

Guided Learning for Role Discovery (GLRD): Framework, Algorithms, and Applications

Sean Gilpin
University of California, Davis
sagilpin@ucdavis.edu

Tina Eliassi-Rad
Rutgers University
eliassi@cs.rutgers.edu

Ian Davidson
University of California, Davis
davidson@cs.ucdavis.edu

ABSTRACT

Role discovery in graphs is an emerging area that allows analysis of complex graphs in an intuitive way. In contrast to community discovery, which finds groups of highly connected nodes, role discovery finds groups of nodes that share similar topological structure in the graph, and hence a common role (or function) such as being a broker or a periphery node. However, existing work so far is completely unsupervised, which is undesirable for a number of reasons. We provide an alternating least squares framework that allows convex constraints to be placed on the role discovery problem, which can provide useful supervision. In particular we explore supervision to enforce i) sparsity, ii) diversity, and iii) alternativeness in the roles. We illustrate the usefulness of this supervision on various data sets and applications.

Categories and Subject Descriptors

H.2.8 [Database Applications]: Data mining; E.1 [Data Structures]: Graphs and networks

General Terms

Algorithms, Design, Performance, Experimentation

Keywords

Role discovery; constrained clustering; graph mining

1. INTRODUCTION

Role discovery is a developing area that allows the simplification of graphs in a user-interpretable way. Consider a graph of n nodes specified in an adjacency matrix \mathbf{A} . Earlier efforts convert this matrix into a new $n \times f$ matrix \mathbf{V} so that each node in the graph has a list of f features [15]. Role discovery is then the computation of converting V so that each node/user is mapped to a combination of roles (denoted by the $n \times r$ matrix \mathbf{G}) and each role is defined with respect to the f features (denoted by the $r \times f$ matrix \mathbf{F}). This

is accomplished by performing a non-negative matrix factor decomposition as shown below:

$$\begin{aligned} & \underset{\mathbf{G}, \mathbf{F}}{\operatorname{argmin}} \quad \|\mathbf{V} - \mathbf{G}\mathbf{F}\|_2 \\ & \text{subject to: } \mathbf{G} \geq \mathbf{0}, \mathbf{F} \geq \mathbf{0} \end{aligned} \quad (1)$$

The $n \times r$ matrix \mathbf{G} when read row-wise indicates which of the r roles each node plays and to what degree. The $r \times f$ matrix \mathbf{F} when read row-wise defines each of the r roles with respect to the f features. The entries in \mathbf{G} and \mathbf{F} are positive real numbers signifying that each node can play each role to varying degrees and that different features define a role in varying degrees. This simplification of graphs into roles is not only intuitive for a domain expert, but it has been shown to be useful in a number of interesting settings including prediction, transfer learning, and sense making [14].

However, all work developed so far has been completely unsupervised in that the domain expert cannot easily inject their expertise and expectations into the simplification. Having this ability is useful in a variety of settings. Consider a domain expert that is looking for the simplest explanation of a graph during their exploratory phase of analysis. Existing work cannot specify how to emphasize this simplicity apart from requiring a small number of roles to be used. Other forms of parsimonious guidance such as requiring a node only be assigned to a few roles or making each role defined by only a small set of features is desirable but currently not possible. Similarly, if a decomposition yields a set of roles that are not actionable, not interesting or already known, the domain expert cannot enforce an *alternative* set of roles. These two recent trends in data mining – exploring the addition of positive and negative guidance – have been shown to have wide-scale application in the data mining literature [2][24]; but to our knowledge have not been applied to role discovery. Hence this work marks the first paper exploring guided role discovery.

Our work makes several contributions to the field of role discovery in graphs:

- We provide a framework to encode guidance as a series of convex optimization problems each of which can be efficiently solved by our alternating least squares (ALS) algorithm. All data sets and code will be made available once the paper is accepted.
- Within our framework we explore guidance in the form of sparsity, diversity and orthogonality/alternativeness but other types of guidance are possible.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

KDD'13, August 11–14, 2013, Chicago, Illinois, USA.
Copyright 2013 ACM 978-1-4503-2174-7/13/08 ...\$15.00.

- We show that sparsity and diversity yield improved performance in terms of predictive accuracy for the identity resolution task across multiple graphs.
- We show that alternative roles exist in social networks (such as in a YouTube graph) and in particular these roles are very different from the known communities in the data.

In the next section, we describe related work and then an algorithm for incorporating convex constraints in non-negative matrix decomposition. Section 4 presents how convex constraints can naturally encode guidance in the form of sparsity and diversity on both the role assignment matrix (\mathbf{G}) and role explanation matrix (\mathbf{F}). We also present how these constraints can encode the notion of alternativeness to find a different set of roles to another set that are for instance non-actionable or trivial. Our experiments, in Section 5, demonstrate the usefulness of these forms of guidance in a number of applications and real-world graphs. We show how sparsity and diversity guidance can improve upon prediction performance for the application of identity resolution via roles. We also show how alternativeness can be used to find an alternative set of roles to the underlying community structure.

2. RELATED WORK

The basis for role discovery in graphs using non-negative matrix factorization (NMF) was first proposed in a series of papers at KDD [15][14]. The method ReFeX [15] described a recursive method to take a $n \times n$ adjacency matrix (A) and compute a set of f salient features for each of the n nodes represented as a matrix V . The RolX method [14] made use of NMF to simplify the features into a set of roles and explored their use for graph matching, sense making and transfer learning. Many previous works had applied NMF to other data mining problems (e.g. [27][18]) but theirs was the first to apply it to role discovery. Other methods for role discovery are not scalable to huge graphs and include Bayesian frameworks using MCMC sampling methods [25] and semi-supervised role labeling [11].

The addition of guidance to matrix decomposition is a relatively new area with most work involving spatial data and properties such as unimodality as we have done for tensors [8]. Of course much work exists on very basic constraints such as non-negativity and minimal rank decompositions. The area of constraints for matrix decomposition takes on several different meanings to our own work. For example in [19] the authors propose the use of labeled information to guide the decomposition. Perhaps the closest to our own work is the use of sparseness constraints in NMF [16].

