

Estimating Local Information Trustworthiness via Multi-Source Joint Matrix Factorization

Liang Ge^{*}, Jing Gao^{*}, Xiao Yu[†], Wei Fan[‡] and Aidong Zhang^{*}

^{*}The State University of New York at Buffalo

[†]University of Illinois at Urbana-Champaign

[‡]Huawei Noah Ark's Lab

{liangge, jing}@buffalo.edu, xiaoyu1@illinois.edu,
david.fanwei@huawei.com, azhang@buffalo.edu

Abstract—We investigate how to estimate information trustworthiness by considering multiple information sources jointly in a latent matrix space. We particularly focus on user review and recommendation systems, as there are multiple platforms where people can rate items and services that they have purchased, and many potential customers rely on these opinions to make decisions. Information trustworthiness is a serious problem because ratings are generated freely by end-users so that many spammers take advantage of freedom of speech to promote their business or damage reputation of competitors. We propose to simply use customer ratings to estimate each individual source's reliability by exploring correlations among multiple sources. Ratings of items are provided by users of diverse tastes and styles, and thus may appear noisy and conflicting across sources, however, they share some underlying common behavior. Therefore, we can group users based on their opinions, and a source is reliable on an item if its opinions given by latent groups are consistent across platforms. Inspired by this observation, we solve the problem by a two-step model – a joint matrix factorization procedure followed by reliability score computation. We propose two effective approaches to decompose rating matrices as the products of group membership and group rating matrices, and then compute consistency degrees from group rating matrices as source reliability scores. We conduct experiments on both synthetic data and real user ratings collected from Orbitz, Priceline and TripAdvisor on all the hotels in Las Vegas and New York City. Results show that the proposed method is able to give accurate estimates of source reliability and thus successfully identify inconsistent, conflicting and unreliable information.

I. Introduction

With proliferation of Internet applications and smartphones, it is becoming much easier to access and create contents than ever before, which leads to huge information exposure. Despite the convenience, such engulfment of information also brings one big challenge – how can we extract meaningful and reliable knowledge from such large volumes of complicated data? One of the fundamental difficulties is that freely-created information is massive in volume, but it is usually of low quality. Especially in user-generated contents, information can be highly noisy, incorrect, misleading and thus unreliable.

As we know, it is difficult to estimate reliability of a particular data source based on its information alone. Only

by comparing information from multiple sources, we are able to quantify trustworthiness and extract key reliable knowledge. In particular, we have the following two key observations for this problem. First, information sources have their "local" expertise, i.e., each source can be reliable on some particular aspects or facts but unreliable on others. Therefore, it is important to give local reliability scores to information sources based on their quality. Second, although at the surface, multiple sources may present conflicting and unaligned information, they must share some common beliefs about the items on which we try to find out the truth. The knowledge shared by multiple sources usually correspond to the truth. Therefore, we propose to unleash the power of multiple data sources by discovering the underlying shared knowledge base among sources and calculating local trustworthiness score accordingly.

Trustworthiness in Ratings. In this paper, we target at a specific application as the first attempt towards this direction. We are concerned with the information trustworthiness issue in online recommendation systems. Nowadays, more and more people leave their opinions about certain products or activities on many websites. The relative openness and widespread success of these review websites attract numerous spammers, who post fraudulent reviews or ratings. Although some efforts have been devoted to detect fake reviews or ratings [10], [11], [7], most of the online review websites are still struggling with fake or fraudulent ratings and reviews.

As different websites attract different sets of spammers, the consensus among multiple sources usually represents the truth. Therefore, most users read reviews from all these sources to get unbiased opinions. For example, before booking a hotel, many people tend to check and compare reviews on Orbitz, Priceline, TripAdvisor and many other trip planning websites. However, it is extremely tedious and time-consuming to look through all the reviews. This motivates us to develop an effective method that *automatically evaluates the reliability of information sources by correlating multiple sources to help users make better decisions*. In this paper, we focus on trustworthiness estimation in review systems by analyzing *user ratings given to a set of items collected from multiple platforms*. We show that only using user-item

rating matrices, we are able to detect inconsistencies across sources and assign a reliability score to each source.

