

Building Ontologies for Cross-domain Recommendation on Facial Skin Problem and Related Cosmetics

Hla Hla Moe

University of Technology (Yatanarpon Cyber City), Pyin Oo Lwin, Myanmar
Email: hla2moe@gmail.com

Win Thanda Aung

University of Computer Studies (Bahan Campus), Yangon, Myanmar
Email: winthanda.monywa@gmail.com

Abstract— Nowadays, recommendation has become an everyday activity in the World Wide Web. An increasing amount of work has been published in various areas related to the recommender system. Cross-domain recommendation is an emerging research topic. This type of recommendations has barely been investigated because it is difficult to obtain public datasets with user preferences crossing different domains. To solve dataset problem, one of the solution is to create different domains. Ontology is playing increasingly important roles in many research areas such as semantics interoperability and knowledge base and creating domain. Ontology defines a common vocabulary and a shared understanding and is applied for real world applications. Ontology is a formal representation of a set of concepts within a domain and the relationships between those concepts. This paper presents an approach for building ontologies using Taxonomic conversational case-based reasoning (Taxonomic CCBR) to apply cross-domain recommendation based on facial skin problems and related cosmetics. For linking cross-domain recommendation, Ford-Fulkerson algorithm is used to build the bridge of the concepts between two domain ontologies (Problems domain as the source domain and Cosmetics domain as the target domain).

Index Terms— recommender system, cross-domain recommendation, ontology, Taxonomic CCBR, semantic concepts

I. INTRODUCTION

Today, different products, services and customers can now comfortably communicate with each other via the internet because of the prosperity of e-commerce that has continuously been improved by the advance of Internet and Web technologies. With the development of e-commerce, customers face a difficulty to choose products or services and find the most suitable ones with them. 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 [10].

The above 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. Various kinds of recommendation system have been constructed [11]. 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. Cross-domain recommendation becomes an emerging research topic today. For instance, a system suggests not only a particular movie but also music CDs and books that are somehow related to that movie.

Chinese character is shaped by the constructure-related stokes, and the hand-written Chinese characters retrieval is reduced to the skeletons recognition. Calligraphic characters as an aesthetic medium contain the sentiment of the author, which makes the recognition more difficult than the usual hand-written characters. The skeletons of calligraphic characters can be acquired in most cases as the rudiment of the first-round retrieval to reduce the globe-searching space and enhance the recall rate. And the second-round retrieval based upon the character regional shape features can guarantee the satisfying retrieval precision.

By definition, cross-domain recommendation is providing recommendations of items in one (source) domain using the preferences expressed on items in a second (target) domain. Another task for cross-domain recommendation is making joint recommendations for items belonging to different domains [12]. Cross-domain recommendation models [14] 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. The benefits of cross-domain recommendation are that they can give more diverse recommendations leading to a higher user satisfaction and engagement addressing cold-start and sparsity problems [12], [14], [15]. However, this type of recommendation has merely been discovered because it is hard to gain public datasets with user preferences

crossing different domains. Building ontologies becomes an important role for solving this problem.

Ontologies have become achingly popular as a way for representing machine-readable knowledge in recent years. In computer science and information science, an ontology formally represents knowledge as a set of concepts within a domain, and the relationships between those concepts. It can be used to reason about the entities within that domain and may be used to describe the domain. In theory, an ontology is a formal, explicit specification of a shared conceptualization. An ontology renders shared vocabulary and taxonomy which models a domain with the definition of objects and/or concepts and their properties and relations. Ontologies are the structural frameworks for organizing information. The creation of domain ontologies is also fundamental to the definition and use of an enterprise architecture framework.

In this paper, an approach for building ontologies using Taxonomic conversational case-based reasoning (Taxonomic CCBR) to apply cross-domain recommendation based on facial skin problems and related cosmetics. Facial skin problems domain and cosmetics domain are required to develop cross-domain recommendations. Two ontologies for problems domain and cosmetics domain have been developed using Protégé editor. Taxonomic CCBR is used to get the feature of personalization because Taxonomic CCBR allows a partial definition of a problem by the user, identifies more clearly user's problem and gives accurate solution by conversation. It can solve one of the weaknesses of cross-domain recommendation system, lack of personalization. Cross-domain recommendation between two domains is developed by using Ford-Fulkerson algorithm building semantic bridge.

