Unsupervised Many-to-Many Object Matching for Relational Data

Tomoharu Iwata, James Robert Lloyd, Zoubin Ghahramani

Abstract—We propose a method for unsupervised many-to-many object matching from multiple networks, which is the task of finding correspondences between groups of nodes in different networks. For example, the proposed method can discover shared word groups from multi-lingual document-word networks without cross-language alignment information. We assume that multiple networks share groups, and each group has its own interaction pattern with other groups. Using infinite relational models with this assumption, objects in different networks are clustered into common groups depending on their interaction patterns, discovering a matching. The effectiveness of the proposed method is experimentally demonstrated by using synthetic and real relational data sets, which include applications to cross-domain recommendation without shared user/item identifiers and multi-lingual word clustering.

Index Terms—Unsupervised Object Matching, Bayesian Nonparametrics, Relational Data, Stochastic Block Model, MCMC

1 Introduction

Object matching is the task of finding correspondences between objects in different domains. Examples of object matching include document alignment [1] and sentence alignment [2], [3] in natural language processing, matching images and annotations in computer vision [4], and matching user identifiers in different databases for cross domain recommendation [5]. Most object matching methods require similarity measures between objects in different domains, or correspondence data for learning the similarity measures. However, similarity measures and correspondence data may not always be available due to cost or privacy issues.

For these situations, a number of unsupervised object matching methods have been proposed recently [6], [7], [8], [9], which can find matchings without correspondence information. These methods find only one-to-one matchings between objects. However, in some applications it is appropriate to find many-to-many matching. For example, multiple English words with the same meaning (e.g. car, automobile, motorcar) might correspond to multiple German words (e.g. Wagen, Automobil). We also might want to find correspondences between groups of people instead of individuals in different social networks.

In this paper, we propose a method for finding many-to-many matchings from multiple networks, or relational data sets. We call the proposed method ReMatch (relational matching). ReMatch assumes that the given multiple networks have common latent groups, where each group exhibits a particular interaction pattern with other groups. Networks from a wide variety of fields fulfill this assumption. Let us consider lexical networks with multiple languages as an example. Each network consists of nodes of words in a language, and nodes are linked when there are relations between them. Synonym groups in a language would have the same relations with groups in another language, e.g. group {car, automobile, motorcar} is connected to {drive, ride} in English, and {Wagen, Automobil} is connected to {fahren, treiben} in German. As further examples, social networks from different research laboratories would share similar relationship patterns among faculty, post-docs and students, and biological networks from different species would have some common components. By assigning objects in different networks to common groups, we can find many-to-many matchings across networks without correspondence information, where objects assigned into the same group are considered to be matched.

We cluster objects into common latent groups using the infinite relational model (IRM) [10]. The IRM is a nonparametric Bayesian extension of the stochastic block model [11], [12], and it finds latent groups from a network without fixing the number of groups in advance. The IRM has been proposed for clustering nodes in a single network. In this paper, we assume that different networks share cluster proportions and interaction patterns between clusters. According to this assumption, nodes in multiple networks are clustered into common groups. Figure 1 shows the generative process of two networks in our model. The cluster proportions and connectivity, which defines interaction patterns between clusters, are shared by two networks. Each node is assigned to a cluster, and the link between two nodes is generated depending on their cluster assignments, and the connectivity. ReMatch does not assume that correspondence information between nodes in different networks is given in advance or even possible to obtain. The number of clusters is automatically inferred using Dirichlet process priors, and ReMatch is applicable to data sets in which the size of each cluster is different across networks, and some clusters might not appear in

• *T. Iwata is with NTT Communication science laboratories, J. Lloyd and Z. Ghahramani are with University of Cambridge*

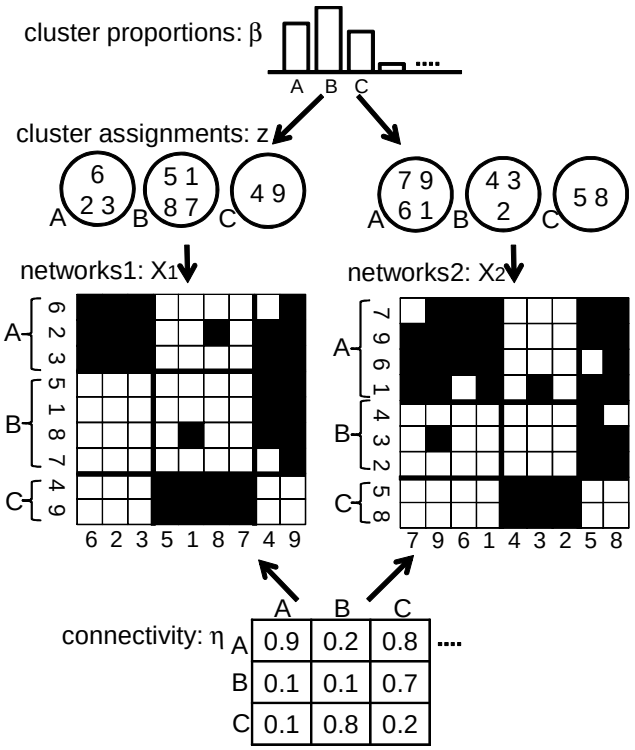


Fig. 1. Generative process of two networks with our model. The cluster proportions and connectivity are shared by both of the networks. From the cluster proportions, a cluster assignment for each node is generated. For example, nodes 6, 2 and 3 are assigned into the first cluster in Network1. The nodes assigned to the same cluster are considered matched; nodes 6, 2 and 3 in Network1 and nodes 7, 9, 6 and 1 in Network2 are matched. Using the cluster assignments and connectivity, links are generated.

some networks. ReMatch can handle multiple networks with different numbers of nodes. Since ReMatch is based on the IRM, it can handle multiple types of nodes, such as a document-word network and a user-item-tag network, as well as a single type, such as a person-person network, and multiple relations, such as 'like' and 'hate' relations. ReMatch corresponds to applying the IRM to a single large network that is constructed by combining all the networks, where connections between different networks are unobserved.

ReMatch can be used for many applications, such as cross-domain recommendation [5], [13], [14], multi-lingual corpus analysis [15], [16], and bioinformatics [17], [18], where we can expect some shared latent clusters and cannot obtain correspondence information. In cross-domain recommendation, we would for example like to recommend books to users in an online movie store. User identifiers are not shared between the stores because of the need to preserve privacy. The given data set is a user-item bipartite network for each store, which represents whether each user has purchased a particular item or not. The books and movies would share clusters, or genres; for example, users who like horror books/movies may tend to like mystery books/movies. ReMatch can find the shared groups, and perform cross-domain recommendations without common user/item identifiers. In multi-lingual corpus analysis, ReMatch can be used for discovering shared topics across languages by applying it to multi-lingual document-word networks without alignments. Most existing techniques in cross-domain recommendation and multi-lingual corpus analysis require alignment information between nodes across domains; for example, user/item identifiers are shared [13], [14], documents are aligned in polylingual topic models [19] and multi-view canonical correlation analysis [20], dictionaries are given, and morphological similarity is assumed in multilingual topic models [15]. However, user identifiers might not be shared between different companies, alignments might not be available in minor languages, and morphological similarity cannot be assumed between languages using different characters such as between English and Japanese.

The paper is organized as follows: In Section 2, we outline related work. In Section 3, we propose a method for discovering clusters shared across multiple networks without node correspondence. In Section 4, we experimentally demonstrate the effectiveness of ReMatch by using synthetic and real relational data sets, which include applications to cross-domain recommendation without shared user/item identifiers, and to multi-lingual word clustering without dictionaries/aligned-texts.

