

Knowledge Transformation from Word Space to Document Space

Tao Li
School of Computer Science
Florida International University
Miami, FL 33199
taoli@cs.fiu.edu

Chris Ding
CSE Department
University of Texas at Arlington
Arlington, TX 76019
chqding@uta.edu

Yi Zhang, Bo Shao
School of Computer Science
Florida International University
Miami, FL 33199
yzhan004,bshao001@cs.fiu.edu

ABSTRACT

In most IR clustering problems, we directly cluster the documents, working in the document space, using cosine similarity between documents as the similarity measure. In many real-world applications, however, we usually have knowledge on the word side and wish to transform this knowledge to the document (concept) side. In this paper, we provide a mechanism for this knowledge transformation. To the best of our knowledge, this is the first model for such type of knowledge transformation. This model uses a nonnegative matrix factorization model $X = FSG^T$, where X is the word-document semantic matrix, F is the posterior probability of a word belonging to a word cluster and represents knowledge in the word space, G is the posterior probability of a document belonging to a document cluster and represents knowledge in the document space, and S is a scaled matrix factor which provides a condensed view of X . We show how knowledge on words can improve document clustering, i.e. knowledge in the word space is transformed into the document space. We perform extensive experiments to validate our approach.

Categories and Subject Descriptors

H.3.3 [Information Search and Retrieval]: Clustering; I.2 [Artificial Intelligence]: Learning; I.5 [Pattern Recognition]: Applications

General Terms

Algorithms, Experimentation, Measurement, Performance, Theory

Keywords

Clustering, Knowledge Transformation

1. INTRODUCTION

As a fundamental and effective tool for data organization, summarization and navigation, clustering has been receiving a lot of attention. Clustering is the problem of partitioning a finite set of points in a multi-dimensional space into classes (called clusters) so

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. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGIR '08, July 20–24, 2008, Singapore.

Copyright 2008 ACM 978-1-60558-164-4/08/07 ...\$5.00.

that (i) the points belonging to the same class are *similar* and (ii) the points belonging to different classes are *dissimilar* [12, 14].

Many clustering algorithms aim at clustering *homogeneous data* where the data points are all of a single type [3]. In many real world applications, however, a typical task often involves more than one type of data points. For example, in document analysis, there are *terms* and *documents*. Generally the different types of data points are not independent to each other and there exist close relationships among them. How to utilize the relationships is a challenging issue. It is difficult for traditional clustering algorithms to utilize those relationships efficiently.

Co-clustering algorithms aim at clustering different types of data simultaneously by making use of the dual relationship information such as the word-document matrix. For instance, bipartite spectral graph partitioning approaches are proposed in [8, 28] to co-cluster words and documents. Cho et al [5] proposed algorithms to co-cluster the experimental conditions and genes of microarray data by minimizing the sum-squared residue. Long et al. [20] proposed a general principled model, called *Relation Summary Network*, to co-cluster the heterogeneous data on a *k-partite graph*.

However, these co-clustering algorithms are unsupervised learning methods, based on the relationship information provided by the word-document matrix only. In many applications, we have some additional/external information. If the additional information is in document space and we are interested in document clustering, this topic is called Semi-supervised Clustering, and has been studied by [24, 4, 26, 6, 2, 7, 19], where the prior knowledge exists in the form of pairwise constraints (In general, however, prior knowledge is not easy to be incorporated into clustering; K-means, Gaussian mixture, and information bottleneck [22] are some of the examples where prior knowledge is difficult to incorporate.).

In many other cases, we have additional information/knowledge on the words side and we wish to see if they can influence/help the clustering of documents. To the best of our knowledge, this problem has not been investigated before. In this paper, we provide a model to show that this can be done. We start with a simple example.

1.1 An Illustrating Example

To demonstrate the usefulness of the additional information in the word space, we give a simple example in Figure 1. The synthetic dataset contains four research article titles as shown in part (a). After removing stop words and words appearing in every article, the dataset is represented as a word-document matrix as shown in part (b). The titles are from two topic areas: Information Retrieval (**IR**) and Computer Vision (**Vision**). If we look at the data matrix directly, $D1$ and $D3$ are similar based on cosine similarity since their dot product is 1 while $D1$ and $D2$ are not similar since

their dot product is 0. Similarly, $D2$ and $D4$ are similar while $D3$ and $D4$ are not similar.

If we run K-means on the simple dataset, $D1$ and $D3$ will be grouped into one cluster while $D2$ and $D4$ will be grouped into another cluster. Clearly, this result is not satisfactory as it can not reflect the topic areas of the titles. Now suppose we have additional knowledge on the words side. In our example, *clustering*, and *classification* belongs to word category **Learning**; *illumination* and *texture* belongs to word category **Graphics**; *webpage* and *hyperlink* belongs to word category **Web**. Using the additional information on word space, we can obtain perfect clustering since $D1$ and $D2$ are associated with word categories **Learning** and **Web**, while $D3$ and $D4$ are associated with word categories **Learning** and **Graphics**. Later on, in Section 2.2, we will illustrate in detail on how knowledge in the word space is transformed into the document space for this example. We will also present the computation results of our proposed method on the example in Section 2.3.

D1: An Algorithm for Hyperlink Clustering
D2: Algorithms for Webpage Classification
D3: Texture Clustering Algorithms
D4: An Algorithm for illumination Classification

(a) A Synthetic Dataset

		Learning		Graphics		Web	
		Clustering	Classification	Illumination	Texture	Webpage	Hyperlink
Vision	D1	1	0	0	0	0	1
	D2	0	1	0	0	1	0
	D3	1	0	0	1	0	0
	D4	0	1	1	0	0	0

(b) Dataset Representation

Figure 1: An Illustrating Example. The standard word-document matrix is the transpose of the table given in Part (b).