To the best of our knowledge the encoding of guidance for role discovery and the encoding of diversity and alternative constraints for NMF as described in this paper has not been addressed before. However, the notion of guided and alternative is popular in the clustering field with work by ourselves and others [2][24].

3. A FRAMEWORK FOR ENCODING GUIDANCE INTO ROLE DISCOVERY

In this section, we discuss our algorithm for solving the guided role-discovery problem. We present a general algo-

rithm that is well-suited for large-scale problems, and is capable of being extended to different forms of guidance. The different supervisions (described in Section 4) are solvable using this algorithm.

Our algorithm for solving the guided role discovery problem is a *constrained* NMF approach used to find the decomposition shown in Equation 2. Like many unconstrained NMF solvers, it uses the alternating least squares approach [23, 4]. Nonnegative least squares is a well-studied problem, and can be utilized to find an NMF solution by solving for one matrix at a time (\mathbf{G} or \mathbf{F}), while holding the other constant which is generally known as alternating least squares (ALS). NMF is known to be intractable; and the ALS approach is not guaranteed to find global solutions but will converge to a local minimum. In this work, we add additional constraints to the problem and therefore need more sophisticated methods.

The method we chose was motivated by gradient projection methods, which are known for being well-suited to quickly finding good but not highly accurate solutions for large problems, by sacrificing some of the theoretical convergence guarantees of methods such as interior point [3]. Projected gradient descent methods can be summarized as those that iteratively find better points by following the gradient of the objective function, and subsequently find the closest point that meets the constraints. Since the objective we are solving is least squares, we have a closed form solution to the unconstrained minimum from which we subsequently find the closest constrained solution. It is known, that for a class of constrained least squares solution, this approach will lead to an exact global solution in one iteration (see Lemma 1).

Therefore, our algorithm has the advantage that each *sub-problem* (but not the entire problem) can be solved exactly by reducing it into an unconstrained least square problem [26][1] and an Euclidean projection problem [10][21], both of which have efficient solutions. Additionally, this approach to optimization (projected gradient descent) has been shown in the past to work well on large-scale problems, at the expense of accuracy, and is used by state of the art solvers [20].

The outline of the remainder of this section is as follows. First, we formally describe the convex constrained NMF problem and discuss how ALS can be used to solve it. Then, we explain how ALS can also be used to solve for individual role assignment vectors, as well as role definition vectors. Finally, we describe how ALS over definition/assignment vectors can be solved using a projection method by first solving an unconstrained least squares problem and then finding the closest point in the constrained space.

The Constrained NMF Problem.

In Equation 2, there are two variables \mathbf{G} and \mathbf{F} that are being simultaneously optimized. If either is treated as a constant, the problem becomes convex and can be solved exactly using any method for solving convex optimization problems. One can alternate between solving for \mathbf{G} and \mathbf{F} this way until convergence. Although each iteration finds a global optimum to this modified problem, the result of this procedure (alternating optimization) is not guaranteed to find a global minimum to the original problem in Equation 2. In the following, we describe our method for transforming

the formulation into a series of convex programming problems, which are generally easy to solve.

$$\begin{aligned}
& \underset{\mathbf{G}, \mathbf{F}}{\text{minimize}} && \|\mathbf{V} - \mathbf{G}\mathbf{F}\|_2 \\
& \text{subject to} && g_i(\mathbf{G}) \leq d_{Gi}, i = 1, \dots, t_G \\
& && f_i(\mathbf{F}) \leq d_{Fi}, i = 1, \dots, t_F \\
& && \text{where } g_i \text{ and } f_i \text{ are convex functions.}
\end{aligned} \tag{2}$$

An ALS Formulation.

Rather than alternating between solving for the entire matrices \mathbf{G} and \mathbf{F} , we can instead solve for one column of \mathbf{G} or one row of \mathbf{F} at a time. This is possible if convex constraints can be specified in terms of these columns, which is the case in this work. Without loss of generality, Equation 3 shows an individual sub-optimization problem in terms of one of the columns of \mathbf{G} , denoted \mathbf{x} .

$$\begin{aligned}
\mathbf{G}_k = & \underset{\mathbf{x}}{\text{minimize}} && \|\mathbf{R} - \mathbf{x}\mathbf{F}_k\|_2 \\
& \text{subject to:} && g_i(\mathbf{x}) \leq d_{Gi}, i = 1, \dots, t_G
\end{aligned} \tag{3}$$

In Equation 3, \mathbf{R} represents the residuals of all other factors not being solved for (sum of outer products of corresponding columns of \mathbf{G} and rows \mathbf{F}). \mathbf{F}_k is the k^{th} row of the role/feature explanation matrix that corresponds to the k^{th} column of the role assignment matrix. So with this formulation, we alternate between learning single role assignments, followed by learning a role definition. Next we explain how we solve the convex constrained problem shown in Equation 3.

Solving The Constrained Least Squares Problem.

Our projection method is as follows. First, solve Equation 3 with all constraints removed using standard least squares solvers. Second, find the closest point to the unconstrained solution, that satisfies the given constraints. This projection method takes advantage of standard and very fast least squares solvers and the subsequent nearest feasible point problem is relatively simple to solve. In addition, Lemma 1 shows that performing these two steps will exactly solve the original problem in Equation 3. Applications of this theorem and its proof can be found in [6][13].

LEMMA 1. *Projection Equivalence Result. The following constrained optimization problem:*

$$\begin{aligned}
& \underset{\mathbf{x}}{\text{minimize}} && \|\mathbf{B} - \mathbf{x}\mathbf{a}\|_2 \\
& \text{subject to:} && c_i(\mathbf{x}) \leq d_i, i = 1, \dots, n
\end{aligned} \tag{4}$$

where c_i are convex functions on x , is equivalent to:

$$\begin{aligned}
& \underset{\mathbf{x}}{\text{minimize}} && \|\mathbf{x}^* - \mathbf{x}\|_2 \\
& \text{subject to:} && c_i(\mathbf{x}) \leq d_i, i = 1, \dots, n
\end{aligned} \tag{5}$$

where \mathbf{x}^* is the optimal to the optimization problem in Equation 4 without constraints.