2 Related work

Unsupervised object matching methods have been proposed, such as kernelized sorting [7], least square object matching [8], matching canonical correlation analysis [6], and variational Bayesian matching [9]. These methods find one-to-one correspondences between objects in two domains, and require that the two data sets contain the same number of objects. On the other hand, the proposed method finds many-to-many correspondences, and can handle multiple domains with different numbers of nodes. Recently, an unsupervised many-to-many object matching method for real-valued data was proposed [21]. Because the method assumes Gaussian noise for input data, it is not well suited for network or relational data which are the focus of this paper.

Latent groups in a single network can be extracted by using probabilistic models, such as the stochastic block model [11], [12], mixed membership stochastic block model [22], infinite relational model [10], and network community detection methods [23]. However, these methods have not been used for discovering shared groups from multiple networks.

ReMatch can be seen as a multi-task learning method for networks. Multi-task learning techniques assumes that common properties are shared among different tasks

or domains. Most multi-task learning methods are developed for supervised learning [24], [25], [26]. Multi-task learning for clustering has been proposed [27], where relationship between clusters of different tasks is learned by using centroids of clusters. It is not applicable to network data because clusters of network data are not defined based on cluster centroids but on connectivities between clusters.

Finding correspondence between multiple networks is related to network de-anonymization [28] given multiple networks. However, this paper is different from the network de-anonymization setting since we try to find cluster-level correspondence instead of node-level correspondence.

3 Proposed method: ReMatch

In the rest of the paper, we assume that the given data are binary bipartite networks, such as user-item and document-word networks, for simplicity. However, ReMatch is applicable to other kinds of networks, such as single-type, multi-type and/or multiple-relation networks.

Suppose that we are given D networks, $\boldsymbol{X} = \{\boldsymbol{X}_d\}_{d=1}^{D}$. Here, $\boldsymbol{X}_d$ is the dth network, or relational data, represented by a $(V_{d1} \times V_{d2})$ matrix, where each element is $x_{dij} = 1$ if nodes i and j are connected, and $x_{dij} = 0$ otherwise. V_{dt} is the number of nodes of type $t \in \{1, 2\}$ in the dth network. In the case of a user-item network, a type 1 node represents a user, and a type 2 node represents an item. The task is to find a many-to-many matching of nodes in an unsupervised fashion given multiple networks.

ReMatch uses an infinite relational model (IRM) for unsupervised matching. We assume that different networks share clusters and their interaction patterns between clusters, or a connectivity matrix $\boldsymbol{\eta}$. The (k, l) element of the $\boldsymbol{\eta}$ matrix, $\eta_{k\ell} \in [0, 1]$, defines the probability of a connection between type 1 nodes belonging to cluster k and type 2 nodes belonging to cluster ℓ. Our model is nonparametric in the sense that the number of possible clusters is countably infinite, and therefore $\boldsymbol{\eta}$ is a doubly infinite matrix.

Each type t has its own common cluster proportions $\boldsymbol{\beta}_t = (\beta_{t1}, \beta_{t2}, \dots)$ that are shared across networks, where $\beta_{tk} \in [0, 1]$ represents the probability that a node is assigned to cluster k in type t, and $\sum_{k=1}^{\infty} \beta_{tk} = 1$. The infinite vector of shared cluster proportions $\boldsymbol{\beta}_t$ is generated by the stick-breaking distribution [29]. For each node i of type t in network d, a cluster assignment $z_{dti} \in \{1, 2, 3, \dots\}$ is drawn according to a discrete distribution with parameter $\boldsymbol{\beta}_t$. The existence of an edge between nodes i and j is determined by $x_{dij} \sim \text{Bernoulli}(\eta_{z_{d1i}, z_{d2j}})$ depending on their cluster assignments.

In summary, ReMatch assumes the following generative process for a set of bipartite networks $\boldsymbol{X}$:

For each cluster for type 1: $k = 1, \dots, \infty$
  For each cluster for type 2: $\ell = 1, \dots, \infty$
    Draw connectivity
    $\eta_{k\ell} \sim \text{Beta}(a, b)$
For each type: $t = 1, 2$
  Draw shared cluster proportions
  $\boldsymbol{\beta}_t \sim \text{Stick}(\gamma)$
For each network: $d = 1, \dots, D$
  For each type: $t = 1, 2$
    For each node: $i = 1, \dots, V_{dt}$
      Draw latent cluster assignment
      $z_{dti} \sim \text{Discrete}(\boldsymbol{\beta}_t)$
  For each node of type 1: $i = 1, \dots, V_{d1}$
    For each node of type 2: $j = 1, \dots, V_{d2}$
      Draw relation
      $x_{dij} \sim \text{Bernoulli}(\eta_{z_{d1i}, z_{d2j}})$

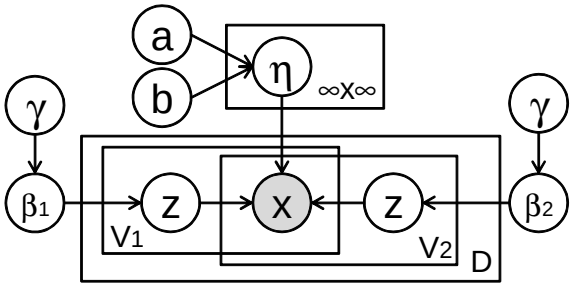


Fig. 2. Graphical model representation of ReMatch for bipartite networks.

Figure 2 shows a graphical model representation of ReMatch, where shaded and unshaded nodes indicate observed and latent variables, respectively.

Figure 3 shows the input and output of ReMatch. ReMatch takes a set of networks as input, where there is no correspondence information between nodes across the networks, and the networks can contain different numbers of nodes. In this example, the size of Network1 is (100×65) and that of Network2 is (80×105). ReMatch discovers clusters shared across the networks, which reveal the block structure when the nodes are sorted according to the cluster assignments as shown in Figure 3b. Each pair of clusters from types $t = 1$ and $t = 2$, (k, l), has its own connectivity $\eta_{k\ell}$ that is shown in the bottom of Figure 3b. The cluster sizes can be different depending on the network. Some clusters might not be used in some networks. For example, cluster $k = 5$ in type $t = 1$ is used only in Network2.

Since ReMatch corresponds to applying the IRM to a combined network as described, we can use the same inference procedure as the IRM. Given the multiple relational data $\boldsymbol{X}$, we infer latent cluster assignments by collapsed Gibbs sampling [10], in which cluster assignments $\boldsymbol{Z} = \{z_{dti}\}$ are sampled while cluster proportions $\{\boldsymbol{\beta}_t\}$ and the connectivity parameters $\boldsymbol{\eta}$ are analytically integrated out.

Given the current state of all but one latent cluster assignment z_j, where $j = (d, t, i)$, a new value for z_j is

