PROXIMITY-BASED GRAPH EMBEDDINGS FOR MULTI-LABEL CLASSIFICATION

Tingting Mu and Sophia Ananiadou
National Centre for Text Mining, University of Manchester, 131 Princess Street, Manchester, M1 7DN, U.K.

Keywords: Dimensionality reduction, Embedding, Supervised, Adjacency graph, Multi-label classification.

Abstract: In many real applications of text mining, information retrieval and natural language processing, large-scale features are frequently used, which often make the employed machine learning algorithms intractable, leading to the well-known problem "curse of dimensionality". Aiming at not only removing the redundant information from the original features but also improving their discriminating ability, we present a novel approach on supervised generation of low-dimensional, proximity-based, graph embeddings to facilitate multi-label classification. The optimal embeddings are computed from a supervised adjacency graph, called multi-label graph, which simultaneously preserves proximity structures between samples constructed based on feature and multi-label class information. We propose different ways to obtain this multi-label graph, by either working in a binary label space or a projected real label space. To reduce the training cost in the dimensionality reduction procedure caused by large-scale features, a smaller set of relation features between each sample and a set of representative prototypes are employed. The effectiveness of our proposed method is demonstrated with two document collections for text categorization based on the "bag of words" model.

1 INTRODUCTION

In information retrieval (IR), text mining (TM) and natural language processing (NLP), research on how to automatically generate a small set of informative features from large-scale features, such as bag of n-grams, are of increasing interest. The goal is not only to reduce the computational cost but also to improve the performance of a followed learning task, which corresponds to the significant problem of dimensionality reduction (DR) in machine learning. Relevant reduction techniques commonly used by IR, TM and NLP researchers include feature selection using wrapper or filter models (Lewis, 1992; Bekkerman et al., 2003; Li et al., 2009), feature clustering (Bekkerman et al., 2003; Dhillon et al., 2003), and latent variable models (Deerwester et al., 1990; Blei et al., 2003).

More sophisticated research for DR has been developed via manifold learning, multidimensional scaling and spectral analysis. These methods generate low-dimensional embeddings so that they preserve certain properties of the original high-dimensional data. Different properties are usually quantified by different objective functions, and the DR problem can thus be formulated as an optimization problem (Kokiopoulou and Saad, 2007). For instance, principal component analysis (PCA) (Jolliffe, 1986) preserves the global structure of the data by maximizing the variance of the projected embeddings. Locally linear embedding (LLE) (Roweis and Saul, 2000) and orthogonal neighborhood preserving projections (ONPP) (Kokiopoulou and Saad, 2007) preserve the intrinsic geometry at each neighborhood by minimizing a reconstruction error. Spectral clustering (SC) analysis (Chan et al., 1994; Shi and Malik, 2000; Luxburg, 2007), Laplacian eigenmaps (LE) (Belkin and Niyogi, 2003), locality preserving projection (LPP) (He and Niyogi, 2003), and orthogonal LPP (OLPP) (Kokiopoulou and Saad, 2007) preserve a certain affinity graph constructed from the original data by minimizing the penalized distances between the embeddings of adjacent points. These methods work in an unsupervised manner, which only preserve the data property in the feature space. Although the unsupervised reduction provides a compact representation of the data, when it is used as a preprocessing step followed by a classification task, it may not always improve the final performance.

When there is extra label (class, category) information available, it is natural to pursue supervised/semi-supervised DR to improve the classification performance. Various DR research has been

Mu T. and Ananiadou S. (2010).
PROXIMITY-BASED GRAPH EMBEDDINGS FOR MULTI-LABEL CLASSIFICATION.
In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval*, pages 74-84
DOI: 10.5220/0003092200740084

conducted for single-label classification task, where each given sample belongs to only one class (He, 2004; Cai et al., 2007a; Yan et al., 2007; Zhang et al., 2007; Kokiopouloua and Saadb, 2009; Sugiyama, 2007; Sugiyama, 2010). Among these methods, Fisher discriminant analysis (FDA) (Fisher, 1936) is the most popular one, which maximizes the between-class scatter while minimizes the within-class scatter of the projected embeddings. These methods work in a similar way of minimizing the penalized distances between the adjacent embeddings, of which the only difference lies on the construction of an adjacency graph and a constraint matrix. Single-label graphs are employed in above methods, where the adjacency is non-zero only when the two points belong to the same class.

Recently, multi-label classification becomes a requirement in NLP, TM and bioinformatics, such as text categorization (Zhang et al., 2008; Tang et al., 2009) and protein function prediction (Barutcuoglu et al., 2006). It allows the given samples to belong to multiple classes. In this case, the above single-label DR methods become inapplicable as there is no clear definition of two samples belonging to the same class, e.g. some of the classes two samples belong to are the same, but not all. Thus, to perform supervised/semi-supervised DR for multi-label classification, one needs to avoid to incorporate such a definition into the computation. Instead, some existing research focuses on construction of different optimization objective functions other than the pernalized distances between intra-class samples in the embedded space, e.g. the reconstruction error of both features and labels (Yu et al., 2006), correlation (Hardoon et al., 2004), independence (Zhang and Zhou, 2007) and mutual information (HildII et al., 2006) between the embeddings and multiple labels. Different from these, a hyper-graph is used to model the multi-label information, and the method replaces the standard Laplacian of LPP with a hyper-graph Laplacian (Sun et al., 2008).

In this paper, we show that, to achieve supervised DR for multi-label classification, one does not need to construct a new optimization objective function, but the penalized distances as used by many existing DR methods (Chan et al., 1994; Shi and Malik, 2000; Luxburg, 2007; Belkin and Niyogi, 2003; He and Niyogi, 2003; Kokiopoulou and Saad, 2007; Fisher, 1936; Yan et al., 2007). Also, to model the multi-label information, it is not necessary to use a hypergraph, but simply a binary label matrix. Multi-label information can be appropriately modelled by discovering the proximity structure between samples in a space spanned by label vectors. Then, supervised embeddings can be computed by using penalizing weights obtained from both label-based and feature-based proximity information. We propose different ways to capture the intrinsic proximity structure based on the multi-label class information, leading to the label-based adjacent graph $\mathbf{W}_Y$. It is then linearly combined with another adjacent graph $\mathbf{W}_X$ representing the geometric structure of features. We also investigate mitigation of the high training cost normally associated with a DR algorithm caused by large number of features. To deal with large-scale features and comparatively large number of training samples, we generate a small set of representative prototypes to compute a set of similarity (or dissimilarity) features (termed as relation features) between each input sample and these prototypes. These new relation features will then be used to generate the embeddings.