This leads to the following algorithm for convex constrained NMF presented in Figure 1. Like ALS for unconstrained NMF, this heuristic is not guaranteed to meet a global optimum, even though all subproblems are solved exactly. However, each step will lead to a reduction in the global objective

(Equation 2). Thus, in practice the algorithm will find local minima that meet all specified constraints.

Inputs:

- \mathbf{V} : Node feature matrix containing n nodes described by f topological structure features.
- $g_i(\mathbf{x}), f_i(\mathbf{x})$: Convex constraints on columns of \mathbf{G} and rows of \mathbf{F} respectively.
- r : Number of roles (methods for learning r described in previous work [14]).

Outputs:

- \mathbf{G} : Role assignment matrix that satisfying all constraints.
- \mathbf{F} : Role definition matrix that satisfying all constraints.

Algorithm:

```

while reconstruction error decreases do
{
  for  $k = 1 \dots r$  //Recalculate each role.
  {
    1. Calculate  $\mathbf{R} = \mathbf{V} - \mathbf{G}_{\bullet(\neq k)}\mathbf{F}_{(\neq k)\bullet}$ 
    2. Calculate  $\mathbf{G}_{\bullet k}$  by solving for  $\mathbf{x}$  as follows:
      (a)  $\mathbf{x}^* = \underset{\mathbf{x}}{\text{argmin}} \|\mathbf{R} - \mathbf{x}\mathbf{F}_{k\bullet}\|_2$ 
      (b)  $\mathbf{G}_{\bullet k} = \underset{\mathbf{x}}{\text{argmin}} \|\mathbf{x}^* - \mathbf{x}\|_2$  s.t.  $g_i(\mathbf{x}) \leq \epsilon_i : \forall_i$ 
    3. Calculate  $\mathbf{F}_{k\bullet}$  by solving for  $\mathbf{x}$  as follows:
      (a)  $\mathbf{x}^* = \underset{\mathbf{x}}{\text{argmin}} \|\mathbf{R} - \mathbf{x}\mathbf{G}_{\bullet k}\|_2$ 
      (b)  $\mathbf{F}_{k\bullet} = \underset{\mathbf{x}}{\text{argmin}} \|\mathbf{x}^* - \mathbf{x}\|_2$  s.t.  $f_i(\mathbf{x}) \leq \epsilon_i : \forall_i$ 
  }
}

```

Figure 1: Our algorithm that will be used to encode all guidances described in Section 4. The algorithm uses a least squares approach and allows additional convex constraints to be added to the NMF formulation.

The advantage of solving for one role at a time rather than the entirety of \mathbf{G} or \mathbf{F} as is generally done with ALS, is that it allows the problem to be broken down into smaller parts that then fit into fast solvers. In general, projection methods have been found to be better suited to larger problems and we found this to be the case as well. Using this method allows us to solve much larger problems than we had previously been able to using standard constrained optimization solvers [8]. The final constrained optimization problem (i.e.,

closest constrained point problem) is simple enough that we find for even medium-sized problems we could utilize high level solvers such as CVX [7][12], which makes experimenting with new types of constraints very simple.

4. FRAMEWORK FOR FLEXIBLE SUPERVISION

In the previous section, we discussed a novel and general algorithm that can easily handle convex constraints. Convex constraints can encode a variety of useful guidances. In this section, we show how they can be used to enforce sparsity, diversity and alternativeness. In the experimental section, we show applications which exploit these forms of guidance.

4.1 Sparsity

The area of sparsity has recently attracted much attention. In a general context, sparsity has been shown to have two main benefits: (1) parsimony and (2) improved predictive performance, with the later being motivated by Occam’s razor. Sparse learning formulations exist for many learning settings such as linear regression (LASSO), Kernel methods (SVM) and covariance estimation.

In our work, we can place sparsity constraints on both the \mathbf{G} or \mathbf{F} matrices leading to an objective function of:

$$\begin{aligned} & \underset{\mathbf{G}, \mathbf{F}}{\operatorname{argmin}} \quad \|\mathbf{V} - \mathbf{GF}\|_2 \\ \text{subject to:} \quad & \mathbf{G} \geq 0, \mathbf{F} \geq 0 \\ & \forall i \quad \|\mathbf{G}_{\bullet i}\|_1 \leq \epsilon_G \\ & \forall i \quad \|\mathbf{F}_{i \bullet}\|_1 \leq \epsilon_F \end{aligned} \quad (6)$$

where ϵ_G and ϵ_F define upperbounds for the sparsity constraints (amount of allowable density).

Previous works have shown the effectiveness of using L1 norm as a penalty in model learning. In our formulation the L1 penalty is encoded as a constraint rather than a penalty in the objective, but it is known that these formulations are theoretically equivalent [5]. However, another twist to our formulation is that we do not constrain the entire matrix but instead constrain each column of \mathbf{G} and each row of \mathbf{F} . This was done because our solver requires constraints to be formulated only over one role vector at a time. The effect of this technical difference is that the sparsity must be more uniformly spread across each role definition or role assignment which is a benefit of this method.

Sparsity constraints on \mathbf{G} and \mathbf{F} have easy to understand intuitive interpretations. If \mathbf{G} is sparse, it means that nodes are assigned to as few roles as possible; and it is possible for some nodes to be assigned to no roles. If \mathbf{F} is sparse, it means that the roles are defined with respect to as few features as possible. Both of these extensions allow for a simple explanation of the data, and lead to improved prediction performance.