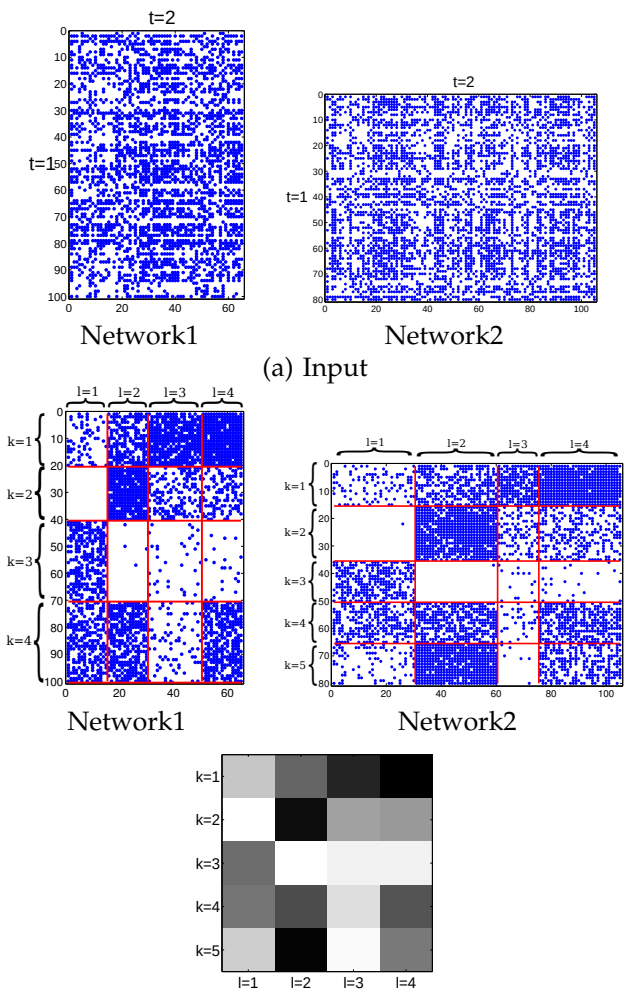


Fig. 3. (a) Input networks. In this case, the inputs are two bipartite networks, where type $t = 1$ nodes are aligned vertically, and type $t = 2$ nodes are aligned horizontally. (b) Outputs of ReMatch: (top) cluster assignments, where the nodes are sorted according to the assignments, and (bottom) the inferred connectivity matrix $\boldsymbol{\eta}$.

sampled from the following probability distribution:

$$p(z_j = k|\boldsymbol{X}, \boldsymbol{Z}^{\setminus j}) \propto \begin{cases} M_{tk}^{\setminus j} \cdot p(\boldsymbol{X}|\boldsymbol{Z}^{\setminus j}, z_j = k) \\ \quad \text{for an existing cluster} \\ \gamma \cdot p(\boldsymbol{X}|\boldsymbol{Z}^{\setminus j}, z_j = k) \\ \quad \text{for a new cluster} \end{cases} \quad (1)$$

where M_{tk} is the number of nodes that are assigned to cluster k in type t, and $\setminus j$ represents the set or value when excluding sample j. The likelihood term is calculated by

$$p(\boldsymbol{X}|\boldsymbol{Z}^{\setminus j}, z_j = k) = \prod_{\ell=1}^{K_{\bar{t}}} \frac{B(N_{k\ell}^{+j} + a, \bar{N}_{k\ell}^{+j} + b)}{B(N_{k\ell}^{\setminus j} + a, \bar{N}_{k\ell}^{\setminus j} + b)}, \quad (2)$$

where $B(\cdot)$ is the beta function, $\bar{t} = 2$ if $t = 1$ and $\bar{t} = 1$ if $t = 2$, K_t is the number of existing clusters, $N_{k\ell}$ is the number of links between clusters k and ℓ, $\bar{N}_{k\ell}$ is the number of non-links between clusters k and ℓ, and $+j$ represents the value when sample j is assigned to cluster k. In the experiments, we fixed the hyperparameters as follows: $\gamma = 1$, $a = 1$, and $b = 1$. We used the last sample of the cluster assignments in the inference for matching.

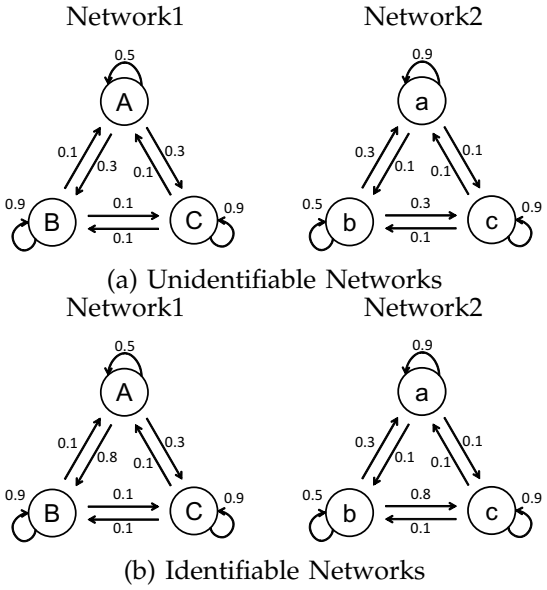


Fig. 4. Cluster connectivities of (a) matching unidentifiable networks and (b) matching identifiable networks. Each node represents a cluster, and the value at the edge represents its connectivity.

Identifiability of matching clusters depends on connectivities between clusters. Suppose that we know the true clusters and true their connectivities for each network. Then, matching is identifiable if the connectivities are matched only with the true matched clusters and they are not matched with other clusters. For example, we have two networks with cluster connectivities as in Figure 4(a). Since two possibilities of matching exist: (A-b, B-a, C-c) and (A-b, B-c, C-a), their matching is unidentifiable. On the other hand, when we have networks with connectivities as in Figure 4(b), only one matching exists (A-b, B-c, C-a), and therefore it is identifiable. Note that even if it is identifiable, the inference might not find the true matching when it falls into a local optimum solution. Since true clusters and connectivities are unknown, we need to infer clusters and connectivities. Therefore, matching performance depends also on how well the true clusters and connectivities can be inferred.

4 Experiments

4.1 Synthetic data

We evaluated ReMatch by using the following three types of synthetic data sets with two networks: Balance, Partial and Dirichlet. Figure 5 shows examples of the synthetic data sets. For the Balance data, the number of nodes in each cluster is constant. In particular, each cluster has 20 nodes and there are four

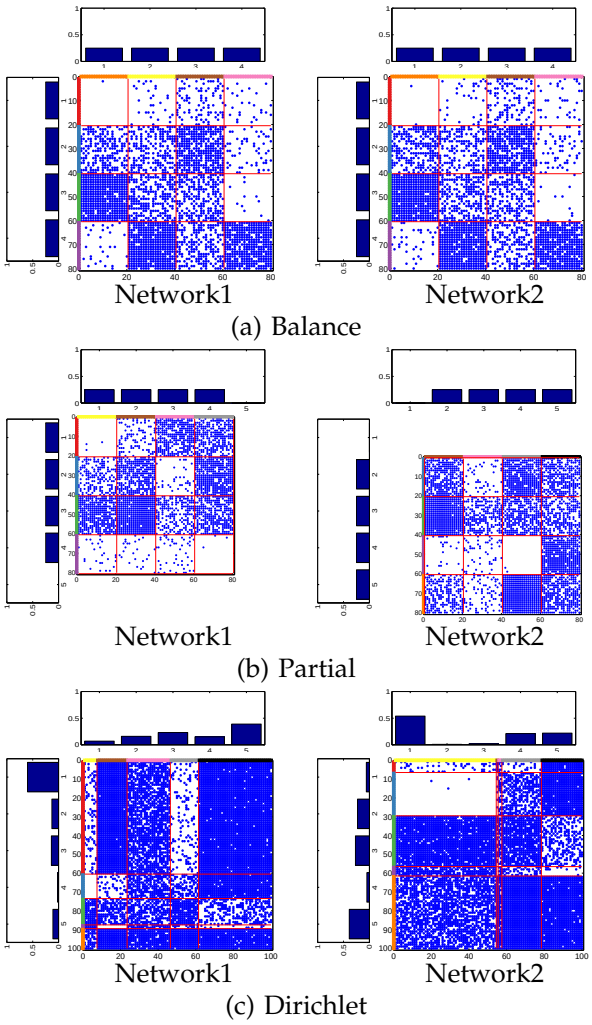


Fig. 5. Examples of synthetic data sets. Nodes are aligned by their cluster assignments. The bar chart shows the cluster proportions of the type in the network.

clusters in each network. For the Partial data, nodes are partially matched. The first cluster is used only in the first network, and the fifth cluster is used only in the second network. The second, third and fourth clusters are shared by both of the networks. For the Dirichlet data, the cluster proportions are generated from a Dirichlet distribution. Therefore, the numbers of nodes are different across clusters. The total number of nodes is 100, and the number of clusters is five. In all of the data sets, the connectivity between clusters was randomly sampled from a Beta$(0.5, 0.5)$ distribution, and the links were generated according to the Bernoulli distribution depending on the assigned clusters.

