The Importance of Similarity Metrics for Representative Users Identification in Recommender Systems

Olga Georgiou, Nicolas Tsapatsoulis

HAL Id: hal-01060646
https://inria.hal.science/hal-01060646

Submitted on 16 Nov 2017

The importance of similarity metrics for representative users identification in Recommender Systems

Olga Georgiou and Nicolas Tsapatsoulis

Cyprus University of Technology,
P.O. Box 50329, 3036, Lemesos, Cyprus,
{olga.georgiou,nicolas.tsapatsoulis}@cut.ac.cy
http://www.cut.ac.cy

Abstract. In this paper we explore the efficiency of recommendation provided by representative users on behalf of cluster members. Clustering is used to moderate the scalability and diversity issues faced by most recommendation algorithms face. We show through extended evaluation experiments that cluster representative make successful recommendations outperforming the K-nearest neighbor approach which is common in recommender systems that are based on collaborative filtering. However, selection of representative users depends heavily on the similarity metric that is used to identify users with similar preferences. It is shown that the use of different similarity metrics leads, in general, to different representative users while the commonly used Pearson coefficient is the poorest similarity metric in terms of representative user identification.

Keywords: representative users, collaborative filtering, recommender systems, similarity metrics, user clustering

1 Introduction

People rely on recommendations related to products, services and so forth in order to find the product or service that suites them mostly [9]. The enormous increase of web products, web services, e-commerce environments and so forth, increases also the demand for new approaches that will help users by providing intelligent recommendations [10]. Market trends shows that online buyers will double, over the coming years [18]. The need for trusted [4] and accurate [2] recommendation now is stronger than ever. A good example is the challenge declared by the Netflix Inc. which was searching for the most improved performance system, for movie recommendations and awarded the winners with the impressive prize of one million dollars [14].

Recommender systems face several challenges such as the scalability when dealing with enormously large amount of data and the requirements demand immediate reaction to online requests. The data sparsity which is another challenge, appears when a new user enters the system and makes it hard to find

similarities with existing users. The diversity challenge [23], describes the problem were users who have similar preference in one category of items they may have totally different judgment on items of another kind, while the synonymy problem, refers mostly to the tendency to name an item or service with many different names causing the similarity estimation algorithms to treat the same item named differently as multiple items [16].

In this paper we attempt to improve recommendation efficiency and eliminate most of the above mentioned problems. Starting with the use of clustering we aim to overcome the scalability problem by seeking users for recommendation within smaller and highly similar clusters instead of an overwhelming number of users [1]. Each cluster is shaped based on current user's preferences (ratings) towards similar data objects (like movies, books, jokes, etc), therefore the diversity problem is also handled since user judgment is based on a specific item each time. The next step, following clusters' formation, is the identification of a representative user per cluster. This user is the one achieving the highest similarity score among all the members of his/her cluster. Representative users provide recommendations towards the other cluster members, succeeding the significant improvement of scalability problem.

Identification of representative users depends heavily on the metric used to compute similarity scores among the cluster members [3], [13]. Our main concern is that both the Pearson correlation coefficient and Cosine based metrics [18] are biased toward the most experience users of a specific item [17]. We propose two modified versions of the Cosine-based correlation coefficient that alleviate this bias and lead to representative users, whose recommendation accuracy is higher.

The remaining of the paper is organized as follows. In Section 2 we present a short literature review on similar topics. The proposed recommendation algorithm and similarity metrics are described in Section 3. Section 4 presents our experimental framework, the details regarding the datasets used and discussion on testing findings. Finally, conclusions are drawn and further work hints are given in Section 5.

2   Literature Review

The market interest for recommendation systems improvement arise from the need of companies to provide more accurate recommendations to their customers and have an additional advantage toward their competitors. An example of market's constant seeking for recommendation improvement is the Netflix challenge as mentioned earlier. It is significant to emphasize the large number of participants and the interest presented from many scientist toward the specific competition, which was one of the reasons that lead the company to declare a second round, with more impressive prizes [14].