4.2 Diversity

In the NMF forms of role discovery, nothing prevents the roles to which nodes are assigned (i.e., the \mathbf{G} matrix) and the role definitions (i.e., the \mathbf{F} matrix) to be highly overlapping. This can be undesirable particularly for the \mathbf{F} matrix since it means all roles are highly similar. This can be overcome by enforcing a diversity requirement so that each role

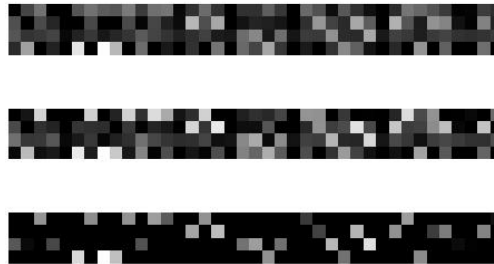


Figure 2: Visualization of diversity constraints on role explanation matrix F (roles \times features) for DBLP dataset. The top matrix shows the unconstrained result; the bottom matrix is constrained to be completely diverse ($\epsilon = 0$); and the middle matrix shows a middle ground. From the top matrix to the bottom matrix, the number of black cells (i.e. zero values) increases since roles definitions must be explained with completely different sets of features.

uses a different set of features (for the \mathbf{F} matrix) and nodes are assigned to different combinations of roles (for the \mathbf{G} matrix).

Our formulation for role allocation diversity (\mathbf{G} matrix) and role definition diversity (\mathbf{F} matrix) makes use of orthogonality as follows:

$$\begin{aligned} & \underset{\mathbf{G}, \mathbf{F}}{\operatorname{argmin}} \quad \|\mathbf{V} - \mathbf{GF}\|_2 \\ \text{subject to:} \quad & \mathbf{G} \geq 0, \mathbf{F} \geq 0 \\ & \forall i, j \quad \mathbf{G}_{\bullet i}^T \mathbf{G}_{\bullet j} \leq \epsilon_G \quad i \neq j \\ & \forall i, j \quad \mathbf{F}_{i \bullet}^T \mathbf{F}_{j \bullet} \leq \epsilon_F \quad i \neq j \end{aligned} \quad (7)$$

where ϵ_G and ϵ_F define upperbounds on how angularly similar role assignments and role definitions can be to each other.

When $\epsilon = 0$, our constraint will exactly match the definition orthogonality, and when $\epsilon \geq 0$ the constraint can be viewed as limiting the angular similarity between two vectors. The effect of combining this constraint with non-negativity constraints is that no role definitions will have any common features and no role assignments will have overlapping populations for $\epsilon = 0$. This is so since $\mathbf{G}_{\bullet i}^T \mathbf{G}_{\bullet j} = 0$ if and only if these two vectors do not share any non-zero entries. Figure 2 shows such an example, where none of the three roles have any overlapping features. In the context of our solver which solves for one vector at a time, this constraint will be linear (a weighted sum).

4.3 Alternative Role Discovery

Recent work on another unsupervised problem, clustering, has explored the area of alternativeness [24, 9]. In that literature, the term alternativeness and orthogonality are used interchangeably, but we only use the term alternativeness for clarity.

The motivation for alternativeness in unsupervised learning is strong. Most interesting problems are on large data

Table 1: Summary of effects of constraints on both role assignment \mathbf{G} and role definitions \mathbf{F} (see Section 4 for formulation of each constraint type).

	Role Assignment	Role Definition
Sparsity	Encourages role assignments to be more definitive. Increasing the strength of constraint reduces the number of nodes that have minority membership in role.	Increases the ability to interpret role definitions by ensuring that the definitions only use features most strongly correlated with each role. Increasing the strength of constraint decreases the likelihood that features with small explanatory benefit are included.
Diversity	Roles cannot have memberships that are too similar. No two roles can have exactly the same membership assignment. Increasing the strength of the constraint limits the amount of allowable overlap in assignments.	Roles cannot have definitions that are too similar. No two roles can have redundant explanations and increasing the strength of constraint ensures that roles must be explained with completely different sets of features.
Alternative	Find a set of roles that lends itself to a different role assignment than a given role assignment. Increasing the strength of constraint, decreases the allowable similarity between the two.	Learn a role definition matrix that is significantly different than a given role definition. Increasing the strength of constraint ensures that the definitions must be very dissimilar.

sets that contain complex phenomena and there may exist multiple explanations of the data. However, most unsupervised learning algorithms expect that there exists only one good explanation of the data and return one explanation.

In many situations, it may be the case that the returned explanation is undesirable since it is either unactionable or not novel. Consider the IMDB (Internet Movies Database) dataset. If the resultant roles map actors to the studios for which they work, then this is not particularly novel. Here, the work on alternative role discovery allows a previously discovered set of role allocations (\mathbf{G}^*) or role definitions (\mathbf{F}^*) to be specified as a counter-example of what not to find. The challenge though is to find another good explanation of the data that is different to those already found.

The optimization problem to find alternative roles is then:

$$\begin{aligned}
 & \underset{\mathbf{G}, \mathbf{F}}{\operatorname{argmin}} \quad \|\mathbf{V} - \mathbf{G}\mathbf{F}\|_2 \\
 & \text{subject to:} \quad \mathbf{G} \geq 0, \mathbf{F} \geq 0 \\
 & \quad \forall i, j \quad \mathbf{G}_{\bullet i}^{*T} \mathbf{G}_{\bullet j} \leq \epsilon_G \\
 & \quad \forall i, j \quad \mathbf{F}_{i \bullet}^* \mathbf{F}_{j \bullet}^T \leq \epsilon_F \\
 & \text{where} \quad \epsilon_G \text{ and } \epsilon_F \text{ define upperbounds on} \\
 & \quad \text{how similar the results can be to } \mathbf{G}^* \\
 & \quad \text{and } \mathbf{F}^*.
 \end{aligned} \tag{8}$$

