Title	UPCV - Distributed recommendation system based on token exchange
Author(s)	Ollikainen, Ville; Mensonen, Aino; Tavakolifard, M.
Citation	Journal of Print and Media Technology Research vol. 2(2013):3, pp. 195 – 201
Date	2013
Rights	Copyright © [2013] IARIGAI. This article may be downloaded for personal use only

Journal of Print and Media Technology Research

3-2013

September 2013

Thematic issue
Content technologies

Guest Editor
Caj Södergård

J. Print Media Technol. Res. 2(2013)3, 125-214

Contents

A word from the Guest Editor

Caj Södergård

VTT - Technical Research Centre of Finland, Espoo

E-mail: Caj.Sodergard@vtt.fi

Content technologies provide tools for processing content to be delivered via any media to the target audience. These tools are applied in numerous ways in media production. Research into content technologies is very active and opens new possibilities to improve production efficiency as well as to enhance the user experience and thereby the business value of media products and services.

This thematic issue focuses on several applications of content technologies. All papers address the user, and the ability to objectively measure and predict the responses various content causes in users is a much needed tool for the media professional. An emerging application proposed in this issue helps journalists find interesting topics for articles from the excessive information available on the internet. Another class of applications dealt with here is recommending content to the users. Relevant recommendations motivate the user to visit and spend time on a web service. Recommenders are therefore important in designing attractive - and monetizable - digital services. As a consequence, this technology is found in many services recommending media items such as music, books, television programmes and news articles. The papers on recommenders in this issue cover the three main methods in the field - content-based, knowledge-based and collaborative - and they bring new perspectives to all three. One such novel perspective which has been evaluated in user studies is that of a portable personal profile.

Most of the included papers are outcomes of the Finnish *Next Media* research program (www.nextmedia.fi) of Digile Oy. Next Media has run from 2010 through 2013 with the participation of 57 companies and eight research organisations. The volume of the program has been substantial; annually around 80 person years with half of the work done by companies and half by research partners. The program has three foci: e-reading, personal media day, and hyperlocal. The papers in this issue represent only a small part of the results of Next Media. As an example, during 2012 the program produced 101 reports, most of which are available on the web.

Even if this thematic issue is centred on work done within the Finnish Next Media program, content technologies are of course studied in many other places around the world. The paper by NTNU in Norway presented here is just one example. Computer and information technology departments at universities and research institutes often pursue content related topics ranging from multimedia "big data" analysis to multimodal user interfaces and user experience. In the upcoming EU Horizon 2020 program, "Content technologies and information management" is a major topic covering eight challenges. This will keep the theme for this thematic issue in the forefront of European research during the years to come.

Caj Södergård, guest editor of this issue of JMTR, holds a doctoral degree in Information Technologies from the Helsinki University of Technology. After some years in industry, he has held positions at VTT as researcher, senior researcher, team manager and technology manager. His work has resulted in several patents and products used in the media field. Currently Caj Södergård is Permanent Research Professor in Digital Media Technologies at VTT.

JPMTR 026 | 1314
UDC 004.42:004.58

Original scientific paper
Received: 2013-08-01
Accepted: 2013-10-25

UPCV - Distributed recommendation system based on token exchange

Ville Ollikainen[1], *Aino Mensonen*[1], *Mozhgan Tavakolifard*[2]

[1] VTT Technical Research Centre of Finland
P. O. Box 1000
FIN-02044 VTT, Espoo, Finland

E-mails: ville.ollikainen@vtt.fi
aino.mensonen@vtt.fi

[2] Department of Computer and Information Science
Norwegian University of Science and Technology
Sem Sælands vei 7-9
N-7491 Trondheim, Norway