Challenges. With the general challenges discussed above, the major difficulties are that opinions on different platforms can be quite diverse as users across platforms are different. It is impossible to simply compare ratings between individual users across multiple sources because user sets are different across websites. On the other hand, summarizing and comparing rating statistics cannot provide any useful knowledge about the information trustworthiness. As users have quite diverse preferences, it is hard to judge a single source's trustworthiness by comparing different ratings received by one item from different websites. As users with various backgrounds and preferences all contribute to the reviews, spammers' entries will be hidden deeply in each item's diverse ratings and will not be uncovered if only a summary is given. To accurately estimate source trustworthiness, we need to identify subtle inconsistencies across sources embedded in multiple rating matrices.

Summary. To address these challenges, we propose a novel two-step procedure, which estimates source reliability of each source on each item. The key idea is that although users are different across sources, they tend to form common groups where people in the same group share similar opinions. Although we cannot align individual users, we are able to connect multiple sources at the group level as underlying user groups, and the latent opinions of each group are consistent across multiple sources.

Based on this fact, we propose to first discover the latent groups from multiple rating matrices by joint matrix factorization and then calculate reliability score of each source on each item based on the degree of consistency in group behavior across multiple sources. Note that even though spammers can also form small groups, the group-level behavior identified by the proposed approach is still dominated by the majority of normal users because spammers are a lot less than normal users, and thus the latent group patterns can be used to detect inconsistent and unreliable information. We design experiments on synthetic data to reveal the insights of the proposed method under different settings. We also crawled real rating data sets: ratings on all the hotels in Las Vegas and in New York City from three travel websites: Orbitz, Priceline and TripAdvisor. We present results to show that the proposed method successfully calculates local reliability scores and identifies reliable information sources for each item.

II. Problem Formulation

In this section, we formally define the problem and present the basic framework to solve the problem. Suppose we are interested in K items. There are M sources that we can obtain ratings about the K items. The s -th source is characterized by a rating matrix V_s , which denotes ratings of K items from N_s users. Note that N_s can be different for different s . The sets of users for the M sources can

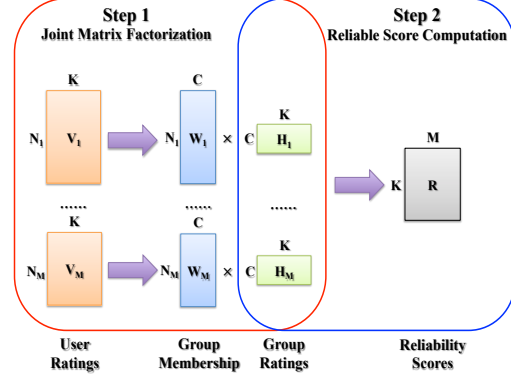


Figure 1: The Flow of the Proposed Method

be different, and thus rows in different V_s are not aligned. The goal is to derive a reliable score matrix R , where each entry r_{ks} denotes the reliable degree of the s -th source on the k -th item. The intuition is that an information source about an item is more likely to be reliable if its ratings are consistent with other sources. However, comparing ratings for individual users is impossible due to noise, sparsity and alignment issues. Therefore, we propose to partition users into groups so that people in the same group share similar rating patterns over items. Reliability scores are then computed based on the consistency degree on group ratings across sources. Specifically, we assume that users can be partitioned into C groups and V_s is the product of W_s and H_s , where

- W_s is an $N_s \times C$ matrix denoting the partition of the N_s users into C groups, where $\sum_{j=1}^C w_{s(ij)} = 1$ and $w_{s(ij)} \geq 0$. Each entry $w_{s(ij)}$ denotes whether user i is in group j .
- H_s is a $C \times K$ matrix denoting the typical ratings given by users in each group on the K items. Each entry $h_{s(jk)}$ denotes the typical rating of item k received from group j and it should be non-negative.

