Chapter 22
Investigation of the Effectiveness of Tag-Based Contextual Collaborative Filtering in Website Recommendation

Reyn Nakamoto, Shinsuke Nakajima, Jun Miyazaki, Shunsuke Uemura, and Hirokazu Kato

22.1 Introduction

As the Internet continues to mature and become more accessible to the common user, the amount of available information increases exponentially. Accordingly, finding useful and relevant information is becoming progressively difficult. Moreover, a lot of the information available—blogs, various types of reviews, and so forth—is highly subjective and thus, hard to evaluate purely through machine algorithms. Being subjective in nature, one person may absolutely love something while the next may loathe the same—no single authority exists. It is in these cases where people—more so than the current ability of machine algorithms—are greatly effective in evaluating and filtering this information.

For this reason, the idea of collaborative filtering (CF) was started, extensively researched, and eventually deployed to relatively good amounts of success. Using the people and the community, recommendations of subjective information can be made through the matching of similar users. Sites such as amazon.com [1] or movielens [6], etc. utilize such recommendation methods, matching users based upon their ratings and then producing recommendations. Through this, CF provides personalized recommendations to the users, while at the same time offering the ability to deal with subjective material. However, the failing of CF is that it does not consider why a user likes something and what the user is interested in now. In other words, CF can recommend relevant sites, but does not know why or when it should be appropriate.

Similarly, online social tagging systems also employ the masses to evaluate and describe information. Instead of relying purely upon machine algorithms, people

Reyn Nakamoto, Shinsuke Nakajima, Jun Miyazaki, and Hirokazu Kato
Graduate School of Information Science, Nara Institute of Science and Technology, Ikoma, Nara, Japan
e-mail: {reyn-n,shin,miyazaki,kato}@is.naist.jp

Shunsuke Uemura
Faculty of Informatics, Nara Sangyo University, Ikoma, Nara, Japan
e-mail: UemuraShunsuke@nara-su.ac.jp

X. Huang et al. (eds.), *Advances in Communication Systems and Electrical Engineering.*

themselves describe some resource—whether they be photos, videos, websites—by using tags, or natural language keywords. People are motivated by differing reasons to tag, but the end result is that these resources become easily discoverable through searching through the metadata provided by tags. These tags provide the who, what, when, where and why—they essentially describe that resource, and at the same time, the reason why it was liked and subsequently tagged. Tagging however, fails to provide what CF does—it has yet to provide a system for producing personalized recommendations.

By combining the advantages of the two systems, we have tag-based contextual collaborative filtering (TCCF), as previously described in [8]. By utilizing personalized recommendations provided by CF and the context provided by tags, TCCF aims to provide effective contextual, personalized recommendations.

In this chapter, we describe the website recommendation system we built using TCCF as a recommendation method and tested against three other recommendation/search methods—plain CF, pure tag searching, and CF with tag filtering. From our user testing, TCCF showed itself to be the most effective method of the group, leading the other recommendation methods in both recommendation precision and in users' general impression of the methods. We show these results and discuss the implications in this chapter.

22.2 Related Work

22.2.1 Collaborative Filtering Systems

Collaborative filtering (CF) is the process whereby the community of users is used to sort out relevant or important information from the nonrelevant or nonimportant information. The process is based upon the idea that if users prefer the same item or items, then their preference will be similar for other items liked by similar users. In other words, a user should like the same items that similar users like. From a wider perspective, once users have recorded their preferences within the systems, subsequent users can benefit from the knowledge within them—hence the collaborative aspect of the method.

CF has been proven to work well under certain domains—mainly entertainment domains—such as usenet recommendations [9], movie recommendations [6], product recommendations [1], and so forth.

Many CF systems rely upon a matrix of numerical ratings from users against some resource. Once enough ratings are in place, a similar rating is calculated between the user and other users. Using this, recommendation can be made by calculating the average of similarity ratings times the ratings other users recorded, and then recommending those resources that have scores above a certain threshold.

However, using only numerical values tells only if a user likes something or not—not *why* a user likes something. Thus, in cases where domains are large, this often leads to issues—for example, while two users may have a similar interests in one topic, they may not share the same for another. In particular, domains like Internet websites fall subject to this—users usually have many topics of interest and matching all interests is very difficult using numerical values alone. In addition to this, users are not always in the mood to see all of their interests—rather, they may be only interested in one or two on that day.