1.2 Organization of the Paper

In this paper, we provide a model to show that additional information/knowledge on the word side can influence/help the clustering of documents. This mechanism allows us to transform knowledge in the word space to the document space. In the following, we present the model in details, and provide a theoretical analysis of the knowledge transfer model. We provide a concrete computational algorithm to solve the model, and also prove the correctness and convergence of the algorithm based on constrained optimization theory. Since the word-document matrix is just an example of two-way data, our knowledge transformation mechanism can apply to any two-way data, such as the DNA microarray data where knowledge on the genes (rows) can be transformed to that of patient tissue samples (columns).

The rest of the paper is organized as follows: Section 2 introduces the basic model for enabling knowledge transformation from the word space to the document space. In particular, Section 2.2 gives a theoretical analysis on the effects of the prior knowledge in the word space; Section 2.3 presents a computational algorithm for solving the model. Section 3 proposes the method for knowledge transformation when knowledge in the word space is in the form of pairwise relations. Section 4 show our experiments on real-world datasets. Finally Section 5 concludes.

2. BASIC MODEL

In this paper, we provide a mechanism for knowledge transformation from the word space to the document space. This model uses a nonnegative matrix factorization model [11, 18]

$$X \approx FSG^T. \quad (1)$$

where X is a $m \times n$ word-document semantic matrix, F is an $m \times k$ nonnegative matrix representing knowledge in the word space, i.e., i -th row of F represents the posterior probability of word i belonging to the k classes, and G is an $n \times k$ nonnegative matrix representing knowledge in document space, i.e., the i -th row of G represents the posterior probability of document i belonging to the k classes. S is an $k \times k$ nonnegative matrix providing a condensed view of X .

We show how information on F help clustering documents on G . We have two different ways to incorporate knowledge in the word space. The first is a categorization of words, represented by a complete specification of F . This is presented in Section 2.1. Another way is partial knowledge on words, for example, two words are known to be highly related and must be grouped into the same word cluster. This is discussed in Section 3.

Our model is similar to the probabilistic latent semantic indexing (PLSI) model [13]. In PLSI, X is treated as the joint distribution between words and documents by the scaling $X \rightarrow \bar{X} = X / \sum_{ij} X_{ij}$ thus $\sum_{ij} \bar{X}_{ij} = 1$. \bar{X} is factorized as

$$\bar{X} \approx WSD^T, \sum_k W_{ik} = 1, \sum_k D_{jk} = 1, \sum_k S_{kk} = 1. \quad (2)$$

where X is the $m \times n$ word-document semantic matrix, $X = WSD$, W is the word class-conditional probability, and D is the document class-conditional probability and S is the class probability distribution.

PLSI provides a simultaneous solution for the word and document class conditional distribution. Our model provides simultaneous solution for clustering the rows and the columns of X . To avoid ambiguity, we impose the orthogonality condition

$$F^T F = I, G^T G = I. \quad (3)$$

which enforce each row of F and G has only one nonzero entry. This form gives a good framework for simultaneously clustering the rows (words) and columns (documents) of X [8, 28, 17].

2.1 Representing Knowledge in Word Space

The prior knowledge in the word space can be represented as F_0 . This information is incorporated into the unsupervised clustering frame as a constraint

$$\min_{F,G,S} \|X - FSG^T\|^2 + \alpha \|F - F_0\|^2, \quad (4)$$

Where $\alpha > 0$ is a parameter which determines the extent to which we enforce $F \approx F_0$. The constraint ensures that the solution for F in the otherwise unsupervised learning problem be close to the prior knowledge F_0 .

The above model is generic and it allows certain flexibility. For example, in some cases, our prior knowledge on F_0 is not very accurate and we use smaller α so that the final results are not dependent on F_0 very much, i.e., the results are mostly unsupervised learning results.

2.2 Analysis: How Knowledge in Word Space is Transformed into Document Space

Here we give a theoretical analysis to show the effects due to F_0 , the prior knowledge in the word space. For this reason, we as-

sume our knowledge is certain and we set $\alpha \rightarrow \infty$; The optimization simplifies to

$$\min_{G,S} \|X - F_0 S G^T\|^2 \quad (5)$$

THEOREM 1. *The optimization of Eq.(5) with orthogonality constraints $G^T G = I$ $F_0^T F_0 = I$, is identical to the optimization of*

$$\max_G \text{Tr} G^T X^T F_0 F_0^T X G. \quad (6)$$

Proof: $J = \|X - F_0 S G^T\|^2 = \text{Tr}(X^T X - 2F_0^T X G S^T + S S^T)$. Now $0 = \partial J / \partial S \Rightarrow S = F_0^T X G$. Thus $J = \text{Tr}(X^T X - G^T X^T F_0 F_0^T X G)$. $\text{Tr} X^T X$ is a constant and the second term is the desired result. \square

By the K -means and Principle Component Analysis (PCA) equivalence theorem [27, 10], the clustering of Eq.(6) uses $X^T F_0 F_0^T X$ as the pairwise similarity, whereas the standard K -means uses $X^T X$ as the pairwise similarity. For the example of Section 1.1,

$$X^T X = \begin{pmatrix} 2 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \\ 1 & 0 & 2 & 0 \\ 0 & 1 & 0 & 2 \end{pmatrix}$$

and the K -means clustering will produce $(D1, D3)$ as a cluster and $(D2, D4)$ as another cluster.

Now with the knowledge F_0 in the word space,

$$X^T F_0 F_0^T X = \begin{pmatrix} 1 & 1 & \frac{1}{2} & \frac{1}{2} \\ 1 & 1 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 1 & 1 \\ \frac{1}{2} & \frac{1}{2} & 1 & 1 \end{pmatrix}$$

where we have set

$$F_0^T = 2^{-1/2} \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \quad (7)$$

Clearly, using this similarity, K -means clustering will generate $(D1, D2)$ as a cluster and $(D3, D4)$ as another cluster.