The first step is to calculate W_s and H_s from the rating matrix V_s for $s = 1, \dots, M$. Although we can decompose V_s separately for each source, it is more reasonable to conduct a joint matrix factorization so that groups in different sources are aligned and can later be easily compared. Therefore, we propose to calculate W_s and H_s as the solutions to the following optimization problem:

$$\min_{\{W_s, H_s\}} \sum_{s=1}^M \|V_s - W_s H_s\|^2 + \alpha \sum_{s,t=1}^M \|H_s - H_t\|^2. \quad (1)$$

The first term tries to minimize matrix factorization error of each source while the second term ensures alignment of groups across sources. Without the second term, different sources might order the groups differently in W_s and H_s , and thus H_s obtained from different sources reside in different subspaces and can not be compared. α is a predefined

Algorithm 1 BCD Algorithm for Reliability Score Computation

Input: Rating matrices from M sources: V_1, \dots, V_M , number of groups C , α ;
Output: Reliable Score Matrix R ;
1: Initialize H_1, \dots, H_M ;
2: **repeat**
3: **for** $s \leftarrow 1$ to M **do**
4: $W_s \leftarrow V_s H_s^T (H_s H_s^T)^{-1}$;
5: $H_s \leftarrow (\alpha M I + W_s^T W_s)^{-1} (W_s^T V_s + \alpha \sum_{t=1, s \neq t} H_t)$;
6: **end for**
7: **until** Convergence criterion is satisfied
8: Compute reliable score matrix R : $r_{ks} = \min_{t \neq s} \delta(\vec{h}_{s(k)}, \vec{h}_{t(k)})$;
9: **return** R

parameter to control how much alignment we would like to enforce on H_s .

After the joint matrix factorization is conducted, we derive the reliability score by computing similarity between sources on their group rating patterns. Let $\vec{h}_{s(k)}$ denote the vector that contains ratings on item k from C groups in the s -th source. Then we compute each entry in the reliability matrix R as $r_{ks} = \min_{t \neq s} \delta(\vec{h}_{s(k)}, \vec{h}_{t(k)})$. Here $\delta(\vec{x}, \vec{y})$ defines similarity measures between two vectors \vec{x} and \vec{y} . The principle is that the s -th source on the i -th item is reliable if its ratings are consistent (similar) across multiple sources and thus the reliability score is high. The flow of the proposed framework is summarized in Figure 1.

III. Methodology

In this section, we describe two methods to solve the proposed reliability estimation problem. The key component of the proposed framework is to solve the joint matrix factorization problem in Eq. (1). It can be seen that the objective function is a convex quadratic function with respect to W_s or H_s when the other variable matrices are fixed. Intuitively, we adopt a block coordinate descent method (referred to as **BCD**) to iteratively update W_s and H_s . By proper initialization, the constraints will be satisfied. As shown in the experimental results, this approach outputs meaningful reliability scores, however, it involves matrix inversion, which could be unstable for large-scale sparse rating matrices. The second approach we proposed is based on the idea of non-negative matrix factorization (referred to as **NMF**), which avoids matrix inversion computation but conducts iterative updates on each entry value. The BCD and NMF algorithms are shown in Algorithms 1 and 2, respectively. Note that I represents a $C \times C$ identity matrix in Algorithm 1. In Algorithm 2, we use A^+ and A^- to represent two non-negative components of the original matrix A such that $A = A^+ - A^-$, $A^+ \geq 0$ and $A^- \geq 0$. In the following, we give analysis on the two algorithms' convergence and time complexity.

Lemma 1: Lines 2-7 of Algorithm 1 converge to a local minimum of the optimization problem in Eq. (1).

Proof: It can be seen that lines 2-7 of Algorithm 1

Algorithm 2 NMF Algorithm for Reliability Score Computation

Input: Rating matrices from M sources: V_1, \dots, V_M , number of groups C , α ;
Output: Reliable Score Matrix R ;
1: Initialize W_1, \dots, W_M and H_1, \dots, H_M ;
2: **repeat**
3: **for** $s \leftarrow 1$ to M **do**
4: **for** $i \leftarrow 1$ to N_s ; $j \leftarrow 1$ to C **do**
5: $w_{s(ij)} \leftarrow w_{s(ij)} \sqrt{\frac{[(V_s H_s^T)^+ + (W_s H_s H_s^T)^-]_{(ij)}}{[(V_s H_s^T)^- + (W_s H_s H_s^T)^+]_{(ij)}}}$;
6: **end for**
7: **for** $j \leftarrow 1$ to C ; $k \leftarrow 1$ to K **do**
8: $h_{s(jk)} \leftarrow h_{s(jk)} \cdot \xi$ where

$$\xi = \sqrt{\frac{[(W_s^T V_s)^+ + (W_s^T W_s H_s)^- + \alpha M (H_s)^- + \alpha \sum_{t=1, t \neq s} (H_t)^+]_{(jk)}}{[(W_s^T V_s)^- + (W_s^T W_s H_s)^+ + \alpha M (H_s)^+ + \alpha \sum_{t=1, t \neq s} (H_t)^-]_{(jk)}}}$$