We evaluated the performance of finding correspondences between groups of nodes in different networks using the matching adjusted Rand index. The adjusted Rand index [30] is used for the evaluation of clustering performance, which quantifies the similarity between inferred clusters and true clusters, and takes the value from -1 to 1, and gives 0 for random clustering. We modified the adjusted Rand index for the many-to-many matching task, where nodes in different networks should be correctly assigned to the same cluster. The matching adjusted Rand index is calculated by

$$MARI = \frac{c_1 + c_2 - \mu}{N_1 N_2 - \mu}, \tag{3}$$

where c_1 (c_2) is the number of node pairs in different networks that are correctly assigned into the same cluster (different clusters) both in the estimated and true assignments, N_d is the number of nodes in network d, and μ is the expected value of $c_1 + c_2$, which is obtained by

$$\mu = \frac{(c_1 + c_3)(c_1 + c_4) + (c_2 + c_3)(c_2 + c_4)}{N_1 N_2}, \tag{4}$$

where c_3 (c_4) is the number of node pairs in different networks that are incorrectly assigned into the same cluster (different clusters) in the estimated assignments but that are assigned into different clusters (the same cluster) in the true assignments.

For the comparison methods, we used IRM+KS, KS and MMLVM explained below. For the IRM+KS, first we discovered clusters by the infinite relational model (IRM) for each network individually, and then found the correspondence between clusters in two networks by using convex kernelized sorting (KS) [31], which is an unsupervised object matching method. It requires that two networks have the same number of clusters. Therefore, we set the number of clusters for the IRM by using the inferred number of clusters by ReMatch, and the IRM is inferred while fixing the number of clusters. Note that the IRM with a fixed number of clusters corresponds to the stochastic block model [11], [12]. The KS method directly finds correspondence between nodes using convex kernelized sorting, where the accuracy is calculated by assuming each node is assigned to a different cluster. The MMLVM is a many-to-many latent variable model [21], which can find a many-to-many object matching given real-valued data. We ran the MMLVM with latent dimensionality $\{1, 2, \cdots, 10\}$, and show the best value.

The average matching adjusted Rand index over 30 experiments is shown in Table 1. ReMatch achieved the highest matching adjusted Rand index for all of the data sets. When IRMs are inferred individually for each network, the discovered clusters can be different among the networks because of local optima or noise in the data. Therefore, even if we try to find correspondences between clusters after the individual IRM inference, there would not be corresponding clusters, and the accuracies by IRM+KS were low. On the other hand, ReMatch alleviates this problem by finding shared clusters simultaneously in multiple networks. Because KS finds only a one-to-one matching, the performance for matching groups becomes low. The MMLVM is better than IRM+KS and KS because the MMLVM simultaneously finds clusters and their matching. However, since the MMLVM assumes real-valued data with Gaussian noise, it is not well suited for binary data, and the

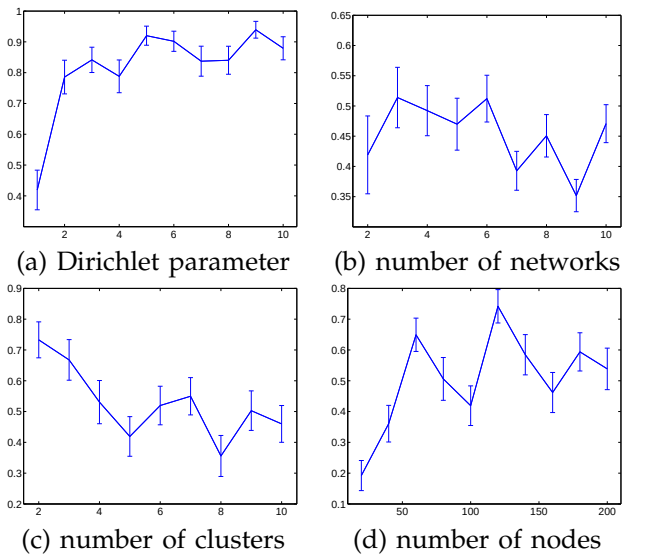


Fig. 6. Average matching adjusted Rand index with (a) different Dirichlet parameters, (b) different numbers of networks, (c) different numbers of clusters and (d) different numbers of nodes for each cluster with the synthetic Dirichlet data sets.

performance of the MMLVM with the best latent dimensionality setting is lower than that of ReMatch that can handle binary data.

Figure 6 shows the matching adjusted Rand index with different parameter settings with Dirichlet data sets. The default setting of the Dirichlet parameter is one, the numbers of networks, clusters and nodes are two, five and 100, respectively. The Dirichlet parameter controls the variance of the number of nodes in each cluster. ReMatch achieved high performance when the Dirichlet parameter was high, or when the distribution of cluster sizes was uniform (a). When the Dirichlet parameter is low, clusters with a small number of nodes are likely to be generated. It is difficult to discover such small clusters because of data scarcity, and also difficult to find matchings. Even when the number of networks was high, the performance was high (b). Since ReMatch find correspondence between clusters, the number of possible correspondences is high when there are many clusters. Therefore, as the number of clusters was increased, the matching adjusted Rand index became low (c). The performance was high when the number of nodes for each cluster was high (d). This is because we can accurately identify the interaction patterns between clusters when their are many nodes per cluster.

Figure 7 shows the cluster assignments over iterations in the inference with the Balance data. Here, node indices are aligned by their true cluster assignments; for example, indices 1 to 20 belongs to cluster 1 for both networks. We started with three clusters, where nodes were assigned randomly to one of the clusters. At the 1st iteration, many nodes were not clustered. At the 2nd iteration, some nodes were clustered and matched

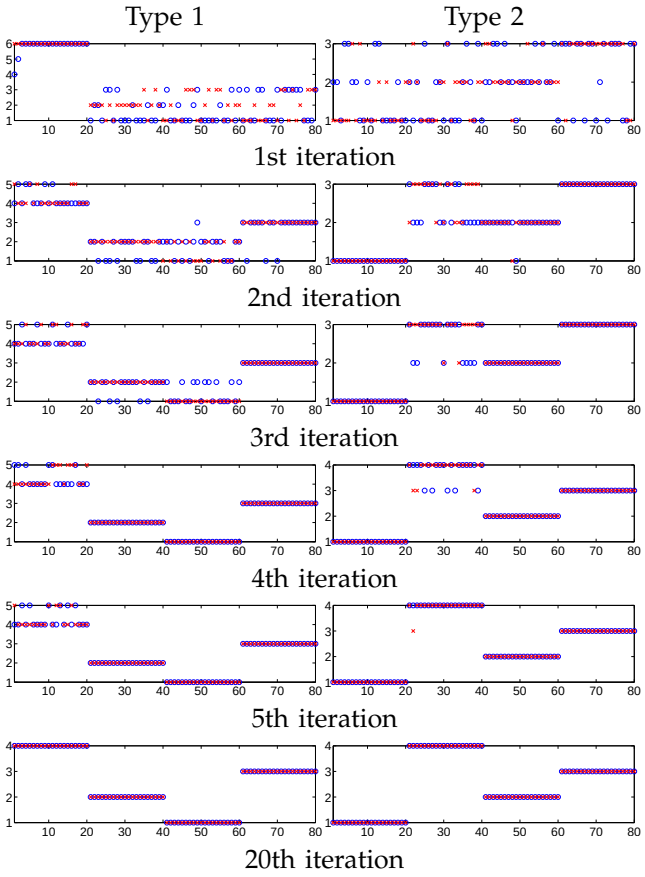


Fig. 7. Cluster assignments over iterations in the inference with a Balance data set. The horizontal axis is the node index, and the vertical axis is the cluster index. Each blue 'o' represents a node in the first network, and each red 'x' represents a node in the second network. Therefore, when 'o' and 'x' are overlapped, nodes from different networks are correctly assigned into the same cluster.