We may see more directly how knowledge in the word space is transformed into the document space. Let the square root of the semi-definite positive matrix be P : $F_0 F_0^T = P^T P$. We have $X^T F_0 F_0^T X = (P X)^T (P X)$ which means we cluster the data using the transformed data

$$\tilde{X} = P X = (F_0 F_0^T)^{1/2} X.$$

For the example in Section 1.1,

$$\tilde{X} = 2^{-1/2} \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} X = 2^{-1/2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

It is obvious that on this transformed data, $D1$ and $D2$ will be clustered into one cluster, $D3$ and $D4$ will be clustered into another cluster. This analysis show directly how the knowledge in the word space is transformed into the document space.

2.3 Computational Algorithm

The optimization problem in Eq. (4) can be solved using the following update rules

$$G_{jk} \leftarrow G_{jk} \frac{(X^T F S)_{jk}}{(G G^T X^T F S)_{jk}}, \quad (8)$$

$$S_{ik} \leftarrow S_{ik} \frac{(F^T X G)_{ik}}{(F^T F S G^T G)_{ik}}. \quad (9)$$

$$F_{ik} \leftarrow F_{ik} \frac{(X G S^T + \alpha F_0)_{ik}}{(F F^T X G S^T + \alpha F F^T F_0)_{ik}}. \quad (10)$$

The algorithm consists of an iterative procedure using the above three rules until convergence:

Initialization. Initialize $F = F_0, G$ to K -means clustering results, and $S = [(F^T F)^{-1} F^T X G (F^T F)^{-1}]_+$.

Update G. Fixing F, S , updating G

Update F. Fixing S, G , updating F

Update S. Fixing F, G , update S

For the example given in Section 1.1, using the above algorithm procedure, we initialize $F = F_0$ as in Eq. (7), after convergence, we obtain

$$G = \begin{pmatrix} 0.0031 & 0.6099 \\ 0.0037 & 0.7903 \\ 0.6108 & 0.0005 \\ 0.7896 & 0.0021 \end{pmatrix}, S = \begin{pmatrix} 0.9997 & 0.9969 \\ 1.0017 & 0.0000 \\ 0.0000 & 1.0017 \end{pmatrix}$$

Thus based on $G, D1$ and $D2$ is grouped together into one cluster and $D3$ and $D4$ is grouped together into another cluster. In addition, S clearly shows the association relationships between topic areas and word categories. In the first column of S , large values in the top two entries indicating that topic cluster 1 (e.g., **Vision** area) is associated with word category 1 (e.g., **Learning**) and word category 2 (e.g., **Graphics**). Similarly, **IR** area is associated with **Learning** and **Graphics** categories. However, if we initialize F randomly, after convergence, we obtain

$$G = \begin{pmatrix} 0.7071 & 0.0000 \\ 0.0000 & 0.7071 \\ 0.7071 & 0.0000 \\ 0.0000 & 0.7071 \end{pmatrix}, S = \begin{pmatrix} 1.7321 & 0.0000 \\ 0.0000 & 1.6762 \\ 0.0003 & 0.4363 \end{pmatrix}.$$

We can not obtain good clustering results from G .

2.4 Algorithm Correctness and Convergence

We prove rigorously two theorems on correctness and Convergence of the algorithm:

THEOREM 2. *The above iterative algorithm converge.*

THEOREM 3. *At convergence, the solution satisfies the Karuch, Kuhn, Tucker optimality condition, i.e., the algorithm converged correctly to a local optima.*

The proof of Theorem 2 is given in the Appendix.

Proof of Theorem 3. Following the theory of constrained optimization [21], we introduce the Lagrangian multipliers λ (a symmetric matrix of size $K \times K$) and minimize the Lagrangian function

$$L(F) = \|X - F S G^T\|^2 + \alpha \|F - F_0\|^2 + \text{Tr}[\lambda(F^T F - I)]. \quad (11)$$

Note $\|X - F S G^T\|^2 = \text{Tr}(X^T X - 2F^T X G S^T + S G^T G S^T F^T F)$. The gradient is

$$\frac{\partial L}{\partial F} = -2X G S^T + 2F(S G^T G S^T + \lambda + \alpha) - 2\alpha F_0. \quad (12)$$

The KKT complementarity condition for the non-negativity of F_{ik} gives

$$[-2X G S^T + 2F(S G^T G S^T + \lambda + \alpha) - 2\alpha F_0]_{ik} F_{ik} = 0. \quad (13)$$

This is the fixed point relation that local minima for F must satisfy.

The standard approach is to solve the couple equations Eq.(13) with constraints using a nonlinear method such as Newton’s method. However, this system of nonlinear equations is generally difficult to solve. In this paper, we provide a much simpler algorithm to compute the solution. From the KKT complementarity condition, sum over i , we have

$$(-F^T XGS^T + (SG^T GS^T + \lambda + \alpha) - \alpha F^T F_0)_{kk} = 0 \quad (14)$$

This gives the diagonal elements of the Lagrangian multiplier

$$\lambda_{kk} = (F^T XGS^T + \alpha F^T F_0 - SG^T GS^T - \alpha)_{kk} \quad (15)$$

For non-diagonal elements, we use the gradient zero condition and obtain,

$$\lambda_{kk'} = (F^T XGS^T + \alpha F^T F_0 - SG^T GS^T - \alpha)'_{kk'} \quad (16)$$

Now substitute λ into Eq.(10), the updating rule for F is identical to the KKT condition of Eq.(13). The correctness of updating rules for G in Eq.(8) and S in Eq.(9) have been proved in [11]. \square

3. OTHER FORMS OF KNOWLEDGE IN WORD SPACE

Sometimes our knowledge in the word space are in the form of a set of pairwise relations. For example, we think *algorithm* and *computation* should be in one class, whereas *economy* and *basketball* should be in different classes. More formally, we have two types of pairwise word association constraints [24]: (1) *Must-link word pairs* encoded by a matrix

$$A = \{(i_1, j_1), \dots, (i_a, j_a)\}, a = |A|,$$

containing pairs of words w_{i_1}, w_{j_1} which are considered similar and must be clustered into the same word cluster, and (2) *Cannot-link word pairs* encoded by a matrix

$$B = \{(i_1, j_1), \dots, (i_b, j_b)\}, b = |B|,$$

where each pair of words are considered dissimilar and are not to be clustered into the same word cluster.