5. EXPERIMENTS

Our experiments demonstrate how constraints on graph role discovery can be useful. Role discovery requires the user to specify the number of roles to use and a set of features for a graph. For the former, we used the Minimum Description Length (MDL) described in [14] to automatically select the number of roles; and for the later, we used the approach described in [15]. We show that role discovery can be used to improve the results of the identity resolution problem between two graphs, and that they can be further improved by using sparsity or diversity constraints. By using sparsity or diversity constraints, we improve the role definitions which leads to more meaningful role assignments and more accurate identity resolutions. See Section 5.1 for these

experiments. We also experimentally verify the solutions to the alternative role discovery formulation presented in Section 4.3 and observe that they indeed produce significantly different results. The purpose of our experimental section is to address the questions:

1. Does adding constraints to the NMF-based role discovery formulation improve the quality of the resulting role explanations and assignments? Figures 3 and 4 show that constraints improve the results of identity resolution.
2. What effects do diversity constraints have on role discovery results? Figures 3 and 4 show how diversity constraints can improve role discovery results even more so than sparsity constraints.
3. Can our alternative role discovery formulation produce significantly different results? Tables 3 and 4 shows that our formulation can produce results that are significantly different than a given set of roles or community assignments respectively.

5.1 Sparse and Diverse Identity Resolution in Co-authorship Graphs

In this experiment, we show that by adding sparsity and diversity constraints to the NMF formulation of role discovery, the resulting role definitions are of higher quality. We measure this improvement in quality indirectly by showing how role definition matrices can be used for resolving identities of nodes across graphs, and that constrained role definitions perform better than unconstrained role definitions for that problem.

From the DBLP data-set [17], we extracted a co-author graph from each of the following related conferences from 2005 to 2009: KDD, ICDM, SDM, CIKM, SIGMOD, VLDB (see Table 2 for detailed information about each co-author graph). We extract a set of relevant structure features for the KDD graph using REFEX [15], and compute these same features for all of the co-author graphs. We subsequently learn a set of role definitions from the KDD graph using

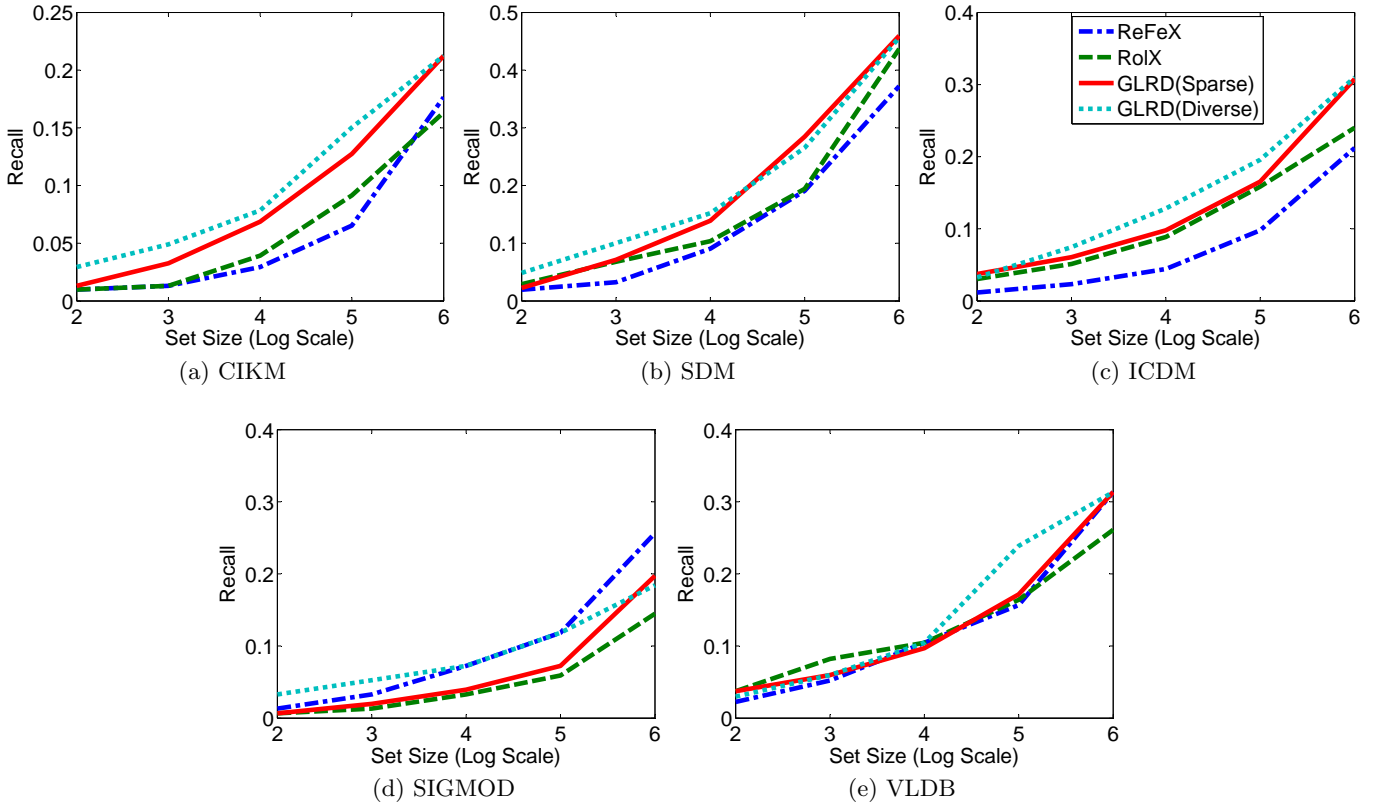


Figure 3: Comparison of role discovery techniques for identity resolution across graphs. Role definitions are learned from the KDD co-authorship graph; then, authors from the other (conference) co-authorship graphs are assigned to these roles using various techniques. In particular, we show results for ReFeX (features only), RolX (unconstrained role discovery), GLRD-Sparse (role discovery with sparsity constraints), and GLRD-Diverse (role discovery with diversity constraints). Authors from each conference are paired with increasing number of nearest neighbors from KDD conference (x-axis) and the resulting recall is reported (y-axis). Across most settings role definitions using sparsity and diversity constraints lead to better identity resolution results than standard unconstrained RolX. For graphs that are most similar in nature to KDD (e.g. ICDM, SDM, CIKM) the transfer of role definitions lead to better results than simply using structural features of nodes directly. Note that the recall values are relatively low because the set sizes (on the x-axis) are small compared to the population size in each graph.