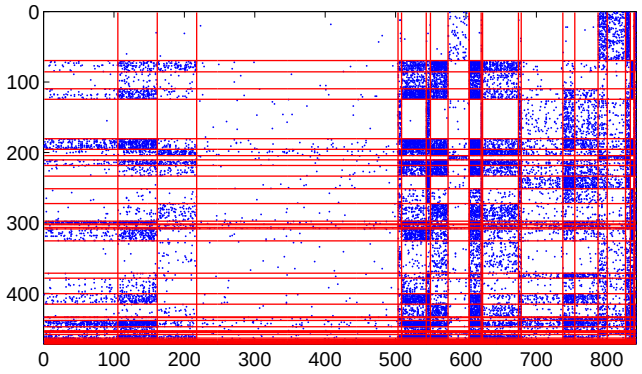
correctly, such as the node 60 to 80 in type 2. At the 5th iteration, nodes except for 1 to 10 in type 1 are correctly matched. At the 20th iteration, all of the nodes were perfectly matched. ReMatch finds clear clusters that are easy to be matched at early stage, and then, it finds other clusters by flexibly changing the number of clusters based on Dirichlet process priors. By simultaneously finding clusters and their matching, we can discourage being trapped into different local optima across multiple networks.

4.2 Cross-domain recommendation

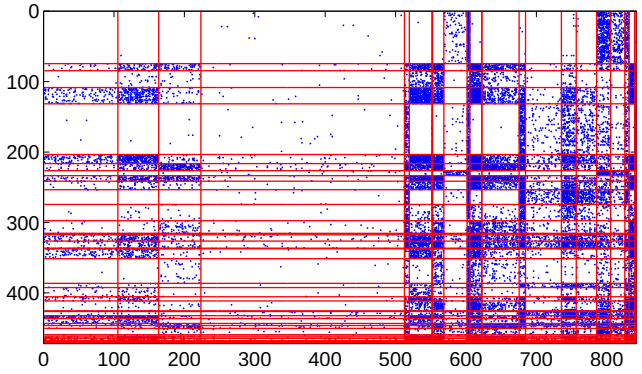
For evaluating ReMatch in a cross-domain recommendation setting, we used Movie data. The Movie data consist of two user-movie networks obtained from MovieLens [32], which is a standard benchmark data set for collaborative filtering. The MovieLens data contained 943 users, 1,682 movies, and 100,000 ratings. First, we split users and movies into two sets. Then, the first (second) user-movie network is constructed by users and

TABLE 1
Average matching adjusted Rand index, and its standard error for the synthetic data sets. Values in bold typeface are the best, or are not statistically different (at the 5% level) from the best as indicated by a paired t-test.

	ReMatch	IRM+KS	KS	MMLVM
Balance	$\mathbf{0.941} \pm 0.012$	0.344 ± 0.069	0.067 ± 0.001	0.736 ± 0.039
Partial	$\mathbf{0.728} \pm 0.050$	0.057 ± 0.040	-0.001 ± 0.002	0.262 ± 0.043
Dirichlet	$\mathbf{0.419} \pm 0.064$	0.065 ± 0.040	0.003 ± 0.001	0.156 ± 0.045



(a) Network1



(b) Network2

Fig. 8. Shared latent group structures discovered by ReMatch in the Movie data. The vertical axis represents the user index, and the horizontal axis represents the movie index.

movies in the first (second) set, where $x_{dij} = 1$ if user i has rated movie j in network d, and $x_{dij} = 0$ otherwise. The two networks do not share any users and movies. Note that collaborative filtering methods would not work for cross-domain recommendation because there are no shared users and items. Cross-domain recommendation methods have been proposed [14], [33]. However, since they assume a Gaussian noise model they are unlikely to perform well on binary data; we empirically demonstrate the poor performance of a similar model with Gaussian noise, MMLVM [21], in Section 4.1. Figure 8 shows the latent group structures inferred by ReMatch, where similar structures were discovered in two different networks.

We evaluated ReMatch in terms of cross-domain recommendation performance. With ReMatch, the probability that user i in network d purchases item j in network d' is calculated by $\eta_{z_{d1i} z_{d'2j}}$. We predicted links in the other network for hidden corresponding users using ReMatch, IRM+KS, User-average, Item-average and Average methods. The User-average method predicts links by the average connectivity of the user, and Item-average method predicts by the average connectivity of the item in the other network. Table 2 shows the results averaged over 30 experiments. For the evaluation measurements, we used the test likelihood, AUC (area under the ROC curve), and accuracy. With all of the measurements, ReMatch achieved the best predictive performance. We did not compare with KS or MMLVM because they require a huge amount of computational time. When using a computer with 2.93GHz CPU, the computational time of ReMatch, KS and MMLVM were 20 minutes, 4 days, and 8 days, respectively, for a single experiment. The computational complexity of KS is cubic in the number of nodes because a linear assignment problem solver is required. Since MMLVM does not assume relational data, it found too many clusters (over 200 clusters) with the Movie data, and thus took a long time to compute.

4.3 Co-clustering words and documents

We used the 20News data to evaluate ReMatch on co-clustering words and documents with multiple text data sets. The 20News data are generated from 20 Newsgroups data set [34] with binary occurrence data for 100 words across 16,242 documents[1]. The documents are categorized into the following four newsgroups: 'computers', 'recreation', 'science' and 'talk'. We randomly sampled two disjoint sets of 1,000 documents from the data with 250 documents from each category, and created two word-document networks.

Figure 9 shows the shared group structure discovered by ReMatch, where the words and documents in the two networks were co-clustered with similar patterns. Table 4 shows the word clustering result by ReMatch, where it discovered 10 word clusters. Even though we did not use the correspondence information between words in different networks, most words were assigned into the same cluster. Figure 10 shows the document clustering result by ReMatch, where it discovered eight document clusters. Each cluster exhibits similar proportions of document categories across the two networks. Some clusters correspond to a particular category, for example most of the documents in cluster $\ell = 3$ are categorized into 'computers'. Table 3 shows the matching adjusted Rand

1. available at http://www.cs.nyu.edu/~roweis/data.html

TABLE 2
Cross-domain recommendation results using the Movie data.

	ReMatch	IRM+KS	User-average	Item-average	Average
Likelihood	$\mathbf{-0.141} \pm 0.000$	-0.189 ± 0.003	-0.191 ± 0.000	-0.211 ± 0.000	-0.235 ± 0.000
AUC	$\mathbf{0.926} \pm 0.000$	0.855 ± 0.005	0.832 ± 0.000	0.748 ± 0.000	0.500 ± 0.000
Accuracy	$\mathbf{0.946} \pm 0.000$	0.931 ± 0.001	0.937 ± 0.000	0.937 ± 0.000	0.937 ± 0.000

TABLE 3
Average matching adjusted Rand index and their standard error for the 20News data.

ReMatch	IRM+KS
$\mathbf{0.082} \pm 0.005$	-0.002 ± 0.003

TABLE 6
Average matching adjusted Rand index and their standard error for the Wikipedia data.

ReMatch	IRM+KS
$\mathbf{0.118} \pm 0.021$	-0.004 ± 0.012

index for the task of matching document categories. ReMatch achieved higher performance than IRM+KS.