We treat them as constraints on the class posterior probability F . A must-link pair (i_1, j_1) implies that the overlap $h_{i_1 k} h_{j_1 k} > 0$ for some class k . Thus $\sum_{k=1}^K h_{i_1 k} h_{j_1 k} = (FF^T)_{i_1 j_1}$ should be maximized. Thus we express the must-link condition as

$$\max_F \sum_{(ij) \in A} (FF^T)_{ij} = \sum_{ij} A_{ij} (FF^T)_{ij} = \text{Tr} F^T A F.$$

For a cannot-link pair (i_2, j_2) , $h_{i_2 k} h_{j_2 k} = 0$ for all k . We treat this as a constraint and minimize $\sum_{k=1}^K h_{i_2 k} h_{j_2 k} = (FF^T)_{i_2 j_2}$, since h_{ik} are nonnegative. We write this condition as

$$\sum_{(ij) \in B} (FF^T)_{ij} = \text{Tr} B F F^T = 0, \quad \text{or} \quad \min_F \text{Tr} F^T B F.$$

Putting these conditions together, we can cast the knowledge transformation model as the following optimization problem

$$\min_{F, G, S} \|X - FSG^T\|^2 + \text{Tr}(-\beta F^T A F + \gamma F^T B F) \quad (17)$$

where β, γ are two positive constants to adjust the strength of the knowledge. A theoretical analysis similar to that of Section 2.2 can be carried out for understanding how knowledge embedded in pairwise relations in A, B can transform into the document space.

The procedure for computing an optimal solution is an iterative algorithm. The updating rules for G, S are the same as in Eqs.(8,9). The updating rules for F is

$$F_{ik} \leftarrow F_{ik} \sqrt{\frac{(XGS^T + \beta A F)_{ik}}{(F(SG^T GS^T + \lambda) + \gamma B F)_{ik}}} \quad (18)$$

where the Lagrangian multiplier k -by- k matrix λ for enforcing the orthogonality $F^T F = I$ is given by

$$\lambda = F^T XGS^T + \beta F^T A F - \gamma F^T B F - SG^T GS^T \quad (19)$$

Similar to Section 2.3, we have

THEOREM 4. *The above iterative algorithm converge.*

THEOREM 5. *At convergence, the solution satisfies the KKT optimality condition, i.e., the algorithm converged correctly to a local optima.*

The proof of Theorem 4 is given in Appendix.

Proof of Theorem 5 We write down the Lagrangian function

$$L_1(F) = \|X - FSG^T\|^2 + \text{Tr}[-\beta F^T A F + \gamma F^T B F + \lambda(F^T F - I)] \quad (20)$$

and obtain the KKT condition for the non-negativity of F :

$$[-2XGS^T + 2F(SG^T GS^T + \lambda) - 2\beta A F + 2\gamma B F]_{ik} F_{ik} = 0 \quad (21)$$

From this, we can obtain the Lagrangian multiplier as in Eq.(19). We can easily see that at convergence, the solution satisfy

$$[-2XGS^T + 2F(SG^T GS^T + \lambda) - 2\beta A F + 2\gamma B F]_{ik} F_{ik}^2 = 0$$

which is identical to the KKT condition: either the first factor is zero, or the F_{ik} is zero. If the first factor is zero, the two equation are identical. If F_{ik} is zero, then F_{ik}^2 is also zero, vice versa. Thus, we have prove that if the iteration converges, the converged solution satisfies the KKT condition, i.e., it converges correctly to a local minima.

4. EXPERIMENTAL RESULTS

4.1 Datasets

We use the following four datasets in our experiments and their characteristics are summarized in Table 1. The Mallet¹ software package is used in our experiments for text processing.

- **DBLP Dataset:** This dataset is obtained from DBLP Computer Science Bibliography². We extract the paper titles published by 552 relatively productive researchers from 9 categories: *database, data mining, software engineering, theory, computer vision, operating system, machine learning, networking, and natural language processing*. For easy comparison purpose, we only consider the publications over the last 20 years (from 1988 to 2007, inclusive). Using the ACM Keywords Taxonomy³, we obtain the category information for terms and use it as the prior knowledge in the word space.

¹It can be downloaded from http://mallet.cs.umass.edu/index.php/Obtaining_MALLET.

²The dblp.xml file is available for download at <http://www.informatik.uni-trier.de/~ley/db/>.

³Available on the page of <http://www.computer.org/portal/pages/ieeecs/publications/author/ACMTaxonomy.html>.

