Analyzing Tag-based Mashups with Fuzzy FCA.

Cheng-Yen Chou and Hsing Mei
Computer Science and Information Engineering Department
Fu Jen Catholic University
{harlan95,mei}@csie.fju.edu.tw

Abstract

Mashing up with APIs has become a critical skill for Web 2.0 developers to construct creative web applications quickly. With the help from descriptive tags, developers can search the target Web APIs easier than before. However, it is still difficult to determine proper tags that best describe Web API. This paper is focusing on analyzing semantic tags and the relationships between each Web APIs. We apply Fuzzy Formal Concept Analysis (FFCA) and Bidirectional Co-occurrence approach to analyze tag relations automatically. As the experiment results shown, Bidirectional Co-occurrence enhanced FFCA model can increase the accuracy of tag mining, and FFCA could speed up the computation. Our Bidirectional Co-occurrence approach is more efficiency and accurate than native approaches.

1. Introduction

Mashup is a web application that merges different elements from more than one source into an integrated service [1]. The concept of mashup was evolved from the web service model toward a highly participated web 2.0 mashup model. Under mashup framework, service providers are also likely to be service requestors. Service providers publish web APIs that allow mashup developers to construct applications easily. Service requestors search web APIs that meet their requirements, and re-compose web APIs into hybrid applications.

Mashup developers face many new challenges, such as to distinguish ambiguous description of Web APIs, to identify and remove useless Web APIs, and so on. It is difficult to reflect the importance of Web APIS by tag-counting based approaches, for example tag cloud [2]. In this paper, we analyze the characteristic of the semantic tag by combine Formal Concept Analysis (FCA) model and Bidirectional Co-occurrence approach. We also use Fuzzy FCA to enhance the computation ability and improve the accuracy.

This paper is organized as follows. In section 2, we review the related works about tag-remix in mashup applications, semantic tag [3], Formal Concept Analysis [4-5], and fuzzy theory. The definition of mashup applications and the Bidirectional Co-occurrence approach are illustrated in section 3. In order to verify the accuracy and efficiency, we introduce the experiment results in section 4. Finally, conclusion is presented in section 5.

2. Related Work

Tag management within mashup applications stands on top of two hot field of information technology: data mining, and social network analysis. Here, we first give a brief introduction to the characteristics and semantic problem for tags. In view of this, FCA is a good model to analyze the relationships between tags. Moreover, fuzzy theory extension improves the drawback of FCA.

2.1. Semantic Tag

Tag, also called keywords, is also a common way of organizing content for future navigation, filtering or search. Tagging is a habit that the users make content with descriptive term [6]. Collaborative tagging [7], known as folksonomy is the process of collaboratively creating and managing tags to share and organize content. Compared with traditional topic indexing system, metadata is not only created by specific experts but also by all users. Different users can make label on content according with one's own characters. Here, we summarize the functionality of tag as following [8]:

1. Identifying what it is about.
2. Identifying what it is.
3. Identifying who owns it.
4. Refining categories.

Among the above four points, the last point is the most critical function of tag. Since the meaning of tags

978-0-7695-3499-2/08 $25.00 © 2008 IEEE
DOI 10.1109/SOSE.2008.16

varies widely from general view to specific view, we need a good model to analyze the relationship between tags.

2.2. Formal Concept Analysis

Formal Concept Analysis (FCA)[4-5] is a data analysis technique based on order theory. It defines formal contexts to represent the relationship between formal objects and formal attributes in a domain, FCA interprets the relationships with concept lattice. For example, Figure 1 shows the FCA result with "Travel" keyword in del.icio.us. From upper formal attribute view, the formal attribute "Travel" is a common attribute of all formal objects. From lower formal object view, the formal object "flights" have many attributes, such as "Flights", "Airfare", "Airlines" and so on.

