Context Aware Cross-domain based Recommendation

Hla Hla Moe, and Win Thanda Aung

***Abstract*—** Nowadays, the way of people access and interact with information has been adapted and modified via the Internet. Consequently, an increasing amount of work has been published in various areas related to the recommender system. However, plenty and abundant room and need for further improvements remain in the effective human decision support in a wide variety of applications. Among them, cross-domain recommendation is an emerging research topic. In this research field, there is lack of an important point to investigate how to gather context aware recommendations and how to control personalization for keeping more user satisfaction and accuracy. The paper presents a context aware cross-domain based recommender system to get customer's personalized facial skin care problem recommending suitable cosmetics, considering customer's contextual features. The system is developed by using Taxonomic conversational case-based reasoning (Taxonomic CCBR), Ford-Fulkerson algorithm and technique for ordering preference by similarity to ideal solution (TOPSIS) method.

***Keywords*—** cross-domain recommendation; contextual features; Taxonomic CCBR; Ford-Fulkerson algorithm; TOPSIS method

I. Introduction

TODAY, the internet began increasing and growing up with tremendous speed. There is no user of the World Wide Web who cannot purchase almost any item being in any country of the world. However, the amount of information and items got extremely huge, leading to an information overload. It became a big problem to find what the user is actually looking for [1]. For doing a right decision, customers still encounter a very time-consuming process in visiting a flood of online retailers, and get worthless information by themselves. Sometimes the contents of Web documents that customers browse have nothing to do with those that they require indeed.

That problem is solved by developing search engines. However search engines can solve the problem partially and personalization of information was not given. So developers found a solution in recommender systems. For filtering and sorting items and information, recommender systems are tools [1]. Recommender systems typically provide the user with a list of recommended items they might prefer, or supply guesses of how much the user might prefer each item [2]. The recommendation process can be content-based, i.e., using features of the item liked by the user to predict what the target user may like [4], or collaborative-based, which finds users with similar item preferences and recommend to the target user items liked by these users [7]. They are used as a base of most modern recommender systems.

Most of recommendation systems provide their recommendations only for items from a single domain. In fact, joint recommendations in multiple domains are sometime required for a customer. For instance, a system suggests not only a particular movie but also music CDs and books that are somehow related to that movie. Cross-domain recommendation is defined as providing recommendations of items in one (source) domain using the preferences expressed on items in a second (target) domain [3], [6]. Another task for cross-domain recommendation is making joint recommendations for items belonging to different domains [5]. Cross-domain recommendation models are classified into adaptive models – which exploit information directly from a source domain to make recommendations in a target domain – and collective models – which are built with data from several domains and potentially can make joint recommendations for such domains [2].

Almost of all cross-domain recommendations can give more diverse recommendations leading to a higher user satisfaction and engagement addressing cold-start and sparsity problems but they are generally less accurate than single-domain recommendations [3], [5], [6], [8], [9]. Moreover, most of the available cross-domain recommender systems suggest items regardless of the contextual conditions which can be important to predict the user's preferences at a particular moment.

In this paper, a context aware cross-domain based recommender system is presented. The system is used to apply the useful application area where facial skin problem is solved with personalization and cosmetics related with this problem are recommended considering customer's contextual features accurately in a particular way. Taxonomic CCBR based on ontological properties is used when the domain is created for managing personalization systematically since it allows for a partial definition of the problem by the user and identifies more clearly user's problem providing accurate solution by conversation. Hence, it improves personalization. The Ford-Fulkerson algorithm is applied to build the bridge of the semantic concepts between source and target domains calculating the accurate maximum flow (weight).

Hla Hla Moe is PhD Candidate from University of Technology (Yatanarpon Cyber City) in Myanmar (e-mail: hla2moe@gmail.com).

Win Thanda Aung is Asistant Lecturer from University of Computer Studies (Bahan Campus) in Myanmar (e-mail: winthanda.monywa@gmail.com).

