover mutation and repairing from a solution of the population.

b) Generate another offspring by selecting at random from the neighbourhood or from population with probability (defined by the user)

4: Use the offsprings to update the reference point. Replace some of the solutions of the current population with worse values for the decomposition approach.
5: These steps are repeated for each solution in population and EP is updated.

6: Calculate the support and confidence for each rule.

7: If the considered rule is valid it is saved in L where L is a set of current Best M rules.

8: Each rule that is frequent is saved in R which is later considered for expansion.

9: Generate the Best M association rules based on minconf and efficiency parameters.

10: return Best M classified association rules
After MOPNAR, we apply the firefly algorithm along with Top K Rules [14] in order to get the best recommendations from the clustered dataset. The Firefly algorithm works as follows,

The Firefly Algorithm is one of the newest meta-heuristics algorithms, therefore there have been written very few articles about it presented. Based on three rules stated in preceding section the pseudo-code of the basic Firefly Algorithm (FA) [15] is illustrated as follows,

```
Begin
Objective function f (x);   x =(x1,......, xd) T
Generate initial population of fireflies xi (i = 1, 2..., n) Light intensity Ii at xi is determined by f (xi)
Define light absorption coefficient g
While (t <MaxGeneration)
For i = 1 : n all n fireflies
    For j = 1 : i all n fireflies
        If (li > lj)
            Move firefly i towards j in d-dimension
        End if
        Attractiveness varies with distance r via exp[¡gr]
        Evaluate new solutions and update light intensity
    End for j
End for i
Rank the fireflies and find the current best
End while
Post process results and visualization
End
```

First each firefly generates an initial solution randomly; Parameters like Light Intensity I, Initial Attractiveness βo, and light absorption coefficient γ are defined.
Then for each firefly, find the brightest firefly among them. Then the less bright firefly move towards the brightest firefly.
When firefly moves or travels its light intensity decreases and its attractiveness among the other firefly will change. Then best firefly will be chosen based on an objective function for the next iteration. This condition will continue until the max iteration is reached.
FA is inspired by the flashing behaviour in the matting phase of fireflies' life cycle in nature. It is developed by Xin-She Yang at Cambridge University in late 2008 [16]. The fundamental function of flashing light in fireflies is to attract a mate. A male or female firefly light glows brighter in order to make it more attractive for a mate. The FA algorithm is presented in algorithm (2). FA uses the following three rules [16]:

- A firefly is attracted to other fireflies regardless of their gender, because all fireflies are uni-gender.
- Attractiveness is proportional to their brightness, thus for any two flashing fireflies, the less– bright one will move towards the brighter one. Both attractiveness and brightness are decreasing as the distance between the two fireflies increases. If no one is brighter than a particular firefly, then it moves randomly
- The brightness or light intensity of a firefly is determined by the objective function of the– optimization problem.

Algorithm 2: Firefly Algorithm

1: Initialize parameters α, β, γ, t=0, Bs=0
2: $P^{(0)} = InitializeFA()$ // Initialize Randomly Firefly population
3: **While** t < Max-Iteration **do**
4: FitnessFA($P^{(t)}$) // calculate fitness value for each solution
5: Bs = BestFA($P^{(t)}$) // order population then find best solution
6: $P^{(t+1)} = MoveFA(P^{(t)})$ // Firefly movement
7: t= t+1
8: $\alpha^{(t)} = NewAlpha$ //calculate new alpha value
9: **End While**
10: **Output** Bs

In Algorithm 2, alpha (α) is the random movement parameter that controls the step length of the random movement, γ is the fixed light absorption coefficient, β is the brightness, t is the iteration number and Bs is the best solution. InitializeFA() function in line (2) is used for initializing the fireflies' population randomly, where each individual contains two attributes; a position and a fitness. The while-loop (lines 3-9) starts with the FitnessFA function in line (4) which is used to calculate the quality of all population solutions. Then, BestFA function in line (5) is used to sort the population of fireflies according to their fitness values. After that, the MoveFA function in line (6) is used to perform a move of the firefly position (the details are presented in algorithm 3). Finally, the NewAlpha function in line (8) is used to decrease the initial value of parameter alpha (α) as the iteration increases. The firefly search process is repeated until we reach Max-Iteration steps. After the loop is terminated, the best solution is obtained.