E-mail: mozhgan@idi.ntnu.no

Abstract

Most conventional recommendation systems are based on service-specific data repositories containing both user and item data. In this paper, we introduce an alternative approach called UPCV (Ubiquitous Personal Context Vectors) that inherently supports distributed computing and distributed data repositories. The principal idea is that each user-item interaction can update the data associated with *both* the user *and* the item. When updating, item data is made to slightly resemble user data and vice versa, leading to increasing similarity between them. Through interactions, similarity will spread from users to items, from items to users, making it possible to inherently provide user-item, item-item, item-user and user-user recommendations. The principle introduced in this paper can be used as a baseline for the design of different types of collaborative recommender systems. The main advantages of this method are that it requires no content analysis, preserves users' privacy and supports scalability. The method was evaluated using data from 1 575 book club members: the members were asked which books they had read and liked. The quantitative analysis indicates that the most promising results are obtained for active readers. However, even for less active readers and without content analysis, the recommendation list tends to be populated by the same authors and/or authors of the same genre that the readers have liked, leading to meaningful recommendations.

Keywords: recommendation, collaborative filtering, distributed computing, cloud computing, scalability, privacy, deniability

1. Introduction and background

Recommendations have become an integral part of successful web services. Recommendation systems are used on one hand for e-commerce (e.g., Amazon) or advertising (e.g., Google) and on the other hand to improve user experience (e.g., Netflix). The more the amount of information in a service increases, the more important it becomes to help users discover what is most relevant for them.

The recommendation problem can be defined as estimating a user's response to new items based on historical information stored in the system, and suggesting novel and original items for which the predicted response for that particular user is high (Desrosiers and Karypis, 2011). Prediction of user interests is traditionnally through demographic data, such as age, sex, income level and matrimonial status. The availability of more data has led to more sophisticated recommending algorithms being proposed in the literature, most commonly classified into two basic categories: content-based and collaborative recommendations. Content-based recommenders are based on representing the items with a set of attributes and using these attributes to find the most relevant content for a particular user. Collaborative recommendations, on the other hand, learn from the behaviour of users. Some of the more recent novel recommendation techniques use data from social networking (Golbeck, 2006; Liu and Lee, 2010) or use hybrid models merging several techniques (Bobadilla et al., 2013).

This work presents a novel method, UPCV (Ubiquitous Personal Context Vectors). The method models each user and item with a set of tokens, each token carrying a random value. Interaction between user and item results in randomly selected tokens being copied from the token set of the user to the token set of the item, and vice versa. Each interaction increases the number of common tokens among these token sets. When the same user interacts with several items, or the same item is involved in interactions with several users, these common token numbers are spread around, resulting in closer similarities among different token sets in the system. Since tokens spread in interactions, it is likely that similarities between two token sets originate from similar user behaviour. No content analysis is required.

Despite significant advances in the field of contemporary recommender systems, there still remain challenges that limit the effectiveness of these systems. Our proposed model targets these challenges by providing:

- **A broader view of user behaviour:**
  User data gathered from a single service has only a narrow coverage of user behaviour.

- **Domain knowledge independency:**
  Some approaches (e.g., Bäck, 2010) suggest storing personal profiles in a database and delivering from there in order to authorize parties providing personalized services. As such, personal profiles support content-based recommenders, matching user interests to what is available in the service. Content-based approaches require knowledge of the domain in order to match user and item data efficiently. Such techniques have a natural limit in the number and type of features associated, whether automatically or manually, with the objects they recommend. There is a frequent need for domain knowledge (of actors and directors in movie recommendations, for example) and occasionally for domain ontologies (Lops et al., 2011).

- **Preserving user privacy:**
  Privacy concerns have been raised both by recommendation systems gathering data from several services (such as Apple IFA; Stampler, 2012) and by recommendations running on social networking sites.

- **Distributed and cloud based computing:**
  In general, recommender systems are based on service-specific data repositories containing both user and item data. Despite recent development in distributed and cloud computing, single repositories pose an inherent problem in terms of scalability.

Moreover, we here report on an evaluation of UPCV based on collecting data from 1575 book club members. The quantitative results indicate that the most promising recommendations are obtained for active readers: readers with more than 28 book selections in the training data would have expected over 50 % probability of obtaining a successful recommendation in a list containing no more than five books.

However, even for less active readers and with no content analysis, the recommendation list tended to be populated by the same authors and/or authors of the same genre that they had liked, leading to meaningful recommendations. The remainder of this paper is structured as follows: The novel recommendation method based on data fusion is described in section 3 and evaluated in section 4. Discussion and conclusions are in sections 5 and 6, respectively.