Consideration of customer's contextual features such as season, place and so on is an important role to suggest the recommendations precisely and accurately. TOPSIS method is used for gathering recommendations suitable with customer's contextual conditions, for finding out the final results. The system expects to understand about the advantages of cross-domain recommender systems adding semantic concepts and to get more accurate, precise and personalized recommendations. Moreover, the system intends to know the interesting results of cross-domain recommendations applying TOPSIS method with customer's contextual features such as season, place, etc and to get more profits for commercial sites if the system is used.

II. Related Work

Cross-domain recommendation is now an interesting research field and hence and applies in many application areas and even mobile environments. Therefore, many applications of cross-domain recommendation approaches become of special interest in many e-commerce and retailer websites because it can increase customers' loyalty. Previous works have been proposed in various kinds of ways. Some of previous cross-domain recommender systems are constructed based on music, music artists and related things [3], [6], [10], [11], [13].

Francesco Ricci et al. proposed an approach [3] that pointed toward integrating and exploiting knowledge on several domains to provide cross-domain items recommendations. It automatically extracted information about two domains available in Linked Data repositories, linked items in the two domains by means of a weighted directed acyclic graph, and performed weight spreading mechanisms on such graph to identify matching items in a target domain (music artists) from items of a source domain (places of interest).

Marius Kaminskas and Francesco Ricci addressed a particular kind of cross-domain personalization task [10], [11] with selecting simultaneously two items in two different domains and recommending them together fitting the user preferences. They also proposed an approach [6] which considered contextual conditions such as the user mood or location. It retrieved music that suited the user's interested place using emotional tags attached by users' population to both music and POIs. It applied a set of similarity metrics for tagged resources to establish a match between music track and POIs.

Matthias Braunhofer et al considered a particular kind of context-aware recommendation task [13] that was illustrated in a mobile travel guide. They designed an approach for that recommendation task by matching music to POIs using emotional tags.

Fabian Abel et al. studied distributed form-based and tag-based user profiles, based on a large dataset aggregated from the Social Web. The performance of several cross-system user modeling strategies in the context of recommender systems is developed and evaluated to solve the cold-start problem and improve recommendation quality [14].

Yue Ni and Yushun Fan proposed an approach [15] for building reference ontologies corresponding to each domain, and then collaboration ontologies were constructed semi-automatically, OWL-S files generated from collaboration ontologies are mapped to BPEL and WSDL files respectively. In that way, the semantic information would be kept in processes and Web services, so that there was a common understanding among cross-domain cooperating enterprises.

Current cross-domain recommender systems does not work using Ford-Fulkerson algorithm. One of the related works using this algorithm is proposed by Aditya Parameswaran et al. But it is a single domain recommendation system. Their goal is to recommend to the students courses that not only help satisfy constraints but that are also desirable developing increasing expressive models for course requirements and presenting a variety of schemes for both checking if the requirements are satisfied, and for making recommendations that consider the requirements [12].

III. System Flow Steps

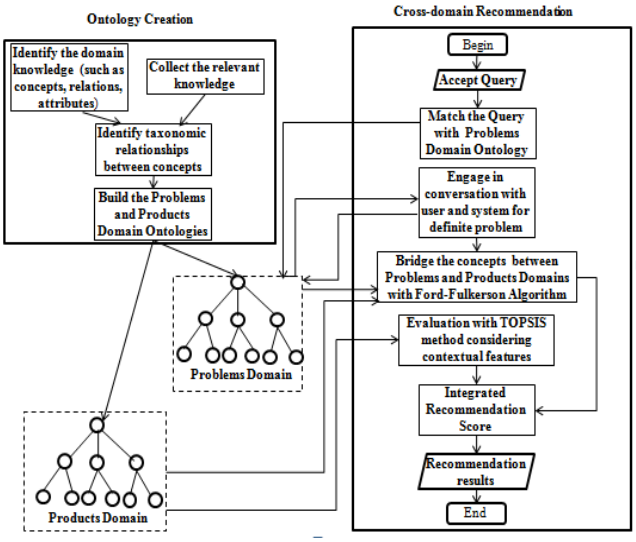

Fig. 1 Flow diagram of context aware cross-domain based recommendation

To provide context aware cross-domain recommendations, recommendation algorithm works as the following steps:

1. User gives the initial query from the system interface.
2. The system retrieves cases by searching, matching and ranking using Taxonomic CCBR.
3. The user and the system engage in a conversation where the system selects, ranks and presents questions to the user.
4. The user refines the problem description by answering questions from the questions that presented by the system.
5. The conversation and retrieval iterate until the system finds the definite problem for the user.
6. According this definite problem, the system links the concepts of the problem from the problem domain and the concepts of the products from the product domain in the weighted directed acyclic graph.

7. The system calculates the weight of each product (target node) applying the Ford-Fulkerson Algorithm.
8. The products are further computed considering customer's contextual features with TOPSIS Method.
9. The system calculates the recommender score of each product by combining the scores from Steps 7 and 8 to get higher user satisfaction and accuracy.
10. The system gives recommendations to the user according to the recommender score.

IV. System Implementation

There are five detail processes as described by the following for implementing the system.

A. Building Ontologies

In the system, ontologies [15] are used to capture knowledge about some domain of interest. Ontology describes the concepts in the domain and also the relationships that hold between those concepts. There are two domains for problems (source) and cosmetics (target). For these domains, the system builds two ontologies using protégé editor.

For problems (source) domain, there are classes: QApairs, Questions, Answers, Problems and Solutions, subclasses: YesNoAnswers and ConceptAnwers (subclasses of Answer), object properties: hasQuestion, hasAnswer, hasProblem, hasSolution, and isNextRelatedTo, data type properties: hasQDescription, hasADescription, hasProblemName, hasIngredients and hasIngValue and individuals for each class.

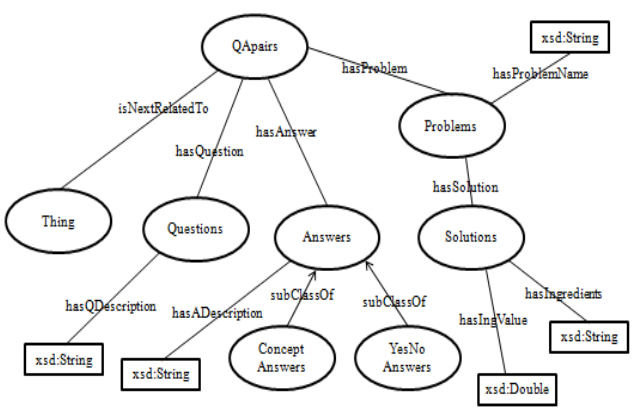

Fig. 2 Architecture of problems domain ontology

For cosmetics (target) domain, there are subclasses of cosmetics: Facial Foam, Toner, Cleansing Cream, Milky Lotion and so on. They have hasIngredients, hasIngredientsValue and hasName data type properties. Moreover, class Cosmetics has to be considered the customer's contextual features with each cosmetic item. Cosmetics class has Place Zone, Age Level, Cosmetics Brand, Season and Price Range subclasses. There is consistsOfPlaceZone object property to connect Place Zone and Country classes.