9: **end for**
10: **end for**
11: **until** Convergence criterion is satisfied
12: Compute reliable score matrix R : $r_{ks} = \min_{t \neq s} \delta(\vec{h}_{s(k)}, \vec{h}_{t(k)})$;
13: **return** R

follow block coordinate descent procedure. At each step, we update the values of one set of variables when the other variables are fixed. By Proposition 2.7.1 in [2], to show the proposed approach converges, we only need to show that a unique minimum is obtained at each step as follows. We first fix the value of $\{H_1, \dots, H_M\}$. Let

$$f(W_s) = -2Tr(H_s^T W_s^T V_s) + Tr(H_s^T W_s^T W_s H_s). \quad (2)$$

We observe that the Hessian matrix of $f(W_s)$ is positive definite, and we can obtain unique minimum of $f(W_s)$ by setting $\frac{\partial f}{\partial W_s} = 0$, which gives the update equation of W_s :

$$W_s = V_s H_s^T (H_s H_s^T)^{-1}. \quad (3)$$

Similarly, H_s can be updated by fixing W_s and $\{H_1, \dots, H_M\} - H_s$. Let

$$f(H_s) = -2Tr(H_s^T W_s^T V_s) + Tr(H_s^T W_s^T W_s H_s) + \alpha \sum_{t=1, t \neq s} (Tr(H_s^T H_s) - 2Tr(H_s^T H_t)). \quad (4)$$

When $\alpha > 1$, $f(H_s)$'s Hessian matrix is positive definite. By setting $\frac{\partial f}{\partial H_s} = 0$, we get the update equation of H_s which obtains unique minimum of $f(H_s)$:

$$H_s = (\alpha M I + W_s^T W_s)^{-1} (W_s^T V_s + \alpha \sum_{t=1, s \neq t} H_t). \quad (5)$$

Since we prove that each step obtains unique minimum, the proof is complete. \blacksquare

Lemma 2: Lines 2-11 of Algorithm 2 converge to a local minimum of the optimization problem in Eq. (1).

Proof: To prove this lemma, we utilize the definition of auxiliary function as defined in [3]. Suppose we try to find x to minimize $f(x)$, and $g(x, x')$ is an auxiliary function of $f(x)$ if the following conditions are satisfied: $g(x, x') \geq f(x)$ and $g(x, x) = f(x)$. Then define x at the $(t+1)$ -th step as $x^{(t+1)} \leftarrow \arg \min_x g(x, x^{(t)})$. This update rule results in

monotonic decrease of $f(x)$: $f(x^{(t+1)}) \leq g(x^{(t+1)}, x^{(t)}) \leq g(x^{(t)}, x^{(t)}) = f(x^{(t)})$.

For this problem, we define two auxiliary functions $g(W_s, W'_s)$ and $g(H_s, H'_s)$ for $f(W_s)$ in Eq. (2) and $f(H_s)$ in Eq. (4). We show that the update rules in lines 5 and 8 of Algorithm 2 minimize these two auxiliary functions respectively when the other variables are fixed. Then updating W_s and H_s will lead to monotonic decrease of the objective function and finally converges to local minimum. Due to space limit, we simply show how the update rule of W_s leads to convergence and then H_s 's convergence can be proved in a similar way. Recall that we break a matrix into two non-negative matrices: $A = A^+ - A^-$. Now we construct the auxiliary function $g(W_s, W'_s)$ for $f(W_s)$ as follows:

$$\begin{aligned} g(W_s, W'_s) = & -2 \sum_{ij} [(V_s H_s^T)^+]_{(ij)} w'_{sij} \left(1 + \log \frac{w_{sij}}{w'_{sij}} \right) \\ & + 2 \sum_{ij} [(V_s H_s^T)^-]_{(ij)} \frac{w_{sij}^2 + w'^2_{sij}}{2w'_{sij}} \\ & + \sum_{ij} [W'_s (H_s H_s^T)^+]_{(ij)} \frac{w_{sij}^2}{w'_{sij}} \\ & - \sum_{ijk} [(H_s H_s^T)^-]_{(jk)} w'_{sij} w'_{sik} \left(1 + \log \frac{w_{sij} w_{sik}}{w'_{sij} w'_{sik}} \right) \end{aligned}$$

