Jointly Modeling Label and Feature Heterogeneity in Medical Informatics

Pei Yang, Arizona State University
Hongxia Yang, Yahoo! Inc.
Haoda Fu, Eli Lilly and Company
Dawei Zhou, Arizona State University
Jieping Ye, University of Michigan
Theodoros Lappas, Stevens Institute of Technology
Jingrui He, Arizona State University

Multiple types of heterogeneity including label heterogeneity and feature heterogeneity often co-exist in many real-world data mining applications, such as diabetes treatment classification, gene functionality prediction, and brain image analysis. To effectively leverage such heterogeneity, in this paper, we propose a novel graph-based model for Learning with both Label and Feature heterogeneity, namely L^2F. It models the label correlation by requiring that any two label-specific classifiers behave similarly on the same views if the associated labels are similar, and imposes the view consistency by requiring that view-based classifiers generate similar predictions on the same examples. The objective function for L^2F is jointly convex. To solve the optimization problem, we propose an iterative algorithm, which is guaranteed to converge to the global optimum. One appealing feature of L^2F is that it is capable of handling data with missing views and labels. Furthermore, we analyze its generalization performance based on Rademacher complexity, which sheds light on the benefits of jointly modeling the label and feature heterogeneity. Experimental results on various biomedical data sets show the effectiveness of the proposed approach.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications - Data Mining; I.5.2 [**Computing Methodologies**]: Pattern Recognition - Design Methodology

General Terms: Design, Algorithms, Performance

Additional Key Words and Phrases: Heterogeneous learning, Multi-label learning, Multi-view learning, Medical informatics

ACM Reference Format:
Pei Yang, Hongxia Yang, Haoda Fu, Dawei Zhou, Jieping Ye, Theodoros Lappas, and Jingrui He, 2015. Jointly Modeling Label and Feature Heterogeneity in Medical Informatics. *ACM Trans. Knowl. Discov. Data.* 0, 0, Article 1 (2015), 25 pages.
DOI:http://dx.doi.org/10.1145/0000000.0000000

This work is partially supported by the NSF (No. IIS1017415), the Army Research Laboratory (No. W911NF-09-2-0053), Region II University Transportation Center (No. 49997-33 25), DARPA (No. W911NF-11-C-0200 and W911NF-12-C-0028), and NSFC (No. 61473123).
Author's addresses: P. Yang, School of Computing, Informatics and Decision Systems Engineering, Arizona State University; email: cs.pyang@gmail.com; H. Yang, Yahoo! Inc.; email: firewater1984@gmail.com; H. Fu, Eli Lilly and Company; email: fu_haoda@lilly.com; D. Zhou, School of Computing, Informatics and Decision Systems Engineering, Arizona State University; email:davidchouzdw@gmail.com ; J. Ye, Department of Computational Medicine and Bioinformatics and Department of Electrical Engineering and Computer Science, University of Michigan; email: jieping@gmail.com; T. Lappas, Howe School of Technology Management, Stevens Institute of Technology; email: tedlappas@gmail.com; J. He, School of Computing, Informatics and Decision Systems Engineering, Arizona State University; email: jingrui.he@gmail.com.
Permission to make digital or hard copies of part or all 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 show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.
© 2015 ACM 1556-4681/2015/-ART1 $15.00
DOI:http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

Many real-world applications exhibit both label and feature heterogeneity, such as text categorization, medical diagnosis, image or video annotation, gene functionality prediction, tag recommendation. On one hand, label heterogeneity means that each example is associated with a set of different class labels. For example, the diabetes patients may receive multiple treatments such as metformin and sulphonylurea which refer to multiple labels; genes may have multiple functionalities which cause them to be associated with multiple diseases. On the other hand, feature heterogeneity means that the data are described by features from multiple views, or information sources. For example, the diabetes patients are characterized by different views of features measuring the long term and short term drug impact; proteins in given species have features that contain diverse information such as gene expression, protein-protein interactions, and sequence similarity, where some features are species-specific, and the others are cross-species.

The major challenge for addressing such problems is how to jointly model the multiple types of heterogeneity in mutually beneficial way. To address this problem, in this paper, we propose a novel graph-based model named L^2F to leverage both label and feature heterogeneity. In particular, L^2F accommodates multiple relationships, such as instance-instance, label-label, and view-view correlations in a principled framework. In this way, it is able to: (1) model the label correlation by requiring that any two label-specific classifiers behave similarly on the same views if the associated labels are similar, and (2) impose the view consistency by requiring that view-based classifiers generate similar predictions on the same examples. To solve the resulting optimization problem, we propose an iterative algorithm based on block coordinate descent. It is guaranteed to converge to the global optimum. Furthermore, different from most existing methods for addressing data heterogeneity, which require complete views and labels, a direct extension of the proposed L^2F model can be used on data with both missing views and missing labels. Therefore, it is widely applicable to real-world applications with incomplete views and incomplete labels.

Moreover, we aim to answer the fundamental question of whether the generalization performance can be improved by jointly modeling both label and feature heterogeneity. Our theoretical analysis based on Rademacher complexity [Shawe-Taylor and Cristianini 2004] shows that the error bound of the proposed model could be improved by utilizing the label correlation and imposing the view consistency. We also empirically demonstrate the effectiveness of L^2F on various data sets compared with state-of-the-art techniques.

The L^2F model can be further extended to a generalized framework which uses the hypergraph to model the multiple types of relationships among the objects in a principled manner. By encoding our prior knowledge on the learning task into the structure of the hypergraph, we could develop various instantiations of the generalized framework for modeling multiple types of heterogeneity.

The main contributions of this paper can be summarized as follows.

(1) A graph-based model named L^2F for jointly learning the label and feature heterogeneity;
(2) A natural extension of L^2F for handling data with missing views and missing labels;
(3) Theoretical analysis showing the benefits of simultaneously leveraging both types of heterogeneity;
(4) Experimental results on a variety of biomedical data sets showing the effectiveness of the proposed approach.

The rest of the paper is organized as follows. After a brief review of the related work in Section 2, we present the proposed L^2F model and its iterative optimization algorithm in Section 3. The generalization performance of L^2F is analyzed in Section 4. Section 5 discusses the extension of L^2F to a hypergraph-based framework. The experimental results on various datasets are shown in Section 6. Finally, we conclude in Section 7.

2. RELATED WORK

In this section, we survey the related work on heterogeneous learning from single or dual heterogeneity, as well as their applications in biomedical domain.

2.1. General Heterogeneous Learning

Multi-label learning studies the problem where each example is associated with a set of labels [Tsoumakas and Katakis 2007; Zhang and Zhou 2014]. The key issue for multi-label learning is how to exploit correlations or dependencies among multiple labels. According to [Zhang and Zhang 2010], existing strategies for label correlation exploitation can be grouped into three categories: first-order, second-order, and high-order approaches. First-order methods assume that labels are independent, and multi-label learning problem can be transformed into a number of independent binary classification problems, e.g., ML-kNN [Zhang and Zhou 2007]. Second-order approaches consider the pairwise relations between labels. Then the multi-label learning problem is transformed into the label ranking problem which aims at properly ranking every relevant-irrelevant label pair for each training instance, e.g., Rank-SVM [Elisseeff and Weston 2001]. Various methods have been proposed for high-order label correlation learning. For example, LEAD [Zhang and Zhang 2010] employed Bayesian network to encode the conditional dependencies of the labels as well as the feature set, with the feature set as the common parent of all labels. LS-ML [Ji et al. 2008] aimed to extract common subspace shared among multiple labels. A hypergraph spectral learning formulation was proposed for multi-label classification, where a hypergraph was constructed to exploit the correlation information among different labels [Sun et al. 2008]. LIFT [Zhang 2011] constructed features specific to each label by conducting clustering analysis on its positive and negative instances, and then performed training and testing by querying the clustering results. MLLOC [Huang and Zhou 2012] assumed that the label correlation may be shared by only a subset of instances rather than all the instances. MAHR [Huang et al. 2012] aimed to discover the label relationship via a boosting approach with a hypothesis reuse mechanism. FaIE [Lin et al. 2014] is a feature-aware label space dimension reduction (LSDR) approach which jointly maximized the recoverability of the original label space from the latent space, and the predictability of the latent space from the feature space. CFT [Li and Lin 2014] adapted the filter tree algorithm for cost-sensitive multi-label classification via constructing the label powerset.

Some multi-label methods work in the semi-supervised setting. CNMF [Liu et al. 2006] exploited unlabeled data as well as label correlations via the constrained nonnegative matrix factorization. TRAM [Kong et al. 2013] studied the problem of transductive multi-label learning by utilizing the information from both labeled and unlabeled data. CSFS [Chang et al. 2014] jointly modeled the sparse feature selection and semi-supervised learning in an optimization framework. TRANS [Guo and Schuurmans 2012] combined large-margin multi-label classification with unsupervised subspace learning. TML [Wang et al. 2011] is probabilistic transductive multi-label approach which simultaneously modeled the labeling consistency between visually similar videos and the multi-label interdependence for each video. iMLCU[Wu and Zhang 2013] is an inductive semi-supervised approach which simultaneously considered pairwise label correlations over labeled data and imposes maximum-margin regularization over unlabeled data.

Related theories for multi-label learning have also been studied and developed. The VC-dimension theory is used to derive the generalization bound for MAHR [Huang et al. 2012], which showed that the hypothesis reuse in MAHR can utilize the label relationship to reduce the capacity of the learning system and thus lead to a better generalization ability. A generic empirical risk minimization (ERM) framework was proposed for large-scale multi-label learning [Yu et al. 2014]. The proposed framework demonstrated better generalization performance for low-rank promoting trace-norm regularization when compared to (rank insensitive) Frobenius norm regularization. A theoretical analysis on multi-label consistency was proposed in [Gao and Zhou 2013]. They proved a necessary and sufficient condition for the consistency of multi-label learning based on surrogate loss functions.

Since multi-label is closely related to multi-task learning, we brief review the related work on multi-task learning. The goal of multi-task learning is to leverage the small amount of labeled data from multiple related tasks to improve the learner for each task. Among others, alternating structure optimization [Ando and Zhang 2005] decomposed the model into the task-specific and task-shared feature mapping; multi-task feature learning [Argyriou et al. 2006] assumed that multiple related tasks share a low-dimensional representation; clustered multi-task learning [Zhou et al. 2011] assumed that multiple tasks follow a clustered structure. Some recent multi-task learning methods are able to deal with irrelevant tasks by assuming that the model can be decomposed into a shared feature structure that captures task relatedness, and a group-sparse structure that detects outliers [Chen et al. 2011; Gong et al. 2012].