22.2.2 Social Tagging Systems

Tagging has been around for some time, albeit known by other terms such as metadata, categorization, labels, and so forth. Tagging is the process of attaching natural language words as metadata to describe a resource like a movie, photo, book, etc. Vocabulary for tagging is usually uncontrolled, so the users themselves can decide what word or combination of words are appropriate.

The current main use of tagging is for the purpose of retrieval, whereby users can search for a tag and the resources tagged with that tag will be returned to the user. In the case of the user who added the tag, they can use tags for later retrieval. For other users, tags serve as a way to discover new resources by searching for whatever tag they are interested in.

In recent years, the advent of social tagging systems have brought tagging back into the limelight. Currently, there are several online social tagging systems that are popular and are the subject of continuing research. They range from website bookmarking such as del.icio.us [4], photo sharing [5], research papers [2], and even people [3]! All of these sites use tagging for many purposes, but in addition to that, they focus on the social networking aspects of tagging to enhance the experience for end users. In their present form, however, tags are generally used for tag searching—user profile matching and subsequent recommendations through this are yet to implemented.

As mentioned before, tags provide the clues as to the context in which a user liked something. These tags are used for several different purposes, including denoting the subject itself, the category, or the refining characteristics of the resource [7]—for example, a picture of a dog would most likely be tagged something like *dog*, *animal*, or maybe *cute*. Thus, tags seem to provide the missing link in CF: they provide the who, what, when, where, and why of a resource—in other words, the context in which the user liked, and subsequently tagged, a resource for later retrieval. Because of this and the similar use of social networking, social tagging systems provide a ideal choice for combination with CF systems.

22.3 TCCF Website Recommendation System

TCCF is the combination of traditional CF systems and social tagging systems that allow for personalized, contextual recommendations. The essential idea is that CF provides personalization, and tags provide the *context* of the users' preferences. We use *context* in the following two ways—first, context as in why a user liked something or why they took the time to tag something. Secondly, we use context as in the user's current state—what interest the user wants to see now. In the first case, it is important to ascertain why the user liked something. Doing so allows for more accurate and more personalized recommendations. In the second case, users often have many interests, but they do not always wish to view all of them all the time. Is it a necessity that we consider what the users' current state is to provide better recommendations. TCCF addresses both of these issues by combining the CF and tagging.

Unlike traditional CF models that use numeric ratings, our TCCF model also uses tags as an indicator of why a user likes something. For example, say we have a website bookmarking system where users can come in and bookmark websites that they enjoy using tags. Normally, the act of bookmarking a website is a strong indicator of whether something is liked. However, the used tags provide the key distinguishing factor from traditional CF systems—the tags attached the resource can be seen as the context in which the user likes the resource. Usually, the user will use tags to describe the resource as the user sees it, and in most cases it would be the context of why they liked something. Thus, from this assumption, we build upon incorporating tags along with CF to provide effective personalized, contextual information recommendations.

We now describe our TCCF method. We explain our method as it is used in our website bookmarking system. In this system, users bookmark websites they like using tags, and subsequently, they can easily retrieve their bookmarks by just searching by the tags. A example scenario is shown in Fig. 22.1. Here, users *A*, *B*, *C*, and *D* are bookmarking the websites they like using tags.

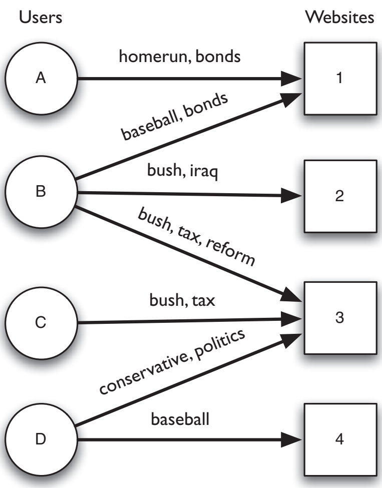

Fig. 22.1 Contextual CF model

22.3.1 TCCF User Similarity Model

Like CF, user similarity is first calculated to determine which users are similar and subsequently, those similar users' preferences are used to find recommendation candidates. Our TCCF user similarity model is based upon both commonly bookmarked websites as well as the tags that they used to bookmark. The TCCF user similarity model between two users A and B is shown in Eqn. 22.1.