Several researchers investigated and evaluated existing collaborative filtering algorithms and techniques. Some of them stroke the area of trusted user identification which presents similarities with the representative user identification problem. However, the target there is to separate malicious users from trusted

ones [15],[22]. Another similarity with the specific research, was the use of clusters. Clustering techniques were initially proposed to address the problem of scalability [5],[21], although their performance were inferior to the straight collaborative filtering approach. This case is not valid anymore, since new clustering based recommender systems show excellent performance [12],[19].

The importance of similarity metrics in collaborative filtering recommender systems is, somehow, underestimated. Emphasis is given either to alternative recommendation techniques like the Top-N recommendation [11] or to the overall performance evaluation [10] and less to the similarity metrics [3]. The most commonly used similarity metrics in collaborative filtering approaches are the Pearson correlation coefficient and variations of the Cosine correlation coefficient [18] and the influence of similarity metrics in recommendation efficiency was examined [17]. The authors proved, through a series of testing experiment, that the performance of these commonly used correlation approaches is rather poor and new similarity metrics must be investigated. This is the area that the current work emphasizes and additionally, it is important to state that the influence of similarity metrics in recommendation performance is much higher in clustering based approaches were clusters are formed and representative users are selected on the basis of these metrics.

3 The Proposed Method

The recommendation problem can be formulated as follows: Let C be the set of users (customers) and let I be the set of all possible items that the users can recommends, such as books, movies, or restaurants. Let also u be a utility function that measures the usefulness (as may expressed by user ratings) of item i to user c_j, i.e., $u : C \times I \to \Re$. The usefulness of all items to all users can be expressed as a matrix U with rows corresponding to users and columns corresponding to items. An entry $u(c_j, i)$ of this matrix may have either positive value indicating the usefulness (rating) of item i to user c_j or a zero value indicating that the usefulness $u(c_j, i)$ has not been evaluated. The recommendation problem can be seen as the estimation of zero values of matrix U from the non-zero ones.

Recommendation in the collaborative filtering approach requires some similarity $r(c_a, c_b)$ between users c_a and c_b to be computed based on the items that both of them evaluated with respect to their usefulness. The most popular approaches for user similarity computation are Pearson correlation and Cosine-based metrics. Both of these methods produce values $r(c_a, c_b) \in [-1\ 1]$. In this paper we propose two alternatives of the Cosine correlation coefficient for similarity computation. However, unlike the classic collaborative filtering approach, recommendations are given only by the representative user toward the other cluster members. The proposed method involves the following steps: (1) the users of an existing recommendation system are partitioned into clusters based on their profile of choices made in the past, (2) for each cluster a representative user is identified (using a variety of similarity metrics), (3) once an active user seeks for

recommendation, its cluster is identified and the corresponding representative user is selected to provide his/her recommendations.

3.1 Clustering using K-Means method

A cluster is a collection of data objects that are similar to one another. The distance measure of the similarity between two objects (users $\underline{c}_i$ and $\underline{c}_j$), influences the shape of clusters, as some elements may be close to one another according to one distance and farther away according to another. A typical similarity metric used in clustering is Euclidean distance; however, when clustering is applied in the context of recommender systems, Pearson and Cosine coefficient based distances are the obvious choices.

The K-means clustering algorithm is widely used in clustering based recommendation systems mainly due to its simplicity [12]. It aims to partition N_c users defined by the user profile vectors $\{\underline{c}_1, \underline{c}_2, ..., \underline{c}_{N_c}\}$ into K clusters ($K << N_c$) $\mathbf{S} = \{S_1, S_2, ..., S_K\}$ in which each user profile vector belongs to the cluster with the nearest mean. The user profile vectors correspond to the rows of the matrix U mentioned earlier.

The K-means algorithm is similar to the expectation-maximization algorithm for mixtures of Gaussians in that they both attempt to find the centers of natural clusters in the data. The optimization criterion is to find the partition $\mathbf{S}$ that minimizes the within-cluster sum of squares (WCSS):

$$\mathbf{S}^o = \underbrace{argmin}_{\mathbf{S}} = \sum_{i=1}^{K} \sum_{\underline{c}_j \in S_i} \|\underline{c}_j - \underline{\mu}_i\|^2 \tag{1}$$

where $\underline{\mu}_i$ is the mean vector of datapoints (user profile vectors) in cluster S_i.

