RESEARCHERID
THOMSON REUTERS

[Rivera, 6(3): March 2019] ISSN 2348 – 8034
DOI- 10.5281/zenodo.2610866 Impact Factor- 5.070

GLOBAL JOURNAL OF ENGINEERING SCIENCE AND RESEARCHES

A SURVEY ON L-INJECTION: TOWARD EFFECTIVE COLLABORATIVE FILTERING USING UNINTERESTING ITEMS

Hemapriya K.E[*1], Nivetha G[2], Sathya M[3], & Shiva Shamyugtha M[4]

[*1,2,3&4]Department of Computer Science and Application, Sri Krishna Arts and Science College

ABSTRACT

We develop a novel framework, named as l-injection, to address the sparsity problem of recommender systems. By vigilant injecting low values to a selected set of unrated user-item pairs in a user-item matrix, we demonstrate that top-N recommendation accuracies of various collaborative filtering (CF) techniques can be significantly and consistently improved. We first adopt the notion of pre-use preferences of users toward a vast amount of unrated items. Using this notion, we identify uninteresting items that have not been rated yet but are likely to receive low ratings from users, and selectively impute them as low values. As our suggested approach is method-agnostic, it can be easily applied to a variety of CF algorithms. Through extensive experiments with three real-life datasets (e.g., Movielens, Ciao, and Watcha), we prove that our solution consistently and universally enhances the accuracies of existing CF algorithms (e.g., item-based CF, SVD-based CF, and SVD++) by 2.5 to 5 times on average. Furthermore, our solution enhance the running time of those CF methods by 1.2 to 2.3 times when its setting produces the best accuracy.

Keywords: *colonial house, typology, vulnerability index.*

I. INTRODUCTION

The goal of recommender systems (RS) is to suggest appealing items to a user by analyzing her prior preferences. As a large number of online applications use RS as a core component, improving the quality of RS becomes a critically important problem to businesses. Among existing solutions in RS, in particular, collaborative filtering (CF) methods have been shown to be widely effective. Based on the past behavior of users such as explicit user ratings and implicit click logs, CF methods exploit the similarities between users' behavior patterns. However, when the fraction of known ratings in a rating matrix R is overly small (so-called data sparsity problem), CF methods tend to suffer. For an R with m users and n items, if we assume that each user has rated k items on average, the fraction of rated items in R is k /n (= m*k/m*n). Asymptotically, such a fraction of rated items in R is extremely small (i.e., k n). It is common for an e-business to sell millions of items with a very long tail, and many users rate very few items (i.e., cold-start users). The goal of this work is to mitigate such a data sparsity problem to improve top-N recommendation accuracies of CF methods.

Rating distributions of three real-life datasets.

Dataset	Low ratings (1 or 2)	High ratings (3, 4, or 5)
Movielens	17%	83%
Ciao	10%	90%
Watcha	13%	87%

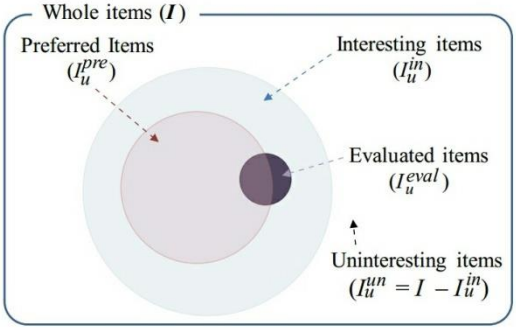
We first argue that ratings in R be often a reflection of the satisfaction of users. Therefore, users tend to rate (high) only the items that they like, and those who are dissatisfied tend not to rate items in R. Corroborating this point, The above table illustrates severe imbalance between low (i.e., 1 or 2) and high (i.e., 3, 4, or 5) ratings from three real-life datasets that we used in our experiments. Note that only a small fraction (i.e., 10–17%) of ratings are low values. Then, a natural question to raise is: how can we identify the unknown opinions of those users who were dissatisfied with and did not leave ratings for items? To answer this question, note that unrated items in R can be classified into three different types:(1) unrated items whose existence users were not aware of, (2) unrated items that users purchased but did not rate, and (3) unrated items that users knew but did not like and did not purchase. We note that the unrated items of the third type, called uninteresting items (denoted by I^{un}), clearly indicate users' latent negative