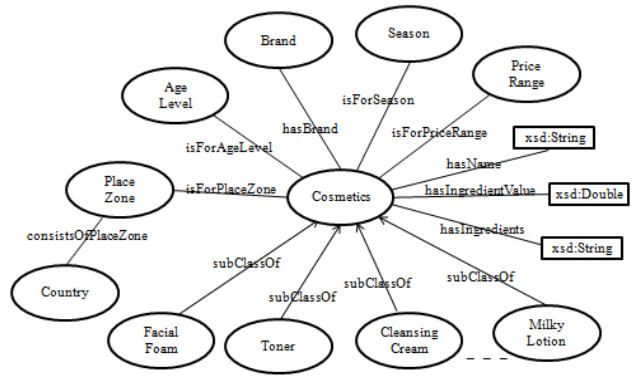

Fig. 3 Architecture of cosmetics domain ontology

B. Obtaining Definite Problem

Taxonomic conversational case-based reasoning (Taxonomic CCBR) [16], [17] is applied based on ontological properties in this stage. The user is unexpected to know exactly which type of problem but the user is required to answer a set of questions to identify more clearly what the problem is. Given information, the retrieval process is initiated whereby all questions in taxonomy relevant to the particular domain are presented to the user. Given the set of questions to choose from, the user can then decide to answer some of these questions. Depending on the answers, the system will try to find cases in which questions were answered in a similar manner. The questions which are present in the retrieved cases but which are still unanswered, yet are related to the problem, are then presented in a rank order to the user. The process continues until the system gets a definite problem.

For case retrieval, Taxonomic theory is divided into two steps taking into account that each question-answer (QA) pair is a set of triples or rather an acyclic directed graph style:

(i) Similarity between question-answer pairs

$$sim(C_{q1}, C_{q2}) = \begin{cases} 1 & if C_{q1} \subseteq C_{q2} \\ (n+1-m)/(n+1+m) & if C_{q2} \subseteq C_{q1} \\ 0 & otherwise \end{cases} \quad (1)$$

where,

Cq1 and Cq2 are concepts

(1) n = number of edges between Cq1 and the root

m = number of edges between Cq1 and Cq2

(ii) an aggregate similarity between the user query Q and a case problem description P to retrieve the most suitable cases

$$sim(Q,P) = \frac{\sum_{i \in Q, j \in P} sim(C_{qi}, C_{qj})}{T} \quad (2)$$

Where, T represents the number of taxonomies.

Example of working in this stage is briefly described. Assume that user query is "I have acnes on my face." From searching step, it is the most similar with "Are they acnes? Yes" question-answer pair in taxonomy. Then the system gives the questions and the user answers those iteratively until the

system gets the definite problem to the user. The working steps are shown in following table.

TABLE I
EXAMPLE OF STEPS FOR OBTAINING USER'S DEFINITE PROBLEM

Step	System Question	User Answer
1	Are they white spot?	No
2	Are they flat spots with dark centre?	No
3	Are they inflammation?	Slight
4	Which size are they?	Small
5	Are they pink?	Yes
6	Are you just before and during the menstrual cycle?	Yes

C. Cross-domain Recommendation

The system finds the relation between source and target domains according to the definite problem from the previous stage in with weighted directed acyclic graph.

In calculation, the weight of relations between instances is identified. In Kirchkoff's Law [18], "everything that leaves the source must eventually get to the sink", how much flowing into the weight of each target node with Ford-Fulkerson algorithm [12], [19]. The weight of target node is calculated by

$$W(v_i) = \sum_{k=1}^{n} f_{k,i}, i > 1 \quad (3)$$

where, n is the number of vertices and f is the weight of the flow. The more the weight is, the better the performance of semantic relation between different domains.