It is easy to prove that $g(W_s, W_s) = f(W_s)$. Also, it can be proved that $g(W_s, W'_s) \geq f(W_s)$ using similar proofs in [9], [17]. Note that $g(W_s, W'_s)$ is convex with respect to W_s . Therefore, while fixing W'_s , we can obtain the minimum of $g(W_s, W'_s)$ by setting its partial derivative to be 0. Solving this gives us the update rule of W_s :

$$w_{sij} = w'_{sij} \sqrt{\frac{[(V_s H_s^T)^+ + W'_s (H_s H_s^T)^-]_{(ij)}}{[(V_s H_s^T)^- + W'_s (H_s H_s^T)^+]_{(ij)}}} \quad (6)$$

In summary, Eq. (6) gives minimum solution to $f(W_s)$'s auxiliary function $g(W_s, W'_s)$ and thus will decrease $f(W_s)$. Similarly, it can be proved that line 8 in Algorithm 2 decreases the objective function. As the objective function is lower bounded, applying the update rules iteratively leads to local minimum. ■

Time Complexity Analysis. In the proposed method, we have N , the maximum number of users among all sources; K , number of items in each source; C , number of groups in each source; and M , number of sources. Both BCD and NMF algorithms conduct iterative updates to solve for W_s and H_s . Suppose the maximum number of steps taken to converge is T , we can derive that the time taken for BCD and NMF to conduct joint matrix factorization is $O(MTC(CN + KN + CK + C^2))$ and $O(MTC(CN + KN + CK + K + N))$ respectively. The time used to compute the reliable score matrix R at the end of both algorithms is $O(KMC)$. As can be seen, both algorithms are linear in terms of maximal number of users and the number of items, yet NMF is faster. In the experiments, the number of iteration T varies from 6 to 100.

IV. Experiments on Synthetic Data

In this section, we present experimental results on synthetic data to show how the proposed algorithms perform in different scenarios.

Data Generation and Evaluation. The synthetic data are generated based on the observations that users can be partitioned into groups in terms of ratings over items. The latent groups and the ratings given by the groups should be consistent across multiple sources. Therefore, we mandate 3 hidden groups in each source. If users belong to the same group, their ratings over K items are drawn from the same distribution. p items are randomly chosen to be the items receiving unreliable information, and their ratings are randomly shuffled and noises are added to their ratings.

The output of BCD and NMF is a reliability score matrix R , where each element (k, j) denotes the reliability score of item k in source j . Since it is more interesting to present some unreliable information detected by the algorithm, we compute the **inconsistency score** for each item across multiple sources as follows: $I(k) = \sigma - \min_{s \in [1, M]} r_{ks}$ where σ is the maximal value in R . If there exist some source emanating unreliable information about an item, the item will receive a high inconsistency score.

We can simulate various situations by tuning two variables: the number of users and the number of items. We also show how the performance changes with respect to parameters C and α . We set the default settings for parameters of the algorithms and variables of the synthetic data generator as follows. There are 400 users and 60 items in total. The rating scale is 1 to 5. Also, we set $p = 3$, $C = 3$ and $\alpha = 2$. While we vary one variable or parameter, we maintain others the same as in the default setting. We present the experimental results in the following way. We partition the items into two sets: items that receive consistent or inconsistent information (referred to as **reliable** and **unreliable sets**), which we know from data generation. Then in each figure, the bar represents the average inconsistency score for each set, and the vertical line denotes the variance of the scores within each set. *The algorithm performs well if the difference of scores in the two sets is big.*