preferences on them. Therefore, it is better not to recommend those uninteresting items. In order to identify such uninteresting items, we propose to use a new notion of pre-use preference, i.e., an impression of items before purchasing and using them. That is, by definition, uninteresting items indicate the items with low pre-use preferences. Unfortunately, the ratings in R do not indicate pre-use preferences but the preferences after using the items, called post-use preference. Based on this novel notion of pre-use preference and uninteresting item, we develop a solution that consists of three steps: (1) infer the pre-use preferences of unrated items by solving the one-class collaborative filtering (OCCF) problem, (2) assign "low" values to uninteresting items in R, yielding an augmented matrix L, and (3) apply existing CF methods to L, instead of R, to recommend top-N appealing items. This simple-yet-novel imputation solution significantly alleviates the data sparsity problem by augmenting R. Extending our prior work, in this work, we develop a more general l-injection to infer different user preferences for uninteresting items for users, and show that l-injection mostly outperforms 0-injection. The proposed l-injection approach can enhance the accuracy of top-N recommendation based on two strategies: (1) preventing uninteresting items from being comprise in the top-N recommendation, and (2) exploiting both uninteresting and rated items to predict the relative preferences of unrated items more accurately. With the first strategy, because users are aware of the existence of uninteresting items but do not like them, such uninteresting items are likely to be false positives if included in top-N recommendation. Therefore, it is effective to ostracize uninteresting items from top-N recommendation results. Next, the second strategy can be clarified using the concept of typical memory based CF methods. Suppose that a few neighbors of a user u rated an item i high but most neighbors of u considered i uninteresting (thus left i unrated in R). In this case, existing CF methods tend to recommend i to user u. However, if many neighbors of u consider i as an uninteresting item, we should avoid recommending i to u. To summarize, our main contributions are as follows:

• We introduce a new notion of uninteresting items, and classify user preferences into pre-use and post-use preferences to identify uninteresting items. • We propose to identify uninteresting items via preuse preferences by solving the OCCF problem and show its implications and effectiveness.

• We propose low-value injection (called l-injection) to improve the accuracy of top-N recommendation in existing CF algorithms.

• We evaluate the proposed solution with three real life datasets, and demonstrate that our solution consistently outperforms baseline CF methods (e.g., item-based CF, SVD-based CF, and SVD++) with respect to accuracy (by 2.5 to 5 times) and running time (by 2.5 to 5 times) on average.



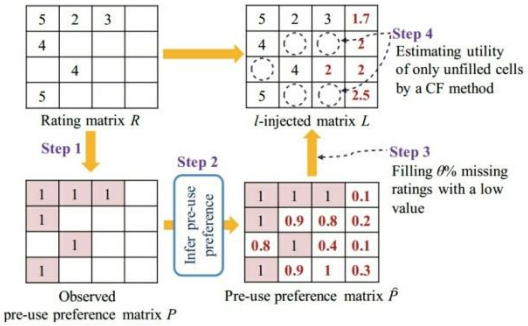
In general, CF methods have been studied under two settings: (1) predicting the ratings of unrated items, and (2) recommending top-N unrated appealing items to users. In this paper, we focus on the top-N recommendation setting, which is more practical in real-world applications.

We first explain some basic notations used throughout this paper. Let U = {u1, . . . , um} be a set of m users, I = {i_1, . . . , i_n} be a set of n items, and r_{ui} be the rating given to item i by user u. A corresponding rating matrix is referred to as R = $(r_{ui})_{m\times n}$, and p_{ui} (respectively q_{ui}) indicates the pre-use (respectively post-use) preference for item i of user u. The pre-use preference p_{ui} is different from a known rating r_{ui} in the rating matrix R, implying post-use preference qui. In theory, both types of preferences p_{ui} and qui exist as a user-item pair (u, i) although they are not always available. Note that it is possible to infer the pre-use preference for item i ∈ I of user u from its external features. After using i, based on the level of her satisfaction, u then assigns a specific score to i, indicating her post-use

preference for i. Therefore, the post-use preference is determined by the inherent features that u had not known before using i.