Algorithm 3: $MoveFA(P^{(t)})$

1: **For** i = 1 **To** n **do** // n is population size
2: **For** j = 1 **To** n **do**
3: $x_i = P_i^{(t)}$.position // array of positions for firefly I at t iteration
4: $x_j = P_j^{(t)}$. Position
5: $$P_i^{(t+1)} = P_i^{(t)}$$
6: **If**(f(x_i)<f(x_j)) **then** //f is attractiveness function
7: $$r_{ij} = \sqrt{\sum_{k=1}^{n} (x_{ik} - x_{jk})^2}$$
8: // move firefly i towards j
9: $x_i^{t+1} = x_i^t + \beta_0 e^{-\gamma r_{ij}^2}(x_j - x_i) + \alpha_t(rand - 0.5)$ // rand is a random number
10: $P_i^{(t+1)}.position = x_i^{t+1}$
11: **End if**
12: **End For**
13: **End For**
14: **Return**($P^{(t+1)}$)

Algorithm 3 shows the steps for the function MoveFA, where line (6) tests the attractiveness (brightness) between two fireflies using the fitness function to determine which firefly is moving and to which one. The firefly with less brightness will move towards the brighter firefly. The rij in line (7) is the distance between any two fireflies i and j at xi and xj positions which is calculated using the Cartesian distance. xik is the kth component of the spatial coordinate xi vector of ith firefly and xjk is the kth component of the spatial coordinate xj vector of jth firefly. The new position xi t+1 of the moving firefly i at t+1 iteration is calculated by line (9) where the step size of the moving firefly i depends on the last two terms which are added to the current position for firefly i at t iteration. The second term is used to control the step size due to the attraction of a firefly towards the intensity of the light (brightness) by neighbouring fireflies. Brightness here is inversely proportional to the distance between the two fireflies due to exponential function characteristics. The brightness is decreasing as the two firefly distance increases. The third term is a randomization vector of random variables, where α is the random movement parameter that controls the step length of the movement. Note that β0 is the attraction factor at rij = 0 and γ is the light absorption coefficient. For most cases β0 = 1, α Є [0, 1] and γ = 1 [13]. Finally, line 14 returns the new population after the movement phase is completed.

The BMPNAR algorithm is used to mine the best M association rules. First the rules are generated with the minsup value. Then the rule expansion is applied to form the final list of rules. The dataset, user number of rules to be generated M and the minconf value are the inputs to the system. As the output we get the time and space required by the algorithm and number of rules generated. Once these best rules are generated they are given to the nature inspired Firefly algorithm for classification [15]. The final output is the Best M classified rules.

Algorithm: Best M Positive Negative Association Rules (BMPNAR)
Input: Datasets, minconf and Number of rules M. Output: Classified Association rules.
1: Perform data pre-processing.

2: Initialize the weight vectors. Generate the initial population with N chromosomes. Initialize the reference point z and the EP.

3: Update: For all N
a) Generate two offsprings by crossover mutation and repairing from a solution of the population.
b) Generate another offspring by selecting at random from the neighbourhood or from population with probability (defined by the user)

4: Use the offsprings to update the reference point. Replace some of the solutions of the current population with worse values for the decomposition approach.

5: These steps are repeated for each solution in population and EP is updated.

6: Calculate the support and confidence for each rule.

7: If the considered rule is valid it is saved in L where L is a set of current Best M rules.

8: Each rule that is frequent is saved in R which is later considered for expansion.

9: Generate the Best M association rules based on minconf and efficiency parameters.

10: Apply the Firefly algorithm.
a) Generate initial population of fireflies.
b) Define objective function.
c) Define light absorption coefficient.
d) Determine light intensity of the firefly by the objective function.
e) For all fireflies if light intensity is greater than any other firefly move it towards that firefly by calculating the distance between them.
f) Evaluate new solutions and update light intensity.

11: return Best M classified association rules
Based on this algorithm, the final results are evaluated on Amazon and Facebook datasets. The next section describes the results in details and compared the results with the standard techniques. It is shown that the proposed algorithm outperforms some of the standard algorithms in terms of recommendation accuracy and delay needed for obtaining the results.