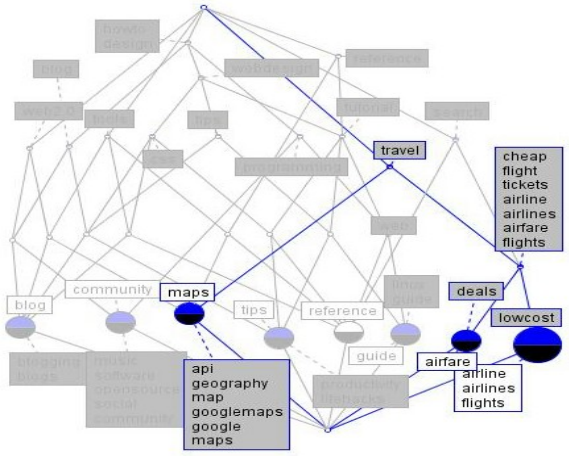

Figure 1. Formal Concept Analysis

In formal definition, a formal context is a triple K = (G,M,I), where G and M are sets, and I is a relation between G and M. The elements of G and M are called formal objects and formal attributes. A formal context can be easily represented by a cross table. The cross table is shown as Table 1. The columns are formal objects and the rows are formal attributes. If there is a relation between formal objects and formal attributes, put a cross in the check.

Table 1. Cross Table

	Flights	Airline	Airlines	Blog	Guide	Maps
Travel	※	※	※			※
Flights	※	※	※			
Airfare	※	※	※			
Maps						※
Guide					※	
Airline	※	※	※			
Airlines	※	※	※			
Cheap	※	※	※			

According to the Mapping Operation and Subconcept-Superconcept definitions in FCA, the cross table can be converted to concept lattice as shown in Figure 1. The white boxes represent formal objects, and the gray boxes represent formal attributes. The attributes are either attached to objects or stand-alone. When constructing a concept lattice, it should start with selecting the minimum cross between objects and attributes in cross table

2.3. Fuzzy Theory

Fuzzy theory was introduced by Professor Lotfi Zadeh in 1965. The basic idea of the fuzzy theory is that fuzzy sets are sets whose elements have degrees of membership. This degree is a real number in the interval [0,1]. In order to enhance the relationships between formal objects and formal attributes in FCA, we use fuzzy formal context and fuzzy mapping operation. As shown in Figure 2, the similarity of Flight and Travel is divided into three parts according to related tags. We will introduce using fuzzy set theory to increase the efficiency of FCA in next section

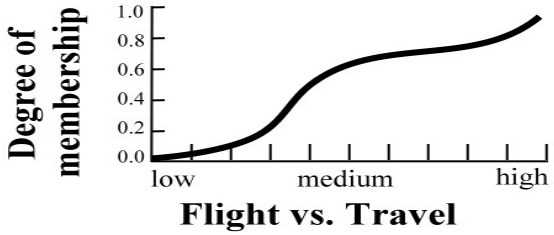

Figure 2. Fuzzy Logic

3. Multi-Layer Bidirectional Co-occurrence Approach

3.1. Precondition Definition

The development of Mashup applications is depends on many factors, such as developers' background, the information of Web APIs, and users' requests. Hence, we define five sets, namely, Developer Profile Set, Web API Set, Web API_{arc} Set, Mashup Applications Set, and Service Request Set, to describe the development process of mashup applications.

- Definition 1: Developer Profile Set (DP)

A Developer Profile, DP = {Tag_{d1}, Tag_{d2} ,…, Tag_{dn}} is a set of tags that describes the developer's background. We define five basic items to represent

DP's attributes: Sex, Age, Activity Range, Profession, and Interest.

- Definition 2. Web API_{arc} Set (WA_{arc})

A Web API, WP = {FeedList, $TagList_{wp}$} includes two sets: FeedList Set and $Taglist_{wp}$ Set. The FeedList, Feedlist = {Input{$feed_{i1}$, $feed_{i2}$,… $feed_{in}$} , Output{$feed_{o1}$, $feed_{o2...}$, $feed_{on}$}} is composed of two sets, Input FeedList and Output FeedList. Generally, the Output FeedList only includes one feed. FeedList is used to be the content of Web API Set. TagList, $TagList_{wp}$ = {Tag_{wp1}, Tag_{wp2},… Tag_{wpn} } is also a list of tags to stand for the representative of Web APIs. Web API Archive, WA_{arc}={WA_1, WA_2,…, Wa_n} is a Web API repository.

- Definition 3. Mashup Applications Set (MA)

Suppose a mashup developer with DP needs one travel service, this service is represented as a Mashup Application Set, MA = { WA_1 , WA_2 ,…, WA_n } briefly as a set of Web API.