4.4 Multi-lingual word clustering

We applied ReMatch to multi-lingual word clustering. The Wikipedia data consists of English and German Wikipedia documents in the following five categories: 'Nobel laureates in Physics', 'Nobel laureates in Chemistry', 'American basketball players', 'American composers' and 'English footballers'. For each category, we sampled 50 documents that appear in both English and German Wikipedia. We used 1,000 frequent words after removing stop-words for each language. There were 150 nodes (documents) for type 1, 1,000 nodes (words) for type 2, and 13,892 relations (word occurrences) on average for each language document-word network.

Figure 11 shows the latent groups discovered by ReMatch, which have different structures in English and German. However, some clusters are shared as shown in Table 5, which contains some examples of the shared word clusters. For example, the first cluster is 'nobel prize', the second is 'basketball', and the third is 'football'. Note that even though some words appear in both English and German, we did not use the correspondence nor morphological similarity information for the inference. ReMatch discovered shared word clusters from English and German documents without any correspondence in documents and words. Table 6 shows the matching adjusted Rand index for the task of matching document categories. ReMatch achieved higher performance than IRM+KS. Figure 12 shows the document clustering result by ReMatch on the Wikipedia data. Each cluster exhibits similar proportions of document categories across English and German networks. For example, in both the English and German networks, all of the documents in cluster $\ell = 1$ were categorized in 'American composers', and all of the documents in cluster $\ell = 3$ were categorized in 'English footballer'.

5 Conclusion

We have proposed a method for unsupervised many-to-many object matching for relational data, which discovers shared latent groups from multiple networks without node correspondence. We have experimentally shown the effectiveness of the proposed method on unsupervised discovery of cluster correspondence, cross-domain recommendation, multi-task co-clustering of multiple document-word networks, and multi-lingual word clustering. In this paper, we exploited a common property among multiple networks of shared latent groups. For future work, we would like to investigate to use other common properties such as heavy-tailed degree distributions and the small world property. A simple extension of the proposed approach is to use other distributions for observation, such as Gaussian and Poisson distributions, which would enable us to handle continuous and count data. The IRM has been extended in many directions, such as the latent feature network model [35] that allows each node to be assigned to multiple clusters, and the dynamic IRM [36] that discovers clusters from time-varying network data. Also, models for finding structured clusters have been proposed, such as trees [37] and multi-way clustering [38]. We would like to apply the framework proposed in this paper to these models for discovering richer latent structures.

References

[1] T. Vu, A. T. Aw, and M. Zhang, "Feature-based method for document alignment in comparable news corpora," in *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 2009, pp. 843–851.

[2] W. A. Gale and K. W. Church, "A program for aligning sentences in bilingual corpora," in *Proceedings of the 29th annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 1991, pp. 177–184.

[3] R. Rapp, "Automatic identification of word translations from unrelated english and german corpora," in *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*. Association for Computational Linguistics, 1999, pp. 519–526.

[4] R. Socher and L. Fei-Fei, "Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR, 2010, pp. 966–973.

[5] B. Li, Q. Yang, and X. Xue, "Transfer learning for collaborative filtering via a rating-matrix generative model," in *Proceedings of the 26th Annual International Conference on Machine Learning*, ser. ICML '09, 2009, pp. 617–624.

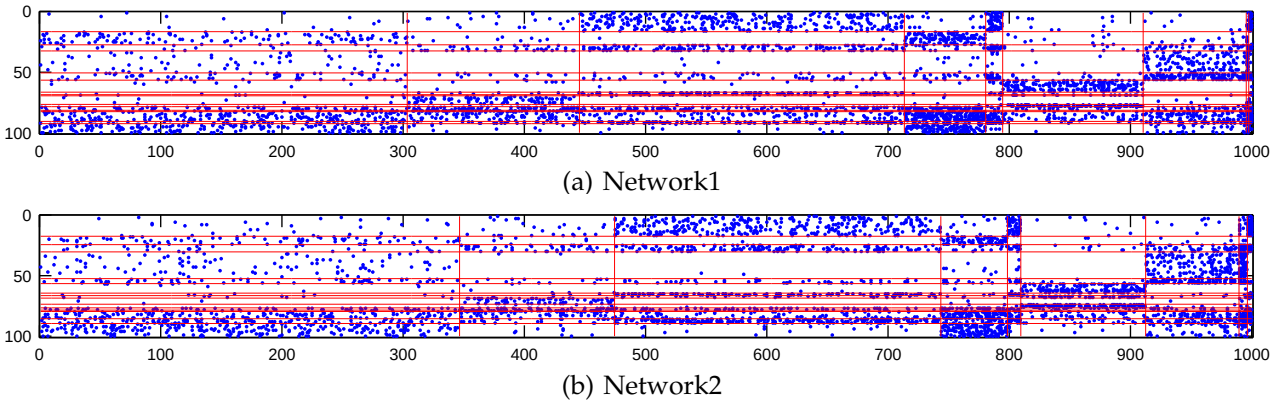


Fig. 9. Shared latent group structures discovered by ReMatch in the 20News data. The vertical axis represents the word index, and the horizontal axis represents the document index.

TABLE 4
Word clustering by ReMatch with the 20News data. Each pair of rows corresponds to a cluster, and the top/bottom row shows words from Network1/Network2, respectively.

Network	Words
1	version files mac dos graphics memory driver video ftp image pc display disk
2	version memory card files dos graphics mac display disk driver ftp image pc
1	evidence jesus war rights bible religion health jews food president israel
2	children war gun religion bible jesus israel
1	computer program software drive phone
2	program computer drive data phone technology
1	moon shuttle medicine mission doctor orbit launch studies solar cancer lunar
2	health medicine orbit launch president disease moon mission patients
1	space research data nasa science technology
2	space research science nasa
1	hockey season baseball hit players league nhl fans won puck
2	games players league won fans hockey nhl puck
1	engine oil insurance water dealer bmw honda
2	engine honda bmw oil dealer
1	team games
2	team baseball season
1	help problem question
2	problem question
1	university world course fact case number state power
2	course world power case fact number
1	email system
2	email help system university
1	god government christian human law earth gun children
2	god state government human christian evidence law rights earth jews water

[6] A. Haghighi, P. Liang, T. Berg-Kirkpatrick, and D. Klein, "Learning bilingual lexicons from monolingual corpora," in *Proceedings of ACL-08: HLT*, 2008, pp. 771–779.
[7] N. Quadrianto, A. J. Smola, L. Song, and T. Tuytelaars, "Kernelized sorting," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 32, no. 10, pp. 1809–1821, 2010.
[8] M. Yamada and M. Sugiyama, "Cross-domain object matching with model selection," in *In Proceedings of Fourteenth International Conference on Articial Intelligence and Statistics*, ser. AISTATS '11, 2011, pp. 807–815.
[9] A. Klami, "Variational Bayesian matching," in *Proceedings of Asian Conference on Machine Learning*, 2012, pp. 205–220.
[10] C. Kemp, J. Tenenbaum, T. Griffiths, T. Yamada, and N. Ueda, "Learning systems of concepts with an infinite relational model," in *AAAI Conference on Artificial Intelligence*, vol. 21, 2006, p. 381.
[11] Y. Wang and G. Wong, "Stochastic blockmodels for directed graphs," *Journal of the American Statistical Association*, vol. 82, no. 397, pp. 8–19, 1987.
[12] K. Nowicki and T. Snijders, "Estimation and prediction for stochastic blockstructures," *Journal of the American Statistical Association*, vol. 96, no. 455, pp. 1077–1087, 2001.
[13] W. Pan, E. Xiang, N. Liu, and Q. Yang, "Transfer learning in collaborative filtering for sparsity reduction," in *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, 2010, pp. 230–235.
[14] Y. Zhang, B. Cao, and D.-Y. Yeung, "Multi-domain collaborative filtering," in *Proceedings of the 26th Conference on Unvertainty in Artificial Intelligence*, ser. UAI. AUAI Press, 2010, pp. 725–731.
[15] J. Boyd-Graber and D. Blei, "Multilingual topic models for unaligned text," in *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 2009, pp. 75–82.
[16] T. Iwata, D. Mochihashi, and H. Sawada, "Learning common grammar from multilingual corpus," in *Proceedings of the ACL 2010 Conference Short Papers*. Association for Computational Linguistics, 2010, pp. 184–188.
[17] C. Wang and S. Mahadevan, "Manifold alignment without correspondence," in *Proceedings of the 21st International Jont Conference on Artifical Intelligence*, ser. IJCAI'09, 2009, pp. 1273–1278.
[18] P. Kirk, J. Griffin, R. Savage, Z. Ghahramani, and D. Wild, "Bayesian correlated clustering to integrate multiple datasets," *Bioinformatics*, 2012.
[19] D. Mimno, H. Wallach, J. Naradowsky, D. Smith, and A. McCallum, "Polylingual topic models," in *Proceedings of the 2009*