2. Related work on item based collaborative filtering method

In this section, we analyse different item based recommendation generation algorithms. We look into different techniques for computing item-item similarities (e.g., item-item correlation vs. cosine similarities between item vectors) and different techniques for obtaining recommendations from them (e.g., weighted sum vs. Regression model).

The item based approach (Sarwar et al., 2001; 2002; Su and Khoshgoftaar, 2009; Linden et al., 2003; Miyahara and Pazzani, 2002; O'Connor and Herlocker, 1999; Xue et al., 2005; Deerwester et al., 1990) looks into the set of items the target user has rated and computes how similar they are to the target item, thereafter selecting the most similar items. At the same time their corresponding similarities are also computed.

Once the most similar items are found, the prediction is then computed by taking a weighted average of the target user's ratings on these similar items. The basic idea in similarity computation between two items i and j is to first isolate the users who have rated both of these items and then to apply a similarity computation technique to determine the degree of similarity. The most popular methods for calculating the similarity are: cosine-based similarity, correlation-based similarity and adjusted cosine similarity. In the following, we briefly describe each similarity method (Sarwar et al., 2001):

- Cosine-based similarity: Two items are represented by two vectors in the m-dimensional user space. The similarity between them is measured by computing the cosine of the angle between these two vectors.

- Correlation-based similarity: Similarity between two items i and j is measured by computing the Pearson-r correlation. To make the correlation computation accurate, we must first isolate the co-rated cases (i.e., cases where the users have rated both).

- Adjusted cosine similarity: Computing similarity by using a basic cosine measure in an item based case has one important drawback: the differences in (rating) scale between different users are not taken into account. The adjusted cosine similarity addresses this drawback by subtracting the corresponding user average from each co-rated pair.

Since the item based approach requires at least one user having rated both items *i* and *j*, the computation is possible only for a limited set, leading to limited coverage which is a common problem in collaborative filtering methods, addressed by e.g., Choi and Suh (2013) and Desrosiers and Karypis (2011).

Once the set of most similar items is isolated (based on the similarity measures), the next step is to look into the target user ratings and use a technique to obtain predicttions.

Two popular approaches are Weighted sum and Regression as explained below (Sarwar et al., 2001):

- Weighted sum: The prediction of an item for a user is computed as the sum of the ratings given by the target user on the similar items. Each rating is weighted by the corresponding similarity between the items. This approach tries to capture how the target user rates the similar items. The weighted sum is scaled by the sum of the similarity terms to make sure the prediction is within the predefined range.
- Regression: The basic idea here is to use the same formula as the weighted sum technique but, instead of using the similar item rating values, this model uses their approximated values based on a linear regression model.

  In practice, the similarities computed using cosine or correlation measures may be misleading in the sense that two rating vectors may be distant (in Euclidean sense), but may yet have very high similarity. In such a case, using the raw ratings of the "so-called" similar item may result in poor prediction.

In summary, the main advantages of item based collaborative filtering methods are that there is no need to consider the content of the items being recommended and that these approaches scale well with co-rated items.

In general, the main shortcomings of these methods are the lack of ability to make recommendations for new users and new items, and the limited scalability for large datasets (Su and Khoshgoftaar, 2009).

3. The proposed method

We now introduce a novel recommendation method based on data fusion, UPCV, most closely related to memory based collaborative filtering in the sense defined in the comprehensive survey of recommender systems by Bobadilla et al. (2013). As such, our approach requires no content analysis.

In UPCV, each user and each item (e.g., a news article) is associated with a set A of tokens x (Equation 1), each represented as a 32-bit integer ($x \in \mathbb{Z}$).

$$A = \{x \in \mathbb{Z} \mid < 0 \leq x < 2^{32}\} \quad [1]$$

When new users or new items appear, their token sets are initialized to contain one single random value. In an interaction between a user i and an item j (e.g., when user i reads the news article j), a small number of tokens X_i are *copied* from the token set of user A_i to the token set of item A_j, see Equation 2.