$$sim_{ccf}(A,B) = \frac{1}{2n}\sum_{k=1}^{n}\{sim(T_{A\to k}, T_{B\to k}) + 1\} \tag{22.1}$$

where $sim(T_{A\to k}, T_{B\to k})$ is the cosine of the tag vectors that user A and B used to bookmark the same website. Essentially, the common bookmark's tag vectors from users A and B are compared. The same is done with all the common bookmarks A has with B. Those values are then averaged to generate the user similarity score. The addition of one in this equation is the incorporation of standard CF—value is given for having common bookmarks, regardless of whether the tag vectors match or not.

For example, in Fig. 22.1, user B, C, and D all bookmarked website 3. However, the similarity score between users B and C would be higher than C and D's because B and C used similar tags to bookmark the same website. Incidentally, C and D's are still higher than A and C due to the existence of a common bookmark.

22.3.2 TCCF Score Prediction Model

Also similar to CF, results are based upon a score prediction that the system generates. Score prediction is basically the numeric representation of how well the system thinks the user will like some resource. The TCCF score prediction model for a website x's score for a user A is as shown in Eqn. 22.2.

$$score_{pred}(A,x) = \frac{\frac{1}{2}\sum_{k=1}^{n}\{sim_{ccf}(A,S_k) * (\max(sim(T_{S_k\to 1}, T_{S_k\to x}), \ldots, sim(T_{S_k\to m}, T_{S_k\to x})) + 1)\}}{\sum_{k=1}^{n} sim_{ccf}(A,S_k)} \tag{22.2}$$

Essentially, all of similar user S_k's bookmarks are considered as recommendation candidates. Each of these candidates' sites' tag vectors are then compared with each of the tag vectors of the common bookmarks that similar user S_k has with user A. The maximum value of these comparisons is taken and then multiplied by $sim_{ccf}(A,S_k)$—the user similarity score between users S_k and A. The process is repeated for all similar users and averaged to form the score predictions. Again, the addition of one in this equation is the incorporation of standard CF to give value to the existence of a similar user's bookmark, regardless of the similarity of the tag vectors.

For example, in Fig. 22.1, user *B* and *C* are similar. Thus, because user *B* has website 1 and website 2 bookmarked, they are candidates for recommendation. However, since website 2's tag vector is similar to the commonly bookmarked website 3's tag vector, its score prediction will be higher than that for website 1. Website 1 could still be recommended, but due to its dissimilar tag vector, it would be ranked lower than website 2.[1]

22.3.3 System Design

Our system is designed around a website bookmarking system not unlike del.icio.us [4]. It has the same basic feature of bookmarking websites using tags as opposed to the traditional directory structure that most browsers use. Users can bookmark websites using whatever tags to describe the website to themselves, and similarly they can search through their bookmarks using the same tags. Additionally, users can discover other peoples bookmarks by searching through all the bookmarks within the system. By providing features as shown in del.icio.us, users were provided an easy motivation to use the system. Many of the users reported having found many interesting links through these base features themselves.

The interface to the system was done through a Firefox browser plug-in as shown in Fig. 22.2. While browsing, the user can easily access the features of the system. They can quickly search for bookmarks as well as easily add new bookmarks.

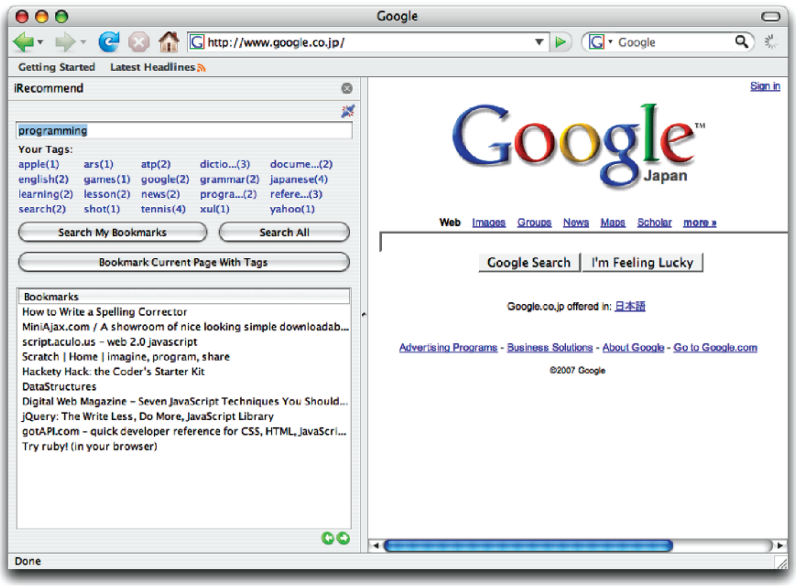

Fig. 22.2 System interface

[1] For further explanation of TCCF, see [8].