Table 1: Document Data Description

DataSet	# of Instances	# of Instance Clusters	# of Words	# of Word Clusters
DBLP	552	9	1000	11
CSTR	550	4	1000	11
Artist	450	15	601	20
BBS	1309	12	1200	12

- CSTR Dataset:** This dataset contains the abstracts of technical reports (TRs) published in the Department of Computer Science at University of Rochester from 1991 to 2007. There are 550 abstracts and they are divided into four research areas: *Natural Language Processing(NLP)*, *Robotics/Vision*, *Systems*, and *Theory*. We also use the category information of terms obtained from ACM Keywords Taxonomy as prior knowledge.
- Artist Dataset:** The genre and style descriptions of famous artists are publicly available on All Music Guide website (<http://www.allmusic.com>). We collect the information of 2431 artists who have both the genre and style descriptions from this website. There are altogether 15 genres (such as Jazz, Rock, Country), and 601 style terms (nouns like Electric Chicago Blues, Greek Folk, and Chinese Pop, as well as adjectives like Joyous, Energetic, and New Romantic). These style terms are classified into 20 categories by domain experts. Each artist is represented as a word vector using the style terms and we cluster the artists into different groups. The genre information is treated as the ground truth for artist clusters and the style category is used as additional information in the word space.
- BBS Dataset:** This is a dataset sampled from the *Bulletin Board Systems (BBS)* data in [15]. A BBS system contains many boards with similar themes. The boards are named to reflect the contents of the articles contained in them [15]. Once a user posts an initial article on a board, the others can show their opinions using reply articles. The initial article and reply articles constitute a topic. People’s behaviors on the BBS usually reflect their interests. For example, people who post articles in the same topic may share similar interests, and people who are interested in the same boards or discussion fields may have something in common (e.g., similar background and education level) [25]. Here we cluster users into groups based on their activities. The user-topic matrix was constructed with the articles each user posted in each topics with *TF-IDF* normalization [1]. The topics are organized into different boards and we use the categories of the topics as external information for clustering users. The board on which a user is most active (e.g., a user has posted the largest number of articles in this board among all boards) is treated as the ground truth for the user clusters.

4.2 Evaluation Measures

To measure the clustering performance, we use accuracy and normalized mutual information as our performance measures. Accuracy discovers the one-to-one relationship between clusters and classes and measures the extent to which each cluster contained data points from the corresponding class. It sums up the whole matching degree between all pair class-clusters. Its value is between $[0, 1]$. Accuracy can be represented as:

$$ACC = \max(\sum_{C_i, L_j} T(C_i, L_j)) / N, \quad (22)$$

where C_i denotes the i -th cluster, and L_j is the j -th class. $T(C_i, L_j)$ is the number of entities which belong to class j are assigned to cluster i . Accuracy computes the maximum sum of $T(C_i, L_j)$ for all pairs of clusters and classes, and these pairs have no overlaps [11]. Generally, the greater accuracy means the better clustering performance.

Normalized mutual information (NMI) is another widely used performance evaluation measure for determining the quality of clusters [23]. For two random variables \mathbf{X} and \mathbf{Y} , the NMI is defined as

$$NMI(\mathbf{X}, \mathbf{Y}) = \frac{I(\mathbf{X}, \mathbf{Y})}{\sqrt{H(\mathbf{X})H(\mathbf{Y})}}, \quad (23)$$

where $I(\mathbf{X}, \mathbf{Y})$ is the mutual information between \mathbf{X} and \mathbf{Y} , and $H(\mathbf{X})$ and $H(\mathbf{Y})$ are the entropies of \mathbf{X} and \mathbf{Y} , respectively. Clearly, $NMI(\mathbf{X}, \mathbf{X}) = 1$ and this is the maximum possible value of NMI. Given a clustering result, NMI in Eq.(23) is estimated as follows:

$$NMI = \frac{\sum_{i=1}^k \sum_{j=1}^k n_{ij} \log(\frac{n_{ij}}{n_i \hat{n}_j})}{\sqrt{(\sum_{i=1}^k n_i \log \frac{n_i}{n})(\sum_{j=1}^k \hat{n}_j \log \frac{\hat{n}_j}{n})}} \quad (24)$$

where n_i denotes the number of data points contained in the cluster $C_i (1 \leq i \leq k)$, \hat{n}_j is the number of data points belonging to the j -th class ($1 \leq j \leq k$), and n_{ij} denotes the number of data points that are in the intersection between the cluster C_i and the j -th class. In general, the larger the NMI value, the better the clustering quality.

4.3 Results Analysis

4.3.1 Experiments Using Prior Knowledge

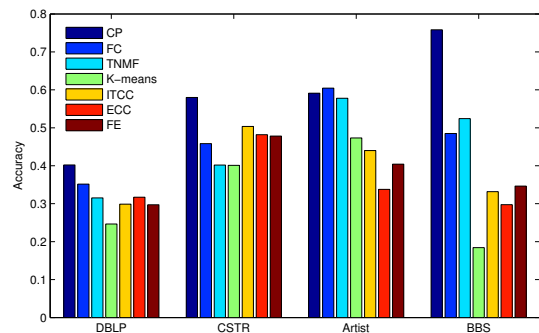


Figure 2: Accuracy results on four datasets

We denote our method utilizing prior knowledge as CP (Clustering with Prior knowledge). In our experiments, we compare our method (CP) with the following methods:

- Four document clustering methods: K-means, Tri-Factor Non-negative Matrix Factorization (TNMF) [11], Information-Theoretic

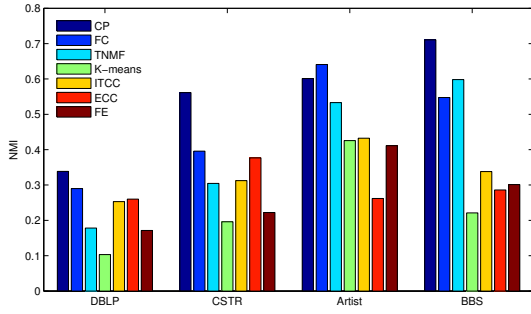


Figure 3: NMI results on four datasets

Co-clustering (ITCC) [9], and Euclidean Co-clustering algorithm (ECC) [5]. These methods do not make use of knowledge in the word space.

- Two simple methods of incorporating prior knowledge. (1) Feature Ensemble (FE): In this approach, we divide the word space into different classes (using prior knowledge) and perform K-means clustering on documents based on each individual word group. Then an ensemble clustering algorithm is applied to obtain a final document clustering results [23]. (2) Feature Centroid (FC): In this approach, we replace each word (e.g., each row of the word-document matrix) using the corresponding category centroid and then perform K-means clustering on the documents. In other words, knowledge in the word space is utilized to perform a dimensionality reduction here.