2 GRAPH EMBEDDINGS

Given a set of data points $\{\boldsymbol{x}_i\}_{i=1}^n$ of dimension d, where $\boldsymbol{x}_i = [x_{i1}, x_{i2}, \ldots, x_{id}]^T$, the goal of DR is to generate a set of optimal embeddings $\{\boldsymbol{z}_i\}_{i=1}^n$ of dimension k ($k \ll d$), where $\boldsymbol{z}_i = [z_{i1}, z_{i1}, \ldots, z_{ik}]^T$, so that the transformed $n \times k$ feature matrix $\mathbf{Z} = [z_{ij}]$ is an accurate representation of the original $n \times d$ feature matrix $\mathbf{X} = [x_{ij}]$, or with improved discriminating power.

2.1 Framework

A graph embedding framework has been proposed as a general platform for developing new DR algorithms (Yan et al., 2007). It minimizes the penalized distances between the embeddings:

$$\min \frac{1}{2} \sum_{i,j=1}^{n} w_{ij} \|\boldsymbol{z}_i - \boldsymbol{z}_j\|_2^2, \tag{1}$$

under the constraint $\mathbf{Z}^T\mathbf{B}\mathbf{Z} = \mathbf{I}_{k\times k}$, where w_{ij} is a weight value to define the degree of "similarity" or "closeness" between the i-th and j-th samples, and $\mathbf{B}$ is an $n \times n$ constraint matrix. Letting $\mathbf{W} = [w_{ij}]$ denote the $n \times n$ symmetric weight matrix, and $\mathbf{D}(\mathbf{W})$ is a diagonal matrix formed by the vector $\mathbf{W} \times \mathbf{1}_{n\times 1}$, Eq. (1) can be rewritten as

$$\min_{\substack{\mathbf{Z}\in R^{n\times k}, \\ \mathbf{Z}^T\mathbf{B}\mathbf{Z}=\mathbf{I}_{k\times k}}} \operatorname{tr}[\mathbf{Z}^T(\mathbf{D}(\mathbf{W}) - \mathbf{W})\mathbf{Z}], \tag{2}$$

of which the output is termed as graph embeddings. Different algorithms define different weight and constraint matrices. The SC analysis in (Luxburg, 2007), called unnormalized SC (USC), employs an identity

matrix as the constraint matrix: $\mathbf{B} = \mathbf{I}_{n\times n}$. The LE and the SC analysis in (Shi and Malik, 2000), called normalized SC (NSC), employs the degree matrix $\mathbf{D}(\mathbf{W})$ as the constraint matrix: $\mathbf{B} = \mathbf{D}(\mathbf{W})$. For these methods, the used weight matrices are determined by a feature-based adjacency graph, which can be constructed by different ways as described in Section 2.3. The optimal solution of Eq. (1) is denoted by $\mathbf{Z}^*$, which is the top k eigenvectors of the generalized eigenvalue problem $(\mathbf{D}(\mathbf{W}) - \mathbf{W})\mathbf{Z}^* = \mathbf{B}\mathbf{Z}^*\mathbf{S}$, corresponding to the k smallest non-zero eigenvalues.

2.2 Out-of-sample Extension

The methods that can be expressed in Eq. (2) only generate embeddings for the n input (training) samples. However, given a different set of m query samples with an $m \times d$ feature matrix $\tilde{\mathbf{X}}$, it is not straightforward to compute the embeddings of new query samples because of the difficulty in recomputing the eigenvector. Various research has been developed on how to formulate the out-of-sample extension (Bengio et al., 2003; Cai et al., 2007a). Since such extension is necessary for DR to facilitate a classification task, we provide in the following the most commonly used extension and another alternative based on least squares model, both using projection technique that assumes the embeddings are linear combinations of the original features, given as $\mathbf{Z} = \mathbf{X}\mathbf{P}$.

2.2.1 Extension 1

The most commonly used way to achieve out-of-sample extension is to directly incorporate $\mathbf{Z} = \mathbf{X}\mathbf{P}$ into Eq. (2), and thus, a set of optimal projections $\mathbf{P}^*$ are obtained by solving the following generalized eigenvalue problem:

$$\mathbf{X}(\mathbf{D}(\mathbf{W}) - \mathbf{W})\mathbf{X}^T\mathbf{P}^* = \mathbf{X}\mathbf{B}\mathbf{X}^T\mathbf{P}^*\mathbf{S}. \quad (3)$$

The embeddings are then computed by $\mathbf{Z} = \mathbf{X}\mathbf{P}^*$ for the training samples, and $\tilde{\mathbf{Z}} = \tilde{\mathbf{X}}\mathbf{P}^*$ for the query samples. LE with such an extension leads to LPP. OLPP imposes the orthogonality condition to the projection matrix, of which the optimal projections are the top k eigenvectors of the matrix $\mathbf{X}(\mathbf{D} - \mathbf{W})\mathbf{X}^T$, corresponding to the k smallest non-zero eigenvalues.

2.2.2 Extension 2

An alternative to achieve out-of-sample extension is to minimize the reconstruction error (Cai et al., 2007a) between the projected features and the computed embeddings $\mathbf{Z}^*$ with a regularization term after solving Eq. (2):

$$\min_{\mathbf{\Lambda}\in R^{d\times k}} \|\mathbf{X}\mathbf{P} - \mathbf{Z}^*\|_{\mathrm{F}}^2 + \alpha\|\mathbf{P}\|_{\mathrm{F}}^2, \quad (4)$$

where $\alpha > 0$ is a user-defined regularization parameter. The optimal least squares solution of Eq. (4) is

$$\mathbf{P}^* = (\mathbf{X}^T\mathbf{X} + \alpha\mathbf{I}_{d\times d})^{-1}\mathbf{X}^T\mathbf{Z}^*. \quad (5)$$

Then, the embeddings of the new query sample can be approximated by $\tilde{\mathbf{Z}} = \tilde{\mathbf{X}}\mathbf{P}^*$.

2.3 Feature-based Adjacency Graph

The embeddings obtained by Eq. (2) preserve the proximity structure between samples in the original feature space. Such a proximity structure is captured by the weight matrix $\mathbf{W} = [w_{ij}]$ of a feature-based adjacency graph, where w_{ij} is non-zero only for adjacent nodes in the graph. There are two principal ways to define the adjacency: (1) whether two samples are the K-nearest neighbors (KNN) of each other; and (2) whether a certain "closeness" measure between two samples is within a given range. There are also different ways to define the weight matrix: (1) Constant value, where $w_{ij} = 1$ if the i-th and j-th samples are adjacent, while $w_{ij} = 0$ otherwise. (2) Gaussian kernel (Belkin and Niyogi, 2003; He and Niyogi, 2003), where $w_{ij} = \exp\left(\frac{-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{\tau}\right)$, and $\tau > 0$. (3) Domain-dependent similarity matrix between the samples (Dhillon, 2001). (4) The optimal affinity matrix in LLE computed by minimizing the reconstruction error between each sample and its KNNs (Roweis and Saul, 2000). All these computations are unsupervised, which only compute $\mathbf{W}$ from the feature matrix $\mathbf{X}$ and preserve the geometric structure of the features.