- Definition 4. Service Request Set (SR_q)

The user's request contains what users want to do, when to do, participants, and other information, such as cost, the location, and so on. We define Service Request Set as SR_q={ Date , Location , Participator , Activity , Note }.

Our problem is to retrieve the set of Web APIs, MA = { WA_k , WA_{k+1} ,…, WA_l }. where $WA_f \in WA_{arc}$ (k-1 < f < l+1) satisfying one of the following two conditions.

1. If users know some information, and they like to find more about their requests. We will discover a lot of useful and detailed information from Developer Profile Set, Service Request Set and FeedList Set in Web API Set.
2. If users have no idea about their requests, and they like to meet their requests. We will search only from TagList Set in Web API Set and Service Request Set.

3.2. Bidirectional Co-occurrence Approach

FCA is a powerful model for expert problems. FCA was only used in expert systems before. The reason. Is that it is very hard to determine the relationships between formal objects and formal attributes. We extend the Co-occurrence approach and integrate it with FCA [9]. The co-occurrence approach is represented by a graph with labeled nodes and undirected weighed edges, as shown in Figure 3. Nodes are tags and edges are the relationships between tags.

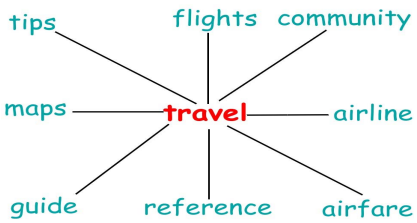

Figure 3. Co-occurrence Approach

We extend co-occurrence approach to bidirectional co-occurrence approach in Figure 4. The bidirectional co-occurrence property is critical to web 2.0 applications, because the development process of mashup applications is in order. A bidirectional co-occurrence graph experiment on del.icio.us is used for illustration. Each node represents a Web API, named tag node, and the center node is user's request, named keyword node. For example, which Web APIs are the users needs if they want to do traveling? They will find out many Web APIs, such as maps, guide, airfare, reference and so on. Users will firstly use Community or Guide Web APIs to understand preliminary traveling plans. However, users cannot search Travel Web API from Community Web API.

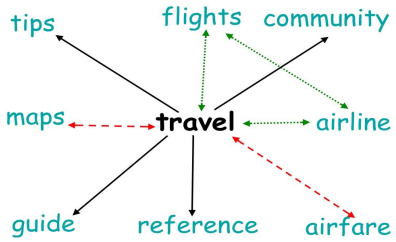

Figure 4. Bidirectional Co-occurrence Approach

Besides, we modify one-layer bidirectional co-occurrence approach to multi-layers bidirectional co-occurrence approach in section 3.3.

3.3. Fuzzy FCA

Fuzzy FCA is combination FCA model and Fuzzy Theory [10-11]. The main drawback of FCA is that the relationships between formal objects and formal attributes are selected by experts. We adopt fuzzy theory to construct multi-layer bidirectional co-occurrence approach. As shown in Figure 5, we use keyword "travel" to search and discover ten tags, including "guide", "reference", "maps", "airfare", "airline", "airlines", "airline", "flights", "community", "blog" and "tips." These ten tags are included in first-layer co-occurrence graph. Secondly, we use these tags to re-search and find other tags, including "tutorial", "geography", "map", "deals", "cheap", "flight", "lowcost", "howto", and so on. These tags are included

in second-layer co-occurrence graph. We find out that the link between nodes "Airline" and "Airlines" is similar. We can infer that the node "Airline" and another node "Airlines" is considered as the same node in this graph.