The rest of the paper is organized as follows. Section 2 describes related works on cross-domain recommender systems briefly. In section 3, system overview of the proposed system is described. Section 4 explains the implementation of the system. Section 5 mentions evaluation of the system and section 6 describes the conclusion.

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. Current cross-domain recommender systems does not work using Ford-Fulkerson algorithm. Previous works have been proposed in various kinds of ways.

Francesco Ricci et al. proposed an approach that 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) [12].

References [6] and [7] showed that Marius Kaminskas and Francesco Ricci proposed an approach 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.

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 [13].

A generic framework to mediate the integration of data collected by several recommender systems was presented by Shlomo Berkovsky et al. They discussed four major types of mediation: cross-user, cross-item, cross-context, and cross representation. Some evaluations had shown that in certain conditions, user modeling data mediation improved the quality of recommendations, especially in the cold start of a recommender system [5].

Yue Ni and Yushun Fan proposed an approach 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 [4].

A web based conversational case-based recommender system for ontology aided metadata discovery [16] was presented by David Leake et al. The approach used Conversational Case-based Reasoning (CCBR), with semantic web markup languages providing a standard form for case representation. They presented the initial efforts in designing and developing ontologies for an Earthquake Simulation Grid, to use these to guide case retrieval.

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 [8].

III. FLOW STEPS OF THE SYSTEM

To provide cross-domain recommendations, the system algorithm works as the following flow 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 his/ her 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 and chooses the products with much weight.
8. Finally, the system gives personalized recommendations to the user according to the recommender score.

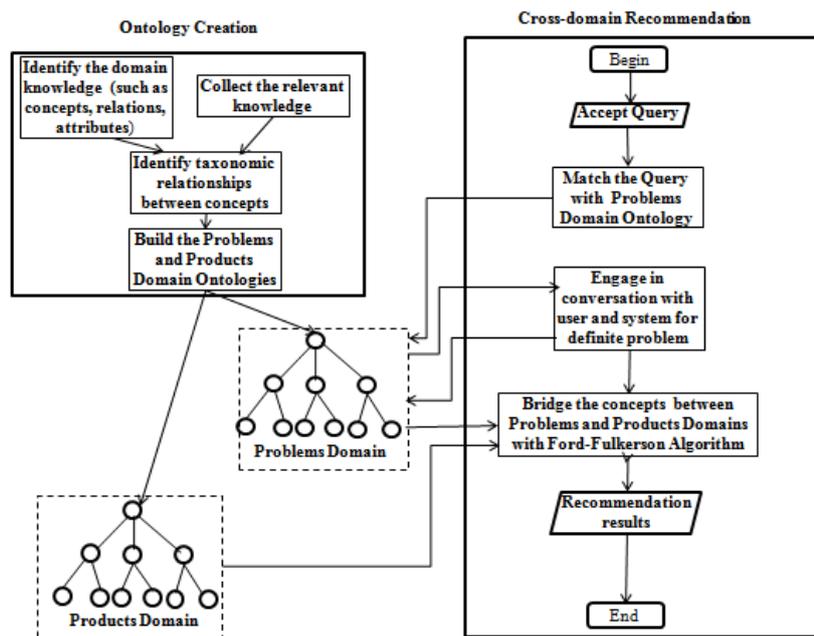


Fig. 1. flow diagram of the system

IV. IMPLEMENTATION OF THE SYSTEM

For implementing the system, there are three detail processes: building ontologies for problems and cosmetics, acquiring the definite problem from user query and recommending the cosmetics related with user problem.

4.1 Building Ontologies for Problems and Cosmetics Domains

Ontologies [4], [16] 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.

Ontological engineering is a field that studies the methods and methodologies for building ontologies. It studies the ontology development process, the ontology life cycle, the method and methodologies for building ontologies and the tools suited and language that support them. It offers a direction towards solving the interoperability problems brought about by semantic

obstacles, such as the obstacles related to the identification of business terms and classes. Ontological engineering is a set of tasks related to the development of ontology for a particular domain. For building ontologies has revealed that ontologists should carry out the following tasks:

Task 1: To build the glossary of terms that identifies the set of terms to be included on the ontology, their natural language definition and their synonym and acronyms.

Task 2: To build concept taxonomies to classify concepts. The output of this task could be one or more taxonomies where the concepts are classified.

Task 3: To build ad hoc binary relation diagrams to identify ad hoc relationships between concepts of the ontology and with the concepts of other ontologies.