2.4 Single-label Adjacency Graph

In content-based image retrieval, to find better image representation, additional label information (relevance feedbacks) is employed to construct a supervised (or semi-supervised with partial label information) affinity graph (He, 2004; Yu and Tian, 2006; Cai et al., 2007a). In an incremental version of LPP (He, 2004) and a supervised version of ONPP (Kokiopoulou and Saad, 2007), a binary labeled data graph is used, that defines the following weight matrix:

$$w_{ij} = \begin{cases} 1 & \text{if } \boldsymbol{x}_i \text{ and } \boldsymbol{x}_j \text{ belong to the same class,} \\ 0 & \text{otherwise.} \end{cases} \quad (6)$$

Such a weight matrix can be further scaled by sizes of different classes:

$$w_{ij} = \begin{cases} \frac{1}{n_s} & \text{if } \boldsymbol{x}_i \text{ and } \boldsymbol{x}_j \text{ belong to the } s\text{th class,} \\ 0 & \text{otherwise,} \end{cases} \quad (7)$$

where n_s denotes the number of training samples belonging to the s-th class (He, 2004; Cai et al., 2007a; Cai et al., 2007b). With Eq. (7), minimizing the pernalized distances between embeddings is equivalent to minimizing the within-class scatter of Fisher criterion (He et al., 2005; Yan et al., 2007). By incorporating the local data structure into FDA, the weight matrix of the local FDA (Sugiyama, 2007) is given by

$$w_{ij} = \begin{cases} \frac{l_{ij}}{n_s} & \text{if } \boldsymbol{x}_i \text{ and } \boldsymbol{x}_j \text{ belong to the } s\text{th class,} \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

By updating the local neighborhood weight matrix with partial label information, the following weight matrix is used for semi-supervised DR (He, 2004; Cai et al., 2007a):

$$w_{ij} = \begin{cases} 1 & \text{if } \boldsymbol{x}_i \text{ and } \boldsymbol{x}_j \text{ belong to the same class} \\ 0 & \text{if } \boldsymbol{x}_i \text{ and } \boldsymbol{x}_j \text{ belong to different classes,} \\ w'_{ij} & \text{if there is no label information,} \end{cases} \tag{9}$$

where w'_{ij} is the weight of a feature-based adjacency graph as discussed in Section 2.3. These methods model the label information by simply considering whether two samples are from the same class. This is unsuitable for multi-label classification, since two samples may share some but not all labels.

3 PROPOSED METHOD

Given a classification dataset of c different classes (categories), we model the class (target) information of the training samples as an $n \times c$ label matrix: $\mathbf{Y} = [y_{ij}] \in \{0,1\}^{n \times c}$, $y_{ij} = 1$ if the i-th sample belongs to the j-th class t_j, and $y_{ij} = 0$ otherwise. The label information is the desired output of the input samples, while the feature information is extracted from the samples so that it can represent the characteristics distinguishing different types of desired outputs. In the original feature space R^d, proximity structures between samples are captured by different adjacency graphs as discussed in Section 2.3. There also exist such structures in the label space $\{0,1\}^c$. Ideally, if the features can accurately describe all the discriminating characteristics, the proximity structures computed from the features and labels should be very similar. However, when processing real dataset, what may happen is that, in the original feature space, the data points that are close to each other may belong to different classes, while on the contrary, the data points that are in a distant to each other may belong to the same class. This subsequently leads to incompatible proximity structures in the feature and label spaces, and thus unsatisfactory classification performance. Aiming at generating a set of embeddings with improved discriminating ability for multi-label classification, we decide to modify the proximity structure of the embedded features based on the label information. This leads to two research issues: (1) how to capture the proximity structure in the label space, (2) how to combine the label-based and feature-based proximity structures.

3.1 Multi-label Adjacency Graph

To model the proximity structure in the multi-label space, our basic idea is to construct an adjacent graph denoted by $G_Y(V,E)$, whose nodes V are the n data points $\{\boldsymbol{y}_i\}_{i=1}^n$ corresponding to the n training samples, where $\boldsymbol{y}_i = [y_{i1}, y_{i2}, \ldots, y_{ic}]^T$. We define the adjacency by including the KNNs of a given node as its adjacent nodes. These KNNs are determined by a certain similarity measure, which is also used as the weight between two adjacent nodes. Different definitions of similarity measures between two nodes determine different adjacent graphs, thus different weight matrices $\mathbf{W}_Y$. In this work, we propose two schemes to compute the similarity between nodes based on the multi-label information: (1) by working in the binary space of labels $\{0,1\}^c$, (2) by working in the transformed real space of labels.

3.1.1 Proximity in Binary Label Space

In the binary label space, all the label vectors $\{\boldsymbol{y}_i\}_{i=1}^n$ are binary strings with the same length. The following string-based distance/similarity can be employed to capture the proximity structure between samples in the label space:

- **Hamming Distance** between two strings of equal length is the number of positions at which the corresponding bits are different, denoted as $\|\boldsymbol{y}_i - \boldsymbol{y}_j\|_H$. This is also the edit distance between two binary strings of equal length. By employing the Gaussian kernel, a Hamming-based similarity between two strings can be obtained:

$$\text{sim}_H(\boldsymbol{y}_i, \boldsymbol{y}_j) = \exp\left(\frac{-\|\boldsymbol{y}_i - \boldsymbol{y}_j\|_H^2}{\tau}\right). \tag{10}$$

  The adjacent graph G_Y constructed from the Hamming distance capture the proximity information between samples based on how many distinct classes they belong to.

- **And-based Similarity** is the size of the intersection between two binary strings, given as

$$\text{sim}_A(\boldsymbol{y}_i, \boldsymbol{y}_j) = \|\boldsymbol{y}_i \wedge \boldsymbol{y}_j\|_1. \tag{11}$$

This provides a measure of "closeness" between two samples by the number of classes they both belong to, which we believe is important to capture the intrinsic structure of the labels. Assuming the importance of a shared class is related to its size in a collection of different sizes of multiple classes, we can further scale the above and-based similarity by

$$\text{sim}_A^{(s)}(\boldsymbol{y}_i, \boldsymbol{y}_j) = \|(\boldsymbol{y}_i \wedge \boldsymbol{y}_j) \cdot \boldsymbol{s}\|_1, \qquad (12)$$

where $\boldsymbol{s} = \left[\frac{1}{n_1}, \frac{1}{n_2}, \ldots, \frac{1}{n_c}\right]^T$ is a scaling vector related to class size.