All data is stored server-side. Additionally, recommendations are also calculated server-side.

22.4 User Testing

The main goal of the testing was to determine the effectiveness of our TCCF algorithm versus already established algorithms. Furthermore, we wanted to emphasize the importance of context when recommending. Thus, recommendations were done on a *pull* basis, where recommendation results are generated at the user's request. This simulates users requesting only the topic they are interested in now. When they search through their bookmarks using tags, up to five recommendation results appear below the user's bookmarks. Depending on the algorithm, these recommendations are generated based upon the searched tags and/or the user's profile.

We tested our TCCF algorithm versus three other search and recommendation methods: plain collaborative filtering (CF), tag searching (Tag), and lastly collaborative filtering with tag filtering (TagCF).

- CF is basic collaborative filtering. First, user similarity is calculated based upon the user profile, using the number of common bookmarks that a user has with another. The higher the number, the higher the user similarity was. Following this, ranking scores for the websites were generated by averaging the ratings of similar users times their respective user similarities.
- Tag searching is akin to popularity-based searching. Basically, the system retrieves all the bookmarks that have been tagged by a certain tag or set of tags. The results are then ordered by the number of users that have that website bookmarked.
- TagCF is CF with tag filtering. CF recommendations are calculated as done with basic CF. Then, instead of displaying the results as is, the system displays only the results that have the tag that the user searched for.
- Lastly, TCCF is as described in Section 22.3. Score predictions were generated through the TCCF model and then only those results that were linked to the searched tag would be displayed.

A total of nine users were selected to participate in testing.

22.4.1 Test Procedure

Users were first asked to bookmark twenty or more webpages to build their user profile. After this, the testing of each recommendation method took place. For each test session, one of the four recommendation methods were chosen at random. They followed the following procedure:

1. Users selected 10 or more tags of their liking.
2. For each of the selected tags, the system generated up to five recommendations. For each of these recommendations, they reviewed the website.
3. If the website was useful to them *in the context of the tag's meaning*, they pushed the *yes* button that appeared in the interface after selecting a recommendation. If not, they would push the *no* button instead.
4. After finishing rating the recommendations, users completed a general survey regarding the session's recommendation method.

22.4.2 Testing Results

Does TCCF give effective recommendations?

We examine whether our TCCF method provides effective recommendations to the user. We asked the user to determine whether the recommended website satisfied the following two conditions:

1. The website is useful to them.
2. The website's content matches the meaning of the tag.

Thus, if the user was requesting *soccer*, but the system generated a useful, but unrelated tennis site, they were to vote *no*. Again, users have many different interests, and a contextual recommendation system should account for this. The precision of top five recommendation results for each method is shown in Fig. 22.3.

In this case, precision is the number of relevant results divided by the number of all results retrieved—i.e. the number of *yes* ratings divided by the total number of ratings.

As can be seen, our TCCF algorithm received the highest precision of all the four selected algorithms. It is followed by TagCF, then Tag, and finally CF.