The reverse also happens, see Equation 3.

$$A_j' = A_j \cup X_i,\ X_i \subseteq A_i \wedge |X_i| \leq {}^{15}/_{100}\ Max\{|A_j|\} \quad [2]$$

$$A_i' = A_i \cup X_j,\ X_j \subseteq A_j \wedge |X_j| \leq {}^{15}/_{100}\ Max\{|A_i|\} \quad [3]$$

We define the maximum cardinality of a token set as 256. Prior to reaching this limit, randomly selected tokens from the receiving token set are deleted so as not to exceed the maximum number. We furthermore limit the number of copied tokens $|X|$ to 15 % of the maximum size of the receiving token set. The selection of this percentage does not appear to be critical since some of our experiments were made using 5 % and 10 % limits, leading to very similar results.

The result of this procedure is an increasing number of common tokens in the respective token sets after each interaction. When the same user interacts with several items, or when the same item is involved in interactions with several users, the token numbers are spread around, resulting in similarities among different token sets in the system. Since tokens spread in interactions, *it is likely that similarities between two token sets originate from similar user behaviour*. Since there is no limit on how far tokens may spread, recommendation coverage is not limited.

The similarity $S(A_i, A_j)$ between the token sets of user i and item j $S(A_i, A_j)$ is measured using the Jaccard similarity measure (Equation 4).

$$S(A_i, A_j) = \frac{|A_i \cap A_j|}{|A_i \cup A_j|} \quad [4]$$

From this point on, the making of recommendations is straightforward, no matter whether they are user-to-user (finding users of similar behaviour), user-to-item (finding items that may interest the user), item-to-item (finding items of similar interest) or item-to-user (finding users who might be interested in an item); it is only necessary to find the token sets (from the total population of size n) that are most similar to the given token set. Thus the recommended item j for a given token set A_i is defined by Equation 5.

$$j = \underset{k=1..n,\, i \neq k}{arg\,max}\; S(A_i, A_k) \qquad [5]$$

In contrast to traditional recommenders, our method inherently supports distributed computing and distributed data repositories. Figure 1 illustrates a concept for storing user data (token set A_i) at the terminal end, while item data (token set A_j) resides at the server end.

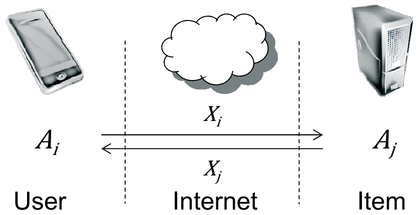

Figure 1: Conceptual illustration of a data transaction between user and item data

Tokens exchanged in a transaction (X_i and X_j respectively) are transferred over the internet. The tokens are fundamentally random numbers and irreversible, they carry no history, thus exchanging the data poses no threat to privacy. This arrangement is motivated by the notion that both parties - the user or the owner of the item(s) - own and have control over their own recommendation data. We consider this a substantial advantage over the mainstream recommendation systems, since it is not only users who may be aware of their privacy, but also enterprises which are reluctant to disclose any business critical information to third parties, such as to an ecosystem owner.

Figure 2 illustrates a concept for obtaining recommendations from a web service. In this case the user submits his/her token set A_i to a web service that has the potential to provide several items (token sets A_j respectively). Such items are recommended for the user which have token sets with the smallest distance to the token set provided.

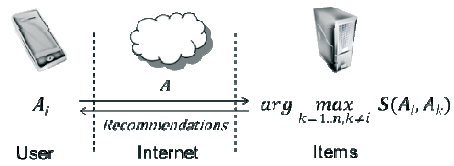

Figure 2: Conceptual illustration for obtaining a recommendation

Once again, all recommendation data for the items in the depicted web service is owned by the web service itself. From the user point of view, the token set A_i is a *de facto* representation of the user profile in an abstract format, basically inherited from other users with somewhat similar behaviour. Therefore, no definite conclusions of personal preferences or history can be drawn.

Furthermore, even with the absence of the user token set A_i, such as in a case of a new user, the depicted web service may generate item-to-item ("see also") recommendations by searching for the smallest distances from the token set of the item j (Equation 6).

$$\hat{j} = \underset{k=1..n,\, j \neq k}{arg\,max}\, S(A_j, A_k) \qquad [6]$$

Since the computation involves only the user terminal and the server in question, the architecture can be considered to inherently comply with internet services and scale up respectively.