- **Søensen's Similarity Coefficient** is a statistic that can be used for comparing the similarity of two binary strings, given as

$$\text{sim}_S(\boldsymbol{y}_i, \boldsymbol{y}_j) = \frac{2\|\boldsymbol{y}_i \wedge \boldsymbol{y}_j\|_1}{\|\boldsymbol{y}_i\|_1 + \|\boldsymbol{y}_j\|_1}, \qquad (13)$$

which is also known as Dice's coefficient. This is equivalent to further scaling the and-based similarity in Eq. (11) by $\frac{2}{\|\boldsymbol{y}_i\|_1 + \|\boldsymbol{y}_j\|_1}$, rather than the inverse of the class size.

- **Jaccard Similarity Coefficient** is another statistic that can be used:

$$\text{sim}_J(\boldsymbol{y}_i, \boldsymbol{y}_j) = \frac{\|\boldsymbol{y}_i \wedge \boldsymbol{y}_j\|_1}{\|\boldsymbol{y}_i \vee \boldsymbol{y}_j\|_1}, \qquad (14)$$

which is also known as Jaccard index. Similarly, this can be viewed as a scaled and-based similarity, of which the used scaling vector has elements equal to $\frac{1}{\|\boldsymbol{y}_i \vee \boldsymbol{y}_j\|_1}$. To compare Eq. (13) and Eq. (14), we have

$$\begin{aligned} & \|\boldsymbol{y}_i \vee \boldsymbol{y}_j\|_1 - \frac{1}{2}(\|\boldsymbol{y}_i\|_1 + \|\boldsymbol{y}_j\|_1) \\ = \; & \|\boldsymbol{y}_i \vee \boldsymbol{y}_j\|_1 - \frac{1}{2}(\|\boldsymbol{y}_i \vee \boldsymbol{y}_j\|_1 + \|\boldsymbol{y}_i \wedge \boldsymbol{y}_j\|_1) \\ = \; & \frac{1}{2}(\|\boldsymbol{y}_i \vee \boldsymbol{y}_j\|_1 - \|\boldsymbol{y}_i \wedge \boldsymbol{y}_j\|_1) \geq 0. \end{aligned}$$

It is obvious that $\text{sim}_S(\boldsymbol{y}_i, \boldsymbol{y}_j) > \text{sim}_J(\boldsymbol{y}_i, \boldsymbol{y}_j) > 0$, when $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ share some classes but not all; and $\text{sim}_S(\boldsymbol{y}_i, \boldsymbol{y}_j) = \text{sim}_J(\boldsymbol{y}_i, \boldsymbol{y}_j) > 0$, when $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ are identical; also $\text{sim}_S(\boldsymbol{y}_i, \boldsymbol{y}_j) = \text{sim}_J(\boldsymbol{y}_i, \boldsymbol{y}_j) = 0$ when $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ do not have any classes in common.

To construct a proximity structure between samples, Hamming distance evaluates the number of "distinct classes", while the rest measures evaluate the number of "shared classes" but with different scalings. For single-label classification, by setting the number of KNNs as n, the weight matrix computed with coefficients in Eq. (11), Eq. (13), and Eq. (14) all lead to Eq. (6), while, the scaled coefficient in Eq. (12) leads to Eq. (7).

3.1.2 Proximity in Projected Label Space

We can also seek the latent similarity between binary label vectors in a transformed and more compact real space. In the first stage, we map each c-dimensional binary label vector $\boldsymbol{y}_i$ to a k_c-dimensional real space ($k_c \leq c$) and obtain a set of transformed label vectors $\{\hat{\boldsymbol{y}}_i\}_{i=1}^n$. One way for achieving this is to employ a projection technique that maximizes the variance of the projections $\hat{\mathbf{Y}} = \mathbf{Y}\mathbf{P}_y$ as

$$\max_{\substack{\mathbf{P}_y \in R^{c \times k_c}, \\ \mathbf{P}_y^T \mathbf{P}_y = \mathbf{I}_{k_c \times k_c}}} \frac{1}{n-1} \sum_{i=1}^{n} \left\| \mathbf{P}_y^T \boldsymbol{y}_i - \frac{1}{n} \sum_{j=1}^{n} \mathbf{P}_y^T \boldsymbol{y}_j \right\|_2^2. \qquad (15)$$

This is actually to apply PCA in the binary label space, mapping the c-dimensional label vectors into a smaller number of uncorrelated directions. The optimal solution of the above maximization problem is the top k_c right singular vectors of the $n \times c$ matrix $(\mathbf{I}_{n \times n} - \frac{1}{n}\boldsymbol{e}\boldsymbol{e}^T)\mathbf{Y}$, corresponding to its largest k_c singular values (Wall et al., 2003). In the second stage of Scheme 2, the similarity between two label vectors is obtained by

$$\text{sim}_P(\boldsymbol{y}_i, \boldsymbol{y}_j) = \exp\left(\frac{-\|\hat{\boldsymbol{y}}_i - \hat{\boldsymbol{y}}_j\|_2^2}{\tau}\right). \qquad (16)$$

Different from scheme 1, the graph G_Y is constructed from the label embeddings $\{\hat{\boldsymbol{y}}_i\}_{i=1}^n$. It should be mentioned that when the problem at hand has a large number of classes, such as text categorization with large taxonomies (Bennett and Nguyen, 2009), the label matrix $\mathbf{Y}$ is usually very sparse due to lack of training samples for some classes. In this case, Scheme 2 is preferred over Scheme 1, as the projected label vectors provide a more compact, simplified and robust representation with reduced noise.

3.1.3 Graph Modification

Let $\mathbf{W}_X$ denote the feature-based weight matrix obtained as discussed in Section 2.3. The following scheme is used to combine the intrinsic label-based and the geometric feature-based proximity structures, leading to a modified weight matrix $\mathbf{W}$:

$$\mathbf{W} = (1-\theta)\frac{\mathbf{W}_X}{\alpha_X} + \theta\frac{\mathbf{W}_Y}{\alpha_Y}, \qquad (17)$$

where $0 \leq \theta \leq 1$ is a user-defined parameter controlling how much the embeddings should be biased by the label information. Here, we scale the two weight matrices $\mathbf{W}_X$ and $\mathbf{W}_Y$ with α_X and α_Y, respectively, which are the means of the absolute values of the non-zero elements in $\mathbf{W}_X$ and $\mathbf{W}_Y$, respectively. The purpose to introduce α_X and α_Y is to control the tradeoff

Table 1: A list of functions used to compute the relation features.