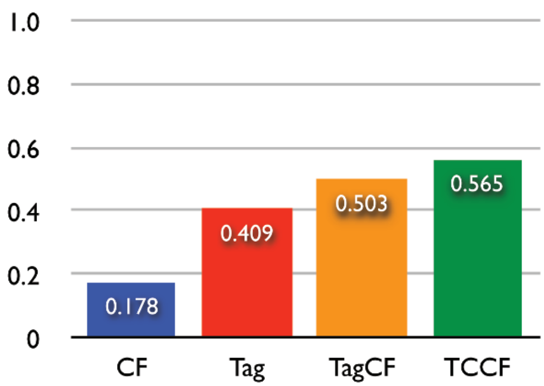

Fig. 22.3 Precision of the top five recommendations

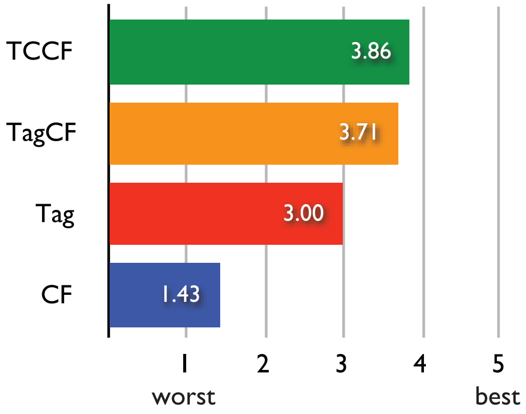

Fig. 22.4 Overall user satisfaction

Does TCCF increase user satisfaction?

From user testing, it would appear that TCCF completes its goal of recommending relevant recommendations when context is considered. Users were also asked to complete a survey after the recommendation rating step was completed. They were asked for their general impression of the algorithm on a scale from 1-5, with five being the best. The results are as shown in Fig. 22.4.

This figure also shows that TCCF comes out on top, although not as far ahead as the precision results. Overall, users were subjectively satisfied with TCCF along with TagCF. Tag searching takes the middle ground again, while CF took last place again.

22.5 Results Discussion

From these results, TCCF provides better recommendations when context is considered. On the other hand, CF does not consider context at either stage. Thus, even though it provided personalized recommendations, CF does not accommodate for changing tag requests. The Tag method considers context in that it produces results related to what the users are interested in now. However, because it does not contain personalization, nor does it consider the context in which the user liked a website, this probably affects why its score is lower. Lastly, TagCF performs reasonably well—it has personalization and considers what the user is interested in at the current moment. It does not, however, consider why the user originally liked a website and this is likely a contributing factor as to why it did not perform better.

Overall, context and personalization are important factors. TCCF accounts for both, and thus, produces more useful results and gives users higher level of satisfaction.

22.6 Conclusions and Future Work

We have described a website bookmarking system using our TCCF algorithm. Our user testing has shown that TCCF is effective in providing effective personalized recommendations when the context is considered. Additionally, user satisfaction was shown to be the highest for TCCF when compared to CF, Tag, and TagCF. From this, it would seem that personalization as well as considering the context is very important when making recommendations to users.

In the future, we plan to further refine the recommendation method. Currently, no score threshold is set for recommendations for the user. Thus, exploration into how high the threshold should be would further improve TCCF's accuracy. We also plan to experiment with altering the weights of tags and CF in the algorithm to find the optimal balance between the two. It is now assumed to be equal. Last, we also intend to experiment with natural language processing to aid in comparing tag vectors with each other.

In terms of system refinement, usability tests must be completed to fully gauge the usefulness versus other information search/recommendation methods. In our test, users were selected and required to follow a set procedure. However, further tests must be completed to find out how the users would react in open-ended tests. We must also determine whether the users would continue to like the system over other currently available search/recommendation methods.

Acknowledgements This research was partly supported by CREST of JST and MEXT (Grant-in-Aid for Scientific Research on Priority Areas #19024058).

References

1. Amazon.com, http://www.amazon.com/
2. Citeulike, http://www.citeulike.org/
3. Consumating.com, http://www.consumating.com/
4. del.icio.us, http://del.icio.us/
5. Flickr, http://www.flickr.com/
6. movielens, http://www.movielens.umn.edu/
7. S. Golder and B. Huberman. The structure of collaborative tagging systems. http://www.hpl.hp.com/research/idl/papers/tags/ tags.pdf
8. Nakamoto R, Nakajima S, Miyazaki J, Uemura S (2007) Contextual collaborative filtering based on tagging information. In *Proceedings of The International MultiConference of Engineers and Computer Scientists* pages 964–969
9. Resnick P, Iacovou N, Suchak M, Bergstorm P, Riedl J (1994) GroupLens: An Open Architecture for Collaborative Filtering of Netnews. In *Proceedings of ACM Conference on Computer Supported Cooperative Work* pages 175–186, Chapel Hill, North Carolina. ACM. http://citeseer.ist.psu.edu/resnick94grouplens.html