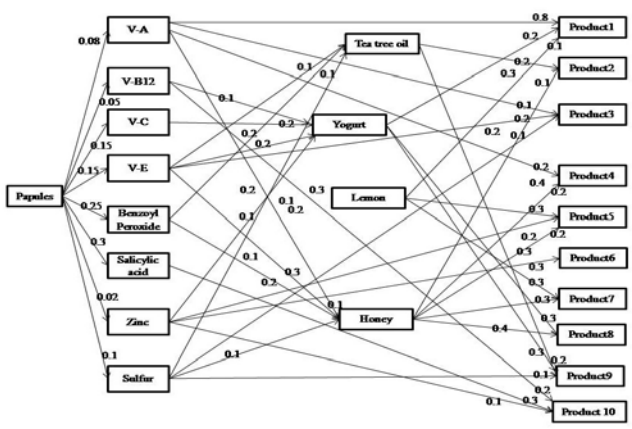

Fig. 4 Example of finding relations between source and target domains by DAG

D. Considering Contextual Features

In this stage, the system calculates the utility value of each candidate product for customer utilizing TOPSIS method, considering customer's contextual features. TOPSIS method is a multi-attribute decision making approach and stands for technique for ordering preference by similarity to ideal solution [20], [21], [22]. It is based on the principle that the solution should have the shortest distance to the best solution and the farthest distance to the worst one. As the mathematical model, P = $\{p_1, p_2,\ldots, p_m\}$ is defined as the vector of the product information and F = $\{f_1, f_2,\ldots, f_n\}$ is defined as the vector of the customer's contextual features. To represent the relevance performance of the product pi in the qualitative feature i, decision matrix can be constructed as the following:

$$D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ d_{m1} & d_{m2} & \cdots & d_{mn} \end{bmatrix} \quad (4)$$

The decision matrix should be normalized following the formula:

$$b_{ij} = d_{ij} \Big/ \sqrt{\sum_{j=1}^{n} d_{ij}^2}, i = 1,2,\ldots,m, j = 1,2,\ldots,n \quad (5)$$

The normalized value bij is limited in [0, 1]. The utility value of the product pi can then be calculated using the formula:

$$R_i = t_i^- \big/ (t_i^- + t_i^+), i = 1,2,..,m \quad (6)$$

where,

$$t_i^+ = \sqrt{\sum_{j=1}^{n} (c_{ij} - c_j^+)^2}, i = 1,2,\ldots,m \quad (7)$$

$$t_i^- = \sqrt{\sum_{j=1}^{n} (c_{ij} - c_j^-)^2}, i = 1,2,\ldots,m \quad (8)$$

In the above equations, n is the number of customer's contextual features, c_{ij} is the weighted normalized decision matrix which is calculated by

$$c_{ij} = w_j b_{ij}, i = 1,2,\ldots,m, j = 1,2,\ldots,n \quad (9)$$

where, wj means the customer's relative need in this feature and c_j^+, c_j^- are the positive and negative ideal solutions:

$$C^+ = \{c_1^+, c_2^+, \ldots, c_n^+\} = \left\{\left(\max_i c_{ij} \mid j \in I\right), \left(\min_i c_{ij} \mid j \in J\right)\right\} \quad (10)$$

$$C^- = \{c_1^-, c_2^-, \ldots, c_n^-\} = \left\{\left(\min_i c_{ij} \mid j \in I\right), \left(\max_i c_{ij} \mid j \in J\right)\right\} \quad (11)$$

The more increase the relative closeness R_i, the more important the utility value of the product p_i.

E. Calculating Recommender Score for Each Product

In this stage, the system calculates the recommender score by using this:

$$Rscore_i = \alpha \cdot W(v_i) + (1-\alpha) R_i, 0 \le \alpha \le 1 \quad (12)$$

According to the recommender score, the system gives the recommendations to the user. The system uses the constant α value 0.6 because this value is the best threshold for the system recommendation.

Finally, by performing the five stages systematically, the algorithm recommends a ranked list with the highest weighted target instances and the customer can obtain the most suitable products and services that matches her personal problem with full of satisfaction.

V. System Evaluation

To evaluate the quality of the recommendation product list, measures of recall and precision have been widely used in the field of recommender systems. Recall measures how many of the products in the actual customer purchase list consist of recommended products, whereas precision measures how many of the recommended products belong to the actual customer purchase list. These measures are simple to compute and intuitively appealing, however, they are in conflict, since increasing the size of the recommendation set will lead to an increase in recall, but to a decrease in precision at the same time. So a widely used combination metric called the 'F-measure' is used as the evaluation, criterion of the experiment in the paper. F-measure gives equal weight to both recall and precision, which can be computed as follow [22]:

$$F-measure = \frac{2 \times \mathrm{Re}\,call \times \mathrm{Pr}\,ecision}{\mathrm{Re}\,call + \mathrm{Pr}\,ecision} \quad (13)$$

In (13), Recall and Precision of the recommender system can be computed respectively, according to the computing method in the literatures, so F-measure can then be computed easily. Obviously, the higher the value of F-measure is, the better the recommendation performance of the system is.