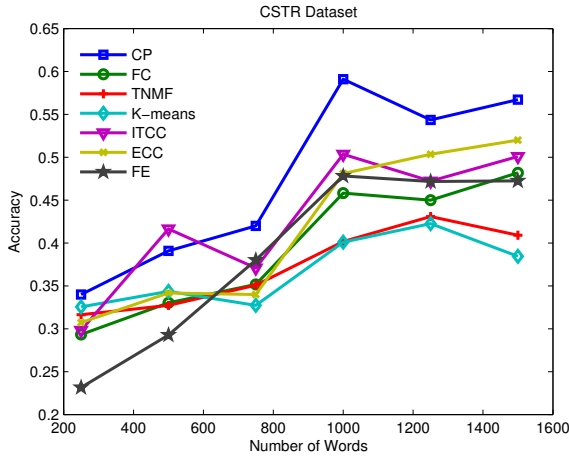


Figure 4: Accuracy results with different numbers of words on CSTR dataset.

Figure 2 shows the experimental results on four datasets using accuracy as the performance measure, and Figure 3 present the NMI results. The results are obtained by averaging 20 runs. From the experimental comparisons, we observe that:

- Prior knowledge in the word space improves clustering results. Our proposed method CP achieves the highest performance on three datasets: DBLP, CSTR and BBS while FC achieves the highest performance on Artist.

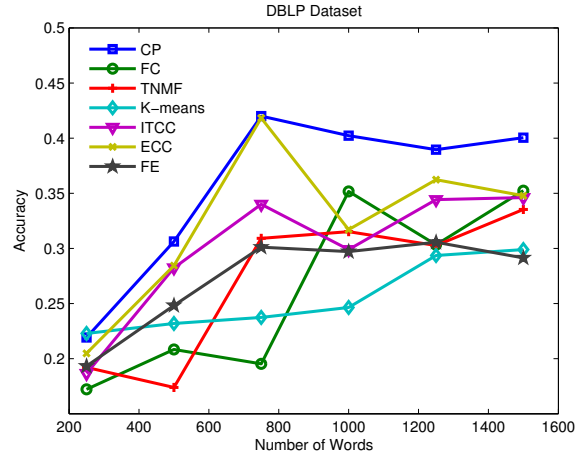


Figure 5: Accuracy results with different numbers of words on DBLP dataset.

- Our proposed method CP can effectively transfer knowledge from the word space to the document space for improving clustering results. On all datasets, the performance of CP is always better than the performances of those methods that do not make use of prior knowledge. In fact, on four datasets, CP ranks first 3 times (it actually outperforms all other methods by a large margin on the three datasets).

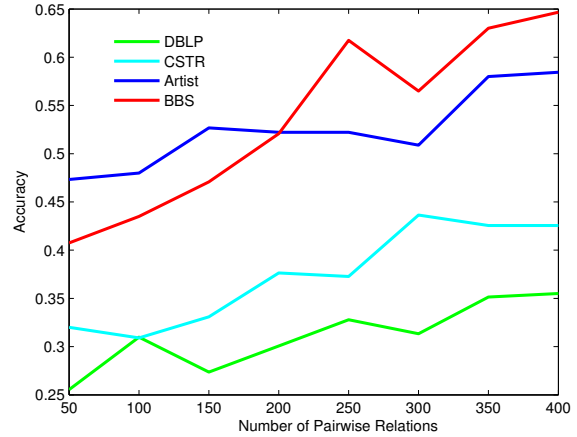


Figure 6: Accuracy results with different numbers of pairwise word relations

4.3.2 Effects of Number of Words

In this section, we perform experiments on DBLP dataset and CSTR dataset to investigate the effects of the size of word space on clustering performance. The word selection is performed using Mallet toolkit with information gain criteria. Figure 4 and Figure 5 show the accuracy results with different numbers of selected words on CSTR dataset and DBLP dataset, respectively. From Figure 4 on CSTR dataset, our proposed method (CP) outperforms all other methods at different word sizes (except one case at 500 words). From Figure 5 on DBLP dataset, CP outperforms all other methods. The two naive ways for incorporating the word space knowl-

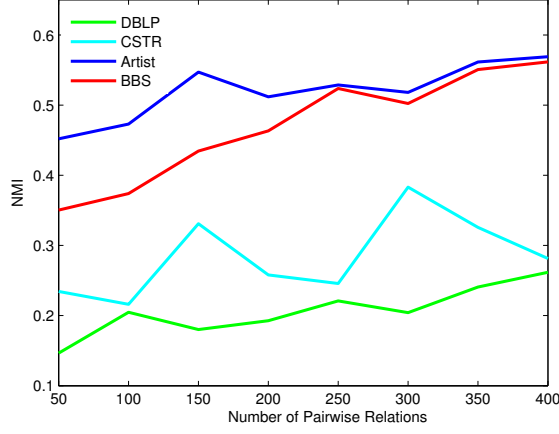


Figure 7: NMI results with different numbers of pairwise word relations

edge, FE and CE, do not perform well, due to the unsophisticated clustering methods used. Similar behavior are also observed for the NMI measure (results are not shown due to lack of space).

4.3.3 Experiments Using Pairwise Relations

In this section, we perform experiments when prior knowledge is in the form of pairwise relations. We use the algorithm presented in Section 3 to transform the pairwise relations in the word space into the document space. The relations are generated as follows: we pick out a pair of words randomly from the word space (the category information of which are available). If this pair of words are in the same word category, then we generate a must-link relation. If they are in different categories, a cannot-link relation is then generated. Figure 6 and Figure 7 present the experimental results of our algorithm with different number of pairwise relations in the word space. In all the experiments, the results are averaged over 20 trials. As you can observe from these Figures, the clustering performance generally improves as the number of pairwise relations increases. The experimental results confirms the ability of our proposed algorithm for transferring the pairwise relations in the word space into the document space which improves clustering results.