Multi-view learning has been studied extensively in the literature. Co-training [Blum and Mitchell 1998] is one of the earliest multi-view learning algorithm. It is proved that the two independent yet consistent views could be used to learn a concept in the probably approximately correct (PAC) framework based on a few labeled and many unlabeled examples. SVM-2K [Farquhar et al. 2005] combined KCCA with SVM in an optimization framework. CoMR [Sindhwani and Rosenberg 2008] is proposed for multi-view learning, which is based on a Reproducing Kernel Hilbert Space (RKHS) with a data-dependent co-regularization norm. The large-margin based method MMH [Chen et al. 2010] aimed to discover a predictive latent subspace representation shared by multiple views. The kernel spectral algorithm [Song et al. 2014] is a nonparametric kernel estimation method for learning multi-view latent variable models. An explicit objective function was introduced to measure the compatibility of learned hypotheses in multi-view learning [Collins and Singer 1999], and the boosting method was used to optimize the function. The PAC generalization bound [Dasgupta et al. 2001] was provided for co-training, which upper-bounded the error of classifiers learned from two views. The view independence assumption is relaxed in [Abney 2002], which suggested that the disagreement rate of two independent hypotheses upper-bounded the error rate of either hypothesis. An information-theoretic framework [Sridharan and Kakade 2008] was proposed for multi-view learning, which showed how to derive incompatibility functions for certain loss functions of interest so that minimizing this incompatibility over unlabeled data helped reduce expected loss on the test data.

More recently, researchers begin to study problems with dual types of heterogeneity. For problems with both task (or domain) and view heterogeneity, a variety of techniques have been proposed to model task relatedness in the presence of multiple views, e.g., [He and Lawrence 2011; Zhang and Huan 2012; Yang and He 2014; Yang et al. 2015; Yang and Gao 2013]. For the problems with both label and view heterogeneity, the multi-view 2DAL [Zhang et al. 2009] method integrated the mechanism of multi-view learning and active learning for multi-label image classification; MVMVL-MM [Fang and Zhang 2012] is based on the large margin framework, which mapped the multi-view data into low-dimensional feature space and simultaneously maximized the dependency between new feature descriptions and the labels; the L^2F approach [Yang et al. 2014] modeled both the view consistency and the label correlations in a graph-based framework. This paper extends our previous work [Yang et al. 2014] substantially by providing the detailed algorithm, theoretical justification and model generalization, as well as the comprehensive empirical evaluations on the biomedical data, which were not specifically presented in the preliminary version.

2.2. Heterogeneous Learning in Biomedical Domain

In biomedical domain, most data collected from different sources are heterogeneous. Take the prediction of causal disease genes as an example, the data may be in the forms of sequence, expression, annotation, etc. According to the survey paper [Piro and Di Cunto 2012], the evidences available for disease genes can be classified into the following categories: text-mining of biomedical literature, functional annotations, pathways and ontologies, phenotype relationships, intrinsic gene properties, sequence data, protein-protein interactions, regulatory information, orthologous relationships and gene expression information. Since different data sources can provide quite complementary disease-relevant information in many cases, they are practically merged to provide better coverage and generalization than any single data source.

Various methods have been proposed to model the associations between gene and disease. To name a few, sequence-based approach [Miozzi et al. 2008] used high-throughput gene-expression data to predict gene function through the 'guilt by association' principle. Network-based approach [Singh-Blom et al. 2013] worked by determining similarity between candidate gene and disease nodes in heterogeneous networks composed of different biological networks. Diffusion-based approach [Li and Patra 2013] conducted the random walk on a heterogeneous network composed of both the protein-protein interaction network and the weighted phenotype network. However, most of these methods ignore either the correlations among multiple labels by treating different labels independently, or the consistency among multiple views by simply concatenating different types of features into one view.

3. THE PROPOSED L^2F MODEL

In this section, we will introduce the proposed L^2F model. The basic idea of L^2F is to encode the label correlation and view consistency in a graph-based model. An iterative algorithm is presented to solve the resulting optimization problem. Furthermore, a direct extension of L^2F can deal with missing labels and missing views.

3.1. Notations and Problem Statements

Let n, m denote the number of examples and labels, respectively. Let $\mathcal{X}$ be an example space, and $\mathcal{L} = \{\mathcal{L}_1, \mathcal{L}_2, \cdots, \mathcal{L}_m\}$ be a finite set of class labels. An example $x \in \mathcal{X}$ is described from V views, i.e., $x = \{x^{(j)}|1 \leq j \leq V\}$ where $x^{(j)}$ is the instance in j^{th} view, which is a feature vector. For the $j^{th}(1 \leq j \leq V)$ view, the feature dimension is denoted by d_j. Each example x is associated with a subset of relevant labels denoted by $\mathcal{L}(x) \in 2^{\mathcal{L}}$. In practice, the relevant labels $\mathcal{L}(x)$ can be denoted by a binary label vector $Y(x) = [Y_1(x), Y_2(x), \cdots, Y_m(x)]$, where $Y_i(x) \in \{1, -1\}(1 \leq i \leq m)$ is defined as

$$Y_i(x) = \begin{cases} 1 & \mathcal{L}_i \in \mathcal{L}(x) \\ -1 & \mathcal{L}_i \notin \mathcal{L}(x) \end{cases}$$

Let $\mathcal{Y} = \{1, -1\}^m$ be the set of all such possible labelings.

Given a data set $\mathcal{D} = \{(x, Y(x)) | x \in \mathcal{X}, Y(x) \in \mathcal{Y}\}$, consisting of n_l labeled examples and n_u unlabeled examples which are i.i.d drawn from some unknown distribution $\mathcal{P}$, our goal is to build a multi-label classifier $h : \mathcal{X} \to \mathcal{Y}$ that optimizes some specific evaluation criterion. Without loss of generality, assume that the labels of the first n_l examples are known. We have $n = n_l + n_u$.

For the compactness of representation, we denote the $i^{th}(1 \leq i \leq m)$ label vector of all the examples by $y_i = [Y_i(x_1), Y_i(x_2), \cdots, Y_i(x_n)]^T \in \mathcal{R}^{n\times1}$. Let $f_{ij} \in \mathcal{R}^{n\times1}$ be the prediction vector of all the examples for the $i^{th}(1 \leq i \leq m)$ label and the $j^{th}(1 \leq j \leq V)$ view. Denote $f = \left[f_{11}^T, \cdots, f_{1V}^T, \cdots, f_{m1}^T, \cdots, f_{mV}^T\right]^T \in \mathcal{R}^{nmV\times1}$. Let $\|A\|_F$ be the Frobenius norm for the matrix A.

3.2. Objective

In L^2F, we model the multiple types of relationship including instance-instance, label-label, and view-view correlations in a graph-based framework. The goal is to maximize the smoothness consistency of the instances together with label correlation and view consistency, and simultaneously minimize the empirical loss on the training data. Thus, the objective is to minimize,

$$J(f) = J_C(f) + \alpha J_L(f) + \beta J_V(f) + \gamma J_{emp}(f) \tag{1}$$

where J_C, J_L, J_V, and J_{emp} correspond to instance consistency, label correlation, view consistency, and empirical classification loss, respectively. The non-negative parameters α, β, and γ balance the importance of the corresponding terms. In the following, we will give a detailed setup of each loss function.

Instance Consistency on the Graph: Let $G_j^{(C)} = \{V_j, E_j\}$ be the K-nearest-neighbors graph for the instances in the j^{th} view, where V_j is the set of instances, and E_j is the set of edges. We connect the instance pair $(x_i^{(j)}, x_k^{(j)})$ if $x_k^{(j)}$ is the K-nearest neighbor of $x_i^{(j)}$. The edge weight is determined by the similarity between the two instances denoted by $k(x_i^{(j)}, x_k^{(j)})(1 \leq i, k \leq n)$, which can be estimated using the instance-feature correlation in various ways (e.g., we use Gaussian RBF function). Let $W_j \in \mathcal{R}^{n\times n}$ be the affinity matrix for the instance-instance graph $G_j^{(C)}$ whose (i,k) element is $k(x_i^{(j)}, x_k^{(j)})$. Define the Laplacian matrix $L_j = D^{-\frac{1}{2}}(D - W_j)D^{-\frac{1}{2}}$ where D is a diagonal matrix whose (i,i) element $D_{ii} = \sum\limits_{k=1}^{n} W_j(i,k)$.

Intuitively, similar instances should have similar predictions. Following the random walk model [Zhou et al. 2004], we model the instance consistency as follows,

$$J_C(f) = \sum_{i=1}^{m}\sum_{j=1}^{V} f_{ij}^T L_j f_{ij} = f^T Q_C f \tag{2}$$

where Q_C is a block diagonal matrix with its entry $[Q_C]_{ij,ij} = L_j$ for $1 \leq i \leq m, 1 \leq j \leq V$. Since the Laplacian matrix L_j is positive semi-definite, Q_C is also positive semi-definite.

Label Correlation: Let $G^{(L)} = \{V, E\}$ be the K-nearest-neighbors graph for the labels, where $V = \mathcal{L}$ is the set of labels, and E is the set of edges. We connect the label pair $(\mathcal{L}_i, \mathcal{L}_k)$ if $\mathcal{L}_k$ is the K-nearest neighbor of $\mathcal{L}_i$. The edge weight is determined by the similarity between the two labels denoted by $k(\mathcal{L}_i, \mathcal{L}_k)(1 \leq i, k \leq m)$, which can be estimated using the example-label correlation in various ways (e.g., we use Gaussian RBF function). Let $S \in \mathcal{R}^{m\times m}$ be the affinity matrix for the label-label graph

$G^{(L)}$ whose (i,k) element is $k(\mathcal{L}_i,\mathcal{L}_k)$. The degree of a label $\mathcal{L}_i$ $(1 \le i \le m)$ is defined as $d_i = \sum\limits_{j=1}^{m} S_{ij}$.