4. Evaluation

4. 1 Data sets

For evaluation of the UPCV method we used data from 1 575 book club members. The members were asked which books they had read and liked. We aimed to predict the books (hereinafter "items") a user might be interested in by hiding part of the questionnaire data from the recommender for use in validation only. We therefore divided the sparse data randomly into training and validation data sets. Based on the training set, we generated recommendations for each member aiming to predict which books the member would have in the validation data set.

4. 2 Data gathering

We arranged an online survey about favourite books. Bonnier Books Finland provided us with a list of 1 041 books that have been available for their book club members. This list was divided into a set of shorter lists, based on author names, A-D first, E-H next, etc. The questionnaire asked respondents to select books they had "read and liked".

They were able to select as many books from as many lists as they wished. A link to the online questionnaire was sent to the book club members. 1 575 book club members responded to the questionnaire. The total number of individual selections was 55 434, leading to an average of 27.6 selections per respondent. The standard deviation was 25.9, indicating that we had both active and inactive readers among the respondents.

Figure 3 illustrates the distribution.

The selections were converted into user-item pairs, shuffled into random order and divided into two groups, each consisting of 27 717 user-item pairs in random or-

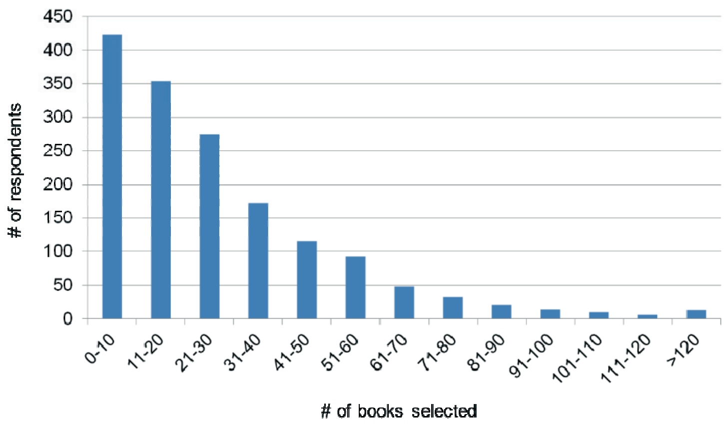

Figure 3: Distribution of the number of books (items) selected by respondents (users)

der. We used the first group as training data, while the second group remained for validation. 1 481 users had at least one selection in both training and validation data.

Token sets for all users and all items were created by entering the user-item pairs of the training data in the recommendation system as an interaction between user and item. We then generated recommendations for each user by searching token sets of the items that had the smallest distance to the token set of the user.

4.3 Quantitative analysis

A recommendation list of length N is considered successful if there is a match between a respective user-item pair in the test data and the N'th recommendation for the user on the list. The search starts from the beginning of the list, on which highest recommendation with smallest distance are provided first.

Figure 4 illustrates that recommendation lists were shorter for users who had a higher number of books selected in the training data. For example, provision of a recommendation list with 5 books would have been achieved with 50 % probability when a user had 28 books in the training data.

Each dot in the figure represents a group of at least 30 users. The figure also illustrates intervals that exclude the upper and lower 10 % of the users in each group, and a best matching trend line (power type).