5. CONCLUSION

In this paper, we provide a model for enabling knowledge transformation from the word space to the document space. Two forms of prior knowledge in the word space (categorization of words and pairwise relations between words) are presented, which can be effectively incorporated into the model and transformed into knowledge in the document space. We give detailed theoretical analysis of the model and propose computational algorithms with rigorous proofs of their correctness and convergence. We experiment on 4 real-world datasets and compare with six other methods. The results show our proposed method consistently outperform other methods. Our model can be applied to any two-way data to effectively transform knowledge from one side to another side. For example, on DNA microarray data, our model can transform knowledge on the gene side (rows of the data matrix) to knowledge on the patient tissue sample side (columns of the data matrix).

Acknowledgments

The project is partially supported by NSF CAREER Award IIS-0546280 and IBM Faculty Research Awards.

6. APPENDIX

6.1 Proof of Theorem 2 in Section 2.3

We use the auxiliary function approach [16]. A function $Z(F, \tilde{F})$ is called an auxiliary function of $L(F)$ if it satisfies

$$Z(F, \tilde{F}) \geq L(F), \quad Z(F, F) = L(F), \quad (25)$$

for any F, \tilde{F} . Define

$$F^{(t+1)} = \arg \min_F Z(F, F^{(t)}). \quad (26)$$

By construction, $L(F^{(t)}) = Z(F^{(t)}, F^{(t)}) \geq Z(F^{(t+1)}, F^{(t)}) \geq L(F^{(t+1)})$. Thus $L(F^{(t)})$ is monotonic decreasing (non-increasing). The key is to find appropriate $Z(F, \tilde{F})$ and its global minima.

We write L of Eq.(11) as

$$L(F) = \text{Tr}[-2F^T XGS^T + (SG^T GS^T + \lambda + \alpha)F^T F - 2\alpha F^T F_0],$$

where we ignore the constraints $X^T X$, $\text{Tr}\lambda$, and $\text{Tr}(F_0^T F_0)$. Now we show that the following function

$$\begin{aligned} Z(F, F') &= -\sum_{ik} 2(F^T XGS^T)_{ik} \\ &+ \sum_{ik} \frac{(F'(SG^T GS^T + \lambda + \alpha))_{ik} F_{ik}^2}{F'_{ik}} - 2\alpha(F^T F_0)_{ik} \end{aligned}$$

is an auxiliary function of $L(F)$. First, it is obvious that when $F' = F$ the equality holds $Z(F, F') = L(F)$. Second, the inequality holds $Z(F, F') \geq L(F)$, because: the second term in $Z(F, F')$ is always bigger than the second term in $L(F)$, due to an inequality

$$\sum_{ik} (AF^T F)_{ik} \leq \sum_{ik} \frac{(F'A)_{ik} F_{ik}^2}{F'_{ik}}$$

for any $A, F, F' \geq 0$ (We skip the proof due to lack of space). Thus the conditions of Eq.(25) are satisfied.

Now according to Eq.(26), we need to find the *global* minimum of $f(F) = Z(F, F')$ fixing F' . We first compute the gradient

$$\begin{aligned} \frac{\partial Z(F, F')}{\partial F_{ik}} &= -2(XGS^T)_{ik} + 2 \frac{[F'(SG^T GS^T + \lambda + \alpha)]_{ik} F_{ik}}{F'_{ik}} - 2\alpha(F_0)_{ik}. \end{aligned} \quad (27)$$

and the Hessian (2nd order derivatives) are

$$\frac{\partial^2 Z}{\partial F_{ik} \partial F_{jk'}} = 2 \frac{[F'(SG^T GS^T + \lambda + \alpha)]_{ik} \delta_{ij} \delta_{kk'}}{F'_{ik}} \quad (28)$$

Thus the Hessian is semi-positive definite, i.e., $Z(F, F')$ is a convex function. It's global minima is obtained by setting $\partial Z / \partial G_{ik} = 0$, we obtain

$$F_{ik} = F'_{ik} \frac{(XGS^T + \alpha F_0)_{ik}}{[F'(SG^T GS^T + \lambda + \alpha)]_{ik}} \quad (29)$$

Now according to Eq.(26), $F^{(t+1)} = F$ and $F' = F^{(t)}$, we recover Eq.(10).

6.2 Proof of Theorem 4 in Section 3

First we find the auxiliary function of $L_1(F)$ in Eq.(20). We can show that the following function

$$Z_1(F, F') =$$

$$\sum_{ik} [-2F'_{ik}(1 + \log \frac{F_{ik}}{F'_{ik}})(XGS^T)_{ik} + \frac{(F'(SG^T GS^T + \lambda))_{ik} F_{ik}^2}{F'_{ik}}] \quad (30)$$

$$+ \beta F'_{ik}(AF')_{ik}(1 + \log \frac{F_{ik} F_{jk}}{F'_{ik} F'_{jk}}) + \gamma \frac{(BF')_{ik} F_{ik}^2}{F'_{ik}}] \quad (31)$$

is an auxiliary function, ignoring the constant terms. Now, the gradient is

$$\frac{\partial Z}{\partial F_{ik}} = -2 \frac{F'_{ik}}{F_{ik}} (XGS^T)_{ik} + 2 \frac{[F'(SG^T GS^T + \lambda)]_{ik} F_{ik}}{F'_{ik}} \quad (32)$$

$$- 2\beta \frac{F'_{ik}(AF')_{ik}}{F_{ik}} + 2\gamma \frac{(BF')_{ik} F_{ik}}{F'_{ik}} \quad (33)$$

and the Hessian is

$$\frac{\partial^2 Z}{\partial F_{ik} \partial F_{jk'}} = [2 \frac{F'_{ik}}{F_{ik}^2} (XGS^T)_{ik} + 2 \frac{[F'(SG^T GS^T + \lambda)]_{ik}}{F'_{ik}}] \quad (34)$$

$$2\beta \frac{F'_{ik}(AF')_{ik}}{F_{ik}^2} + 2\gamma \frac{(BF')_{ik}}{F'_{ik}}] \delta_{ij} \delta_{kk'} \quad (35)$$