Based on the graph $G^{(L)}$, we model the label correlations by requiring that any two label-specific classifiers behave similarly on the same views if the associated labels are similar. In specific, if two labels $\mathcal{L}_i$ and $\mathcal{L}_k$ are similar, the label-specific classifiers f_{ij} and f_{kj} should keep close to each other on the same j^{th} view. Therefore, we model the correlation among multiple labels as follows,

$$J_L\left(f\right) = \sum_{j=1}^{V}\sum_{i,k=1}^{m} S_{ik}\left\|\frac{f_{ij}}{\sqrt{d_i}} - \frac{f_{kj}}{\sqrt{d_k}}\right\|_F^2 = f^T Q_L f \tag{3}$$

where Q_L is a block matrix with its entry,

$$\left[Q_L\right]_{ij,kj} = \begin{cases} 2\left(1 - S_{ik}/d_i\right) I_{n\times n}, & i = k \\ -2S_{ik} I_{n\times n}/\sqrt{d_i d_k}, & i \neq k \end{cases}$$

for $1 \le i,k \le m, 1 \le j \le V$. Since $f^T Q_L f \ge 0$, Q_L is positive semi-definite.

View Consistency: In order to maximize the view consistency, we require that for any view pairs, the difference of predictions resulting from their view-based classifiers should keep small as much as possible. Hence, we model the consistency among multiple views as follows,

$$J_V\left(f\right) = \sum_{i=1}^{m}\sum_{j,k=1}^{V} \left\|f_{ij} - f_{ik}\right\|_F^2 = f^T Q_V f \tag{4}$$

where Q_V is a block matrix with its entry,

$$\left[Q_V\right]_{ij,ik} = \begin{cases} 2\left(V-1\right) I_{n\times n}, & j = k \\ -2I_{n\times n}, & j \neq k \end{cases}$$

for $1 \le i \le m, 1 \le j,k \le V$. Since $f^T Q_V f \ge 0$, Q_V is positive semi-definite.

Empirical Loss: Various empirical loss functions, such as hinge loss, least square loss, logistic loss, and etc., can be used to measure the consistency with known label information.

Overall Objective: In summary, the overall goal is to minimize the following objective function:

$$\begin{array}{l} J\left(f\right) = J_C\left(f\right) + \alpha J_L(f) + \beta J_V\left(f\right) + \gamma J_{emp}\left(f\right) \\ = f^T\left(Q_C + \alpha Q_L + \beta Q_V\right) f + \gamma J_{emp}\left(f\right) \\ = f^T Q f + \gamma J_{emp}\left(f\right) \end{array} \tag{5}$$

where $Q = Q_C + \alpha Q_L + \beta Q_V$.

A nice property of the proposed method is that its objective function is joint convex as shown in the following theorem.

THEOREM 3.1 (CONVEXITY). *When using convex empirical loss, the objective function in Eq. 5 is convex with respect to f.*

PROOF. Since all of Q_C, Q_L, and Q_V are positive semi-definite, Q is also positive semi-definite. Hence, $f^T Q f$ is convex with respect to f. Therefore, when the empirical loss function $J_{emp}(f)$ is convex, the overall objective function in Eq. 5 is also convex. □

3.3. Optimization

When using least square loss as empirical loss function, the objective function in Eq. 5 can be solved analytically. For the least square loss, we have

$$J_{emp}\left(f\right) = \sum_{i=1}^{m}\sum_{j=1}^{V} \left\| f_{ij} - y_i \right\|_F^2 = f^T Q_{emp} f - 2f^T p + q \tag{6}$$

where Q_{emp} is block diagonal matrix with its entry $\left[Q_{emp}\right]_{ij,ij} = I_{n\times n}$, p is a block vector with its entry $\left[p\right]_{ij} = y_i$, and q is a constant block vector with its entry $\left[q\right]_{ij} = y_i^T y_i \cdot 1_{n\times 1}$ for $1 \le i \le m, 1 \le j \le V$. Obviously, Q_{emp} is positive semi-definite.

Then, the objective function in Eq. 5 can be rewritten into

$$\begin{aligned} &J\left(f\right) = J_C\left(f\right) + \alpha J_L(f) + \beta J_V\left(f\right) + \gamma J_{emp}\left(f\right) \\ &= f^T\left(Q_C + \alpha Q_L + \beta Q_V + \gamma Q_{emp}\right) f - 2\gamma f^T p + \gamma q \\ &= f^T Q_A f - 2\gamma f^T p + \gamma q \end{aligned} \tag{7}$$

where $Q_A = Q_C + \alpha Q_L + \beta Q_V + \gamma Q_{emp}$. Obviously, Q_A is positive semi-definite. By taking derivative of Eq. 7 with respect to f, we have

$$\nabla_f J\left(f\right) = 2Q_A f - 2\gamma p = 0 \Rightarrow f^* = \gamma Q_A^{-1} p \tag{8}$$

Optimization using block coordinate descent: Since $Q_A \in \mathcal{R}^{nmV\times nmV}$, the space complexity of the above method is $O\left(n^2m^2V^2\right)$. To reduce the space complexity, we resort to block coordinate descent (BCD) method [Luo and Tseng 1992; Tseng 2001] to iteratively solve the optimization problem. We first rewrite the objective in Eq. 7 as follows

$$\begin{aligned} &J\left(f\right) = J_C\left(f\right) + \alpha J_L(f) + \beta J_V\left(f\right) + \gamma J_{emp}\left(f\right) \\ &= \sum_{i=1}^{m}\sum_{j=1}^{V} f_{ij}^T L_j f_{ij} + \alpha \sum_{j=1}^{V}\sum_{i,k=1}^{m} S_{ik}\left(\frac{f_{ij}^T f_{ij}}{d_i} - \frac{2f_{ij}^T f_{kj}}{\sqrt{d_i d_k}} + \frac{f_{kj}^T f_{kj}}{d_k}\right) \\ &+\beta \sum_{i=1}^{m}\sum_{j,k=1}^{V}\left(f_{ij}^T f_{ij} - 2f_{ij}^T f_{ik} + f_{ik}^T f_{ik}\right) \\ &+\gamma \sum_{i=1}^{m}\sum_{j=1}^{V}\left(f_{ij}^T f_{ij} - 2f_{ij}^T y_i + y_i^T y_i\right) \end{aligned} \tag{9}$$

By setting the first-order derivative of Eq. 9 with respect to f_{ij} $\left(1 \le i \le m, 1 \le j \le V\right)$ to zero, we have,

$$H_{ij} f_{ij}^* = p_{ij} \tag{10}$$

where

$$H_{ij} = 2L_j + \left[4\alpha\left(1 - \frac{S_{ii}}{d_i}\right) + 4\beta\left(V-1\right) + 2\gamma\right] I_{n\times n}$$

and

$$p_{ij} = 4\alpha \sum_{k=1,k\neq i}^{m} \frac{S_{ik}}{\sqrt{d_i d_k}} f_{kj} + 4\beta \sum_{k=1,k\neq j}^{V} f_{ik} + 2\gamma y_i$$

Prediction: For the test example, the final prediction is the expectation of predictions resulting from view-based classifiers. For the example x, the prediction for its i^{th} label is as follows

$$h_i\left(x\right) = \text{sgn}\left(\frac{1}{V}\sum_{j=1}^{V} f_{ij}^*\left(x\right)\right) \tag{11}$$

ALGORITHM 1: The L^2F Algorithm based on BCD

```
Input:
  multi-view multi-label dataset $\mathcal{D} = \{(x, y(x)|x \in \mathcal{X}, y(x) \in \mathcal{Y}\}$,
  parameters: $\alpha, \beta, \gamma, n_{iter}$.
Output:
  predicted labels for the test data.
Initialize $y_i (1 \leq i \leq m)$ for each example to its true label for training data, 0 for test data;
Compute the instance-instance Laplacian matrices $\{L_j | 1 \leq j \leq V\}$;
Compute the label-label affinity matrix $S$;
for $t = 1 : n_{iter}$ do
  for $i = 1 : m$ do
    for $j = 1 : V$ do
      Keep the other block fixed, and update $f_{ij}$ by Eq. 10;
    end for;
  end for;
end for;
return predicted labels for the test data by using Eq. 11.
```

The L^2F algorithm based on block coordinate descent method is shown in Algorithm 1. Next, we will discuss the convergence property, the time and space complexity of the proposed algorithm.

THEOREM 3.2 (CONVERGENCE). *The L^2F algorithm converges to the global optimum.*

PROOF. By taking second-order derivative of Eq. 9 with respect to f_{ij}, we have $\nabla^2_{f_{ij}} J(f) = H_{ij}$. Obviously, the Hessian matrix H_{ij} is positive semi-definite. Therefore, the objective J_f is block-wise convex with respect to $f_{ij}\,(1 \leq i \leq m, 1 \leq j \leq V)$. According to [Luo and Tseng 1992], block coordinate descent method converges to the local optimum when the objective is block-wise convex.

Based on Theorem 3.1, the overall objective in Eq. 5 is joint convex with respect to f. For a joint convex function, a local minimum is also a global minimum. Hence, our algorithm based on block coordinate descent method converges to the global optimum. □

The space complexity for instance-instance Laplacian matrix and label-label affinity matrix are $O(n^2)$ and $O(m^2)$, respectively. To sum up, the space complexity of the L^2F algorithm is $O(Vn^2 + m^2)$.

To solve Eq. 10, a straightforward way is to compute $f^*_{ij} = H^{-1}_{ij} p_{ij}$. Based on the selected matrix inversion algorithm [Don Coppersmith 1990], the time complexity of computing the inverse matrix is $O(n^c)$ where $2.373 \leq c \leq 3$. Hence, the time complexity for the whole algorithm is $O(n_{iter} V m n^c)$. However, if L_j is a large-size matrix, the computation of inverse matrix is inefficient. Note that L_j is sparse since it is the Laplacian matrix based on K-nearest-neighbors graph. Therefore, for the sparse and large-size matrix L_j, it is much more efficient to solve Eq. 10 by the iterative conjugate gradient type algorithms [Sun et al. 2009] such as LSQR [Paige and Saunders 1982], which can take advantage of the sparsity to accelerate the convergence procedure.

3.4. Learning from Data with Missing Views and Labels

Many real-world applications often face the challenges of data with missing views or labels. Generally speaking, missing label means that some labels of the examples are incomplete, and missing view means that all the features in certain view are missing for some examples. Given the incomplete views or labels, the learning problem becomes more challenging. A natural extension of our method can deal with these missing value problems.

Missing Views: To tackle the missing view problem, we need to change the formulation of the instance-instance affinity matrices $\{W_j|1 \leq j \leq V\}$.