II. PROPOSED APPROACH

While existing CF methods only employ user preferences on rated items, the proposed approach employs both pre-use and post-use preferences. Specifically, the proposed approach first infers pre-use preferences of unrated items and identifies uninteresting items I^{un}_u. Then, it enriches the rating matrix by exploiting uninteresting items. The existing CF methods equipped with our approach not only benefit from the enriched matrix and but also exclude the uninteresting items from top-N recommendation. Lastly, we analyze the benefits of the proposed approach on account of improving the accuracy greatly. The main challenges of our approach are as follows: (1) how to identify uninteresting items among unrated items and (2) how to exploit uninteresting items discerned in CF methods. To address the first challenge, we infer pre-use preferences for all unrated items and find the unrated items whose pre-use preferences are low. For the second challenge, we build an augmented matrix where some missing entries are imputed by low values if their corresponding items are considered uninteresting. The augmented matrix can be applied to any CF method (thus making our approach method-agnostic), which enables existing CF methods to benefit from uninteresting items in their top-N recommendation.



This picture depicts the overall processes of the proposed approach. First, we build a pre-use preference matrix $P = (p_{ui})m*n$ by examining a rating matrix $R = (r_{ui})m*n$. It is set as one if $r_{ui} \in R$ has been already rated (i.e., u should have liked i if she bought i) (Step 1). It is the highest because p_{ui} is set as a real value. Next, we infer pre-use preference scores on "unrated" user-item pairs (u, i) (i.e., p_{uj} = null) based on other observed pre-use preferences (i.e., $p_{ui} = 1$) and add them in P, which becomes $\hat{P}$ (Step 2). Based on $\hat{P}$, we identify uninteresting items for each user and build a low-value injected matrix $L = (l_{ui})m*n$ (Step 3). That is, if r_{ui} in R is unrated and item i is an uninteresting item for user u, it is imputed by l_{ui}. In the proposed approach, i is determined as the uninteresting item for u if the pre-use preference score $\hat{p}_{ui}$ is ranked in the bottom θ% in $\hat{P}$. The augmented matrix L thus includes both the original ratings for rated items and the imputed ratings for uninteresting items. Lastly, existing CF algorithms are applied to the augmented matrix L (Step 4). We recommend top-N items by predicting the post-use preferences of empty entries (dotted circles). In the following subsections, we explain each step in detail.

III. INFERRING PRE-USE PREFERENCES

It is straightforward to determine a pre-use preference p_{ui} if a user u has already rated an item i (i.e., r_{ui} is not null). This is because i may have been interesting to u at first consideration, i.e., $I^{eval}_u \subseteq I^{in}_u$. As such, we set the pre-use preference p_{ui} as one. However, when u has not rated i (i.e., r_{ui} = null), it is non-trivial to determine p_{ui}. Therefore, it is essential to infer pre-use preferences p_{ui} if r_{ui} is unrated. To address this challenge, we borrow the framework of the one-class collaborative filtering (OCCF) problem. The OCCF problem occurs when a rating score is unary such as clicks, bookmarks, and purchases so that a cell $r_{ij} \in R$ has a null value or a single value indicating "yes." The ambiguity arises from the interpretation of unrated items. That is, it is difficult to distinguish negative and positive examples that co-exist among unrated items. Some unrated items can be positive because the user is not aware of

their existence. On the other hand, some are negative because the user knows about the items but dislikes them. Therefore, she determines not to use them. This problem setting also happens when we infer pre-use preferences for unrated items. That is, known pre-use preferences for rated items have positive values (i.e., $p_{ui} = 1$) and missing pre-use preferences for unrated items are ambiguous. we observe that both unlabeled positive examples ($I^{in}{}_u - I^{eval}{}_u$) and negative examples ($I^{un}{}_u$) co-exist in the set of items whose pre-use preferences are unknown ($I - I^{eval}{}_u$). We thus employ the OCCF method to infer pre-use preferences. In we also demonstrate that the OCCF method is the most effective to infer users' pre-use preferences. The basic idea of the OCCF method is to treat all unrated items as negative examples and to assign weights to quantify the relative contribution of these examples. In our situation, the OCCF method assigns 0 to every p_{ui} whose value is null in P and determines weight w_{ui} by three schemes: uniform, user-oriented, and item-oriented schemes. In this paper, we employ the user-oriented scheme, which was the best performer . The underlying principle of the user-oriented scheme is essentially that as a user rates more items, she is more likely to dislike unrated items. That is, it computes the weight w_{ui} in proportion to the number of items rated by u: $w_{ui} = P\ i\ p_{ui}$. The OCCF method finally updates $p_{ui} \in P$ using their corresponding weights. We treat the updated values as the inferred pre-use preference scores. To update these values, the OCCF method employs the weighted alternating least squares (wALS) method in building singular value decomposition (SVD) with a rating matrix and its weight matrix. It infers the preference scores for each user's unrated items via the SVD model. The wALS method decomposes a matrix P into two low-rank matrices X and Y while optimizing an objective function £(X, Y). The matrix P represents observed pre-use preferences in our case, i.e., $P = (p_{ui})m*n$. The matrices X and Y represent the latent features of users and items, respectively. The objective function is represented as follows:

$£(X, Y) = X\ u\ X\ i\ w_{ui}\{(p_{ui} - XuY^T{}_i)\ 2 + \lambda(kX_u(\cdot)k\ 2\ F + \lambda kY_i(\cdot)k^2{}_F)\}$

where p_{ui} and w_{ui} are the entries in the observed pre-use preference matrix P and its weight matrix W, respectively.

The vector X_u is the u-th row of matrix X, and the vector Yi is the i-th row of matrix Y . The two vectors represent the features of user u and item i. In addition, k·kF denotes the Frobenius norm and λ is a regularization parameter. In order to factorize matrix P, the OCCF method first assigns random values to elements in matrix Y , and updates elements in matrix X by optimizing the objective function.

IV. IDENTIFYING UNINTERESTING ITEMS

Once pre-use preferences of unrated items are computed, we can identify uninteresting items. Based on the pre-use preference scores inferred by the OCCF method, the uninteresting items of user u are defined as follows:

$I^{un}\ u\ (\theta) = \{i|\rho(\hat{}pui) \leq \theta, r_{ui} = null\}$ where $\rho(\hat{}p_{ui})$ indicates the percentile rank of $p\hat{}_{ui}$ among all user-item pairs whose ratings are missing in R. For example, I un u (20) indicates that we assign all unrated items whose percentile ranks of pre-use preference scores are at the bottom 20% as uninteresting items. We do not use an absolute cut-off value for pre-use preference scores because the OCCF method is originally designed for computing users' relative preferences. In addition, we adjust the parameter θ to obtain the best accuracy for top-N recommendation. If θ is set high, a large number of unrated entries are injected with low values, leading to a less sparse rating matrix. On the other hand, if θ is set low, we may not be fully utilizing the benefit of uninteresting items as only a small number of unrated entries are injected. The simple use of relative cut-off based on the percentile rank works well.

V. L-INJECTION

We now propose a novel method to impute missing ratings, named as l-injection, such that we assign a "low" value to $r_{ui} \in R$ if an item i is determined as uninteresting for a user u. This is because u would not be satisfied with an uninteresting item i even if recommended. By filling a rating matrix R with low values, we can build a new "denser" matrix that contains low value ratings as well as actual user ratings. We call this augmented matrix an l-injected matrix $L = (l_{ui})m*n$, where entry l_{ui} is defined as follows:

is follows:

$$l_{ui} = \begin{cases} r_{ui} & \text{if } u \text{ has rated } i; \\ v_{ui} & \text{if (1) } u \text{ has not rated } i \text{ and} \\ & \text{(2) } i \text{ is an uninteresting item to } u; \\ null & \text{otherwise} \end{cases}$$