Measures	Functions
Minkowski Distance	$r_{ij} = \left(\sum_{t=1}^{d} \lvert x_{it} - p_{jt}\rvert^{P}\right)^{\frac{1}{P}}$
Dot Product	$r_{ij} = \sum_{t=1}^{d} x_{it} p_{jt}$
Cosine Similarity	$\frac{\sum_{t=1}^{d} x_{it} p_{jt}}{\lVert \boldsymbol{x}_i\rVert_2 \times \lVert \boldsymbol{p}_j\rVert_2}$
Polynomial Kernel	$r_{ij} = \left(\sum_{t=1}^{d} x_{it} p_{jt} + 1\right)^{\rho}$
Gaussian Kernel	$r_{ij} = \exp\left(-\frac{\lVert \boldsymbol{x}_i - \boldsymbol{p}_j\rVert_2^2}{\sigma^2}\right)$
Pearson Correlation	$r_{ij} = \frac{1}{d}\sum_{t=1}^{d}\left(\frac{x_{it}-\mu_t^x}{\sigma_t^x}\right)\left(\frac{p_{jt}-\mu_t^p}{\sigma_t^p}\right)$

between $\mathbf{W}_X$ and $\mathbf{W}_Y$ only with one parameter θ. Using the above combined weight matrix in Eq. (2), we achieve supervised implementation when $\theta > 0$, while unsupervised when $\theta = 0$. It is worth to mention that when $\theta = 1$ no feature structure is considered, and the computed embeddings are forced to preserve the structure in the label space. This may lead to overfitting when there exist erroneously labeled samples. Thus, an appropriate selection of the degree parameter θ is required by the users, given a specific classification task.

3.2 Computation Reduction

With the out-of-sample extension 1, one needs to compute the (generalized) eigen-decomposition of a $d \times d$ matrix, which has a computational cost around $O(\frac{9}{2}d^3)$ (Steinwart, 2001). With the extension 2, one needs to compute the inverse of a $d \times d$ matrix, which has a computational cost around $O(d^{2.376})$ (Coppersmith and Winograd, 1990). This is often unacceptably high with large-scale features $d \gg n$. To overcome this, we employ a set of relation values, such as distance, similarity and correlation, between each sample and $p \leq n$ prototypes as the new input features of the DR algorithm, when dealing with large-scale tasks ($d \gg n$). In Table 1 , we list several relation measures that can be used to compute these relation features. Previous research (Pekalska and Duin, 2002; Pekalska et al., 2006) has already shown that (dis)similarities between the training samples and a collection of prototype objects can be used as input features to build good classifiers. This means that, for each sample, its (dis)similarities to prototypes possess comparable discriminating ability to its original features. Thus, we expect the discriminating ability of the embeddings computed from the relation values should be similar to that of the embeddings computed from the original features.

To obtain prototypes from training samples, different methods can be used (Huang et al., 2002; Mollineda and andE. Vidal, 2002; Pekalska et al., 2006), among which random selection is the simplest (Pekalska et al., 2006). Existing results show that, by directly employing the dissimilarities between each sample and the prototypes as the input feature of a linear classifier, different prototype selection techniques lead to quite similar classification performance as the number of used prototypes increases, even including the random selection (Pekalska et al., 2006). This means, when the number of used prototypes is large enough, the discriminating ability of the relation values between samples and the selected prototypes does not vary much with respect to different selected prototypes.

In this work, we employ the following prototype selection scheme: Letting p denote the number of selected prototypes, we use the ratio $0 < \beta = \frac{p}{n} \leq 1$ as a user-defined parameter to control the size of prototypes. When $\beta \geq 50\%$, we simply pick up p training samples by random as prototypes. When $\beta < 50\%$, we perform the k-center clustering analysis for data points belonging to the same class, by employing the Gonzalez's approximation algorithm (Gonzalez, 1985). As the objective of the k-center clustering analysis is to group a set of points into different clusters so that the maximum intercluster distance is minimized, the obtained cluster centers (heads) can reliably summarize the distribution of the original data. Such a procedure is repeated c times for c different classes. For each class c_i, a set of resulting cluster heads are obtained from the analysis and are used as the prototypes, denoted as H_i. Let $\mathfrak{P}$ denote the total set of obtained prototypes and p denote the size of $\mathfrak{P}$, we have $\mathfrak{P} = H_1 \bigcup H_2 \bigcup \cdots \bigcup H_c$, and $p = |\mathfrak{P}|$. Let $\mathbf{P} = [p_{ij}]$ denote the $p \times d$ feature matrix for the p obtained prototypes, $\mathbf{R} = [r_{ij}]$ denote the $n \times p$ relation matrix between the n training samples and the p prototypes, and $\tilde{\mathbf{R}}$ the $m \times p$ relation matrix between the m query (test) samples and the p prototypes. We use $\mathbf{R}$ to replace $\mathbf{X}$ in Eqs. (2, 3 and 5), and $\tilde{\mathbf{Z}} = \tilde{\mathbf{R}}\mathbf{P}^*$.

4 EXPERIMENTS

In order to empirically investigate our proposed proximity-based embeddings for multi-label classification, two text categorization problems with large-scale features are studied, of which the used document collections are briefly described as follows.

Reuters Document Collection. The "Reuters-21578 Text Categorization Test Collection" contains articles taken from the Reuters newswire[1], where

[1] http://archive.ics.uci.edu/ml/support/Reuters-21578+Text+Categorization+Collection

Table 2: Performance comparison using the Reuters dataset.

	corn	grain	wheat	acq	earn	ship	interest	money-fx	crude	trade	Average
LE	0.851	0.902	0.845	0.924	0.956	0.845	0.826	0.847	0.861	0.795	0.865
SLE	0.907	0.957	0.902	0.960	0.983	0.878	0.849	0.885	0.900	0.888	**0.911**
USC	0.846	0.902	0.865	0.923	0.955	0.858	0.827	0.852	0.868	0.807	0.870
SUSC	0.907	0.956	0.902	0.959	0.983	0.882	0.855	0.885	0.911	0.875	**0.912**
OLPP	0.882	0.948	0.869	0.936	0.973	0.870	0.829	0.870	0.871	0.862	0.891
SOLPP	0.910	0.956	0.896	0.960	0.983	0.866	0.850	0.885	0.904	0.884	**0.909**

each article is designated into one or more semantic categories. A total number of 9,980 articles from 10 overlapped categories were used in our experiments. We randomly divide the articles from each category into three partitions with nearly the same size, for the purpose of training, validation and test. This leads to 3,328 articles for training, and 3,326 articles for validation and test, respectively, where around 18% of these articles belong to 2 to 4 different categories at the same time, while each of the rest belongs to a single category.