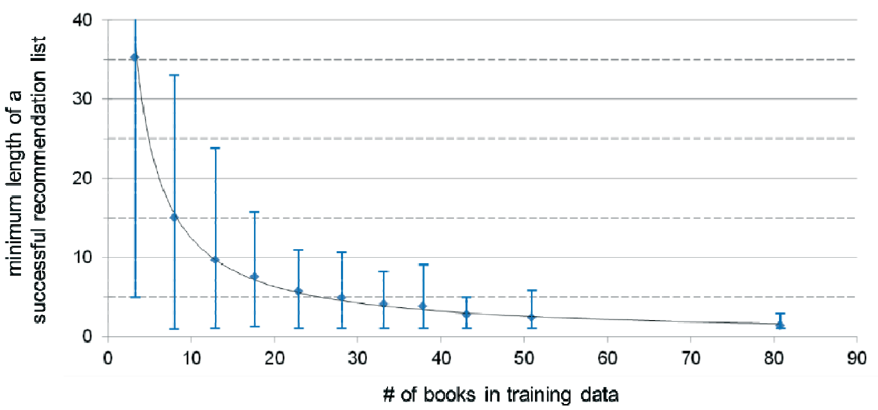

Figure 4: Length of a successful recommendation list (vertical axis) when a user had a certain number of books selected in the training data (horizontal axis)

4.4 Qualitative observations

Regarding recommendations made for users with only a small number of selections in the training data, the quantitative results indicate that a rather long list would be necessary for proper recommendation. However, even in these cases it seems that the same authors and/or authors of the same genre they liked has a tendency to populate the recommendation list. Figure 5 illustrates one randomly selected example.

Although the quantitative length of a successful recommendation is 10, most books in the recommendation list belong to the same genres (thrillers, crime fiction) as two out of the three books in the training data.

Training data

Thomas Harris: Hannibal Rising
Nikolai Gogol: Dead Souls
John Grisham: The Litigators

Recommendation list

1	John Grisham: The Associate
2	John Grisham: The Appeal
3	Mary Clark Higgins: I Heard That Song Before
4	Karin Slaughter: Beyond Reach
5	Kathy Reichs: Cross Bones
6	Joy Fielding: Lost
7	Sandra Brown: Chill Factor
8	Mary Clark Higgins: The Shadow of Your Smile
9	Kathy Reichs: Break no Bones
10	*J. R. R. Tolkien: The Lord of the Rings*

Validation data

Sofi Oksanen: Purge
Arto Paasilinna: A Charming Mass Suicide
J. R. R. Tolkien: The Lord of the Rings
Leo Tolstoi: Anna Karenina

Figure 5: Training and validation data for a randomly selected user with a small number of selections (3) in training data

5. Discussion

We have described a simple collaborative recommender method based on token exchange and designed to protect privacy and support scalability in a distributed architecture. We have evaluated its performance by applying it to questionnaire data from a book club survey.

The quantitative results were better for the most active users. Active readers with more than 28 book selections in the training data had over 50 percent probability of obtaining a successful recommendation in a list containing no more than five books.

Even for less active readers, the recommender system provided selections of a similar genre based on the training data related to the user. We emphasize that these result were obtained *without content analysis*. The only data a book received initially was a single token containning one random integer.

The results indicate that the tokens successfully distributed by interactions among users and items, together with comparison of various token sets, can provide meaningful insights for recommendations.

6. Conclusions and future work

Our study provides an overview of and evaluation results for our proposed approach, which is based on exchanging tokens. Use of this method can easily be extended to other application areas since it is not dependent on any particular assumption about the application area. Exchanging tokens in social networking sites, for instance, might lead to similar tokens for people in the closest social network and - consequently - a higher probability of their receiving similar recommendations.

The method is efficient enough to learn from fairly few interactions, as described in the previous example of a reader with only a couple of books in the training data. With its short required learning time, the method can be beneficial for temporary content, such as news articles.

Generally, there seems to be a trade-off between privacy, trust and recommendation quality. If a service has a comprehensive view of the behaviour and preferences of the user, recommendations may become very accurate. However, in this case, the user must have an indisputable trust in the service, otherwise privacy is compromised. Our approach has an inherent advantage in this respect, since registering any history of actual interactions is not required. Therefore, as a last resort for privacy, the tokens may be said to have been inherited from any interaction; our approach provides a fair degree of deniability.

Further studies are necessary to investigate how to assign tokens to item properties (e.g., keywords of a news article, or its semantic network). Rather than an interaction with the article itself, reading would initiate a series of interactions with various properties. Recommendations would then aggregate results by finding items with the most highly ranked properties for the user.

Moreover, we aim to use considerably larger data for evaluation. It is likely that related data mining studies (e.g., Segond and Borgelt, 2011) may further help in improving and optimizing UPCV.