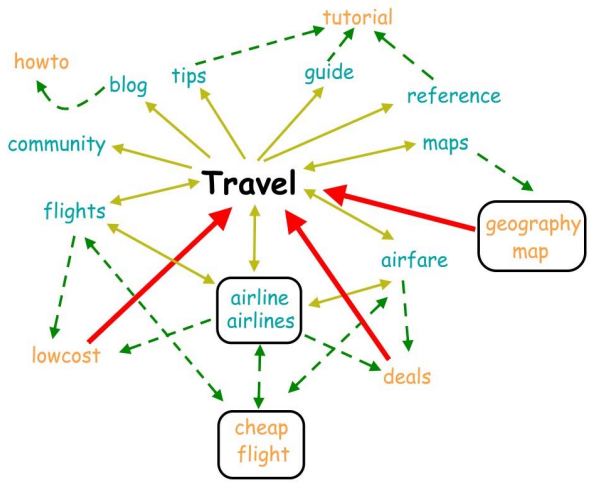

Figure 5. Multi-layer Bidirectional Co-occurrence Graph

In the first-layer bidirectional co-occurrence graph, these tags generally are common tags. Hence, we adopt second-layer bidirectional co-occurrence graph to filter these common tags. Besides, we find some interesting phenomenon. Firstly, there is a direct link between keyword node "Travel" and tags node "Flights" and "Airfare." Besides, tag node "Cheap" has an indirect relationship between keyword node "Travel" and tags node "Flights" and "Airfare." In the last, although tag node "Lowcost" is belonged to isolated node, there is a relationship from tag node "Lowcost" to keyword node "Travel." Thus, we define three fuzzy rules for this special phenomenon in Figure 6.

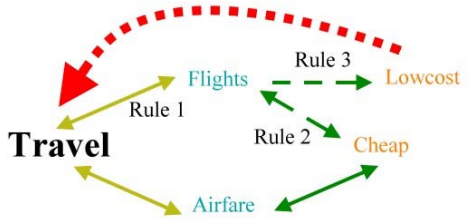

Figure 6. Fuzzy Rules for Multi-layer Bidirectional Co-occurrence Approach

- Definition 5. Fuzzy FCA Rules for Multi-layer Bidirectional Co-occurrence Approach.

If developers use developer's Service Request Set SR_q to find $Taglist_{wp} = \{Tag_{(1,1)}, Tag_{(1,2)},\ldots, Tag_{(1,n)}\}$ in FeedList. Our model use $Tag_{(1,1)}$ finds out $Taglist_2 = \{Tag_{(2,1)}, Tag_{(2,2)},\ldots, Tag_{(2,n)}\}$. Therefore, there is a membership function u(g , m), g is SR_q and m is $Tag_{(m,n)}$. We define three fuzzy rules for this membership function as follows,

1. Rule1: If SR_q finds $Tag_{(1,n)}$ and $Tag_{(1,n)}$ finds SR_q, then the membership function u(g , m) is high.
2. Rule2: If $Tag_{(1,n)}$ finds $Tag_{(2,n)}$ and $Tag_{(2,n)}$ finds $Tag_{(1,n)}$,then the membership function u(g , m) is medium.
3. Rule 3: If $Tag_{(2,n)}$ only finds $Tag_{(1,n)}$,then the membership function u(g , m) is low.

These three Fuzzy FCA rules for multi-layer bidirectional co-occurrence approach reduce the computing cost than the traditional FCA approach. Moreover, the covered attribute is similar than the original FCA approach.

4. Experiments

This section presents the experiment environment and results obtained. The aim of this research is to understand the Fuzzy FCA approach is more efficient than the native (count-based) approache. The assumptions of our approaches are: (1) each Web APIs for mashup applications is tag-based; and (2) the service request from developer profile is related to tags in Web APIs. We attempt to validate these assumptions with some experiments.

4.1. Experiment Environment

There are many Web APIs platform for developers, such as Yahoo Pipes, ProgrammableWeb. Even though these web sites are very useful to developing new applications, it is still difficult to perform large scale mashup experiments. In view of this, we conduct our experiment by simulating the process of mashup application with content aggregation. To validate our experiments, the following are used for evaluation parameters and experiment environment setups.

1. Accuracy: Tag Count
2. Data Source: del.icio.us, flickr, D_TO_F (del.icio.us to flickr) and F_TO_D (flickr to del.icio.us)
3. Evaluation Parameter: Single keyword.
4. Approaches: Count-based, FCA and FFCA.

The accuracy is that the amount of tags, which are related to data items. D_TO_F and F_TO_D in data source are mashup applications. We ever test many keywords and find out the result is not only related to tags but also title, databases, systems and so on.

4.2. Result Analysis

Figure 7 shows the result of count-based and FCA approach. Regardless of del.icio.us, flickr, F_TO_D or D_TO_F, FCA approach is more accuracy. Among of these, the result in del.icio.us is excellent than in flickr because del.icio.us is a fully tag-based system [12]. The accuracy in F_TO_D is superior to D_TO_F. The reason is that the impact of del.icio.us for tag is more powerful than flickr for tag.