Task 4: To build the concept dictionary, which mainly includes the concepts instance for each concept, their instance and class attributes and their ad hoc relations?

Task 5: To describe in detail each ad hoc binary relation that appears on the ad hoc binary relation diagram and on the concept dictionary.

Task 6: To describe in detail each instance attribute that appears on the concept dictionary.

Task 7: To describe in detail each class attribute that appears on the concept dictionary.

Task 8: To describe in detail each constant and to produce a constant table. Constants specify information related to the domain of knowledge, they always take the same value, and are normally used in formulas.

OWL ontologies have similar components to Protégé frame based ontologies. However, the terminology used to describe these components is slightly different from that used in Protégé. An OWL ontology consists of Individuals (Instances), Properties, and Classes, which roughly correspond to Protégé Instances, Slots and Classes.

In the system, for problems (source) domain, there are classes: QApairs, Questions, Answers, Problems and Solutions, subclasses: YesNoAnswers and ConceptAnswers (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. Object property isNextRelatedTo is mainly used to obtain the taxonomy structure of the problems from related questions and answers in building this domain ontology.

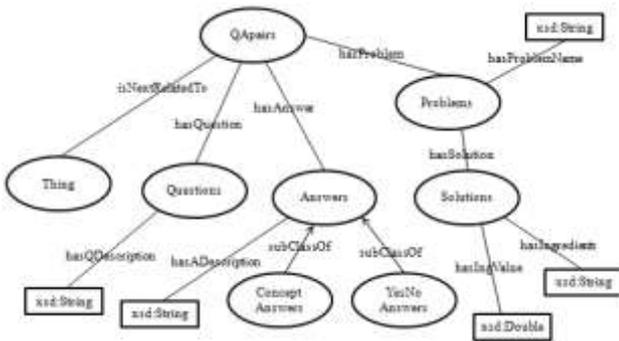


Fig. 2. Architecture of the Problems Domain Ontology

```

<owl:Class rdf:ID="Answers"/>
<owl:Class rdf:ID="QApairs"/>
<owl:Class rdf:ID="Problems"/>
<owl:Class rdf:ID="ConceptAnswers">
<rdfs:subClassOf
rdf:resource="#Answers"/>
</owl:Class>
<owl:Class rdf:ID="Solutions"/>
<owl:Class rdf:ID="Questions"/>
<owl:Class rdf:ID="YesNoAnswers">
<rdfs:subClassOf
rdf:resource="#Answers"/>
</owl:Class>
    
```

Fig.3. Example of classes in 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.

Contextual features are considered with Place Zone, Age Level, Cosmetics Brand, Season and Price Range subclasses. There is consistsOfPlaceZone object property to connect Place Zone and Country classes.



Fig. 4. Architecture of the Cosmetics Domain Ontology

```

<owl:Class rdf:ID="Cosmetics"/>
<owl:Class rdf:ID="PriceRange">
<rdfs:subClassOf>
<owl:Class rdf:ID="ContextualFeatures"/>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="PlaceZone">
<rdfs:subClassOf
rdf:resource="#ContextualFeatures"/>
</owl:Class>
<owl:Class rdf:ID="MakeUp">
<rdfs:subClassOf rdf:resource="#Cosmetics"/>
</owl:Class>
    
```

Fig. 5. Example of classes in Cosmetics Domain Ontology

4.2 Obtaining Definite Problem from User Query

In this stage, Taxonomic conversational case-based reasoning (Taxonomic CCBR) [1], [3] is applied based on ontological properties. The user is not expected to know exactly which type of problem the user has but the user is required to answer a set of questions such that the system identifies more clearly what his/her problem is. Given information related to the domain, the retrieval process is initiated whereby all questions in taxonomy relevant to that 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 provided, the system will try to find cases in which questions were answered in a similar manner. A similarity measure is used to rank cases. 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 case which includes a definite problem, personalizing the solution to her needs.

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

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

where,

C_{q1} and C_{q2} are concepts

n = number of edges between C_{q1} and the root

m = number of edges between C_{q1} and C_{q2}

(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.

From these steps, the system can determine that the user problem is “Papules”.

Table 1. Example of steps for acquiring 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

4.3 Finding 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. According to Kirchhoff’s Law [2], “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 [8], [18]. It is simple and gives accurate weight of each target node. 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.