EEP Document Collection. A collection of documents is supplied by Education Evidence Portal (EEP)[2], where each document is a quite lengthy full paper or report (approximately 250 KB on average after converting to plain text). Domain experts have developed a taxonomy of 108 concept categories in the area and manually assigned categories to documents stored in the database. This manual effort has resulted in 2,157 documents, including 1,936 training documents and 221 test documents, where 96% of these documents were assigned 2 to 17 different categories, while only one category for the rest.

Used Features. The numerical features for classification were extracted as follows: We first applied Porter's stemmer[3] to the documents, then, extracted word uni-grams, bi-grams, and tri-grams from each documents. For the Reuters document collection, after filtering the low-frequency words, the tf-idf values of 24,012 word uni-grams are used as the original features. This leads to a $3,328 \times 24,012$ feature matrix $\mathbf{X}$ for the training samples, while, a $3,326 \times 24,012$ feature matrix $\tilde{\mathbf{X}}$ for the query sample, in both the validation and test procedures. For the EEP document collection of full papers, the corresponding binary values of the word uni-grams, bi-grams, and tri-grams, representing whether the terms occurred in the documents, are used as the original features. This leads to a $1,936 \times 176,624,316$ feature matrix $\mathbf{X}$ for the training samples, while, a $221 \times 176,624,316$ feature matrix $\tilde{\mathbf{X}}$ for the test samples.

Table 3: Performance comparison using the EPP dataset. Cat. 1-5 are the five largest classes containing the most samples.

	Cat. 1	Cat. 2	Cat. 3	Cat. 4	Cat. 5	Average
LE	0.646	0.544	0.690	0.553	0.554	0.355
SLE	0.662	0.561	0.752	0.579	0.538	**0.394**
USC	0.646	0.554	0.691	0.563	0.494	0.346
SUSC	0.671	0.566	0.717	0.557	0.557	**0.410**
OLPP	0.652	0.556	0.710	0.589	0.564	0.424
SOLPP	0.677	0.574	0.712	0.616	0.550	**0.457**

4.1 Experimental Setup

In this paper, we propose different ways to construct the multi-label graph so that it can be used by Eq. (2) to obtain the proximity-based embeddings. The proposed graph is applied to two settings of the framework, corresponding to LE and USC, respectively. Our proposed extension 2 is used to compute embeddings for new query samples, for both LE and USC. We also applied extension 1 with orthogonal projections, leading to OLPP. When the feature-based adjacency graph in Section 2.3 is used, unsupervised DR is achieved, leading to the standard LE, USC, and OLPP; when our multi-label graph is used, supervised DR is achieved, leading to the supervised extension of LE, USC, and OLPP denoted as SLE, SUSC, and SOLPP. We also compare our method with another unsupervised DR method, latent semantic analysis (LSI) (Kim et al., 2005), and three existing supervised DR methods for multi-label classification, including canonical correlation analysis (CCA) (Hardoon et al., 2004), multi-label DR via dependence maximization (MDDM) (Zhang and Zhou, 2007), and multi-output regularized feature projection (MORP) (Yu et al., 2006). Among these existing methods, LSI defines an orthogonal projection matrix to enable optimal reconstruction by minimizing the error in terms of $\|\mathbf{X} - \mathbf{X}\mathbf{P}\mathbf{P}^T\|_{\mathrm{F}}^2$, LE, USC and OLPP optimizes Eq. (2) using a feature-based weight matrix, CCA and MDDM maximize the correlation coefficient and the Hilbert-Schmidt independence criterion

[2]http://www.eep.ac.uk

[3]http://tartarus.org/ martin/PorterStemmer/

Table 4: Comparison of the macro F_1 score for different methods. The proposed methods are marked by *, and (U) denotes unsupervised, (S) supervised.

Method (U/S)		Raw N/A	LSI (U)	LE (U)	USC (U)	OLPP (U)	CCA (S)	MORP (S)	MDDM (S)	SLE* (S)	SUSC* (S)	SOLPP* (S)
Reuters	F_1	0.890	0.828	0.865	0.870	0.891	0.878	0.900	0.900	0.911	**0.912**	0.909
	k	24,012	1800	1800	1800	1800	1800	1800	1800	1800	1800	1800
EPPI	F_1	0.332	0.387	0.355	0.346	0.424	0.390	0.394	0.385	0.394	0.410	**0.457**
	k	176,624,316	300	100	200	150	500	500	200	100	100	100

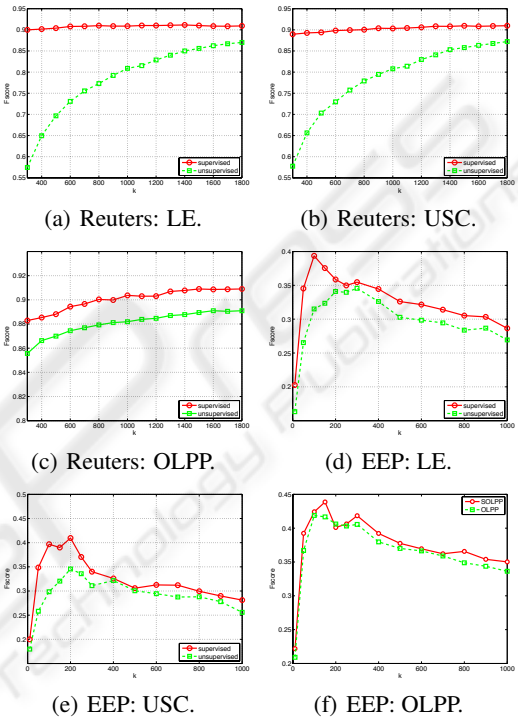
between the projected features and the labels, respectively, and MORP minimizes the reconstruction error of both features and labels.

To obtain the feature-based adjacency graph, two types of KNN-graph were used, one with the Gaussian kernel weight and the other with constant binary weight, which were also used as $\mathbf{W}_X$ to obtain our multi-label graph. All the model parameters, including the number of KNNs, the regularization parameter α of out-of-sample extension 2, the parameter β to control the number of prototypes, the number of lower-dimensional embeddings k, the degree parameter θ, and the width parameters of the Gaussian kernels, were tuned by grid search, using the validation set for the Reuters data and 3-fold-cross validation with the training set for the EEP data. To reduce the computational complexity of the DR procedure caused by large-scale features, the Euclidean distance was employed to compute the prototype-based relation features for the Reuters data, while, the inner-product for the EEP data.