IV. RESULTS AND ANALYSIS

We used this algorithm (BMPNAR) for performing data mining on amazon and facebook datasets. These datasets consists of the user buying patterns and the friends connection of the users, the application of the proposed algorithm is evaluated on these datasets together and following results are obtained,

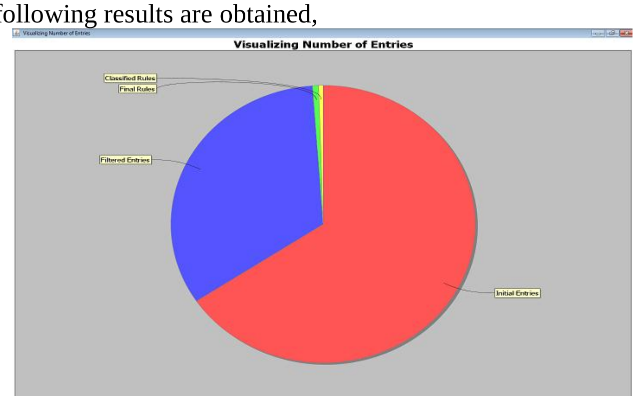

Figure 1: Number of rules for facebook 1000 and Amazon 1000 entries dataset

Similar comparison is done on other datasets of facebook and Amazon, and the following results were obtained,
Item with user ID 8 will buy the item 7 with max quantity of 1.0
Item with user ID 7 will buy the item 8 with max quantity of 2.0
Item with user ID 1 will buy the item 7 with max quantity of 5.0

Item with user ID 3 will buy the item 7 with max quantity of 0.0

From the results, it is recommended that which user from the social media dataset will be buying which particular item from the ecommerce dataset with how much maximum quantity. For example, from the output we can observe that the user number 8 will buy item number 7 with a max quantity of 1, while the user number 3 might not buy item number 7, as its max quantity is 0, thus there are both positive and negative rules which can be observed by the user of the system.

We then evaluated the accuracy and delay needed by our system with the standard algorithms and obtained the following results,

Table 1: Delay comparison between different algorithms

Dataset size	Delay (ms) Kmeans+Apriori	Delay (ms) FCM+Apriori	Delay (ms) Proposed + Apriori	Delay (ms) Proposed+Top K Rules	Delay (ms) Proposed + MOPNAR	Delay (ms) Proposed+MOPNAR+Top K Rules+Firefly
1000	1.56	1.72	1.09	0.87	0.74	0.54
2000	3.78	3.54	2.44	1.95	1.59	1.2
5000	5.91	5.87	3.93	3.14	2.59	1.93
10000	12.54	11.27	7.94	6.35	5.11	3.88
20000	22.8	20.66	14.49	11.59	9.35	7.08

The following table indicates the accuracy of result evaluation for the algorithms, we evaluated the accuracy by first manually finding out the correct recommendations for the datasets and then comparing the number of correct recommendations given by the algorithms,

Table 2: Accuracy comparison of the algorithms

Dataset size	Accuracy (%) Kmeans+Apriori	Accuracy (%) FCM+Apriori	Accuracy (%) Proposed + Apriori	Accuracy (%) Proposed+Top K Rules	Accuracy (%) Proposed + MOPNAR	Accuracy (%) Proposed+MOPNAR+Top K Rules+Firefly
1000	86.2	87.5	91.42	93.02	95.76	98.66
2000	87.5	87.8	92.26	93.88	96.46	99.51
5000	88.6	88.2	93.05	94.69	97.16	99.96
10000	89.7	89.1	94.11	95.76	98.23	99.97
20000	89.9	91.2	95.32	96.99	99.83	99.98

We find that the accuracy has been improved by more than 8% in average when compared with the accuracy of the existing algorithms. Thus, we can suggest that our technique can be used by researchers for obtaining better recommendations in multi domain recommender systems

V. CONCLUSION

The proposed algorithm is tested on social media and e-commerce datasets and it is found that the delay improvement as compared to other algorithms is more than 10% and the accuracy improvement is more than 8%. This work can be applied to any dataset as well, provided the dataset and the user profile information is properly modelled into the system.

REFERENCES

[1] X. Tao, Y. Li, and N. Zhong, "*A customized philosophy display for web data gathering,*" IEEE Transactions on Knowledge and Data Engineering, Vol. 23, No. 4, pp. 496– 511, April 2011.

[2] K. L. Skillen, L. Chen, C. D. Nugent, M. P. Donnelly, W. Burns, and I. Solheim, "*Ontological client demonstrating and semantic standard based thinking for personalisation of assistance on-request benefits in inescapable conditions*," Future Generation Computer Systems, Vol. 34, pp. 97 – 109, 2014, Special Section: Distributed Solutions for Ubiquitous Computing and Ambient Intelligence.

[3] L. Zhao, R. Ichise, "*Cosmology reconciliation for connected information*," Journal on Data Semantics, Vol. 3, No. 4, pp. 237–254, 2014.