where v_{ui} is to be defined below. We now develop various methods to impute missing ratings to uninteresting items. To determine v_{ui}, a simple way is to fill it with zero. This imputation means that a user does not like uninteresting items at all. Alternatively, because uninteresting items are generally less preferred than rated items, we can also fill a "low" value by under-estimating the average of known ratings. Note that the proposed approach works regardless of the choice of underlying CF methods as it simply replaces the original rating matrix R by the l-injected matrix L. The proposed approach is orthogonal to existing CF methods, which is one of our key strengths. It is also possible to develop other imputation methods to reflect the characteristic of uninteresting items. Because our intention is to evaluate the effectiveness of using uninteresting items for top-N recommendation, we leave more sophisticated modeling for l-injection as our future work. The proposed approach can improve existing CF methods with three aspects. First, when CF methods are applied, uninteresting items are excluded from the recommendation list. While existing CF methods consider all items whose ratings are missing as the candidates for top-N recommendation, we essentially avoid uninteresting items from top-N recommendation. That is, the l-injected matrix can prevent uninteresting items from top-N recommendation. Second, the l-injected matrix includes a higher number of ratings (including ratings with low values) for uninteresting items than the original rating matrix. The CF algorithms equipped with an l-injected matrix are able to understand users' preferences more accurately. Lastly, because the number of candidates for top-N recommendation essentially reduces, the computational cost can also decrease in top-N recommendation. We further discuss a key difference from existing work. Similar to an l-injected matrix, PureSVD also assigns zero to missing ratings. However, PureSVD has no regard for identifying uninteresting items and simply fills zero values to "all" missing ratings. That is, PureSVD simply regards all unrated items as uninteresting items. In addition, PureSVD considers the items with missing ratings as candidates for top-N recommendation. In clear contrast, the l-injection selectively fills the uninteresting items $I^{un}{}_u$ with low values, to help understand user preferences more precisely, and exclude uninteresting items from top-N recommendation. In Section 4, we demonstrate the superiority of our approach over PureSVD.

VI. WHY DOES THE L-INJECTED MATRIX HELP?

We argue that an l-injected matrix helps improve the accuracy of any CF method. To present the ground for our argument, we discuss the effect of our approach when applied to two popular CF methods: item-based collaborative filtering method (ICF) and SVD-based method (SVD). ICF predicts a rating $\hat{l}_{ui}$ for a target item i of a user u by referencing her ratings on those items similar to the item i as follows:

$$\hat{l}_{ui} = \frac{\sum_{j \in S_i} \{l_{uj} * sim(i,j)\}}{\sum_{j \in S_i} sim(i,j)}$$

where S_i is a set of (up to) k items which have most similar rating patterns to the items for which item i of user u is known. If there are less than k items evaluated by u, Si includes that number of items only instead of k. In addition, let sim(i, j) denote the similarity between items i and j in terms of users' rating patterns. In this paper, we adopt Pearson's correlation coefficient as the well-known similarity measure.

VII. RELATED WORK

In general, CF methods are categorized into two approaches: memory-based and model-based. First, memory based methods predict the ratings of a user using the similarity of her neighborhoods, and recommend the items with high ratings. Second, model-based methods build a model capturing users' ratings on items, and then predict her unknown ratings based on the learned model. Most CF methods, despite their wide adoption in practice, suffer from low accuracy if most users rate only a few items called the data sparsity problem. This is because the number of unrated items is significantly more than that of rated items. To address this problem, some existing work attempted to infer users' ratings on unrated items based on additional information such as clicks and bookmarks. However, these works require an overhead of collecting extra data, which itself may have another data sparsity problem. Compared to, our proposal does not require any extra data and solely works on account of an existing rating matrix. In addition, to improve the accuracy of top-N recommendation, other works leverage both ratings and the fact

whether a user evaluates an item or not. For instance, SVD++, builds an extended SVD model exploiting both information. The conditional restricted Boltzmann machine (RBM) and constrained probabilistic matrix factorization (PMF) also account for both information in learning their models. However, these approaches are based on a simple assumption such that a user would dislike all unrated items. On the other hand, we strive to discern a subset of unrated items that users truly dislike. Therefore our proposal yields improvements in accuracy compared to existing methods finally, several CF methods have been proposed to fill missing ratings with a particular value in order to improve the accuracy. They also simply assume that a user would dislike all unrated items. Based on this assumption, PureSVD fills all missing ratings with zeros, and then makes prediction using both known ratings and zero ratings. Steck assigns a low value to all missing ratings, and then makes recommendation by learning a multinomial mixture model. By filling all missing ratings with low values, however, this approach could mistakenly assign low values to the items that users might like, thereby affecting an overall accuracy in recommendation. Our preliminary work infers uninteresting items and builds 0-injected matrix. Because the 0-injected matrix includes the ratings inferred from uninteresting items, it can infer latent user preferences more accurately. However, because 0-injection simply considers all uninteresting items as zero, it may neglect to the characteristics of users or items. In contrast, l-injection not only maximizes the impact of filling missing ratings but also considers the characteristics of users and items, by imputing uninteresting items with low pre-use preferences.