Thus the Hessian is semi-positive definite, i.e., $Z(F, F')$ is a convex function. It's global minima is obtained by setting $\partial Z / \partial G_{ik} = 0$, we obtain

$$\frac{F'_{ik}}{F_{ik}} (XGS^T + \beta AF')_{ik} = \frac{F_{ik}}{F'_{ik}} (F'(SG^T GS^T + \lambda + \gamma BF'))_{ik} \quad (36)$$

or

$$F_{ik}^2 = F_{ik}'^2 \frac{(XGS^T + \beta AF')_{ik}}{(F'(SG^T GS^T + \lambda) + \gamma BF')_{ik}} \quad (37)$$

Now according to Eq.(26), $F^{(t+1)} = F$ and $F' = F^{(t)}$, we recover

$$F_{ik} \leftarrow F_{ik} \sqrt{\frac{(XGS^T + \beta AF)_{ik}}{(F(SG^T GS^T + \lambda) + \gamma BF)_{ik}}} \quad (38)$$

7. REFERENCES

- [1] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley Longman, 1999.
- [2] S. Basu, M. Bilenko, and R. J. Mooney. A probabilistic framework for semi-supervised clustering. In *Proceedings of ACM SIGKDD*, pages 59–68, 2004.
- [3] P. Berkhin. Survey of clustering data mining techniques. Technical report, Accrue Software, San Jose, CA, 2002.
- [4] M. Bilenko, S. Basu, and R. Mooney. Integrating constraints and metric learning in semi-supervised clustering. *Proc. Int'l Conf. Machine Learning (ICML2004)*, 2004.
- [5] H. Cho, I. Dhillon, Y. Guan, and S. Sra. Minimum sum squared residue co-clustering of gene expression data. In *Proceedings of The 4th SIAM Data Mining Conference*, pages 22–24, April 2004.
- [6] D. Cohn, R. Caruana, and A. McCallum. Semi-supervised clustering with user feedback. Technical Report TR2003-1892, Cornell University, 2003.
- [7] I. Davidson and S. Ravi. Clustering under constraints: Feasibility results and the k-means algorithm. In *Proceedings of SIAM Data Mining Conference*, 2005.
- [8] I. S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. *Proceeding of ACM SIGKDD*, 2001.
- [9] I. S. Dhillon, S. Mallela, and D. S. Modha. Information-theoretical co-clustering. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'03)*, pages 89–98, 2003.
- [10] C. Ding and X. He. K-means clustering and principal component analysis. *Int'l Conf. Machine Learning (ICML)*, 2004.
- [11] C. Ding, T. Li, W. Peng, and H. Park. Orthogonal nonnegative matrix tri-factorizations for clustering. In *Proceedings of ACM SIGKDD*, pages 126–135, 2006.
- [12] J. Hartigan. *Clustering Algorithms*. Wiley, 1975.
- [13] T. Hofmann. Probabilistic latent semantic indexing. *Proc. ACM Conf. on Research and Develop. IR (SIGIR)*, pages 50–57, 1999.
- [14] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.
- [15] Z. Kou and C. Zhang. Reply networks on a bulletin board system. *Phys. Rev. E*, (67), 2003.
- [16] D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems 13*, Cambridge, MA, 2001. MIT Press.
- [17] T. Li. A general model for clustering binary data. In *KDD*, pages 188–197, 2005.
- [18] T. Li and C. Ding. The relationships among various nonnegative matrix factorization methods for clustering. In *Proceedings of the 2006 IEEE International Conference on Data Mining (ICDM 2006)*, pages 362–371, 2006.
- [19] T. Li, C. Ding, and M. Jordan. Solving consensus and semi-supervised clustering problems using nonnegative matrix factorization. In *Proceedings of the 2007 IEEE International Conference on Data Mining (ICDM 2007)*, pages 577–582, 2007.
- [20] B. Long, X. Wu, Z. M. Zhang, and P. S. Yu. Unsupervised learning on k-partite graphs. In *Proceedings of ACM SIGKDD*, pages 317–326, 2006.
- [21] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer-Verlag, 1999.
- [22] N. Slonim and N. Tishby. Document clustering using word clusters via the information bottleneck method. In *SIGIR*, pages 208–215, 2000.
- [23] A. Strehl and J. Ghosh. Cluster ensembles - a knowledge reuse framework for combining multiple partitions. *The Journal of Machine Learning Research*, 3:583–617, March 2003.
- [24] K. Wagstaff, C. Cardie, S. Rogers, and S. Schroedl. Constrained k-means clustering with background knowledge. *ICML*, 2001.
- [25] F. Wang, T. Li, and C. Zhang. Semi-supervised learning via matrix factorization. In *Proceedings of 2008 SIAM International Conference on Data Mining*, 2008.
- [26] E. Xing, A. Ng, M. Jordan, and S. Russell. Distance metric learning, with application to clustering with side-information. *NIPS*, 2002.
- [27] H. Zha, C. Ding, M. Gu, X. He, and H. Simon. Spectral relaxation for K-means clustering. *NIPS*, pages 1057–1064, 2002.
- [28] H. Zha, X. He, C. Ding, M. Gu, and H. Simon. Bipartite graph partitioning and data clustering. *CIKM*, 2001.