Network	$ V $	$ E $	k	$ LCC $	$\#CC$
VLDB	1,306	3,224	4.94	769	112
SIGMOD	1,545	4,191	5.43	1,092	116
CIKM	2,367	4,388	3.71	890	361
SIGKDD	1,529	3,158	4.13	743	189
ICDM	1,651	2,883	3.49	458	281
SDM	915	1,501	3.28	243	165

Table 2: Information about DBLP co-author networks for each conference. Data was collected for five years (2005-2009). $|V|$ =number of vertices, $|E|$ =number of edges, k =average degree, $|LCC|$ =size of largest connected component, $\#CC$ =number of connected components.

standard RolX [14] as well as the sparse and diverse versions of GLRD. For each of these competing role definitions, we assign each vertex from each graph to the roles whose function they most exhibit. As a baseline, we also explore author identification without roles by using the raw graph features as described in ReFeX.

We use the role assignments to resolve the identities of vertices from each graph (namely, ICDM, SDM, CIKM, SIGMOD, and VLDB) to the vertices in the KDD graph. Without loss of generality, assume we are resolving identity of authors from the KDD graph to the authors in ICDM graph. For each author in both conferences, we select the corresponding row vector from the node by role matrix \mathbf{G}_{kdd} and find the k closest neighbors (row vectors) from \mathbf{G}_{icdm} . If the original author from KDD graph is present in the set of k closest neighbors, we count the result as a match. We repeat this experiment using sparsity and diversity constraints on \mathbf{F}_{kdd} . We also repeat the experiment using the ReFeX features, comparing author feature vectors from \mathbf{V}_{kdd} and \mathbf{V}_{icdm} . Figures 3 and 4 shows how the different decompo-

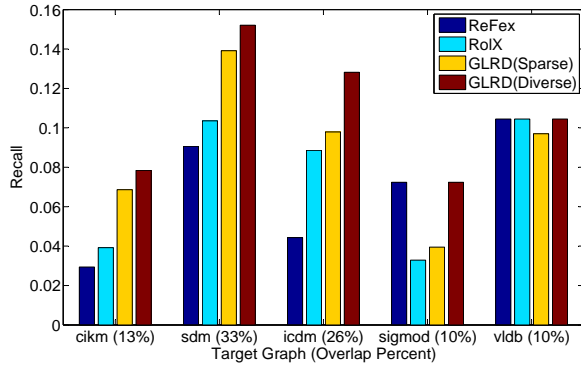


Figure 4: Comparison of role discovery techniques for identity resolution experiments. Authors from each conference paired with the nearest 32 neighbors from KDD conference; the resulting recall accuracy is reported. The percentage number (on the x-axis) is the fraction of authors that overlap between the two conferences. Nearly all experiments show better results with sparsity and diversity constraints except when the authors do not share similar roles in the two conferences (SIGMOD and VLDB).

sition methods compare in this setting for all graphs paired with KDD.

Our method of utilizing role discovery results for the author identification task is described formally in the following set of steps:

1. Extract features from co-authorship graphs to get graph features (e.g. $\mathbf{V}_{kdd}, \mathbf{V}_{icdm}$) using ReFeX.
2. From the graph features matrix \mathbf{V}_{kdd} perform role discovery to obtain \mathbf{G}_{kdd} and \mathbf{F}_{kdd} .
3. Transfer the role definition matrix \mathbf{F}_{kdd} (role by feature matrix) to other graphs (e.g. \mathbf{V}_{icdm}) by solving Equation 9.

$$\mathbf{G}_{icdm} = \min_{\mathbf{G}} \|\mathbf{V}_{icdm} - \mathbf{G}\mathbf{F}_{kdd}\|_2 \text{ s.t. } \mathbf{G} \geq 0 \quad (9)$$

Our experiments with graph identity-resolution show that diversity and sparseness constraints almost universally improve the quality of learned role-definition matrix. This is not unexpected since there is a long tradition in machine learning of using sparsity to prevent overfitting. As mentioned previously we can view diversity as enforcing sparsity since a diverse set of roles as per our definition do not share many overlapping features and hence each role definition is concise.

Figure 3 shows that role definitions learned using sparsity and diversity outperform standard unconstrained role discovery (RoIX) in almost every setting and problem parameterization. Figure 4 more clearly shows the general trend by considering the results for a particular problem parameterization. In that figure, we observe that diversity constraints lead to the most improvement over RoIX, while sparsity improvements are lesser. We also observe that transferring the KDD role definitions to some graphs (like VLDB and SIGMOD) does not compare well to the baseline method that does not use any roles (such as ReFeX). We believe this is because the same participants in conferences such as VLDB

and SIGMOD do not have a similar role to the ones they play in KDD; and hence, using the raw features (without roles) produces better results.

We believe that sparsity improves the quality of role definitions by reducing the ability of unconstrained NMF-based role discovery to overfit the problem. Features that only slightly add to the definition of a role are more likely to be explaining noise; and by forcing those values to zero, we end up with more robust definitions. Furthermore, the diversity constraints help by removing redundancy in role definitions, which leads to definitions that are more easily comparable. For example, if a feature is used to define every role, then it is not essential in defining any of them.

5.2 Alternative Roles

In this section, we show that our alternative role discovery formulation (presented in Section 4.3) can discover significantly different role definitions, as well as show that the formulation can be used to improve the role definitions when there are ground-truth communities. In Table 3, we show the difference between an alternative role discovery result and an original role definition found using unconstrained role discovery (via RoIX). In Table 4, we show that we can use our formulation to get more consistent assignments of roles when ground-truth communities are known.