Suppose that the j^{th} view information is missing for the instance $x_k^{(j)}$, which is denoted as $x_k^{(j)} \in \varnothing$. In this case, we compute the instance-instance similarity regarding the instance $x_k^{(j)}$ by borrowing the strength from other views as follows

$$W_j(i,k) = \frac{1}{V_k} \sum_{v=1, x_k^{(v)} \notin \varnothing}^{V} k(x_i^{(v)}, x_k^{(v)})$$

where V_k is the number of non-missing views for the example x_k. Note that we suppose that each example has at least one non-missing view.

Missing Labels: To tackle the missing label problem, we need to change the computation of the label-label affinity matrix S.

Suppose that for example x, its k^{th} label $Y_k\,(x)$ is missing. In this case, we estimate its label by borrowing the strength from its nearest neighbors. First, by letting $\alpha = \beta = 0$ in Eq. 10 and averaging the predictions from view-based classifiers, we can obtain the predictions as follows

$$f_k = \frac{\gamma}{V} \sum_{j=1}^{V} (L_j + \gamma I)^{-1} y_k$$

Then, the missing label for example x can be estimated as $max\{0, sgn(f_k(x))\}$. Finally, we can use the smoothed labels to re-compute the affinity matrix S.

Co-existing of Missing Views and Labels: When the missing views and missing labels co-exist, the learning task becomes more difficult. But our proposed model can handle this issue by simply combining the above two operations.

4. THEORETIC ANALYSIS

In this section, we analyze the generalization performance of the proposed approach, which shows the benefits of simultaneously modeling label and feature heterogeneity. To be specific, we will demonstrate that the upper bound of empirical Rademacher complexity together with the error bound of the proposed L^2F model can be reduced by incorporating the label correlation and enhancing the view consistency.

We first construct a Reproducing Kernel Hilbert Space (RKHS) for the proposed method, and then analyze its Rademacher complexity and error bound.

4.1. An RKHS

Let $\mathcal{H}$ be the space of functions with the norm defined as $\|f\|_{\mathcal{H}}^2 = f^T Q_C f$. Based on $\mathcal{H}$, we define $\tilde{\mathcal{H}}$ to be the space of functions with the norm $\|f\|_{\tilde{\mathcal{H}}}^2 = \|f\|_{\mathcal{H}}^2 + \alpha f^T Q_L f + \beta f^T Q_V f = f^T Q f$. Suppose that Q_C, Q_L, Q_V, Q are invertible [1]. The following theorem will show that both $\mathcal{H}$ and $\tilde{\mathcal{H}}$ are RKHS.

THEOREM 4.1 (RKHS). *Both $\mathcal{H}$ and $\tilde{\mathcal{H}}$ are RKHS with kernel matrix $K = Q_C^{-1}$, and $\tilde{K} = Q^{-1} = [Q_C + \alpha Q_L + \beta Q_V]^{-1}$, respectively.*

PROOF. Since Q_C is positive semi-definite, according to Theorem 4 in [Smola and Kondor 2003], $\mathcal{H}$ is a Reproducing Kernel Hilbert Space with the kernel matrix $K =$

[1] When it is singular, a practical approach is to add a small regularization term to it such as $\lambda I(\lambda \geq 0)$.

Q_C^{-1}. Likewise, since Q is positive semi-definite, $\tilde{\mathcal{H}}$ is also a Reproducing Kernel Hilbert Space with the kernel matrix $\tilde{K} = Q^{-1} = \left[Q_C + \alpha Q_L + \beta Q_V\right]^{-1}$. □

Hence, based on Theorem 4.1, the overall objective in Eq. 5 can be reduced to standard supervised learning problem:

$$f^* = \arg\min_{f \in \tilde{\mathcal{H}}} \|f\|_{\tilde{\mathcal{H}}}^2 + \gamma J_{emp}(f) \tag{12}$$

4.2. Generalization Performance

Let $\mathcal{F} := \{f \in \tilde{\mathcal{H}} : \|f\| \leq r\}$ denote the ball of radius r in $\tilde{\mathcal{H}}$. According to Theorem 4.12 in [Shawe-Taylor and Cristianini 2004], we can obtain the following theorem regarding the empirical Rademacher complexity of the proposed method.

THEOREM 4.2 (RADEMACHER COMPLEXITY). *The empirical Rademacher complexity of the proposed L^2F method is upper bounded by:*

$$\hat{R}(\mathcal{F}) \leq \frac{2r}{nmV} \sqrt{tr\left(\left[Q_C + \alpha Q_L + \beta Q_V\right]^{-1}\right)} \tag{13}$$

Note that α and β balance the importance of label correlation and view consistency in the overall objective, respectively. For simplicity, let $\hat{R}(\mathcal{F}_{\alpha=\beta=0})$ and $\hat{R}(\mathcal{F}_{\beta=0})$ correspond to the empirical Rademacher complexity for $\alpha = \beta = 0$, and $\beta = 0$ in Eq. 13, respectively.

THEOREM 4.3 (RADEMACHER COMPLEXITY REDUCTION). *For the proposed L^2F method, the upper bound of empirical Rademacher complexity can be reduced by incorporating the label correlation and enhancing view consistency, i.e.,*

$$\hat{R}(\mathcal{F}) \leq \hat{R}(\mathcal{F}_{\beta=0}) \leq \hat{R}(\mathcal{F}_{\alpha=\beta=0}) \tag{14}$$

PROOF. Note that both of Q_C and Q_L are positive semi-definite and symmetric. Suppose Q_C has eigenvalues

$$\upsilon_1 \geq \cdots \geq \upsilon_t \geq 0$$

and αQ_L has eigenvalues

$$\rho_1 \geq \cdots \geq \rho_t \geq 0$$

and $Q_C + \alpha Q_L$ has eigenvalues

$$\mu_1 \geq \cdots \geq \mu_t \geq 0$$

According to Weyl's inequality, the following inequality holds for $i = 1, \cdots, t$:

$$\upsilon_i + \rho_t \leq \mu_i$$

Then, we have

$$tr\left(\left[Q_C + \alpha Q_L\right]^{-1}\right) = \sum_{i=1}^{t} \frac{1}{\mu_i} \leq \sum_{i=1}^{t} \frac{1}{\upsilon_i} = tr\left(Q_C^{-1}\right) \tag{15}$$

Likewise, we have

$$tr\left(\left[Q_C + \alpha Q_L + \beta Q_V\right]^{-1}\right) \leq tr\left(\left[Q_C + \alpha Q_L\right]^{-1}\right) \tag{16}$$

By Eq. 15 and Eq. 16, we can reach the final conclusion as follows:

$$\hat{R}(\mathcal{F}) \leq \hat{R}(\mathcal{F}_{\beta=0}) \leq \hat{R}(\mathcal{F}_{\alpha=\beta=0})$$

□

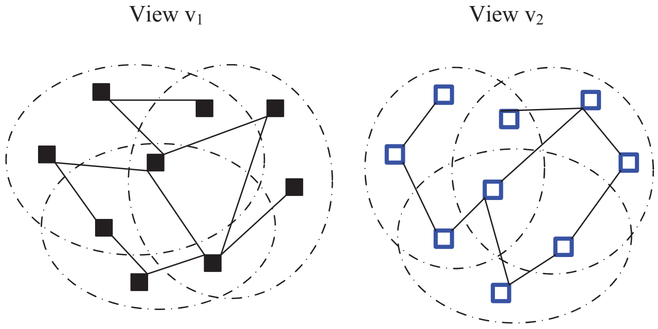


Fig. 1: A hypergraph model where the nodes represent the instances, and the simple edges (solid line) encode the instance-instance correlations, and the hyperedges (dotted ellipse) encode the instance-label correlations. The graphs corresponding two different views are shown in the left and right panels, respectively.

Note that Q_L encodes the correlation among multiple labels, while Q_V encodes the consistency among multiple views. From Theorem 4.3, we can see that the Rademacher complexity of the proposed L^2F method decreases by incorporating the label correlation and view consistency into the overall objective as defined in Eq. 5.

An application of Theorem 4.9 in [Shawe-Taylor and Cristianini 2004] together with Theorem 4.2 show that:

THEOREM 4.4 (ERROR BOUND). *With probability at least $1 - \delta(0 \leq \delta \leq 1)$, the generalization error of prediction function f is upper-bounded as follows:*

$$\mathcal{E}_{\mathcal{D}}[f(x)] \leq J_{emp}(f) + \frac{2r}{nmV}\sqrt{tr\left(Q^{-1}\right)} + 3\sqrt{\frac{\ln\left(2/\delta\right)}{2nmV}} \tag{17}$$

Theorem 4.4 suggests that the error bound of our proposed method can be improved due to the reduction of Rademacher complexity.

5. MODEL GENERALIZATION AND DISCUSSION

In this section, we extend the model proposed in Section 3 to a generalized framework which uses the hypergraph to model multiple types of relationships among the objects including the instance-instance, instance-label, and view-view correlations. Some specific instantiations of the generalization framework will also be discussed, as well as the relationships between them.

Let $G = \{N, E\}$ be the graph where N is the set of instances, and E is the set of edges. E consists of two subset of edges, i.e., $E = E_c \cup E_l$, where E_c is the set of simple edges and E_l is the set of hyperedges. Thus G can be viewed as the combination of two subgraphs, i.e., $G = G_c \cup G_l$, where $G_c = \{N, E_c\}$ and $G_l = \{N, E_l\}$ are the simple graph and hypergraph, respectively. In comparison with simple graph which can only represent the pairwise relationship, the hypergraph is capable of representing more complex relationships among the objects [Zhou et al. 2007]. The illustration of the hypergraph framework is shown in Figure 1. Note that the simple edges (solid line) encode the instance-instance correlations, while the hyperedges (dotted ellipse) encode the instance-label correlations. The graphs corresponding to different views have the same hyperedge set but different simple edge set.

For an edge $e \in E_c$ connecting the instance pair (x_i, x_j), its weight is determined by the similarity between the two instances. The similarity matrix for the subgraph G_c is denoted by W_c. The degree matrix for G_c is denoted by D_c which is diagonal matrix containing the degrees of the nodes, i.e., $D_c(i,i) = \sum_j W_c(i,j)$.