3.2 Similarity metrics

The aim of this paper is to show that the similarity metric that is used for the computation of a representative user r_i of cluster S_i affects the overall recommendation efficiency. Let us denote $\underline{c}_i^{I_c}$ and $\underline{c}_j^{I_c}$ the user profile vectors for users c_i and c_j computed on the set of items I_c that both of them evaluated. A typical metric used for the similarity computation of users c_i and c_j is the Cosine correlation coefficient:

$$r(c_i, c_j) = \frac{\underline{c}_i^{I_c} \cdot \underline{c}_j^{I_c}}{\|\underline{c}_i^{I_c}\|\|\underline{c}_j^{I_c}\|} \tag{2}$$

where "·" denotes the dot product of two vectors and $\|\underline{x}\|$ denotes the Euclidean norm of vector $\underline{x}$.

In an actual situation, different users may use different rating scales, which the vector Cosine similarity cannot take into account. To address this drawback, the adjusted Cosine similarity coefficient known as Pearson correlation coefficient is used:

$$r(c_i, c_j) = \frac{(\underline{c}_i^{I_c} - \overline{c}_i) \cdot (\underline{c}_j^{I_c} - \overline{c}_j)}{\|\underline{c}_i^{I_c} - \overline{c}_i\| \|\underline{c}_j^{I_c} - \overline{c}_j\|} \tag{3}$$

where $\overline{c}_i$ and $\overline{c}_j$ are the average ratings over all rated items of users c_i and c_j respectively.

The use of Pearson or Cosine correlation coefficient for representative user identification creates bias toward very active users, the ones that rated lot of items. For further explanation, consider a recommender system with N_c users, N_I items, and a cluster S_i containing N_{S_i} users. It is common that $N_{S_i} << N_c$ and $N_{S_i} << N_I$; are the users that are classified into many clusters and the number of cluster members is much smaller than the total number of items. As a result, an average user rates a limited number of items, assume that there is a very active user who rated much more items that the average user. Since both the Cosine and Pearson coefficients are computed based on the commonly rated items, there is an increased probability for a very active user to present similarity (even with a low score) with all cluster members. As a result the specific user will be selected as cluster's representative, even though his/her actual similarity with many of the cluster members is low. In order to overcome this limitation we propose the *min_norm* Pearson coefficient as follows:

$$r(c_i, c_j) = \frac{(\underline{c}_i^{I_c} - \overline{c}_i) \cdot (\underline{c}_j^{I_c} - \overline{c}_j)}{\|\underline{c}_i^{I_c} - \overline{c}_i\| \|\underline{c}_j^{I_c} - \overline{c}_j\|} \frac{N_I^{ij}}{min(N_I^i, N_I^j)} \tag{4}$$

where N_I^{ij} is the number of items commonly rated by users i and j, N_I^i is the total number of items rated by user i and N_I^j is the total number of items rated by user j.

In cases where $N_I^i << N_I^j$ or $N_I^j << N_I^i$ the *min_norm* Pearson coefficient creates bias against very active users making it difficult to create dense clusters. The *max_norm* Pearson coefficient is proposed to moderate this problem by decreasing the similarity value in cases where there is a large difference among the total number of ratings per user:

$$r(c_i, c_j) = \frac{(\underline{c}_i^{I_c} - \overline{c}_i) \cdot (\underline{c}_j^{I_c} - \overline{c}_j)}{\|\underline{c}_i^{I_c} - \overline{c}_i\| \|\underline{c}_j^{I_c} - \overline{c}_j\|} \frac{N_I^{ij}}{max(N_I^i, N_I^j)} \tag{5}$$

It is shown in the experimental results session that both representative users selected either by using the *max_norm* or *min_norm* Pearson coefficient perform better that the classic Pearson coefficient.

4 Experimental evaluation

The aim of experimental evaluation is to prove the efficiency of the proposed similarity metrics method, for representative user identification. The recommendation efficiency of representative users identified, with various similarity metrics, is measured based on the experiments conducted. The quality of the overall recommendation method is also benchmarked against no recommendation.