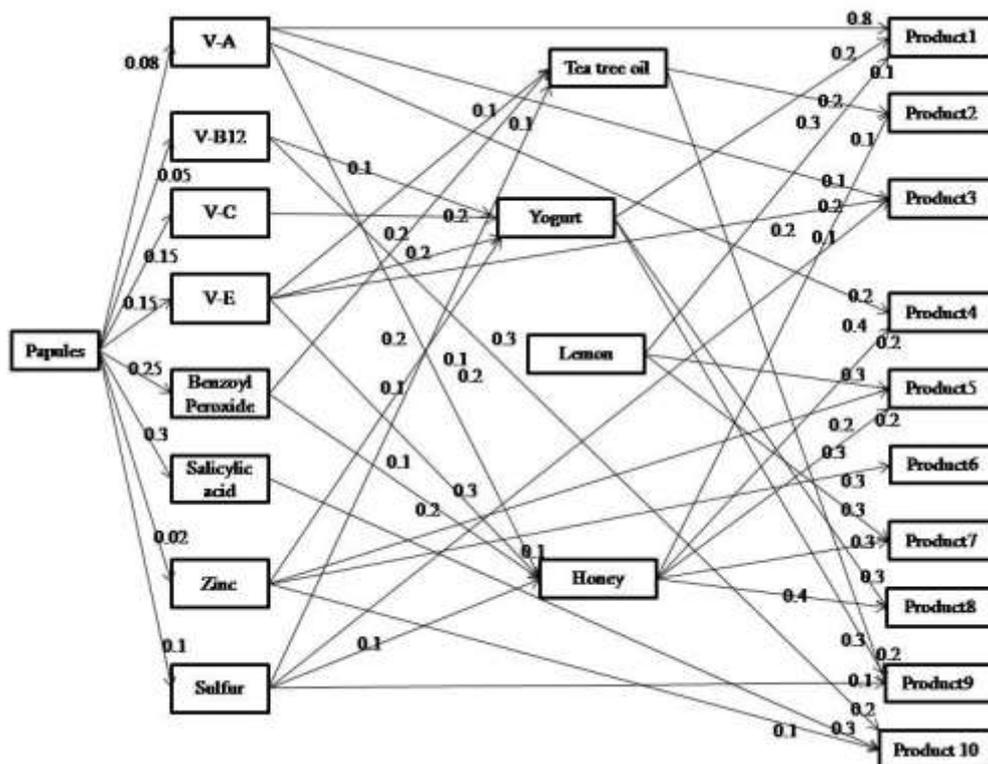


Fig. 6. example of finding relations between source and target domains by DAG

By applying the Ford-Fulkerson Algorithm with the above example figure, the system gets the flow weight of each product if there is relation. The flow weight of the products is shown as Table 2.

Table 2. Table of sample product weight

Product P_i	Flow Weight $W(v_i)$
1	0.38
2	0.45
3	0.53
4	0.58
5	0.52
6	0.32
7	0.60
8	0.70
9	0.60
10	0.37

4.4 Recommendation Example

To give recommendation to the user, the more the flow weight of each product is, the better the performance of relation between two domains. And thus, the example of recommendation product is as the following Table 3.

Table 3: Table of recommendation product

Recommendation Product ID	Flow Weight of Product
8	0.70
7	0.60
9	0.60
4	0.58
3	0.53
5	0.52
2	0.45
1	0.38
10	0.37
6	0.32

V. EVALUATION OF THE SYSTEM

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 [9], [17]:

$$F - measure = \frac{2 \times Recall \times Precision}{Recall + Precision} \tag{4}$$

In the above formula, 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 the value of constant α and precision, recall and F-measure value is illustrated in Fig: 5.

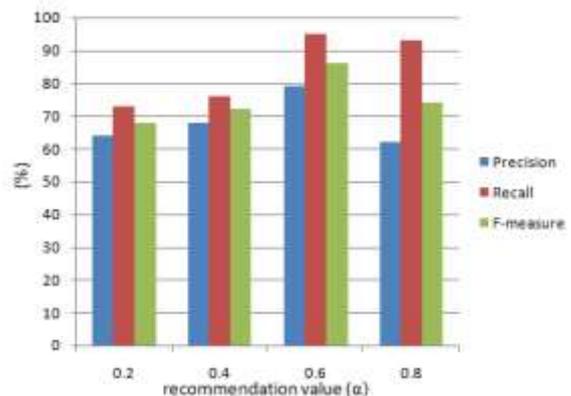


Fig. 5. the relation between the value of constant α and precision, recall and F-measure

VI. CONCLUSION

The paper presents an approach for building ontologies to provide cross-domain recommendation based on facial skin problem and related cosmetics using Taxonomic CCBR and Ford-Fulkerson algorithm. 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. As the future work, we will improve the recommender model and make it to provide semantic personalized services considering customer's contextual features based on the present work.