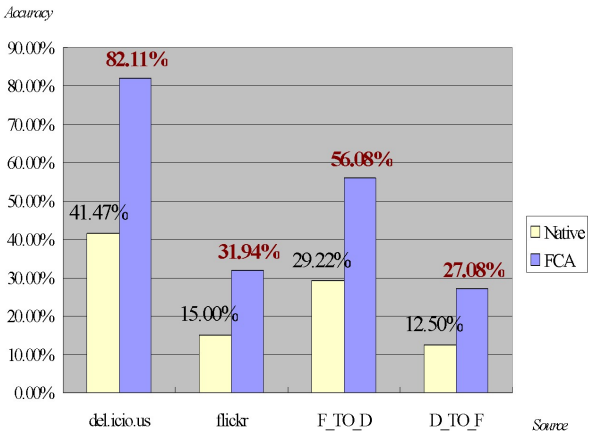

Figure 7. Native vs. FCA Result

Table 2 shows the result of FCA and FFCA. Computing cost is number of concepts in FCA. Covered-Attributes rate is the rate (number of covered attributes / number of total attributes). The computing cost on FFCA with rule 1 is lowest and the on FCA is highest. Although the covered-attributes rate on FFCA with rule 1 is higher than others, the number of covered attributes only includes one attribute, "Travel".

Table2. FCA vs. FFCA Result

	FCA	Rule 1	Rule 2	Rule 3
#Objects	10	10	10	10
#Attributes	40	1	9	16
Computing Cost	31	2	17	19
#Covered Attributes	15	1	5	12
Covered-Attributes Rate	0.3750	1.0000	0.5556	0.7500

FFCA result with rule 3 in Figure 8 is very similar to the one in Figure 1. There are five formal objects including maps, airlines, flights, airline and airfare to share common formal attribute "Travel". The system will cost minimum computing cost to do complicated suggestion jobs.

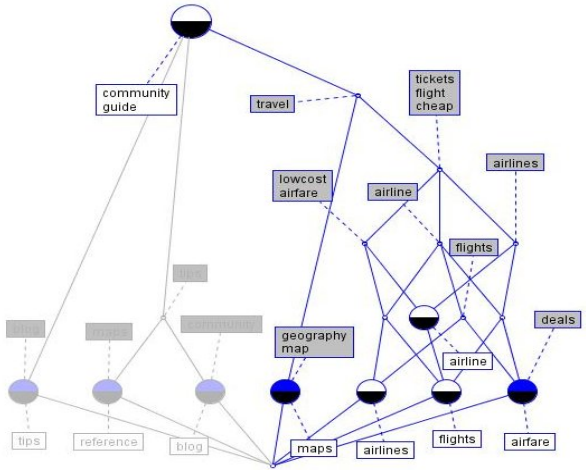

Figure 8. FFCA Result with Rule 3

Compared Figure 1 FCA approach with "Travel" keyword on del.icio.us with Figure 9, we find out that the keyword node, "Vacation" is below the tag node, "Travel". If the mashup developers want to search more information about Vacation Web API, the system will suggest the mashup developers to use Travel Web API.

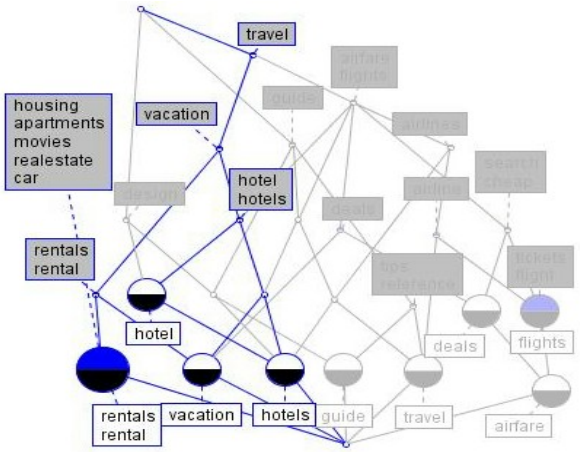

Figure 9. FCA with "Vacation" Keyword on del.icio.us

Figure 10 shows the result of FCA approach on F_TO_D. The order of mashup application is from flickr so the formal objects come from flickr. The covered attributes are covered with formal attributes in del.icio.us and flickr.