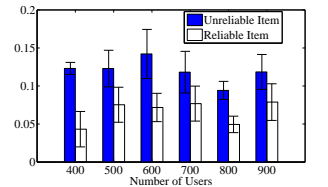
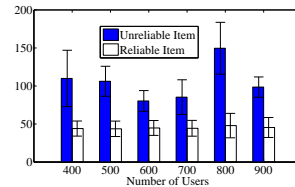


Figure 2: BCD w.r.t. #Users Figure 3: NMF w.r.t. #Users

Number of Users and Items. Figures 2 and 3 demonstrate the scores obtained from BCD and NMF with different number of users. Figures 4 and 5 show the performance of BCD and NMF on synthetic data with different number of items. Both BCD and NMF successfully distinguish the reliable items from the unreliable items. As seen from

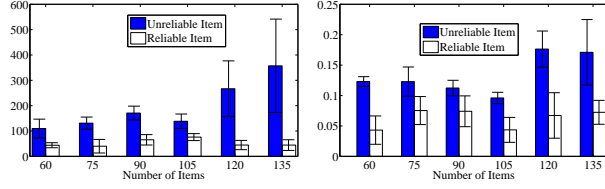


Figure 4: BCD w.r.t. #Item

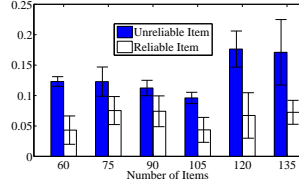


Figure 5: NMF w.r.t. #Item

Figures 2 and 3, both BCD and NMF produce separable results even when the number of users varies. Since we focus on the group behavior instead of individual behavior, the performance of BCD and NMF is not sensitive to the number of users. Figures 4 and 5 show that as the number of items increases, the performance of BCD and NMF improves in general. The improvement is due to the fact that the size of reliable sets becomes larger while the size of unreliable set remains unchanged. Therefore, the discrepancy between unreliable and reliable sets becomes larger, leading to better performance for both BCD and NMF algorithms.

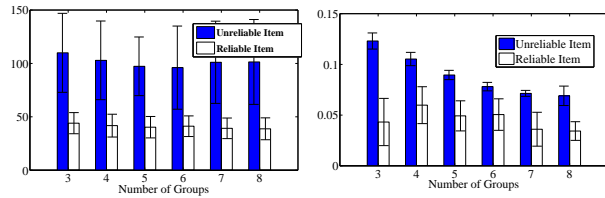


Figure 6: BCD w.r.t. C

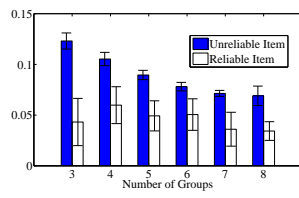


Figure 7: NMF w.r.t. C

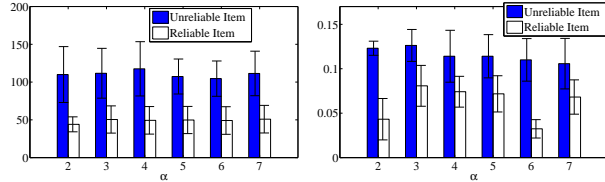


Figure 8: BCD w.r.t. α

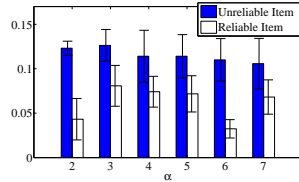


Figure 9: NMF w.r.t. α

Parameter Sensitivity. Figures 6 and 7 show the performance of BCD and NMF in terms of the parameter C . As seen from the figures, both BCD and NMF algorithms are not sensitive to the choices of C , i.e., they both produce very separable results. Figures 8 and 9 show the performance of BCD and NMF with respect to parameter α , where smaller α leads to better results as the difference between unreliable and reliable sets becomes larger.

V. Experiments on Hotel Rating Data Sets

In this section, we show how the proposed methods issue meaningful alerts on unreliable information through case studies on hotel rating data sets.