Each hyperedge $e \in E_l$ corresponds to a label and consists of all the instances relevant to this label. The degree of a hyperedge e, denoted as $\delta(e)$, is the number of nodes in e. The degree of a node $v \in N$, denoted as $\delta(v)$, is defined as $\delta(v) = \sum_{\{e \in E_l | v \in e\}} w(e)$ where $w(e)$ is the weight associated with the hyperedge e. The diagonal matrix forms for $\delta(v)$, $w(e)$ are denoted as D_l, H_l, respectively. The node-edge incidence matrix $C_l \in \mathcal{R}^{|N| \times |E_l|}$ is defined as $C_l(v,e) = 1$ if $v \in e$, and $C_l(v,e) = 0$ otherwise. The similarity matrix for the subgraph G_l is defined as $W_l = C_l H_l C_l^T$.

Based on the graph G, we propose a generalized framework to model both label and feature heterogeneity. Note that the number of instances, features, and labels are denoted as n, d, and m, respectively. Let $X \in \mathcal{R}^{d \times n}$, $Y \in \mathcal{R}^{n \times m}$, and $F \in \mathcal{R}^{n \times m}$ be matrices for the instances, labels, and predictions, respectively. The objective is to minimize:

$$J(X, Y, F, G) = \Omega(X, Y, F, G) + \gamma J_{emp}(Y, F) \tag{18}$$

where $J_{emp}(Y, F)$ is the empirical classification loss, and $\Omega(X, Y, F, G)$ is the regularization term used to model multiple types of relationships among the objects including the instance-instance, instance-label, and view-view correlations, which are closely related to the structure of the graph G. Next we will introduce two instantiations of the generalized framework defined in Eq. 18. For simplicity, we only consider the single-view scenario in the following discussions.

One instantiation is to model the instance-instance and instance-label correlations on the subgraphs, G_c and G_l, respectively. Note that the subgraph G_c encodes the instance-instance correlation. Intuitively, similar instances should have similar predictions. Therefore, we model the instance consistency as $J_c(F) = tr(F^T L_c F)$ where L_c is the Laplacian matrix for the graph G_c. For the subgraph G_l, if two instances share more common hyperedges, they will be more similar. Likewise, if two hyperedges share more common instances, they will also be more similar. In this regard, the hypergraph G_l captures the instance-label correlation. Hence we can model instance-label correlations as $J_l(F) = tr(F^T L_l F)$ where L_l is the Laplacian matrix for the graph G_l. In this case, Eq. 18 can be rewritten into

$$J(X, Y, F, G) = \mu tr(F^T L_c F) + \omega tr(F^T L_l F) + \gamma J_{emp}(Y, F) \tag{19}$$

where μ and ω are non-negative parameters to balance the contributions of different terms. We can prove that Eq. 1 is equivalent to Eq. 19 in single-view setting after some algebraic operations.

Another instantiation is to model both instance-instance and instance-label correlations in the hypergraph G. For the hypergraph G, we can consider putting different weights on the edges to encode our prior knowledge on relations among the subgraphs. For example, we can assign the weights to the edges of the graph G such as $\begin{bmatrix} \mu H_c & 0 \\ 0 & \omega H_l \end{bmatrix}$. We propose to model both instance-instance and instance-label correlations encoded in the graph G as $J(F) = tr(F^T L F)$ where L is the Laplacian matrix for the graph G. In this case, Eq. 18 can be reformulated into

$$J(X, Y, F, G) = tr(F^T L F) + \gamma J_{emp}(Y, F) \tag{20}$$

Since the solutions to both Eq. 19 and Eq. 20 are closely related to the corresponding Laplacian matrices, i.e., L_c, L_l and L. Next we will discuss some relationships between these two instantiations in term of their Laplacian matrices.

THEOREM 5.1. *For $G = G_c \cup G_l$ with the diagonal weight matrix for the hypergraph as $H = \begin{bmatrix} \mu H_c & 0 \\ 0 & \omega H_l \end{bmatrix}$, the unnormalized Laplacian matrix of the hypergraph is weighted sum of the unnormalized Laplacian matrices of the subgraphs, i.e.,*

$$L = \mu L_c + \omega L_l \tag{21}$$

PROOF. For the subgraph G_l, the unnormalized Laplacian matrix is defined as $L_l = D_l - W_l = D_l - C_l H_l C_l^T$. For the subgraph G_c, the unnormalized Laplacian matrix is defined as $L_c = D_c - W_c$. Because simple edge is a special case of hyperedge, W_c can be rewritten into $W_c = C_c H_c C_c^T$ in the same form with W_l, where C_c is the node-edge incidence matrix and H_c is the weight matrix of the edges for graph G_c.

For the hypergraph G, we have

$$\begin{aligned}
&L = D - W = D - CHC^T \\
&= \mu D_c + \omega D_l - \left[\, C_c \;\; C_l \,\right] \begin{bmatrix} \mu H_c & 0 \\ 0 & \omega H_l \end{bmatrix} \begin{bmatrix} C_c^T \\ C_l^T \end{bmatrix} \\
&= \mu D_c + \omega D_l - \left(\mu C_c H_c C_c^T + \omega C_l H_l C_l^T\right) \\
&= \mu \left(D_c + C_c H_c C_c^T\right) + \omega \left(D_l + C_l H_l C_l^T\right) \\
&= \mu L_c + \omega L_l
\end{aligned}$$

□

The normalized Laplacian matrix is defined as $L^{sym} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}} = I - D^{-\frac{1}{2}} L D^{-\frac{1}{2}}$. However, the equation $L^{sym} = \mu L_c^{sym} + \omega L_l^{sym}$ is usually not satisfied. One special case is when $D_c = D_l$, we have $L^{sym} = 2(\mu L_c^{sym} + \omega L_l^{sym})$, which follows from

$$\begin{aligned}
&\mu L_c^{sym} + \omega L_l^{sym} = \mu D_c^{-\frac{1}{2}} L_c D_c^{-\frac{1}{2}} + \omega D_l^{-\frac{1}{2}} L_l D_l^{-\frac{1}{2}} \\
&= \frac{1}{2} D^{-\frac{1}{2}} \left(\mu L_c + \omega L_l\right) D^{-\frac{1}{2}} = \frac{1}{2} D^{-\frac{1}{2}} L D^{-\frac{1}{2}} \\
&= \frac{1}{2} L^{sym}
\end{aligned}$$

The above discussions indicate that if the unnormalized Laplacian matrices are used in both Eq. 19 and Eq. 20, they will lead to the equivalent solutions. But such a conclusion cannot be made for the normalized Laplacian matrices.

It is worth noting that a variety of instantiations of the generalization framework are possible depending on the structure of the hypergraph, as well as the weights putting on the hyperedges. The strength of the proposed generalization framework is that it allows us to model multiple types of correlations, such as instance-instance, label-label, and view-view correlations in a principled way by encoding our prior knowledge of the learning task into the hypergraph structure.

6. EXPERIMENTS

The theoretical analysis in Section 4 shows the advantages of modeling label and feature heterogeneity in our framework. In this section, we aim to empirically verify the effectiveness of the proposed algorithm in comparison with a variety of state-of-the-art approaches.

6.1. Datasets and Setup

Four multi-label datasets on the biomedical domain are used to test the performance of our proposed algorithm.

The first dataset is the Medical dataset [Pestian et al. 2007]. The Computational Medical Center organized Medical NLP Challenge [2] with a rich set of medical text

[2] http://www.computationalmedicine.org/challenge/index.php

Table I: Statistics of Different Datasets.

Dataset	Instances	Features	Labels	Cardinality	Density	Diversity	Training	Test
Medical	978	1449	45	1.245	0.028	94	333	645
Tudiabetes	10521	9064	20	2.715	0.136	1665	7364	3157
Genbase	662	1186	27	1.252	0.046	32	463	199
Diabetes	8812	16	30	1.290	0.043	45	6168	2644

corpus. This dataset is actually a collection of patient symptom histories, diagnosis and prognoses reported to the insurance companies.

The Tudiabetes dataset is a collection of about 300,000 posts belonging to 21,285 threads crawled from Tudiabetes forum [3]. The posts are organized into 20 categories such as type 1 diabetes, type 2 diabetes, insulin pump, etc. Each forum user may send the posts in multiple categories. In this dataset, users are instances and categories are labels. This is a multi-label setting, since the same user can belong to many categories. The features for each user are characterized by the text of his posts.

Both of the Medical and Tudiabetes datasets are text data. Based on the raw text, we generate two views of features as follows: one corresponds to the TF-IDF features; another corresponds to the latent topics obtained by applying probabilistic latent semantic analysis on the term counts, where the number of latent topics is set to 100.

Genbase [Diplaris et al. 2005] is a biomedical dataset for protein function classification. In Genbase, each instance is a protein, and each label is a protein class which it belongs to. The function of a protein is directly related to its structure. The proteins are represented with two views of features, i.e., patterns and profiles. Patterns are short amino acid chains that have a specific order, while profiles are computational representations of multiple sequence alignments using hidden Markov models. Proteins are grouped into several families according to the functions they perform. All proteins contained in a family feature a certain structural relation, thus having similar properties. Some proteins belong to more than one class, thus the problem could be defined as a multi-label classification problem.

The Diabetes dataset was obtained from a big biomedical company. This dataset is a collection of symptom and treatment information regarding diabetes patients. Each patient may receive multiple treatments which can be regarded as the labels. The patients are described with two views of features measuring the long and short term drug impact, respectively. For example, glycated hemoglobin (A1c or HbA1c) is a measurement of blood glucose level. A1c measures the average past 3 month blood glucose value. Comparing with the A1c, fasting blood glucose (FBG) measure the short term drug impact. The FBG measures the current fasting blood glucose value. Both views of measurements are key variables for treating diabetes patients.

Both Medical and Genbase datasets are available online [4]. Table I shows the properties of different datasets. Label cardinality is the average number of labels per instance. Accordingly, label density normalizes label cardinality by the the number of labels. Label diversity is the number of distinct label combinations observed in the dataset [Zhang and Zhou 2014].

6.2. Evaluation Metric and Comparison Algorithms

We use both F_1-score and accuracy as the evaluation metrics to test the performance of the proposed method.