4.1   Evaluation framework

Experimental evaluation of the proposed method was done with the aid of two widely used datasets. The first dataset was collected from Jester online Jokes Recommender System [6] contains over than 4.1 million continuous ratings $[-10.00 \quad 10.00]$ of 100 jokes from 73421 users. It is one of the most widely used dataset for evaluating algorithms for recommender systems. The second dataset was collected from the GroupLens research project   [8] and consists of 943 users (demographical information's are also provided for each user), 1682 movies (more details related to the type of movie are also provided) and a record file of 100000 ratings with a rating scale 1 to 5, with 5 being the highest score. In both cases we divided the datasets ratings into 60% for training and 40% for testing producing two matrics $T_r \in \Re^{N_c x J_1}$ and $T_s \in \Re^{N_c x J_2}$ respectively.

Our experiments where conducted on a typical PC, running Windows XP, and with the aid of Matlab platform (`http://www.mathworks.com/`). First, the K-means algorithm (Matlab implementation) was used to partition the N_c users into a varying number of clusters. For each cluster the user with the highest similarity with the other cluster members was selected as the representative user of the cluster. Similarity among users was measured by comparing their ratings on the training set with the aid of Pearson coefficient, max_norm and min_norm similarity metrics (see equations (3), (5), (4) respectively). In general three different representative users per cluster were identified; however, in several cases, especially in the Joke dataset, the identified representative users coincide.

Recommendations of the representative users were considered the rated items (jokes / movies) with value higher than a threshold T. We selected $T = 3$ for the Joke dataset and $T = 2$ for the GroupLens dataset; the main reason for T restriction in both cases, was the ranging ranges of the two datasets. however, the threshold value does not actually affect the experiments, recommendation efficiency will remain. Removing the thresholds will lead to negative recommendations which is out of scope for this paper. Let c_i^j be the i-th member of the j-th cluster and c_r^j be the representative of that cluster. Let us, also, denote the set of items rated by c_i^j as $\mathbf{I}_i^j$ and the set of high rated items of the same user as $\mathbf{H}_i^j$ (obviously $\mathbf{H}_i^j \subseteq \mathbf{I}_i^j$). For each cluster member two values are computed: the satisfaction without recommendation s_i^j and the satisfaction after receiving recommendation by the cluster representative $\widetilde{s}_i^j$. The last value was computed using the three different similarity metrics mentioned earlier. The s_i^j and $\widetilde{s}_i^j$ values are computed as follows:

$$s_i^j = \frac{Car(\mathbf{H}_i^j)}{Car(\mathbf{I}_i^j)} \qquad \widetilde{s}_i^j = \frac{Car(\mathbf{H}_i^j \bigcap \mathbf{H}_r^j)}{Car(\mathbf{I}_i^j) \bigcap \mathbf{H}_r^j)} \tag{6}$$

where $Car(\mathbf{X})$ denotes the cardinality of set $\mathbf{X}$.

4.2   Results and discussion

Figure 1 shows the recommendation quality, in the GroupLens dataset, as a function of the number of clusters for the three similarity metrics used. For

comparison we plot also the user satisfaction without receiving any recommendation. The representative user identified with the *max_norm* metric clearly provides more accurate recommendations than the representatives, of the other two methods. The *min_norm* metric also outperforms the Pearson coefficient in all cases. Nevertheless, Figure 1 shows that recommendation provided by representative users is effective despite the similarity metric used; user satisfaction is always higher, in cases were recommendations was provided, compared to user satisfaction were no recommendations were provided. An exception occurs in cases where very few clusters are created and recommendations provided only from representative users identified by the Pearson correlation coefficient. Partition into small number of clusters, leads to very sparse and non-homogeneous clusters. As a consequence the identification of representative users, is risky and prone to error.