Data Sets. The two data sets are the ratings of hotels crawled from three popular travel websites: Orbitz, Priceline and TripAdvisor. We choose two popular cities: Las Vegas and New York City. The three websites have different number of hotels in the two cities and we crawl all the

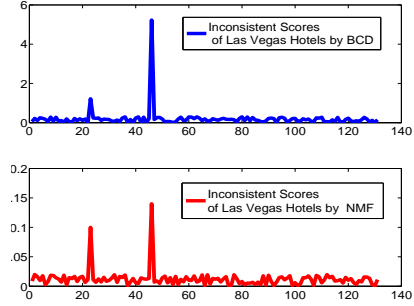


Figure 10: Inconsistent Score Distribution-Las Vegas

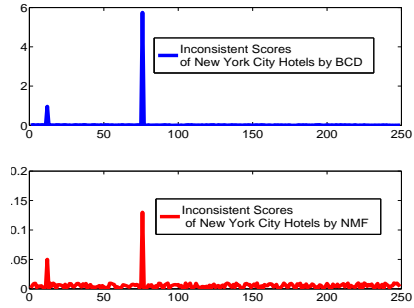


Figure 11: Inconsistent Score Distribution-NYC

ratings of the common hotels among the three websites. The data sets are crawled between March 7 and March 9, 2012. For Las Vegas, the number of users for Orbitz, Priceline and TripAdvisor are 34735, 2530 and 100037 respectively. For New York City, the number of users from Orbitz, Priceline and TripAdvisor are 10259, 3096 and 117582, respectively.

Results and Evaluation. We apply BCD and NMF on the above data sets. Figures 10 and 11 show the inconsistent score distributions of Las Vegas and New York City hotels produced by BCD and NMF, respectively. There are several observations that can be drawn from the figures. Firstly, most hotels in Las Vegas and New York city receive consistent ratings, indicating that the information about most hotels in three websites are reliable. Secondly, several hotels' ratings are significantly inconsistent across sources comparing with the other hotels. Among all hotels, two hotels, i.e., The Carriage House in Las Vegas and The Jane Hotel in New York City, receive very high inconsistency scores from both algorithms, and we investigate whether there indeed exist unreliable information in their ratings.

For The Carriage House Hotel in Las Vegas, users from Tripadvisor give incredibly high ratings. The Carriage House ranks the 12-th of all 282 hotels in Las Vegas, higher than many world-renowned hotels. Of its 379 ratings, only 24 (6.3%) give less than 2, 207 (54.6%) give 5, and 126 (33.2%) give 4. We might say that this is a nice place to stay if we only have information from TripAdvisor. However, the other sources tell a different story. In Yelp, 50% of guests gives ratings less than 3. In Priceline, the overall

rating is 4.1 and many of the guest have some complaints about this hotel. In Booking, of 233 ratings, 57 (24.4%) gives ratings less than 4. From the detail comments on this hotel from Yelp, Priceline and Booking, The Carriage House in Las Vegas shows some unattractive features and most of the negative reviews are quite consistent across multiple sources (e.g., loud, rude, dirty). As there exist inconsistencies between TripAdvisor and other sources on this hotel's ratings, the proposed method successfully detects it and can alert potential customers on the information trustworthiness of the ratings.

Similarly, for The Jane Hotel in New York City, in TripAdvisor, of 374 ratings, 253 (67.7%) give ratings more than 4. However, in Yelp, only 104 (61.5%) of 169 give ratings more than 4. Priceline gives average rating of 3.3. In Booking, out of 382 ratings, only 177 (46.3%) give more than 4. Again, we can see the inconsistencies in ratings on this hotel. We omit the details about the reviews due to space limit.

VI. Related Work

Broadly speaking, our work is related to spam detection in various applications: online auction website (e.g., Ebay) [12], social networks (e.g., Twitter) [8], and product reviews (e.g., Amazon) [10]. Most of these studies aim to detect spam and spammers based on one particular source of data. Some of them formulate the problem as a supervised learning task where manually labeled spams and normal messages are required for training. Our work is different in that we estimate the reliability of information by correlating multiple sources in an unsupervised manner.

Another relevant topic is trustworthiness analysis [18], [5], [19], [16], [1], which tries to find the truth about some questions or facts given multiple conflicting sources. Usually the truth is a weighted combination of multiple pieces of information where weights are derived for multiple sources based on their reliability. The proposed approaches differ in the factors taken into account to estimate trustworthiness (e.g., difficulty of the question [5] or two types of errors [19]), as well as the applications (e.g., Wikipedia [1] or text collections [16]). Different from these truth discovery approaches, our goal is to provide a *local* trustworthiness score for each source on each item. Furthermore, we target at identifying subtle discrepancies based on group-level truths uncovered from different sources on different sets of objects, whereas the previous truth discovery studies seek to find the truth of each object from its multiple observations given by different sources.