We have recommendation value (α) that makes decision to recommend the products (i.e. flow weight value). The value of α will influence the value of F-measure and recommendation quality of the system. In order to select the suitable value of α, the relevant experiment has been done. Based on the initial analysis of the experiment, the relation between α and the F-measure value is illustrated in Fig. 5.

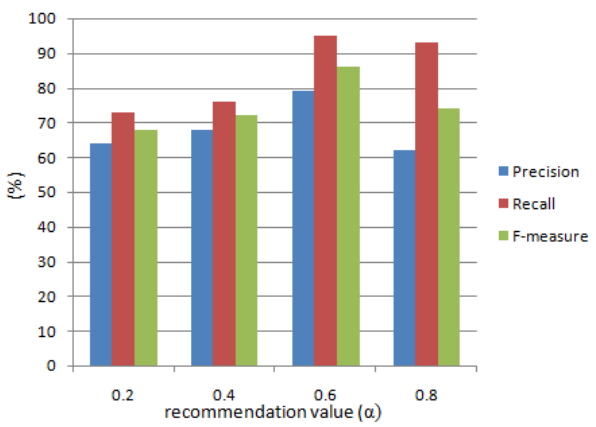

Fig. 5 The relation between the value of constant α and Precision, Recall and F-measure

VI. Conclusion

The paper presents an approach of context aware cross-domain recommendations for getting a user's definite problem and recommending relevant cosmetics based on Taxonomic CCBR and Ford-Fulkerson algorithm and TOPSIS method. The system tends to build the framework for recommending cosmetics (target domain) related to customer's skin care problems (source domain) because skin care is the most interesting area for people today. The system is user-friendly and more accurate than the other related works. It gives more personalized recommendations and makes more profits for commercial sites. Therefore, the system becomes an interesting and successful recommender system taking the advantages of ground-truth theory and application area. Furthermore, relevant experiments have been done to verify the effectiveness of product recommender algorithm in terms of F-measure criteria between accepted recommendations and offered recommendations.

Acknowledgment

My Sincere thanks to my supervisor Dr. Win Thanda Aung, Lecturer, University of Computer Studies (Bahan Campus), Myanmar for providing me an opportunity to do my research work. I express my thanks to my Institution namely University of Technology (Yatanarpon Cyber City) for providing me with a good environment and facilities like internet books, computers and all that as my source to complete this research. My heart-felt thanks to my family, friends and colleagues who have helped me.

References

[1] D. Asanov, "Algorithms and Methods in Recommender Systems", Berlin Institute of Technology, Berlin, Germany, 2011

[2] I. Fernández-Tobías, I. Cantador, M. Kaminskas and F. Ricci, "Cross-domain recommendr sysems: A servey of the State of the Art", Escuela Politécnica Superior, Universidad Autónoma de Madrid, 28049 Madrid, Spain and Faculty of Computer Science,Free University of Bozen-Bolzano, 39100 Bolzano, Italy, 2012

[3] I. Fernández-Tobías, I. Cantador, M. Kaminskas and F. Ricci, "A Generic Semantic-based Framework for Cross-domain Recommendation", Escuela Politécnica Superior, Universidad Autónoma de Madrid, 28049 Madrid, Spain and Faculty of Computer Science,Free University of Bozen-Bolzano, 39100 Bolzano, Italy, 2011

[4] K. Hoashi, K. Matsumoto, and N. Inoue, "Personalization of user profiles for content-based music retrieval based on relevance feedback", In Proceedings of the eleventh ACM international conference on Multimedia, pages 110–119, New York, NY, USA, 2003. ACM. http://dx.doi.org/10.1145/957013.957040