Acknowledgements

In addition to the authors, a number of people have made the study possible: thus, we thank Minna Kokka (Bonnier Books Finland) and Katri Grenman (VTT) for arranging the survey and our colleagues at VTT, especially Raimo Launonen, Caj Södergård and Atte Kortekangas, for contributing to the development of UPCV, and Bidyut Patra and Alain Boyer for most valuable comments in finalizing this paper.

We also thank VTT's Open Smart Spaces programme (especially Heikki Ailisto), the Tekes Ubicom programme (especially Pekka Yrjölä), the Digile Oy research programme Next Media (especially Eskoensio Pipatti), and the ENPI project Digital Sphere - all these projects and the respective persons for making the UPCV development possible.

References

Bobadilla, J., Ortega, F., Hernando, A. and Gutiérrez, A., 2013. Recommender systems survey. *Knowledge-Based Systems*, 46(0), pp. 109-132

Bäck, A., 2010. Käyttäjän itsensä hallitsema semanttinen profiili verkkopalveluiden personointiin. *GT-lehti*, No. 4, 2010. Espoo: VTT, pp. 25-27

Choi, K. and Suh, Y., 2013. A new similarity function for selecting neighbors for each target item in collaborative filtering. *Knowledge-Based Systems*, 37(0), pp. 146-153

O'Connor, M. and Herlocker, J., 1999. Clustering items for collaborative filtering. *Proceedings of the ACM SIGIR Workshop on Recommender Systems*. UC Berkeley, pp. 121-128

Deerwester, S. T., Dumais, S., Landauer, T. K., Furnas, G. W. and Harshman, R., 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6), pp. 391-407

Desrosiers, C. and Karypis, G., 2011. *A Comprehensive Survey of Neighborhood-based Recommendation Methods*. Boston, MA: Springer US

Golbeck, J., 2006. *Generating predictive movie recommendations from trust in social networks*. Berlin: Springer-Verlag

Linden, G., Smith, B. and York, J., 2003. Amazon.com recommendation - Item-to-item collaborative filtering *IEEE Internet Computing*, 7(1), pp. 76-80

Liu, F. and Lee, H. J., 2010. Use of social network information to enhance collaborative filtering performance. *Expert Systems with Applications*, 37(7), pp. 4772-4778

Lops, P., De Gemmis, M. and Semeraro, G., 2011. *Content-based Recommender Systems: State of the Art and Trends*. Boston, MA: Springer US

Miyahara, K. and Pazzani, M. J., 2002. Improvement of Collaborative Filtering with the Simple Bayesian Classifier Information Processing Society of Japan. *IPSJ Journal*, 43(11), pp. 3429-3437

Segond, M. and Borgelt, C., 2011. *Item Set Mining Based on Cover Similarity*. Berlin, Heidelberg: Springer

Stampler, L. 2012. Here's Everything We Know about IFA, The iPhone Tracking Technology in Apple's iOS 6. *Business Insider*, October 15, 2012. [online] Available at http://www.businessinsider.com/everything-we-know-about-ifa-and-tracking-in-apples-ios-6-2012-10 [Accessed 23 July 2013]

Sarwar, B., Karypis, K., Konstan, J. and Riedl, J., 2001. Item-based collaborative filtering recommendation algorithms. *Proceedings of the 10th international conference on World Wide Web*, pp. 285-295

Sarwar, B., Karypis, K., Konstan, J. and Riedl, J., 2002. Recommender Systems for Large-Scale E-Commerce: Scalable Neighborhood Formation Using Clustering. *Proceedings of the Fifth International Conference on Computer and Information Technology*

Su, X. and Khoshgoftaar, T. M., 2009. A Survey of Collaborative Filtering Techniques. *Advances in Artificial Intelligence*, vol. 2009, Article ID 421425. doi:10.1155/2009/421425

Xue, G.-R., Lin, C., Yang, Q., Xi, W., Zeng, H.-J., Yu, Y. and Chen, Z., 2005. Scalable Collaborative Filtering Using Cluster-based Smoothing. *Proceedings of the ACM SIGIR Conference*, pp. 114-121