VIII. CONCLUSIONS

In this paper, we proposed a novel approach, l-injection, for uninteresting items by using a new notion of pre-use preferences. This approach not only significantly alleviates the data sparsity problem but also effectively prevents those uninteresting items from being recommended. Because the proposed approach is method-agnostic, it can be easily applied to a wide variety of existing CF methods. Through comprehensive experiments, we successfully demonstrated that the proposed approach is effective and practical, dramatically improving the accuracies of existing CF methods (e.g., item-based CF, SVD-based CF, and SVD++) by 2.5 to 5 times. Furthermore, our approach improves the running time of those CF methods by 1.2 to 2.3 times when its setting produces the best accuracy.

REFERENCES

1. *W. Hwang, J. Parc, S. Kim, J. Lee, and D. Lee, "Told you i didn't like it: Exploiting uninteresting items for effective collaborative filtering," in Proc. of IEEE ICDE, 2016, pp. 349–360.*
2. *G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," IEEE Transactions on Knowledge and Data Engineering, vol. 17, no. 6, pp. 734–749, 2005.*
3. *Y. Koren et al., "Matrix factorization techniques for recommender systems," IEEE Computer, vol. 42, no. 8, pp. 30–37, 2009.*
4. *B. Sarwar et al., "Item-based collaboration filtering recommendation algorithms," in Proc. of IEEE WWW, 2001, pp. 285–295.*
5. *S. Zhang et al., "Using singular value decomposition approximation for collaborative filtering," in Proc. of IEEE CEC, 2005, pp. 257–264.*
6. *P. Resnick et al., "Grouplens: an open architecture for collaborative filtering of netnews," in Proc. of ACM CSCW, 1994, pp. 175–186.*
7. *P. Cremonesi et al., "Performance of recommender algorithms on top-n recommendation tasks," in Proc. of ACM RecSys, 2010, pp. 39–46.*
8. *R. Pan et al., "One-class collaborative filtering," in Proc. of IEEE ICDM, 2008, pp. 502–511.*
9. *V. Sindhwani et al., "A family of non-negative matrix factorization for one-class collaborative filtering," in Proc. of ACM RecSys, 2009.*
10. *J. Ha et al., "Top-n recommendation through belief propagation," in Proc. of ACM CIKM, 2012, pp. 2343–2346.*
11. *N. Srebro and T. Jaakkola, "Weighted low-rank approximations," in Proc. of AAAI ICML, 2003, pp. 720–727.*
12. *J. Breese et al., "Empirical analysis of predictive algorithms for collaborative filtering," in Proc. of UAI, 1998, pp. 43–52.*

13. J. Tang, H. Gao, and H. Liu, "mtrust: discerning multi-faceted trust in a connected world," in "Proc. of WSDM", pages = 93–102, year = 2012.
14. H. Steck, "Training and testing of recommender systems on data missing not at random," in Proc. of ACM KDD, 2010, pp. 713–722.
15. Z. Gantner et al., "Mymedialite: A free recommender system library," in Proc. of ACM RecSys, 2011, pp. 305–308.
16. R. Bell and Y. Koren, "Lessons from the netflix prize challenge," ACM SIGKDD Explorations Newsletter, vol. 9, no. 2, pp. 75–79, 2007.
17. Y. Koren, "Factorization meets the neighborhood: a multifaceted collaborative filtering model," in Proc. of ACM KDD, 2008, pp. 426– 434.
18. H. Steck, "Item popularity and recommendation accuracy," in Proc. of ACM RecSys, 2011, pp. 125–132.
19. H. Ma et al., "Effective missing data prediction for collaborative filtering," in Proc. of ACM SIGIR, 2007, pp. 39–46. [20] J. Lee, D. Lee, Y. Lee, W. Hwang, and S. Kim, "Improving the accuracy of top-n recommendation using a preference model," Inf. Sci., vol. 348, pp. 290–304, 2016.
20. Z. Huang et al., "Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering," ACM TOIS, vol. 22, no. 1, pp. 116–142, 2004.
21. J. Liu et al., "Personalized news recommendation based on click behavior," in Proc. of ACM IUI, 2010, pp. 31–40.
22. S. Niwa et al., "Web page recommender system based on folksonomy mining for ITNG '06 submissions," in Proc. of IEEE ITNG, 2006, pp. 383–393.