As support vector machines (SVMs) have shown success in text categorization (Bennett and Nguyen, 2009), a linear SVM was employed to obtain the multi-label classification performance of different types of embeddings. The macro average of the F_1 scores of all classes is computed for performance evaluation and comparison. For each category, the F_1 score is computed by $F_1 = \frac{2\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$, where $\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}}$, $\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}}$, TP denotes true positive, TN denotes true negative, FP denotes false positive and FN denotes false negative.

4.2 Results and Analysis

Different types of multi-label graph in Section 3.1 were tried for SLE, SUSC and SOLPP, of which performance varies from 0.902 to 0.912 for the Reuters data, and from 0.387 to 0.457 for the EEP data. It is observed that the best performance was mostly achieved with $\mathbf{W}_X$ defined by the KNN-graph with the Gaussian kernel weight, and $\mathbf{W}_Y$ computed from the projected label vectors. We compare our SLE, SUSC and SOLPP using this best performing multi-label graph with LE, USC and OLPP using their best performing feature-based graph (KNN-graph with the Gaussian kernel weight), respectively, in Table 2 and Table 3 for both datasets, as well as Figure 1 for different values of the resulting dimensionality of embeddings. It can be seen from Table 2, Table 3 and Figure 1, our supervised multi-label graph generate embeddings with better discriminating power, as compared with the unsupervsied feature-based graph. We also show the impact of the tradeoff between the feature and label structures in Figure 2, for different methods and datasets. Different optimal values of θ were reached for different used values of k. Appropriate combination of the label and feature information can improve the performance obtained by solely using one type of information on its own.



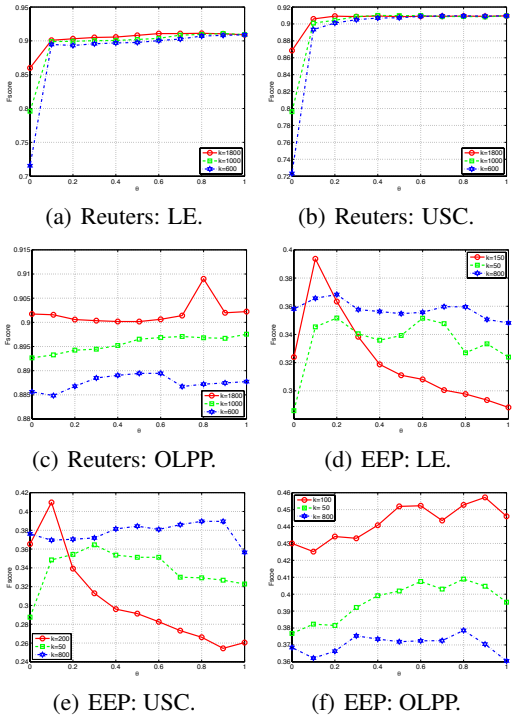
(a) Reuters: LE. (b) Reuters: USC.

(c) Reuters: OLPP. (d) EEP: LE.

(e) EEP: USC. (f) EEP: OLPP.

Figure 1: Performance with respect to the reduced dimension k for different methods and datasets.

We compare the macro F_1 scores of our proposed supervised DR methods with that of four existing unsupervised DR methods and three existing supervised DR methods, as well as that of the original features, denoted as raw features, without applying any DR



(a) Reuters: LE. (b) Reuters: USC.

(c) Reuters: OLPP. (d) EEP: LE.

(e) EEP: USC. (f) EEP: OLPP.

Figure 2: Impact of the tradeoff between the feature and label structures controlled by θ, for different methods.

method in Table 4. The original CCA and MORP both impose the orthogonality condition on the embeddings. It is noticed in the experiments the original CCA and MORP performed unsatisfactorily for both datasets. However, by imposing the orthogonality condition on the projections instead, the performance has been greatly improved, which is reported in Table 4. The results show that most supervised DRmethods perform better than the unsupervised ones in terms of classification performance. Our proposed methods provides the highest classification performance for both datasets (see Table 4).

We also show the show the reduction of computational cost using the prototype-based relation features, as compared with the original features. To compute the embeddings based on Eq. (3) or Eq. (5) for the EEP data using the original features, one needs to decompose or compute the inverse of a $176,624,316 \times 176,624,316$ matrix. This makes it impossible to collect the classification results in a reasonable time. For the Reuters data, although with comparatively smaller size of features, it still took long time (more than 7,000 Sec. using MATlAB with computer of 2.8G CPU and 4.0 GB Memory) to obtain results using the original features. By using the prototype-based relation features, the computing time of these methods was greatly reduced to less than 400 Sec. using MATLAB with the same computer, for both datasets.

5 CONCLUSIONS

In this paper, we have developed algorithms for supervised generation of low-dimensional embeddings with good discriminating ability to facilitate multi-label classification. This is achieved by modelling the proximity structure between samples with a multi-label graph constructed from both feature and multi-label information. Working in either a binary label space or a projected real label space, different similarity measures have been used to compute the weight values of the multi-label graph. By employing the weighted linear combination of the feature-based and label-based adjacency graphs, the tradeoff between the category and feature structures can be adjusted with a degree parameter. To further reduce the computational cost for classification with a large number of input features, we seek the optimal projections in a prototype-based relation feature space, instead of the original feature space. By incorporating the label information into the construction of the adjacency graph, performance of LE, USC, and OLPP has been improved by 2% to 5% for the Reuters data, and by 7% to 18% for the EEP data. Our current method is applicable to discrete output value (classes). Research on how to extend this to supervised learning task with continuous output values, such as regression, is in procedure. The proposed method is a general supervised DR approach for multi-label classification, which should find more applications in IR, TM, NLP and bioinformatics.

ACKNOWLEDGEMENTS

This research is supported by Biotechnology and Biological Sciences Research Council, BBSRC project BB/G013160/1 and the JISC sponsored National Centre for Text Mining, University of Manchester, UK.

REFERENCES

Barutcuoglu, Z., Schapire, R. E., and Troyanskaya, O. G. (2006). Hierarchical multi-label prediction of gene function. *Bioinformatics*, 22(7):830–836.

Bekkerman, R., Tishby, N., Winter, Y., Guyon, I., and Elisseeff, A. (2003). Distributional word clusters vs. words for text categorization. *Journal of Machine Learning Research*, 3:1183–1208.

Belkin, M. and Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396.