[3] http://www.tudiabetes.org/forum

[4] http://mulan.sourceforge.net/datasets.html

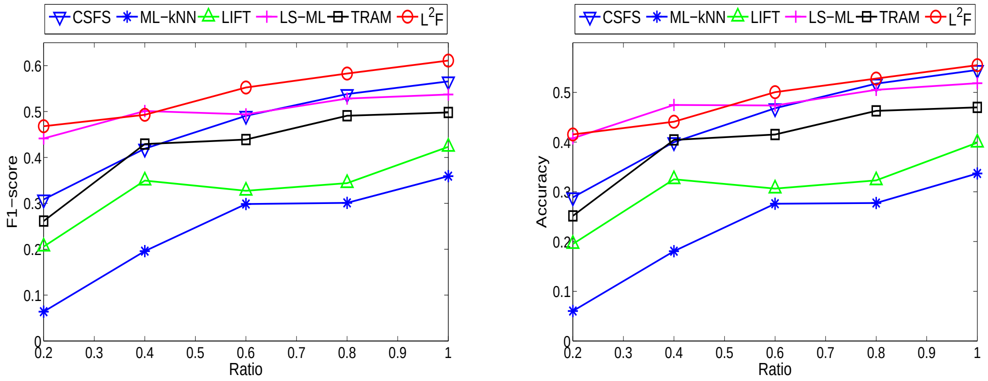


Fig. 2: Performance (left: F_1-score, right: accuracy) varies with ratio on Medical.

F_1-score [Godbole and Sarawagi 2004] is the harmonic mean of precision and recall, which is defined as follows:

$$F_1 = \frac{1}{n_u} \sum_{k=n_l+1}^{n_l+n_u} \frac{2\,|\mathcal{L}(x_k) \cap \mathcal{Z}(x_k)|}{|\mathcal{L}(x_k)| + |\mathcal{Z}(x_k)|}$$

where $\mathcal{Z}(x) = \{\mathcal{L}_i | h_i(x) = 1, 1 \leq i \leq m\}$ is the predicted label set for example x. Note that the larger value of F_1-score is indicating better performance.

Accuracy [Godbole and Sarawagi 2004] for each instance is defined as the proportion of the predicted correct labels to the total number of labels for that instance. Overall accuracy is the average across all test instances, which is defined as follows,

$$Accuracy = \frac{1}{n_u} \sum_{k=n_l+1}^{n_l+n_u} \frac{2\,|\mathcal{L}(x_k) \cap \mathcal{Z}(x_k)|}{|\mathcal{L}(x_k) \cup \mathcal{Z}(x_k)|}$$

Note that the larger value of accuracy is indicating the better performance.

The proposed L^2F method is compared with a variety of multi-label learning algorithms including: 1) first-order approach ML-kNN [Zhang and Zhou 2007]; 2) feature-based approach LIFT [Zhang 2011]; 3) subspace learning approach LS-ML [Ji et al. 2008]; 4) transductive learning approach TRAM [Kong et al. 2013]; 5) semi-supervised method CSFS [Chang et al. 2014].

L^2F is given the multi-view data, whereas the other methods are given the concatenated features from all the views. The parameters are tuned for each algorithm using cross-validation on the training data. Then, the models are builded on the training data with the optimal parameters, and then evaluated on the test data. We repeat the experiments ten times for each dataset and report the average performance.

6.3. Performance Evaluation

The comparison results for the datasets are shown in Figures 2-5. In each figure, x-axis represents the ratio which is used to randomly sample a subset of instances from the training data, and y-axis denotes the performance such as F_1-score and accuracy.

First of all, a common trend observed from these figures is that the performance of all the algorithms usually perform better when the ratio increases. It is reasonable because more training instances help build a robust classifier.

The results show that L^2F performs better than the other algorithms in most cases. The performance of ML-kNN [Zhang and Zhou 2007] is somewhat limited due to the

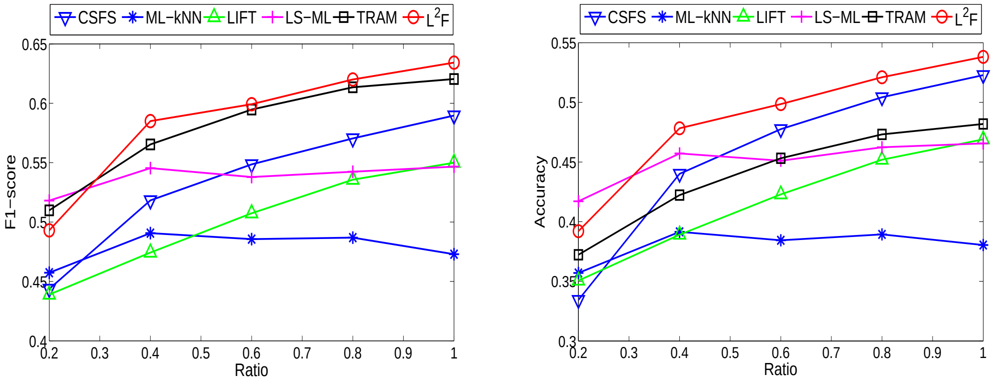


Fig. 3: Performance (left: F_1-score, right: accuracy) varies with ratio on TuDiabetes.

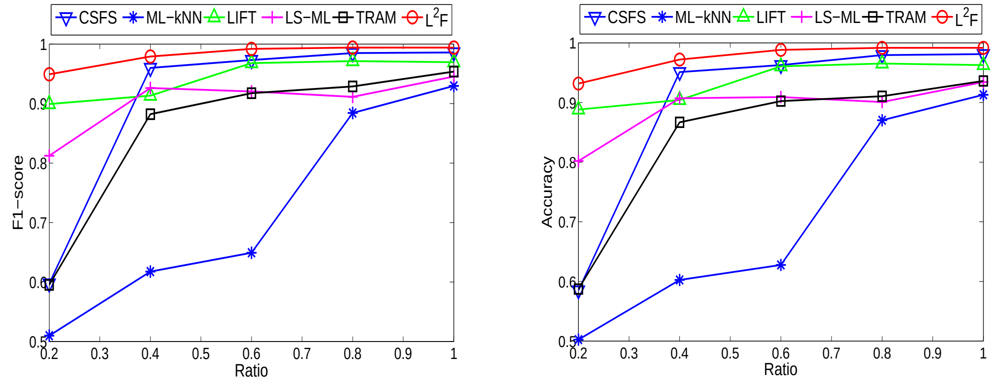


Fig. 4: Performance (left: F_1-score, right: accuracy) varies with ratio on Genbase.

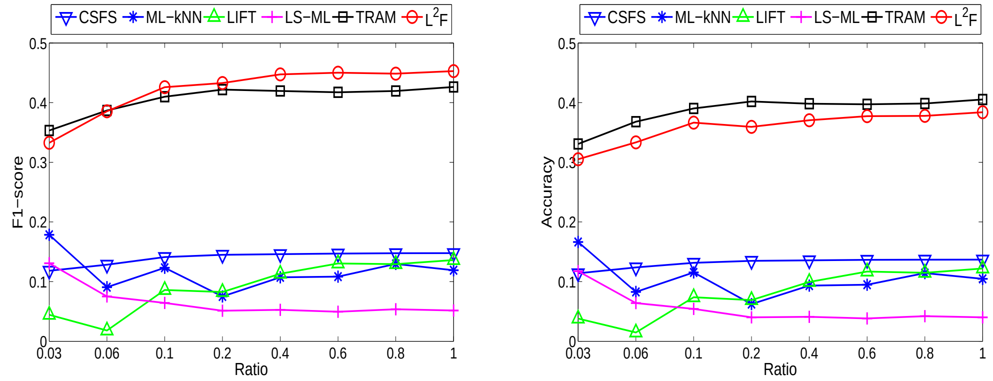


Fig. 5: Performance (left: F_1-score, right: accuracy) varies with ratio on Diabetes.

fact that ML-kNN is a first-order approach which ignores the correlation among multiple labels. In contrast, all the other algorithms usually perform better than ML-kNN by leveraging the label correlations. In principle, the classifiers induction process of LIFT [Zhang 2011] is similar to ML-kNN. But LIFT improves upon ML-kNN by building the classifier on each label with label-specific features instead of the original ones. LS-ML [Ji et al. 2008] learns a common subspace shared among multiple labels, which helps improve the learning performance for the multi-label data. However, since its objective function is non-convex, the performance of LS-ML would be limited by the local optimum problem. Different from other approaches, TRAM [Kong et al. 2013] is a tranductive multi-label learning method which tries to exploit the information from both labeled and unlabeled data. They formulate the transductive multi-label classification as an optimization problem of estimating label concept compositions. CSFS [Chang et al. 2014] is also a semi-supervised method which conducts the sparse feature selection by leveraging the unlabeled data. The results show that unlabeled data can provide helpful information to build the multi-label classifier.

In comparison with the other methods, the key advantage of L^2F is that it models both label and feature heterogeneity in a principled framework. First, by leveraging the consistency among multiple views, the view-based classifiers can mutually improve each other. On the contrary, all the other comparison methods do not consider the view consistency, simply concatenating features from different views cannot gain much additional improvement. Second, by considering the correlation among multiple labels, the performance of label-specific classifiers in L^2F can benefit from each other. In the next subsection, we will also show how the performances of L^2F vary with the trade-off parameters, α and β, which control the weight of label correlation and view consistency, respectively. Another competency of L^2F is that it is capable of finding the global optimum due to the joint convexity of the objective function.

In addition, we have a few different observations on the Diabetes dataset. First, we find that the performance of the proposed algorithm is not very sensitive to different ratios in the range of 0.2-1.0. Hence, we gradually decrease the ratio from 0.2 to 0.03, and find that its performance begins to worsen. It suggests that the proposed method can perform well on this data set even though very few instances are used for training. Second, the performance of four algorithms, i.e., ML-kNN, LS-ML, LIFT and CSFS, are poor, indicating that this is a more challenging task. In contrast, both TRAM and L^2F perform better than the other methods. This might due to the fact both TRAM and L^2F take advantage of the unlabeled data. TRAM utilizes the unlabeled data in a tranductive way, while L^2F leverages the smoothness consistency among the nearest instances. But all things have pros and cons. The result of CSFS suggests that the performance cannot be guaranteed to improve with the help of unlabeled data.