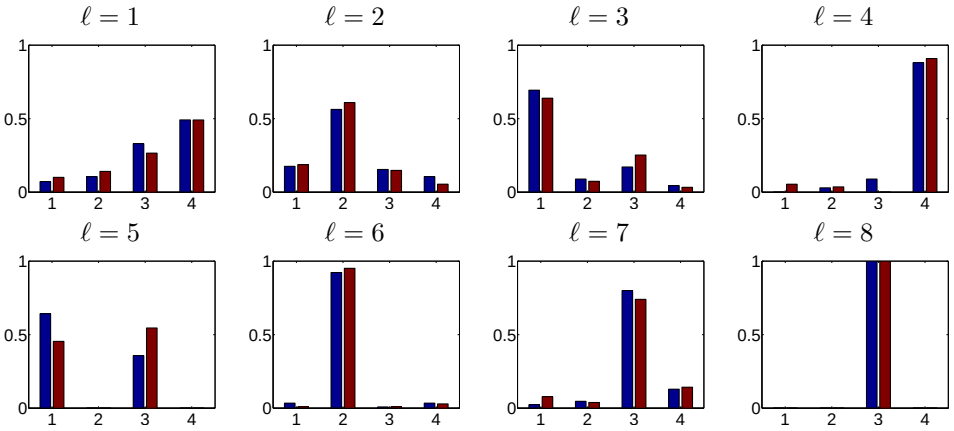
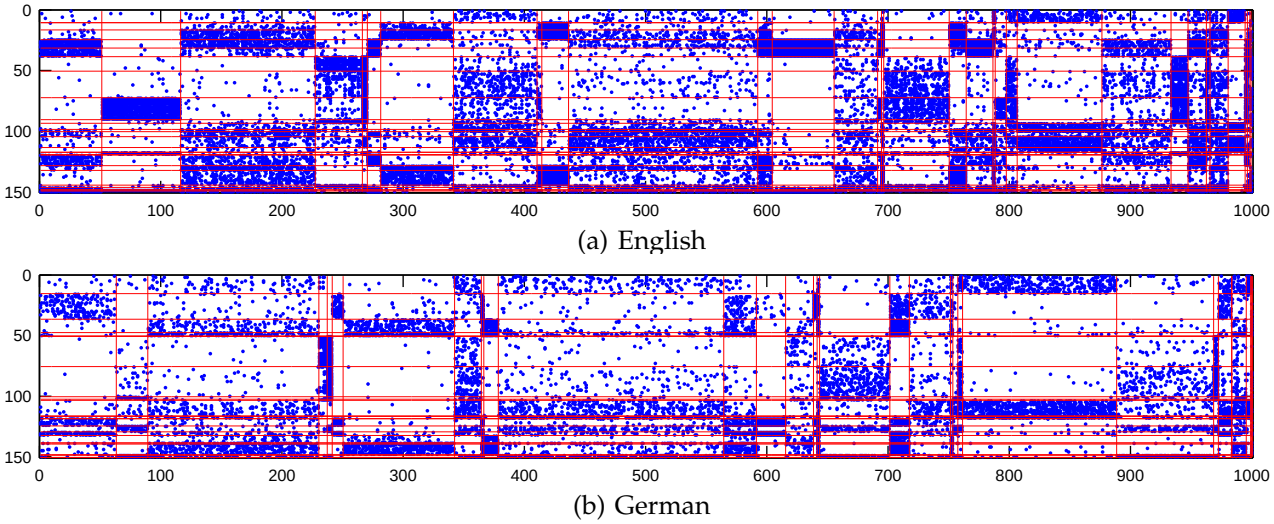


Fig. 10. Document clustering by ReMatch with the 20News data. The x-axis in each figure shows the category index: (1) computers, (2) recreation, (3) science, and (4) talk. The y-axis in each figure of cluster ℓ shows the probability that documents assigned to cluster ℓ are labeled with the category. The left blue bar shows the probability in the first network, and the right red bar shows that in the second network.



(a) English

(b) German

Fig. 11. Shared latent group structures discovered by ReMatch in the Wikipedia data. The vertical axis represents the document index, and the horizontal axis represents the word index.

Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2. Association for Computational Linguistics, 2009, pp. 880–889.

[20] J. Rupnik and J. Shawe-Taylor, "Multi-view canonical correlation analysis," in *Conference on Data Mining and Data Warehouses (SiKDD 2010)*, 2010.

[21] T. Iwata, T. Hirao, and N. Ueda, "Unsupervised cluster matching via probabilistic latent variable models," in *AAAI '13: Proceedings of the 27th AAAI Conference on Artificial Intelligence*, 2013.

[22] E. Airoldi, D. Blei, S. Fienberg, and E. Xing, "Mixed membership stochastic blockmodels," *The Journal of Machine Learning Research*, vol. 9, pp. 1981–2014, 2008.

[23] J. Leskovec, K. J. Lang, and M. Mahoney, "Empirical comparison of algorithms for network community detection," in *Proceedings of the 19th international conference on World wide web*, ser. WWW '10. New York, NY, USA: ACM, 2010, pp. 631–640.

[24] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[25] R. Ando and T. Zhang, "A framework for learning predictive structures from multiple tasks and unlabeled data," *The Journal of Machine Learning Research*, vol. 6, pp. 1817–1853, 2005.

[26] T. Evgeniou, C. Micchelli, and M. Pontil, "Learning multiple tasks with kernel methods," *Journal of Machine Learning Research*, vol. 6, no. 1, p. 615, 2006.

[27] J. Zhang and C. Zhang, "Multitask Bregman clustering," *Neurocomputing*, vol. 74, no. 10, pp. 1720–1734, 2011.

[28] M. Hay, G. Miklau, D. Jensen, D. Towsley, and P. Weis, "Resisting structural re-identification in anonymized social networks," *Proceedings of the VLDB Endowment*, vol. 1, no. 1, pp. 102–114, 2008.

[29] J. Sethuraman, "A constructive definition of Dirichlet priors," *Statistica Sinica*, vol. 4, pp. 639–650, 1994.

[30] L. Hubert and P. Arabie, "Comparing partitions," *Journal of Classification*, vol. 2, no. 1, pp. 193–218, 1985.

[31] N. Djuric, M. Grbovic, and S. Vucetic, "Convex kernelized sorting," in *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

[32] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl, "An algorithmic framework for performing collaborative filtering," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1999, pp. 230–237.

[33] S. Gao, H. Luo, D. Chen, S. Li, P. Gallinari, and J. Guo, "Cross-domain recommendation via cluster-level latent factor model," in *Machine Learning and Knowledge Discovery in Databases, Lecture Notes in Computer Science*, 2013, pp. 161–176.