In our first experiment, we explore the difference between the roles of the original and alternative role discovery. Using the KDD co-authorship graph, we find a set of roles and constrain a new solution to have a significantly different role definition (\mathbf{F} matrix). We then compare the results by assigning each vertex to its most dominant role in both results to create two separate partitions of the vertices. We then measure the difference between the two partitions using Jaccard distance. Table 3 shows that all of the Jaccard distances are far from 0 meaning that the alternative role assignments are very different than the original ones. Figure 5 illustrates the alternative roles found in the largest connected component of the KDD coauthorship graph. Note, the reader can zoom in on this figure to read the names of each author. The following is a description of the original roles and the roles that GLRD(Alternative) found. These descriptions are based on sense-making analysis [14]. As the descriptions show these roles are capturing alternative concepts.

	R1(alt)	R2(alt)	R3(alt)	R4(alt)
R1	0.946	0.510	0.762	0.913
R2	1.000	0.971	0.810	0.739
R3	1.000	0.7942	1.000	1.000
R4	0.345	0.991	1.000	0.982

Table 3: Jaccard distance matrix comparing original role assignments (rows) to alternative role assignments (columns). Jaccard distance of 0 represents an exact match between clustering and 1 represents no overlap. The relative error for the two decompositions was similar: 0.12% and .5% (where relative error is $error = \|\mathbf{V} - \mathbf{G}\mathbf{F}\|/\|\mathbf{V}\|$).

Original Roles:

Role 1: Nodes here have high eccentricity. These are periphery nodes.

Role 2: Nodes here have high eccentricity and high clustering coefficient. These are periphery nodes that are cliquy.

Role 3: Nodes here have high degree and high clustering coefficient. These are highly connected cliquy nodes.

Role 4: Nodes here have high PageRank, high degree, and high biconnected components numbers. These are globally central stars and brokers.

Alternative Roles:

Role 1: Nodes here have high PageRank and high biconnected component numbers. These are globally central and brokers.

Role 2: Nodes here have high clustering coefficient but not high eccentricity. These are non-periphery nodes that are cliquy.

Role 3: Nodes here have high eccentricity and high clustering coefficient. These are periphery nodes that are cliquy.

Role 4: Nodes here have high eccentricity and high degree. These are periphery nodes that are locally stars.

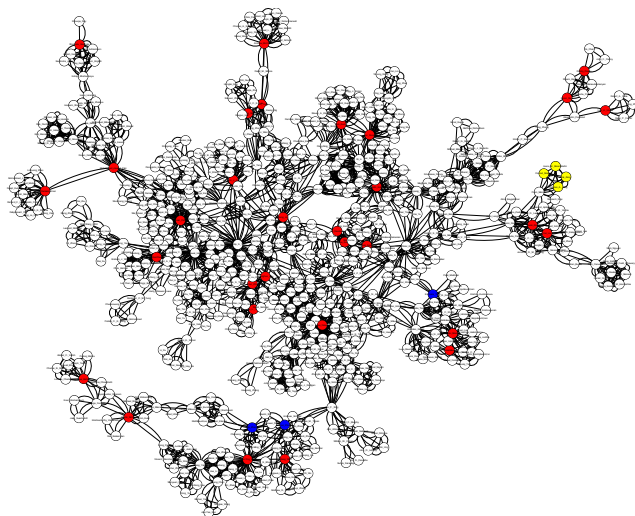


Figure 5: A visualization of our alternative role discovery results for the KDD co-authorship graph’s largest connected component. All the colored nodes belong to the same primary role under the original factorization. However, they belong to different primary roles under the alternative factorization, as indicated by the various colors. We observe that the alternative roles are able to separate the 3 blue “local-star” nodes (namely, Jun Zhu, Lei Zhang, and Evimaria Terzi) from the red “global-broker” nodes (namely, Christos Faloutsos, Heikki Mannila, Vipin Kumar, etc). The alternative roles also separate out the 4 yellow “periphery-cliquy” nodes. Note, the reader can zoom in on this figure to read the names of each author.

We next experiment with a YouTube dataset, which is a network of users with known ground-truth communities [22].

This graph was created by crawling the YouTube site in 2007 and creating directed edges between a pair of users a and b if a ’s profile page linked to b ’s profile page. Ground-truth communities were assigned by collecting all users belonging to the same group, which were pages that allowed communications between users on given topics. The graph has 1,134,890 vertices, 2,987,624 edges, and 8,385 communities. We selected all communities with over 100 users of which there were 105. The largest community has 2,217 users.

There is an inherent complementarity between role discovery and community detection. The former is about structural similarity; while the latter is based on proximity in the graph. Role discovery finds functions/roles of users but does not find the communities themselves. However, there may be multiple interesting sets of communities within the same network and those communities may be characterized by very different roles. In this experiment, we encode the set of ground-truth communities for which our role discovery technique should find roles.

The way we encode the YouTube ground-truth communities into our analysis is by providing the communities as \mathbf{G}^* to our alternative role discovery formulation. This will force our discovered roles to have a role assignment that is different than ground-truth communities, which matches the semantic relationship between the two problems.

To evaluate the effectiveness of this result we measured the proportion of members in each community belonging to each role. We then calculated the standard deviation over all such communities per role and report the results in Table 4. The assumption for this evaluation is that each role should be equally represented in each community. Our results show that the alternative role discovery formulation can indeed be used to normalize the roles with respect to a set of ground-truth communities. After applying sense-making [14], the six roles that our GLRD(Alternative) finds are as follows:

Alternative Role 1: Nodes here are global hubs. They have high PageRank values, high out-degrees, and high biconnected component numbers.

Alternative Role 2: Nodes here are on the periphery of the graph. They have higher than default eccentricity.

Alternative Role 3: Nodes here are authorities. They have high PageRank values and high in-degrees.