6.4. Parameter Sensitivity

We study the parameter sensitivity on the Medical dataset. α and β are used to weigh the importance of label correlation and view consistency, respectively. We tune α and β on the grid $2^{[-4:1:4]}$. The left panel of Figure 6 shows the performance varies with α. In comparison with $\alpha = 0$, the algorithm performs better when α increases, and the best result occurs when $\alpha = 1$, which indicates that modeling label correlation could significantly improve the multi-label learning performance. However, if α is very large such as $\alpha = 16$, the label correlation part will dominate the entire objective function, making the model hard to keep certain level of accuracy. Nevertheless, the performance is robust over a wide range of values for α. The right panel of Figure 6 shows a similar trend for β, which suggests that the learner could benefit from enhancing the consistency among multiple views.

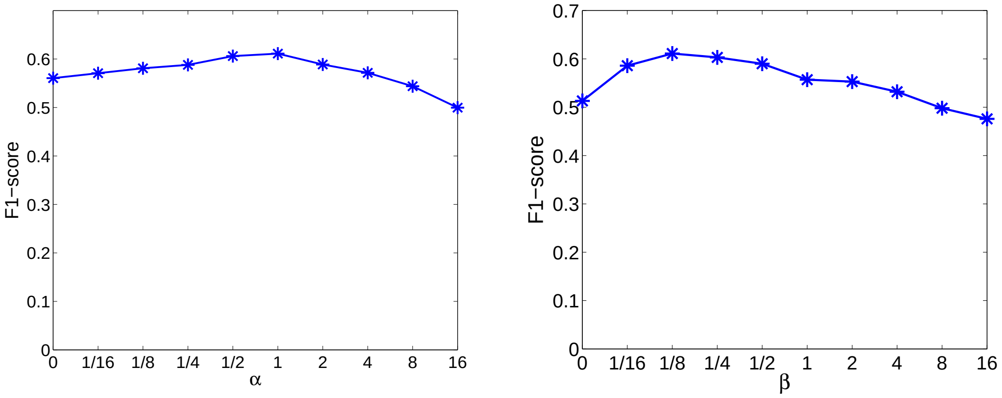


Fig. 6: F_1-score varies with α (left) and β (right) (log_2 scale).

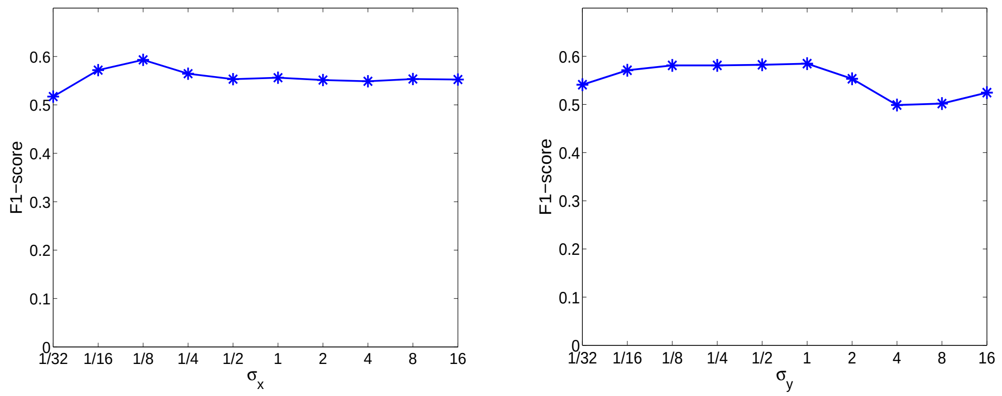


Fig. 7: F_1-score varies with σ_x (left) and σ_y (right) (log_2 scale).

We use the RBF kernel to estimate both the instance-instance and label-label similarities, which is defined as $k(x_i, x_j) = \exp\left(-\frac{|x_i - x_j|^2}{2\sigma^2}\right)$ where σ is the width parameter. We tune the width for instance kernel denoted by σ_x and label kernel denoted by σ_y on the grid $2^{[-5:1:5]}$. The performance varying with σ is shown in Figure 7. The results show that the performance is robust over a wide range of σ values.

γ is used to control the weight of empirical loss. We tune γ on the grid $2^{[-4:1:4]}$. The result is shown in the left panel of Figure 8. As expected, the performance is poor when $\gamma = 0$, and the F_1-score first increases and then decreases when γ is increased.

As a result, we tune the parameters on each dataset using standard cross-validation.

6.5. Convergence

The L^2F algorithm uses an iterative procedure to solve the optimization problem. Theorem 3.2 guarantees that L^2F converges to a global optimum. Here, we empirically study the convergence property of L^2F algorithm on the Medical dataset. The result is shown in the right panel of Figure 8. From this figure, we can see that L^2F converges fast and its performance becomes stable after 10 iterations. Thus, we terminate the algorithm after a maximum of 15 iterations.

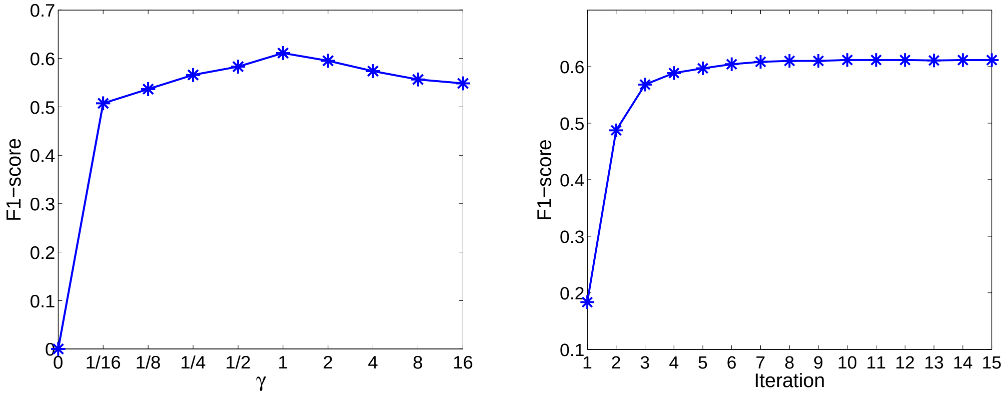


Fig. 8: F_1-score varies with γ (log_2 scale) (left) and iteration (right).

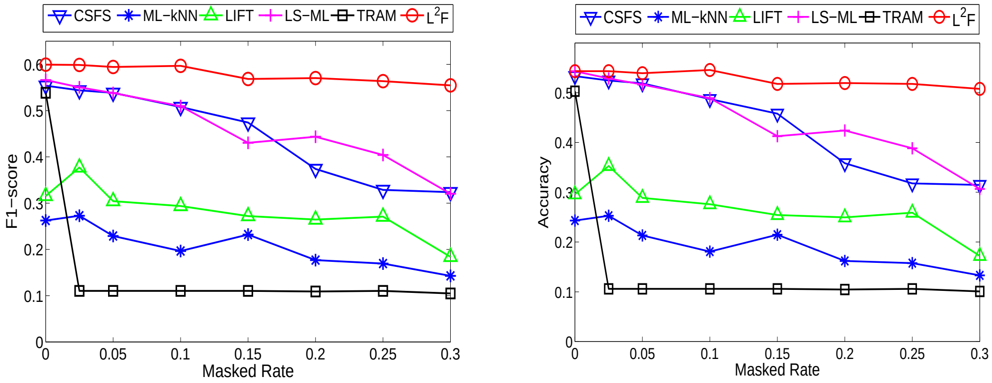


Fig. 9: Performance (left: F_1-score, right: accuracy) varies with masked rate of labels.

6.6. Learning from Missing Views and Labels

In this subsection, we aim to verify the robustness of the L^2F method to the missing views and missing labels on the Medical dataset.

Missing Labels: To generate the dataset with missing labels, we first randomly select a percentage of examples from the training data. The ratio between the number of selected examples and that of total training examples is denoted by the masked rate r, which is adjusted in the range $r = \{0, 0.025, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3\}$. Then, for each example, we randomly select one of its positive labels and mask it as missing label.

Figure 9 shows the performances of different algorithms vary with masked rate of labels. A common trend is that all the algorithms usually perform worse when the masked rate increases in most cases. It is reasonable that as the masked rate increases, more noise will be introduced into the training data rendering more difficulty to the learning task. As shown in the figure, L^2F is more robust to the noisy data in comparison with the other algorithms. Its performance remains stable over a wide range of masked rates. In contrast, the performance curve of TRAM [Kong et al. 2013] drops sharply when the labels of a very small percentage of instances are masked as missing, such as $r = 0.025$. This result suggests that the transductive learning method TRAM is

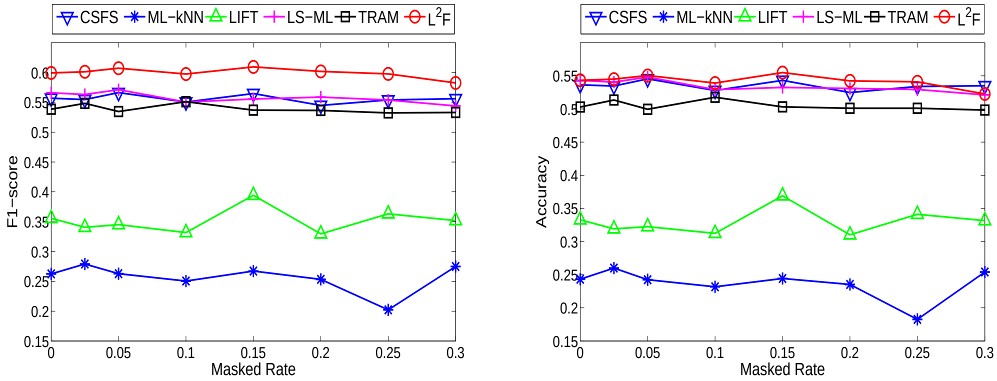


Fig. 10: Performance (left: F_1-score, right: accuracy) varies with masked rate of views.