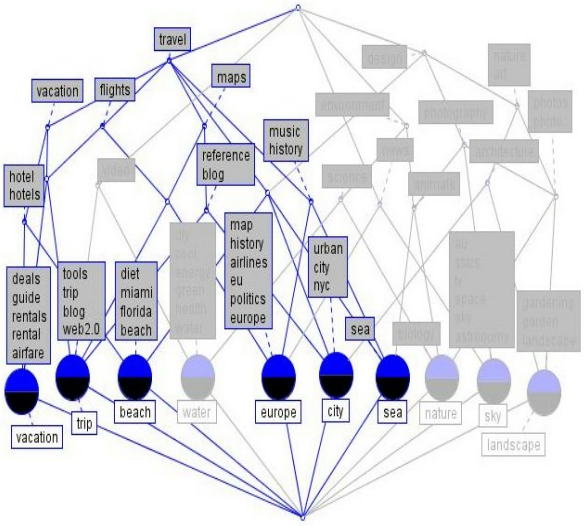

Figure 10. FCA on F_TO_D

5. Conclusion

In this paper, we proposed a tag-analysis approach for mashup applications. We integrated FCA model and Bidirectional Co-occurrence approach. FCA is used to analyze data classification. The domain experts in each field have to judge each formal attribute by themselves. This is not objective when we want to search Web APIs for mashup applications by tags. Hence, we modify the original co-occurrence approach to Bidirectional Co-occurrence approach. Moreover, we use FFCA reduce the computation of FCA. In conclusion, using Bidirectional Co-occurrence approach for mashup applications is not only better than the native method, but also provide a more convenient and scalable development process.

6. References

[1] C. Bussler, "Is Semantic Web Technology Taking the Wrong Turn?," Internet Computing, IEEE, vol. 12, pp. 75-79, 2008.

[2] Grigory Begelman, Philipp Keller, and Frank Smadja, "Automated Tag Clustering Improving search and exploration in the tag space" In Proceeding of 15th International WWWW2006 Conference on Collaborative Web Tagging. Edinburgh, 2006.

[3] D. Fensel, F. van Harmelen, I. Horrocks, D. L. McGuinness, and P. F. Patel-Schneider, "An ontology infrastructure for the Semantic Web," Intelligent Systems, IEEE [see also IEEE Intelligent Systems and Their Applications], vol. 16, pp. 38-45, 2001.

[4] R. W. Bernhard Ganter, Formal Concept Analysis mathematical Foundations: Springer, 1999.

[5] U. Priss, "Formal Concept Analysis in Information Science," Annual Review of Information Science and Technology, vol. 40, pp. p. 521-543, 2006.

[6] G. Hope, G. Hope, T. Wang, and S. Barkataki, "Convergence of Web 2.0 and Semantic Web: A Semantic Tagging and Searching System for Creating and Searching Blogs," in Semantic Computing, 2007. ICSC 2007. International Conference on, 2007, pp. 201-208.

[7] FD P. Smitz, "Collaborative Web Tagging (Inducing ontology from Flickr tags)," in WWW2006, Edinburgh, 2006.

[8] S. A. Huberman, Golder, and B. A., "The Structure of Collaborative Tagging Systems." Information Dynamics Lab, HP Labs. Visited November 24, 2005.

[9] E. Michlmayr, S. Cayzer, and P. Shabajee, "Learning User Profiles from Tagging Data and Leveraging them for Personal(ized) Information Access " in 16th International World Wide Web Conference: Tagging and Metadata for Social Information Organization, Coleman , Banff, Alberta, CANADA, 2007.

[10] W. Zhou, W. Zhou, Z. Liu, and Y. Zhao, "Concept Hierarchies Generation for Classification using Fuzzy Formal Concept Analysis," in Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2007. SNPD 2007. Eighth ACIS International Conference on, 2007, pp. 50-55.

[11] Q. T. Tho, S. C. Hui, A. C. M. Fong, and T. H. Cao, "Automatic fuzzy ontology generation for semantic Web," in Knowledge and Data Engineering, IEEE Transactions on, 2006, pp. 842-856.

[12] V. Zwol and Roelof, "Flickr: Who is Looking?," in Web Intelligence, IEEE/WIC/ACM International Conference, Fremont, CA, USA, , 2007, pp. 184-190