Alternative Role 4: Nodes here are very cliquy. They have high clustering coefficients.

Alternative Role 5: Nodes here are local hubs. They have high out-degrees and high biconnected component numbers.

Alternative Role 6: Nodes here are the majority of the population; they are the “regular” folks. They have a local neighborhood that is more cliquy than expected but otherwise nothing special stands out.

6. CONCLUSION

Role discovery is an emerging and important area of graph mining. It looks at discovering nodes that perform similar functions in networks, but do not necessarily belong to the same community. Existing work so far has been completely

Roles	1	2	3	4	5	6
Original	7.85	7.93	8.70	2.35	9.81	7.57
Alternate	5.06	6.34	5.34	3.81	8.62	5.88

Table 4: For each role, we report the standard deviations of role proportions over all communities. The result shows that our alternative role discovery formulation can be used to find roles whose members are better distributed across a set of interesting communities. The values are scaled by 10^2 .

unsupervised. We propose a framework that allows incorporating convex constraints into NMF to allow a rich set of guided role discovery formulations.

In particular we explore three types of guidance: sparsity, diversity and alternativeness. Sparsity and diversity can be used to create simpler and more interpretable role definitions and role allocations. Also they can reduce overfitting and produce better predictive results for matching authors between the KDD conference and a variety of other conferences provided they perform similar roles in both conferences. The notion of alternativeness has been explored in the clustering literature and is useful if the given explanation is not valid and an alternative is required. Here we show that not only do alternative roles exist in co-author networks, but that we can find an alternative to the community structure in a very large YouTube graph.

We view our work as a framework for incorporating guidance into role discovery and in the future will explore other forms of guidance such as partial role definitions and extend our solvers to tensor settings to allow evolving roles.

7. ACKNOWLEDGMENTS

The authors gratefully acknowledge support of this research via ONR grants N00014-09-1-0712, N00014-11-1-0108 and NSF Grant NSF IIS-0801528. This work was also supported in part by IARPA via AFRL Contract No. FA8650-10-C-7061 and in part by DAPRA under SMISC Program Agreement No. W911NF-12-C-0028.

8. REFERENCES

- [1] B. W. Bader, T. G. Kolda, et al. Matlab tensor toolbox version 2.5, January 2012.
- [2] S. Basu, I. Davidson, and K. Wagstaff. *Constrained Clustering: Algorithms, Applications and Theory*. Prentice Hall, 2008.
- [3] A. Beck and M. Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.
- [4] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons. Algorithms and applications for approximate nonnegative matrix factorization. *Computational Statistics and Data Analysis*, 52(1):155–173, 2007.
- [5] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, NY, USA, 2004.
- [6] R. Bro and N. D. Sidiropoulos. Least squares algorithms under unimodality and non-negativity constraints. *J. of Chemometrics*, 12(4):223–247, 1998.
- [7] CVX Research. CVX: Matlab software for disciplined convex programming, version 2.0 beta, Sept. 2012.
- [8] I. Davidson, S. Gilpin, and P. B. Walker. Behavioral event data and their analysis. *DMKD*, 25(3):635–653, 2012.
- [9] I. Davidson and Z. Qi. Finding alternative clusterings using constraints. In *ICDM*, pages 773–778, 2008.
- [10] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra. Efficient projections onto the l_1 -ball for learning in high dimensions. In *ICML*, pages 272–279, 2008.
- [11] H. Furstenuau and M. Lapata. Semi-supervised semantic role labeling. In *EACL*, pages 220–228, 2009.
- [12] M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In *Recent Advances in Learning and Control*, volume 371, pages 95–110. Springer-Verlag, 2008.
- [13] W. Heiser and P. Kroonenberg. Dimensionwise fitting in PARAFAC-CANDECOMP with missing data and constrained parameters. Technical Report PRM 97-01, University of Leiden, The Netherlands, 1997.
- [14] K. Henderson, B. Gallagher, T. Eliassi-Rad, H. Tong, S. Basu, L. Akoglu, D. Koutra, C. Faloutsos, and L. Li. RolX: Structural role extraction & mining in large graphs. In *KDD*, pages 1231–1239, 2012.
- [15] K. Henderson, B. Gallagher, L. Li, L. Akoglu, T. Eliassi-Rad, H. Tong, and C. Faloutsos. It’s who you know: Graph mining using recursive structural features. In *KDD*, pages 663–671, 2011.
- [16] P. O. Hoyer. Non-negative matrix factorization with sparseness constraints. *JMLR*, 5:1457–1469, 2004.
- [17] M. Ley. DBLP, computer science bibliography. <http://www.informatik.uni-trier.de/~ley/db/>.
- [18] T. Li and C. Ding. The relationships among various nonnegative matrix factorization methods for clustering. In *ICDM*, pages 362–371, 2006.
- [19] H. Liu, Z. Wu, X. Li, D. Cai, and T. Huang. Constrained nonnegative matrix factorization for image representation. *PAMI*, 34(7):1299–1311, 2012.
- [20] J. Liu, S. Ji, and J. Ye. *SLEP: Sparse Learning with Efficient Projections*. Arizona State University, 2009.
- [21] J. Liu and J. Ye. Efficient Euclidean projections in linear time. In *ICML*, pages 657–664, 2009.
- [22] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *IMC*, pages 29–42, 2007.
- [23] P. Paatero and U. Tapper. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126, 1994.
- [24] Z. Qi and I. Davidson. A principled and flexible framework for finding alternative clusterings. In *KDD*, pages 717–726, 2009.
- [25] M. Somaiya, C. Jermaine, and S. Ranka. Mixture models for learning low-dimensional roles in high-dimensional data. In *KDD*, pages 909–918, 2010.
- [26] L. N. Trefethen and D. Bau. *Numerical Linear Algebra*. SIAM, 1997.
- [27] F. Wang, T. Li, X. Wang, S. Zhu, and C. Ding. Community discovery using nonnegative matrix factorization. *DMKD*, 22(3):493–521, 2011.