[5] F. Abel, E. Herder, G. Houben, N. Henze and D. Krause, "Cross-system User Modeling and Personalization on the Social Web", Web Information Systems, TU Delft, The Netherlands and IVS Semantic Web Group & L3S Research Center, Leibniz University Hannover, Germany, 2011

[6] M. Kaminskas and F. Ricci, " Location Adapted Music Recommendation Using Tags", Free University of Bolzano, Piazza Domenicani 3, 39100 Bolzano, Italy, 2011

[7] I. Konstas, V. Stathopoulos, and J. M. Jose, "On social networks and collaborative recommendation", In Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, pages 195–202, New York, NY, USA, 2009. ACM.

[8] M. Azak, "Crossing: A Framework To Develop Knowledge-based Recommenders In Cross Domains", MSc thesis, Middle East Technical University, 2010

[9] J. Tang, S. Wu, J. Sun, and H. Su, "Cross-domain Collaboration Recommendation", Department of Computer Science and Technology, Tsinghua University, IBM TJ Watson Research Center, USA, August, 2012

[10] M. Kaminskas and F. Ricci, "Matching Places of Interest With Music", Free University of Bozen-Bolzano, 39100 Bolzano, Italy, 2009.

[11] M. Kaminskas, "Matching Information Content with Music", Free University of Bozen-Bolzano, 39100 Bolzano, Italy, 2009.

[12] A. Parameswaran, P. venetis and H. Garcia-Monlina, Recommendation Systems with Complex Constraints: A Course Recommendation Perspective", ACM Transactions on Information Systems, Standford University, January 2010.
[13] M. Braunhofer, M. Kaminskas and F. Ricci, "Recommending Music for Places of Interest in a Mobile Travel Guide", Free University of Bozen-Bolzano, Piazza Domenicani 3, Bolzano, Italy, 2011.
[14] F. Abel, E. Herder, G. Houben, N. Henze and D. Krause, "Cross-system User Modeling and Personalization on the Social Web", Web Information Systems, TU Delft, The Netherlands and IVS Semantic Web Group & L3S Research Center, Leibniz University Hannover, Germany, 2011.
[15] Y. Ni and Y. Fan, "Ontology Based Cross-Domain Enterprises Integration and Interoperability" Department of Automation, Tsinghua University, Beijing 100084 and Guilin Air Force Academy, Guilin 541003, China, 2008.
[16] K. M. Gupta, "Taxonomic Conversational Case-Based Reasoning", in Proc. ICCBR 2001, LNAI 2080, 2001, pp. 219-133.
[17] C. Abela and M. Montebello, "PreDiCtS: A Personalised Service Discovery and Composition Framework", Dept. Computer Science and AI, University of Malta, 2006.
[18] V. Vatter, "Graphs, Flows and the Ford-Fulkerson Algorithm", August 12, 2004.
[19] http://en.wikipedia.org/wiki/Ford-Fulkerson_algorithm
[20] J. Wei, "TOPSIS Method for Multiple Attribute Decision Making with Incomplete Weight Information in Linguistic," Journal of Convergence Information Technology, Volume 5, Number 10, December 2010, doi: 10.4156/jcit.vol5. issue 10.23
[21] O. Jadidi, F. Firouzi and E. Bagliery, "TOPSIS Method for Supplier Selection Problem", World Academy of Science, Engineering and Technology 71, 2010
[22] Z. Zeng, "A Personalized Product Recommender System based on Semantic Similarity and TOPSIS Method", Journal of Convergence Information Technology, Volume 6, Number 7, July 2011, doi: 10.4156/jcit.vol6.issue7.39
http://dx.doi.org/10.4156/jcit.vol6.issue7.39

Hla Hla Moe is a Ph.D (IT) candidate from University of Technology (Yatanarpon Cyber City). She received master degree in computer science from Computer University (Sittway). The field of her thesis is cross-domain recommendation for personalized semantic services. She is working as a teacher in Computer University (Sittway) and her research interested areas are web mining, semantic web and personalization.