The conclusions drawn above also hold for the Joke dataset, however, it is shown in Figure 2 that the difference in recommendation efficiency for the three methods is smaller. In addition the overall recommendation efficiency of Jokes dataset is much higher than of the GroupLens dataset. Both observations can be explained by carefully examining the nature of the two datasets. In the Joke dataset the number of items N_I is small (100) and the variation in the number of items evaluated by each user is also small (the minimum number of items evaluated by a single user is 60 while the maximum is 100). In contrary the GroupLens dataset contains much more items (1682) and much less users (943), resulting a very large (minimum 20, maximum 737) variation in the number of items evaluated by each user. For this reason the representative users identified by the different similarity metrics rarely coincide in the GroupLens dataset while coincidence is very common in the Joke dataset.
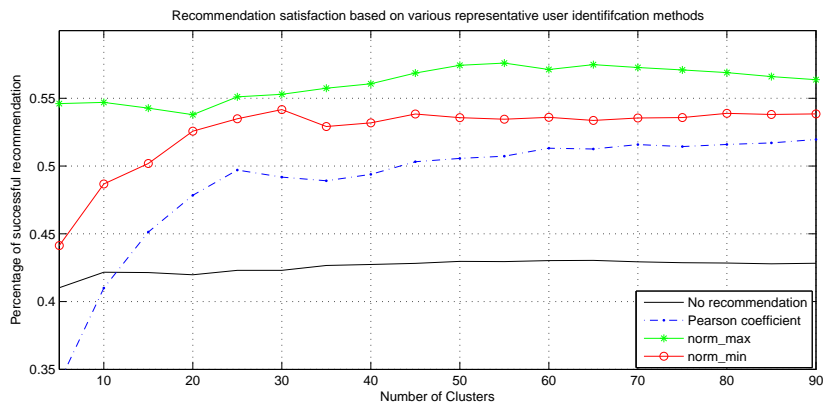

Fig. 1. GroupLens dataset: Recommendation quality using one representative per cluster identified using the Pearson coefficient, the *max_norm* and the *min_norm* similarity metrics. Benchmarking against no recommendation is also shown.

Fig. 2. Jokes Dataset 1: Recommendation quality using one representative per cluster identified using the Pearson coefficient, the *max_norm* and the *min_norm* similarity metrics. Benchmarking against no recommendation is also shown.
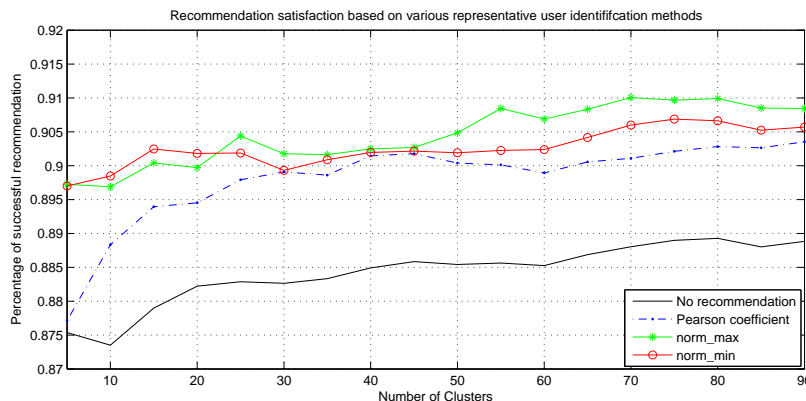
5   Conclusion and further work

Recommender systems are becoming a valuable tool for E-commerce on the Web as well as for information retrieval in general. They are being stressed by the huge volume of customer data in existing corporate databases, and will be stressed even more by the increasing volume of customer data available on the Web. In order to keep up with this information explosion the scalability of recommender systems must be improved while the time complexity of recommendation algorithms must be kept low. These are the main reasons the clustering based recommendation systems gain attention. In this paper, we have proposed two different similarity metrics for representative user identification. Representative users provide recommendations on behalf and for the other cluster members. It was shown in this paper that the selection of an appropriate similarity metric for representative user identification affects the recommendation efficiency of the overall system. It was experimentally justified, with the aid of two widely used datasets, that the proposed modifications to the Pearson coefficient method, lead to much better recommendation efficiency.

Future work includes testing the proposed method in sparse datasets like the one of the book recommendation system [23]. The creation of dense clusters in such datasets is challenging. In addition Top-N recommendation methods will be investigated using all cluster members recommendations instead of the ones of clusters' representatives. Finally, methods for choosing the best cluster representatives will be also explored.

References

1. Braak, P. t., Abdullah, N., Xu, Y.: Improving the Performance of Collaborative Filtering Recommender Systems through User Profile Clustering. In: the 2009 IEEE/ACM Int'l Joint Conference on Web intelligence and intelligent Agent Technology, vol. 03, pp. 147–150. IEEE Computer Society, Washington (2009)

2. Carenini, G.: User-Specific Decision-Theoretic Accuracy Metrics for Collaborative Filtering. In: Proceedings of the 'Beyond Personalization' Workshop, Intelligent User Interfaces (IUI'05), pp. 26-30 (2005).