Bengio, Y., Paiement, J., Vincent, P., Delalleau, O., Roux, N. L., and Ouimet, M. (2003). Out-of-sample extensions for LLE, Isomap, MDS, eigenmaps, and spectral clustering. In *Proc. of Neural Information Processing Systems, NIPS*.

Bennett, P. N. and Nguyen, N. (2009). Refined experts: improving classification in large taxonomies. In *Proc. of the 32nd Int'l ACM SIGIR conference on Research and development in information retrieval*.

Blei, D. M., Ng, A. Y., Jordan, M., and Lafferty, J. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:2003.

Cai, D., He, X., and Han, J. (2007a). Spectral regression: A unified subspace learning framework for content-based image retrieval. In *Proc. of the ACM Conference on Multimedia*.

Cai, D., He, X., and Han, J. (2007b). Spectral regression for efficient regularized subspace learning. In *Proc. of the International Conf. on Data Mining, ICDM*.

Chan, P. K., Schlag, M. D. F., and Zien, J. Y. (1994). Spectral k-way ratio-cut partitioning and clustering. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 13(9):1088–1096.

Coppersmith, D. and Winograd, S. (1990). Matrix multiplication via arithmetic progressions. *Journal of Symbolic Computation*, 9:251–280.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407.

Dhillon, I. S. (2001). Co-clustering documents and words using bipartite spectral graph partitioning. In *Proc. of the 7th ACM SIGKDD International Conf. on Knowledge discovery and data mining*, pages 269–274, San Francisco, California, US.

Dhillon, I. S., Mallela, S., and Kumar, R. (2003). A division information-theoretic feature clustering algorithm for text classification. *Journal of Machine Learning Research*, 3:1265–1287.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188.

Gonzalez, T. F. (1985). Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38:23–306.

Hardoon, D. R., Szedmak, S. R., and Shawe-taylor, J. R. (2004). Canonical correlation analysis: An overview with application to learning methods. *Neural Computation*, 16(12):2639 – 2664.

He, X. (2004). Incremental semi-supervised subspace learning for image retrieval. In *Proc. of the ACM Conference on Multimedia*.

He, X. and Niyogi, P. (2003). Locality preserving projections. In *Proc. of Neural Information Processing Systems 16, NIPS*.

He, X., Yan, S., Hu, Y., Niyogi, P., and Zhang, H. (2005). Face recognition using laplacianfaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(3):328–340.

HildII, K. E., Erdogmus, D., Torkkola, K., and Principe, J. C. (2006). Feature extraction using information-theoretic learning. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 28(9):1385–1392.

Huang, Y., Chiang, C., Shieh, J., and Grimson, W. (2002). Prototype optimization for nearest-neighbor classification. *Pattern Recognition*, (6):12371245.

Jolliffe, I. T. (1986). *Principal Component Analysis*. Springer-Verlag, New York, NY.

Kim, H., Howland, P., and Parl, H. (2005). Dimension reduction in text classification with support vector machines. *Journal of Machine Learning Research*, 6:3753.

Kokiopoulou, E. and Saad, Y. (2007). Orthogonal neighborhood preserving projections: A projection-based dimensionality reduction technique. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29(12):2143–2156.

Kokiopouloua, E. and Saadb, Y. (2009). Enhanced graph-based dimensionality reduction with repulsion laplaceans. *Pattern Recognition*, 42:2392–2402.

Lewis, D. D. (1992). Feature selection and feature extraction for text categorization. In *Proc. of the workshop on Speech and Natural Language*, pages 212–217, Harriman, New York.

Li, S., Xia, R., Zong, C., and Huang, C.-R. (2009). A framework of feature selection methods for text categorization. In *Proc. of the Joint Conf. of the 47th Annual Meeting of the ACL and the 4th Int'l Joint Conf. on Natural Language Processing of the AFNLP*, pages 692–700, Suntec, Singapore. Association for Computational Linguistics.

Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and Computing*, 17(4).

Mollineda, R. and andE. Vidal, F. F. (2002). An efficient prototype merging strategy for the condensed 1-nn rule through class-conditional hierarchical clustering. *Pattern Recognition*, (12):27712782.

Pekalska, E. and Duin, R. (2002). Dissimilarity representations allow for building good classifiers. *Pattern Recognition Letters*, (8):943–956.

Pekalska, E., Duin, R., and Paclik, P. (2006). Prototype selection for dissimilarity-based classifiers. *Pattern Recognition*, (2):189–208.

Roweis, S. T. and Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326.

Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(8):888–905.

Steinwart, I. (2001). On the influence of the kernel on the consistency of support vector machines. *Journal of Machine Learning Research*, 2:67–93.

Sugiyama, M. (2007). Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis. *Journal of Machine Learning Research*, 8:1027–1061.

Sugiyama, M. (2010). Semi-supervised local fisher discriminant analysis for dimensionality reduction. *Machine Learning*, 78(1-2):35–61.

Sun, L., Ji, S., and Ye, J. (2008). Hypergraph spectral learning for multi-label classification. In *Proc. of the 14th ACM SIGKDD International Conf. on Knowledge Discovery and Data Mining*, pages 668–676, Las Vegas, Nevada, USA.

Tang, L., Rajan, S., and Narayanan, V. K. (2009). Large scale multi-label classification via metalabeler. In *Proc. of 18th Int'l Conf. on World Wide Web*.

Wall, M. E., Andreas, R., and Rocha, L. M. (2003). Singular value decomposition and principal component analysis. *A Practical Approach to Microarray Data Analysis*, pages 91–109.

Yan, S., Xu, D., Zhang, B., Zhang, H., Yang, Q., and Lin, S. (2007). Graph embedding and extensions: A general framework for dimensionality reduction. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29(1):40–51.

Yu, J. and Tian, Q. (2006). Learning image manifolds by semantic subspace projection. In *Proc. of the ACM Conference on Multimedia*.

Yu, S., Yu, K., Tresp, V., and Kriegel, H. (2006). Multi-output regularized feature projection. *IEEE Trans. on Knowledge and Data Eigeneering*, 18(12):1600–1613.

Zhang, W., Xue, X., Sun, Z., Guo, Y., and Lu, H. (2007). Optimal dimensionality of metric space for classification. In *Proc. of the 24th International Conf. on machine learning, ICML*, volume 227, pages 1135–1142.

Zhang, Y., Surendran, A. C., Platt, J. C., and Narasimhan, M. (2008). Learning from multitopic web documents for contextual advertisement. In *Proc. of 14th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*.

Zhang, Y. and Zhou, Z. (2007). Multi-label dimensionality reduction via dependence maximization. In *Proc. of the 23rd National Conf. on Artificial intelligence*, volume 3, pages 1503–1505, Chicago, Illinois.