[34] K. Lang, "Newsweeder: Learning to filter netnews," in *Proceedings of the Twelfth International Conference on Machine Learning*, 1995, pp. 331–339.

[35] K. Miller, T. Griffiths, and M. Jordan, "Nonparametric latent

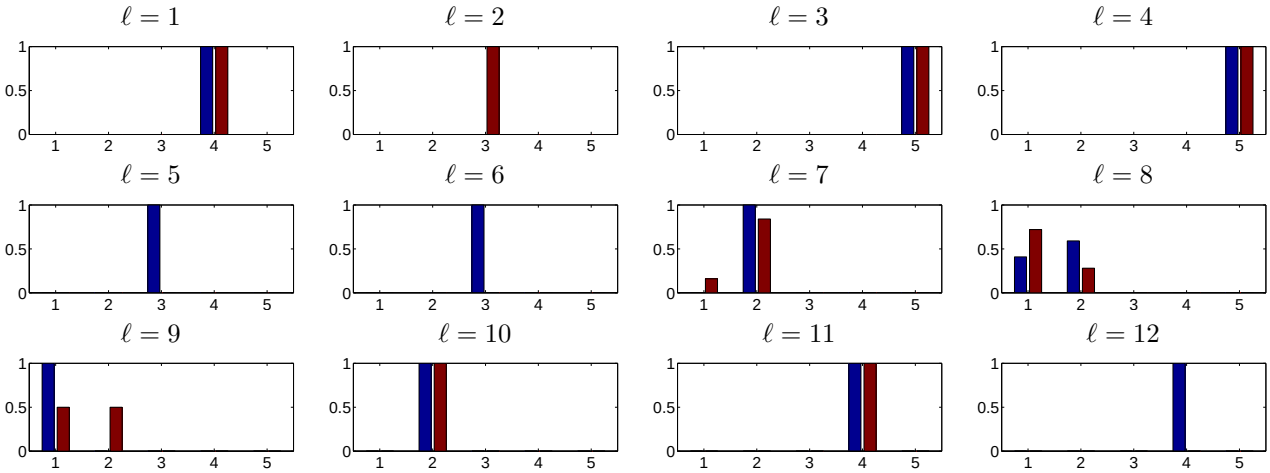


Fig. 12. Document clustering by ReMatch with the Wikipedia data. The x-axis in each figure shows the category index: (1) Nobel laureates in Physics, (2) Nobel laureates in Chemistry, (3) American basketball players, (4) American composers, and (5) English footballers'. The y-axis in each figure of cluster ℓ shows the probability that documents assigned to cluster ℓ are labeled with the category. The left blue bar shows the probability in English, and the right red bar shows that in German.

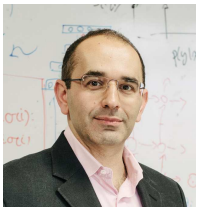
TABLE 5
Multi-lingual word clustering by ReMatch in the Wikipedia data. Each pair of rows corresponds to a cluster, and the top/bottom row shows words from English/German, respectively.

Lang	Words
EN	prize laureates fields nobel
DE	nobelpreis preisverleihung nobelstiftung hochschullehrer
EN	basketball nba draft weight sportspeople guard pro kg lb averaged ncaa
DE	draft basketball cm rebounds punkte forward playoffs lakers pick
EN	youth cup goals clubs premier counted footballers app gls fa caps manchester uefa soccerbase correct friendly substitute arsenal
DE	englischer fc united tore kader angegeben league manchester englische premier tor fa
EN	robertson thompson duncan teammate consecutive bio champion pick playoffs assists wilkins mvp rebounds ron rookie highlights
DE	punkten philadelphia assists zweimal bester spitzname finals bill year player guard celtics serie olympischen holte spielern rookie
EN	planck einstein shockley lamb yang jensen braun franck laue stark landau purcell alvarez glaser gabor townes millikan bardeen
DE	grundlagen theoretischen theoretische assistent zeitschrift verfasste ernst materie naturwissenschaften quantenmechanik friedrich
EN	steve break joe terry twice broke chris suffered performances captain ball fifth agreed shot offer county missed chance finish
DE	george lee folgen mittlerweile erstes beste erfolgreichen gary martin dritten west gespielt durchbruch brown tony young acht van
EN	chemist otto carl chemical frederick harold kurt chemists doi irving stanley biographical fellows friedrich porter fischer lecture todd
DE	chemie chemiker datensatz pnd individualisierter vorhanden biochemiker
EN	transfer cole draw soccer defeat campbell stadium euro neville beckham charity liverpool ham wembley eriksson promotion fee
DE	wm schoss unterschrieb pfund arsenal chris taylor absolvierte foto wayne spielzeit chelsea minuten steve cup englands partie fans
EN	parents berlin lawrence jewish columbia interested taught contributed contributions graduated mitchell francisco victor student
DE	auszeichnungen heiratete school institute starb life biographie royal high weltkrieg american wuchs kalifornien amerikanischer
EN	coach championship johnson kevin retired champions round seasons tournament teams finals championships
DE	besten vertrag berufen erreichte bevor beendete liga johnson jugend aktiv leistung sommer gewinnen finale verlor trikotnummer

feature models for link prediction," *Advances in Neural Information Processing Systems (NIPS)*, vol. 22, 2009.
[36] K. Ishiguro, T. Iwata, N. Ueda, and J. Tenenbaum, "Dynamic infinite relational model for time-varying relational data analysis," *Advances in Neural Information Processing Systems*, vol. 23, 2010.
[37] C. Blundell, Y. W. Teh, and K. A. Heller, "Bayesian rose trees," in *Proceedings of the 27th Conference on Unvertainty in Artificial Intelligence*, ser. UAI. AUAI Press, 2011.
[38] P. Shafto, C. Kemp, V. Mansinghka, and J. B. Tenenbaum, "A probabilistic model of cross-categorization," *Cognition*, vol. 120, no. 1, pp. 1–25, 2011.

Tomoharu Iwata received the B.S. degree in environmental information from Keio University in 2001, the M.S. degree in arts and sciences from the University of Tokyo in 2003, and the Ph.D. degree in informatics from Kyoto University in 2008. In 2003, he joined NTT Communication Science Laboratories, Japan. From 2012 to 2013, he was a visiting researcher at University of Cambridge, UK. He is currently a senior research scientist (distinguished researcher) at Learning and Intelligent Systems Research Group of NTT Communication Science Laboratories, Kyoto, Japan. His research interests include data mining, machine learning, information visualization, and recommender systems.

James Robert Lloyd is a post doctoral researcher at the University of Cambridge working with Zoubin Ghahramani in the machine learning group. He studied mathematics, specialising in statistics, at the University of Cambridge where he also obtained his PhD in 2015. His research has included work on Gaussian processes, networks, automatic regression model building and model criticism.



Zoubin Ghahramani is Professor of Information Engineering at the University of Cambridge, where he leads the Machine Learning Group. He studied computer science and cognitive science at the University of Pennsylvania, obtained his PhD from MIT in 1995, and was a postdoctoral fellow at the University of Toronto. His academic career includes concurrent appointments as one of the founding members of the Gatsby Computational Neuroscience Unit in London, and as a faculty member of CMU's Machine Learning Department for over 10 years. He has published over 200 papers, receiving 27,000 citations (an h-index of 73). His current research interests include statistical machine learning, Bayesian nonparametrics, scalable inference, probabilistic programming, and building an automatic statistician. He has held a number of leadership roles as programme and general chair of the leading international conferences in machine learning: AISTATS (2005), ICML (2007, 2011), and NIPS (2013, 2014). In 2015 he was elected a Fellow of the Royal Society.