[4] A. Hawalah, M. Fasli, "*Dynamic client profiles for web personalisation*," Expert Systems with Applications, Vol. 42, No. 5, pp. 2547 – 2569, 2015.

[5] D. Le-Phuoc, H. Nguyen Mau Quoc, Hung Ngo Quoc, T. Tran Nhat, and M. Hauswirth, "*The chart of things: A stage towards the live learning diagram of associated things*," Web Semantics: Science, Services and Agents on the World Wide Web, Vol. 3738, pp. 25 – 35, 2016.

[6] J. Wang, Yi Zhang, "*Opportunity demonstrate for internet business proposal: Right item; ideal time*," Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, New York, NY, USA, SIGIR '13, pp. 303– 312, ACM, 2013.

[7] K. Christidis, G. Mentzas, "*A subject based recommender framework for electronic commercial center stages*," Expert Systems with Applications, Vol. 40, No. 11, pp. 4370 – 4379, 2013.

[8] J. Lu, Dianshuang Wu, M. Mao, Wei Wang, and G. Zhang, "*Recommender framework application advancements: A study*," Decision Support Systems, Vol. 74, pp. 12 – 32, 2015.

[9] N. Manouselis, H. Drachsler, R. Vuorikari, H. Hummel, and R. Koper, *Recommender Systems in Technology Enhanced Learning*, pp. 387– 415, Springer US, Boston, MA, 2011.

[10] A. Klanja-Milievi, B. Vesin, M. Ivanovi, and Z. Budimac, "*E-learning personalization dependent on half breed proposal methodology and learning style distinguishing proof*," Computers and Education, Vol. 56, No. 3, pp. 885 – 899, 2011.

[11] D. Mladeni, M. Grar and M. Grobelnik, "*Client profiling for intrigue centered perusing history*," Proceedings of the SKIDD, ACM, 2005.

[12] A. K. Shingarwade, P. N. Mulkalwar, "*Reducing Cluster Formation Delay for Real Time Data using Automatic Bisecting Hierarchical Clustering",* Proceeding of Third National Conference on "Recent Advances in Science and Engineering" (NC-RACE 18), pp 1-8, Oct 2018.

[13] D. Martn, A. Rosete, J. Alcala-Fdez,and F. Herrera, "*A New Multiobjective Evolutionary Algorithm for Mining a Reduced Set of Interesting Positive and Negative Quantitative Association Rules*", IEEE Trans. Evol. Computing, Vol. 18, NO. 1, Feb 2014.

[14] A. K. Shingarwade, P. N. Mulkalwar, *"Development of Top K Rules For Association Rule Mining on E-Commerce Dataset***"**, Proceeding of Global Journal of Engineering Science & Researches, pp 304-309, Feb 2019.

[15] X. S. Yang, "Nature-Inspired Metaheuristic Algorithms", Luniver Press, 2008.

[16] X. S. Yang, "*Firefly algorithms for multimodal optimisation"*, Proc. 5th Symposium on Stochastic Algorithms, Foundations and

Applications, (Eds. O. Watanabe and T. Zeugmann), Lecture Notes in Computer Science, 5792: 169-178, 2009.

[17] A. K. Shingarwade, P. N. Mulkalwar, "*Study of Text Content Mining for E-Commerce Web Sites*" International Journal of Advanced Research in Computer Science, Vol 8, No. 4, May – June 2017.

[18] A. K. Shingarwade, P. N. Mulkalwar, *"Comparative Analysis of Clustering Techniques Based on Validity Measures"*, Proceedings of the International Conference on Recent Trends in Science & Technology ICRTST Special Issue No. 25, pp 139-144, 2018.

Authors Profile

A.K. Shingarwade, at present working as Head and Assistant Professor in Computer Science at College of Management and Computer Science, Yavatmal (MS), India. He is awarded with M.Sc. in Computer Science in 1998 by SRTMU, Nanded and MBA in 2005 by YCMOU, Nashik. He is currently a research scholar and doing research in Data Mining and published some research papers in National and International Journals.

Dr. P. N. Mulkalwar, at present working as Head and Associate Professor in Computer Science at Amolakchand Mahavidyalaya, Yavatmal (MS), India. He is awarded with M.Sc. (1983), M. Phil (1995), PGDCS (1987) and Ph. D. (2012) by RTM Nagpur University, Nagpur. He has a teaching experience of 30 years in Computer Science and published various research papers in National and International Journals. SGBAU Amravati has recognised him as a research supervisor in computer Science.