3. Chan, Y. S., Tou, Ng. H.: MAXSIM: A Maximum Similarity Metric for Machine Translation Evaluation. In: Proceedings of ACL-08: HLT, pp 55-62 (2008).

4. O'Donovan, J., Smyth, B.: Trust in recommender systems. In: 10th Int'l Conference on Intelligent User Interfaces (IUI'05), pp. 167–174. ACM, New York (2005)

5. Ester, M., Kriegel, H.-P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: 2nd International Conference on Knowledge Discovery and Data Mining, pp. 226-231, Portland, OR (1996)

6. Goldberg, K., Roeder, T., Gupta, D., Perkins, C.: Eigentaste: A Constant Time Collaborative Filtering Algorithm. In: Information Retrieval 4(2), pp. 133-151 (2001).

7. Goldberg, D.: Genetic Algorithms. Addison-Wesley, 1989.

8. The GroupLens Research Project, `http://www.grouplens.org/`

9. Herlocker, J. L., Konstan, J. A., Riedl, J. T.: Explaining Collaborative Filtering Recommendations. In: Proceedings of the 2000 ACM conference on Computer supported cooperative work, pp. 241-250 (2000).

10. Herlocker, J. L., Konstan, J. A., Terveen, L. G., Riedl, J. T.: Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems 22(1), 5–53 (2004)

11. Jamali, M. and Ester, M. 2009. Using a trust network to improve top-N recommendation. In: Proceedings of the 3rd ACM Conference on Recommender Systems (RecSys'09), pp. 181–188 (2009)

12. Kim, K., Ahn, H.: A recommender system using GA K-means clustering in an online shopping market. Expert Systems with Applications: An International Journal 34(2), 1200–1209 (2008)

13. McLaughlin, M. R., Herlocker, J. L.: A collaborative filtering algorithm and evaluation metric that accurately model the user experience. In: 27th Annual International ACM Conference on Research and Development in Information Retrieval (SIGIR'04), pp. 329–336. ACM, New York (2004)

14. Netflix Inc. `http://sacramento.bizjournals.com/sacramento/stories/2009/09/21/daily3.html` (Last visited on: 24/May/2010)

15. Sarda, K., Gupta, P., Mukherjee, D, Padhy, S., Saran, H.: A Distributed Trust-based Recommendation System on Social Networks. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 160–168 (2008).

16. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering. In: 5th International Conference on Computer and Information Technology. ICCIT, (2002)

17. Spertus, E.: Evaluating Similarity Measures: A Large-Scale Study in the Orkut Social Network. In: Proceedings of the 16th international conference on World Wide Web, pp. 131–140 (2005).

18. Su, X., Khoshgoftaar, T. M.: A Survey of Collaborative Filtering Techniques. In: Advances in Artificial Intelligence, vol. 2009, Article ID 421425, Hindawi Publishing Corporation (2009)
19. Truong, K., Ishikawa, F., Honiden, S.: Improving Accuracy of Recommender System by Item Clustering. IEICE - Transactions on Information and Systems E90-D(9), 1363–1373 (2007)
20. Vogiatzis, D., Tsapatsoulis, N.: Modeling User Networks in Recommender Systems. In: 3rd International Workshop on Semantic Media Adaptation and Personalization (SMAP'08), pp. 106–111. IEEE Computer Society, Washington, DC (2008)
21. Zhang, T., Ramakrishnan, R., Livny, M.: BIRCH: an efficient data clustering method for very large databases. In: the ACM SIGMOD International Conference on Management of Data, vol. 25, pp. 103-114, Montreal, Canada (1996)
22. Zhang, F., Bai, L., Gao, F.: A User Trust-Based Collaborative Filtering Recommendation Algorithm. In: Information and Communications Security. LNCS, vol. 5927, pp. 411–424. Springer, Heidelberg (2009)
23. Ziegler, C.-N., McNee, S. M., Konstan, J. A., Lausen, G.: Improving Recommendation Lists Through Topic Diversification. In: 14th International World Wide Web Conference (WWW '05), pp. 22–32. Chiba, Japan (2005).