VII. Conclusions

We proposed to tackle the problem of source reliability estimation in user review systems by exploring the correlations among multiple review sources. We formulated the problem as a joint matrix factorization procedure where different sets of users from different sources are partitioned into

common groups and rating behavior of groups is assumed to be consistent across sources. Two effective optimization methods based on block coordinate descent and non-negative matrix factorization principles were developed to solve the joint matrix factorization problem, and we analyze their convergence and time complexity. After original rating matrices were effectively decomposed, we compute reliability score per item per source according to the degree of its group ratings being consistent with those from other sources. Experimental results on synthetic data demonstrated that the proposed method is capable of giving robust and accurate reliability estimates for rating systems. In real rating datasets collected from three popular travel planning websites on Las Vegas and New York City hotels, the proposed method found meaningful suspicious rating behavior from around 400,000,000 ratings given by more than 150,000 users.

References

- [1] B. Adler and L. de Alfaro. A Content-driven Reputation System for the Wikipedia. In *Proc. of WWW*, 2007.
- [2] D. P. Bertsekas. Non-Linear Programming. *Athena Scientific*, 2nd Edition, 1999.
- [3] C. Ding, T. Li, W. Peng, and H. Park. Orthogonal Nonnegative Matrix Tri-factorizations for Clustering. In *Proc. of KDD*, 2006.
- [4] C. Ding, T. Li, and W. Peng. On the Equivalence between Non-negative Matrix Factorization and Probabilistic Latent Semantic Indexing. *Computational Statistics Data Analysis*, 2008.
- [5] A. Galland, S. Abiteboul, A. Marian, and P. Senellart. Corroborating Information from Disagreeing Views. In *Proc. of WSDM*, 2010.
- [6] E. Gaussier and C. Goutte. Relation between PLSA and NMF and implications. In *Proc. of SIGIR*, 2005.
- [7] S. Lam and J. Riedl. Shilling Recommender Systems for Fun and Profit. In *Proc. of WWW*, 2004.
- [8] K. Lee, J. Caverlee, and S. Webb. Uncovering Social Spammers: Social Honeypots+Machine Learning. In *Proc. of SIGIR*, 2010.
- [9] D. Lee and H. Seung. Algorithms for Non-negative Matrix Factorization. In *Proc. of NIPS*, 2001.
- [10] E. P. Lim, V. A. Nguyen, N. Jindal, B. Liu, and H.W. Lauw. Detecting Product Review Spammers Using Rating Behaviors. In *Proc. of CIKM*, 2010.
- [11] B. Mehta. Unsupervised Shilling Detection for Collaborative Filtering. In *Proc. of AAAI*, 2007.
- [12] S. Pandit, D. H. Chau, S. Wang, and C. Faloutsos. Netprobe: A Fast and Scalable System for Fraud Detection in Online Auction Networks. In *Proc. of WWW*, 2007.
- [13] J. Rennie and N. Srebro. Fast Maximum Margin Matrix Factorization for Collaborative Prediction. In *Proc. of ICML*, 2005.
- [14] N. Srebro and T. Jaakkola. Weighted Low Rank Approximation. In *Proc. of ICML*, 2003.
- [15] X. Su and T. M. Khoshgoftaar. A Survey of Collaborative Filtering Techniques. *Advances in Artificial Intelligence*, 2009.
- [16] V. Vydiswaran, C. Zhai and D. Roth. Content-driven trust propagation framework. In *Proc. of KDD*, 2011.
- [17] F. Wang, T. Li, C. Zhang. Semi-supervised Clustering via Matrix Factorization. In *Proc. of SDM*, 2008.
- [18] X. Yin, J. Han, and P.S. Yu. Truth Discovery with Multiple Conflicting Information Providers on the Web. In *Proc. of KDD*, 2007.
- [19] B. Zhao, B. Rubinstein, J. Gemmel and J. Han. A Bayesian approach to discovering truth from conflicting sources for data integration. In *Proc. of VLDB Endowment*, 2012.