REFERENCES

- [1] Kalyan Moy Gupta, (2001), "Taxonomic Conversational Case-Based Reasoning", in Proc. ICCBR 2001, LNAI 2080, pp. 219-133.
- [2] Vince Vatter, (2004), "Graphs, Flows and the Ford-Fulkerson Algorithm".
- [3] Charlie Abela and Matthew Montebello, (2006), "PreDiCtS: A Personalised Service Discovery and Composition Framework", Dept. Computer Science and AI, University of Malta.
- [4] Yue Ni and Yushun Fan, (2008), "Ontology Based Cross-Domain Enterprises Integration and Interoperability", Department of Automation, Tsinghua University, Beijing 100084 and Guilin Air Force Academy, Guilin 541003, China.
- [5] Shlomo Berkovsky, Tsvi Kuflik and Francesco Ricci, (2008), "Mediation of User Models for Enhanced Personalization in Recommender Systems", University of Haifa, Haifa, Israel, Free University of Bozen-Bolzano, Italy.
- [6] Marius Kaminskas and Francesco Ricci, (2009), "Matching Places of Interest With Music", Free University of Bozen-Bolzano, 39100 Bolzano, Italy.
- [7] Marius Kaminskas, (2009), "Matching Information Content with Music", Free University of Bozen-Bolzano, 39100 Bolzano, Italy.
- [8] Aditya Parameswaran, Petros Venetis and Hevtor, (2010) Garcia-Monlina, "Recommendation Systems with Complex Constraints: A Course Recommendation Perspective", ACM Transactions on Information Systems, Stanford University.
- [9] Chein-Shung, Hwang, (2010), "Genetic algorithms for feature weighting in multi-criteria recommender systems", Journal of Convergence Information Technology, vol 5, no 8, pp: 126-136.
- [10] Ziming Zeng, (2011), "A Personalized Product Recommender System based on Semantic Similarity and TOPSIS Method", Journal of Convergence Information Technology, Volume 6, Number 7, doi: 10.4156/jcit.vol6.issue7.39.
- [11] Daniar Asanov,(2011), "Algorithms and Methods in Recommender Systems", Berlin Institute of Technology, Berlin, Germany.
- [12] Ignacio Fernández-Tobías, Iván Cantador , Marius Kaminskas and Francesco Ricci, (2011), "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.
- [13] Fabian Abel, Eelco Herder, Geert-Jan Houben, Nicola Henze and Daniel Krause, (2011), "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.
- [14] Ignacio Fernández-Tobías, Iván Cantador , Marius Kaminskas and Francesco Ricci, (2012), "Cross-domain recommendr systems: 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.
- [15] Jie Tang , Sen Wu, Jimeng Sun, and Hang Su, (2012), "Cross-domain Collaboration Recommendation", Department of Computer Science and Technology, Tsinghua University, IBM TJ Watson Research Center, USA.
- [16] Mehmet S. Akatas, Marlon Pierce, Geoffrey C. Fox and David Leake, "A Web based Conversational Case-Based Recommender System for Ontology aided Metadata Discovery", Community Grids Labs and Computer Science Department, Indiana University, Bloomington, IN 47404, USA.
- [17] Bing Liu, "Web Data Mining", Department of Computer Science, University of Illinois at Chicago, 851 s. Morgan Street, Chicago, IL60607-7053, USA.
- [18] http://en.wikipedia.org/wiki/Ford-Fulkerson_algorithm

Authors' Profiles



Hla Hla Moe: Research student for doctor degree for Information Technology in University of Technology (Yatanarpon Cyber City), major in Computer Science, Software Engineering. 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.

Win Thanda Aung: Asistant Lecturer from University of Computer Studies (Bahan Campus), interested in web mining.

How to cite this paper: Hla Hla Moe, Win Thanda Aung, "Building Ontologies for Cross-domain Recommendation on Facial Skin Problem and Related Cosmetics", International Journal of Information Technology and Computer Science(IJITCS), vol.6, no.6, pp.33-39, 2014. DOI: 10.5815/ijitcs.2014.06.05