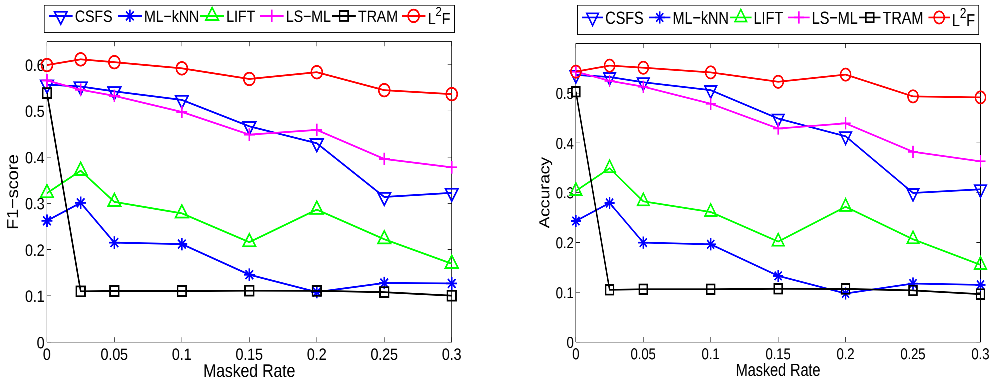


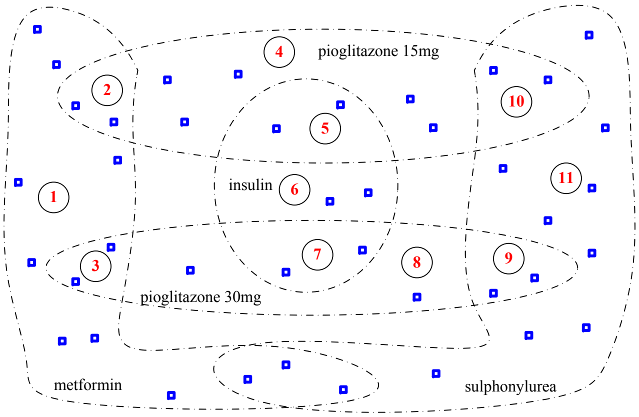
Fig. 11: Performance (left: F_1-score, right: accuracy) varies with masked rate of both views and labels.

sensitive to the missing labels because the noisy label information is likely to mislead the learning system in the transductive setting.

Missing Views: To generate the dataset with missing views, we first randomly select a percentage of examples from the training data. Likewise, the masked rate is adjusted in the range $r = \{0, 0.025, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3\}$. Then, for each example, we randomly select one of its views and mask it as missing view.

Figure 10 shows the performances of different algorithms vary with masked rate of views. In this figure, we can see that the performance of all the algorithms is not very sensitive to the missing values in most cases. Two reasons could account for this phenomenon. For the comparison algorithms except L^2F, though some views are missing, the features in other views are concatenated to build the classifiers. For L^2F, we borrow the strength from other views to reconstruct the instance-instance affinity matrix, which is then used to build the learning system.

Co-existing of Missing Views and Labels: By combining the above two operations as we done for the missing labels and views, we can generate the dataset with both missing labels and missing views. The results are shown in Figure 11. Since the



No	Instance	A1C	FBG
1	1027	8.72	11.42
2	227	8.68	11.69
3	177	9.36	12.78
4	727	9.29	12.26
5	135	9.80	12.67
6	623	9.42	12.20
7	240	9.26	11.88
8	1982	8.86	11.34
9	53	7.97	8.78
10	29	7.86	8.36
11	446	8.05	8.94

Fig. 12: A fraction of hypergraph constructed from the Diabetes data, where the hyperedge in dotted polyline represents the treatment which corresponds to a set of patients received this treatment. The average A1C and FBG values of the patients received the corresponding treatments are shown in the table.

algorithms are not very sensitive to the missing views in our setting, we can see that the results shown in this figure are similar to those in Figure 9.

6.7. Visualization of the Hypergraph on Diabetes Dataset

Taking the Diabetes data as an example, we explore the correlations among multiple labels, as well as the feature distributions in different views.

Figure 12 shows a fraction of the hypergraph constructed from the Diabetes data. In this figure, the hyperedge in dotted polyline represents the label (treatment) which corresponds to a set of nodes (patients) received this treatment. There are five types of treatments shown in the figure, i.e., insulin, metformin, sulphonylurea, pioglitazone 15mg and pioglitazone 30mg. Insulin is a hormone which helps to regulate blood sugar. It is known that insulin is prescribed for patients with type 1 diabetes and for patients with type 2 diabetes who have not responded so well on oral medication. Metformin is commonly used as a first line treatment for type 2 diabetes, which is the only available diabetes medication in the biguanides class of drugs. Sulphonylureas are the class of antidiabetic drug for type 2 diabetes, which work by increasing the amount of insulin the pancreas produces and increasing the working effectiveness of insulin. Pioglitazone is an antidiabetic drug used to increase the insulin sensitivity. From the figure, we can see that there is no correlations between insulin and either metformin or sulphonylurea. This is reasonable because insulin is usually used for patients with type 1 diabetes, while both metformin and sulphonylurea are commonly used for patients with type 2 diabetes. The results suggest that multiple types of labels are locally correlated in the Diabetes dataset, and learning system can benefit from exploring the correlations among different labels.

The right panel of the figure further shows the data distributions of features in different views. In this table, the second column (instance) refers to the total number of patients received the corresponding treatments; the last two columns (A1C and FBG) refer to the average A1C and FBG values of these patients, which measure the long term and short term drug impact, respectively.

7. CONCLUSION

In this paper, we propose a graph-based approach L^2F for learning from both label and feature heterogeneity. L^2F is robust to missing value problems, such as missing labels and missing views. An iterative algorithm is presented to solve the convex problem, which is guaranteed to converge to the global optimum. We analyze its performance in terms of its generalization error rate, which shows the benefit of jointly modeling the dual heterogeneity. Furthermore, a generalized framework based on L^2F allows us to model multiple types of heterogeneity by incorporating our prior knowledge of the learning task into the hypergraph structure. The comparison experiments with state-of-the-art methods demonstrate the effectiveness of the proposed algorithm.

One of our on-going work is to extend the proposed framework to the semi-supervised setting. It is expected that the performance could be improved by picking out the informative examples and rebuilding the learning system, which is particularly challenging due to both label and feature heterogeneity. Also, we will explore different pathways to model the multiple types of correlation relationships among different labels, and investigate how and when the learning performance can be improved by leveraging multiple types of correlations in the heterogeneous scenarios.

REFERENCES

Steven P. Abney. 2002. Bootstrapping. In *ACL*. 360–367.

Rie Kubota Ando and Tong Zhang. 2005. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research (JMLR)* 6 (2005), 1817–1853.

Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. 2006. Multi-Task Feature Learning. In *NIPS*. 41–48.

Avrim Blum and Tom Mitchell. 1998. Combining Labeled and Unlabeled Data with Co-Training. In *COLT*. 92–100.

Xiaojun Chang, Feiping Nie, Yi Yang, and Heng Huang. 2014. A Convex Formulation for Semi-Supervised Multi-Label Feature Selection. In *AAAI*. 1171–1177.

Jianhui Chen, Jiayu Zhou, and Jieping Ye. 2011. Integrating low-rank and group-sparse structures for robust multi-task learning. In *KDD*. 42–50.

Ning Chen, Jun Zhu, and Eric P. Xing. 2010. Predictive subspace learning for multi-view data: a large margin approach. In *NIPS*.

Michael Collins and Yoram Singer. 1999. Unsupervised models for named entity classification. In *EMNLP*. 100–110.

Sanjoy Dasgupta, Michael L. Littman, and David A. McAllester. 2001. PAC Generalization Bounds for Co-training. In *NIPS*. 375–382.

Sotiris Diplaris, Grigorios Tsoumakas, Pericles A. Mitkas, and Ioannis P. Vlahavas. 2005. Protein Classification with Multiple Algorithms. In *Panhellenic Conference on Informatics*. 448–456.

Shmuel Winograd Don Coppersmith. 1990. Matrix Multiplication via Arithmetic Progressions. *J. Symb. Comput. (JSC)* 9, 3 (1990), 251–280.

André Elisseeff and Jason Weston. 2001. A kernel method for multi-labelled classification. In *NIPS*. 681–687.

Zheng Fang and Zhongfei (Mark) Zhang. 2012. Simultaneously Combining Multi-view Multi-label Learning with Maximum Margin Classification. In *ICDM*. 864–869.

Jason D. R. Farquhar, David R. Hardoon, Hongying Meng, John Shawe-Taylor, and Sándor Szedmák. 2005. Two view learning: SVM-2K, Theory and Practice. In *NIPS*.

Wei Gao and Zhi-Hua Zhou. 2013. On the consistency of multi-label learning. *Artif. Intell. (AI)* (2013), 22–44.

Shantanu Godbole and Sunita Sarawagi. 2004. Discriminative Methods for Multi-labeled Classification. In *PAKDD*. 22–30.

Pinghua Gong, Jieping Ye, and Changshui Zhang. 2012. Robust multi-task feature learning. In *KDD*. 895–903.

Yuhong Guo and Dale Schuurmans. 2012. Semi-supervised Multi-label Classification - A Simultaneous Large-Margin, Subspace Learning Approach. In *ECML PKDD*. 355–370. DOI:http://dx.doi.org/10.1007/978-3-642-33486-3_23

Jingrui He and Rick Lawrence. 2011. A Graph-based Framework for Multi-Task Multi-View Learning. In *ICML*. 25–32.

Sheng-Jun Huang, Yang Yu, and Zhi-Hua Zhou. 2012. Multi-label hypothesis reuse. In *KDD*. 525–533.

Sheng-Jun Huang and Zhi-Hua Zhou. 2012. Multi-Label Learning by Exploiting Label Correlations Locally. In *AAAI*. 1–7.

Shuiwang Ji, Lei Tang, Shipeng Yu, and Jieping Ye. 2008. Extracting shared subspace for multi-label classification. In *KDD*. 381–389.

Xiangnan Kong, Michael K. Ng, and Zhi-Hua Zhou. 2013. Transductive Multilabel Learning via Label Set Propagation. *IEEE Trans. Knowl. Data Eng. (TKDE)* (2013), 704–719.

Chun-Liang Li and Hsuan-Tien Lin. 2014. Condensed Filter Tree for Cost-Sensitive Multi-Label Classification. In *ICML*. 423–431.

Yongjin Li and Jagdish C. Patra. 2013. Genome-wide inferring gene-phenotype relationship by walking on the heterogeneous network. *Bioinformatics* 26, 9 (2013), 1219–1224.

Zijia Lin, Guiguang Ding, Mingqing Hu, and Jianmin Wang. 2014. Multi-label Classification via Feature-aware Implicit Label Space Encoding. In *ICML*. 325–333.

Yi Liu, Rong Jin, and Liu Yang. 2006. Semi-supervised Multi-label Learning by Constrained Non-negative Matrix Factorization. In *AAAI*. 421–426.

Z. Q. Luo and P. Tseng. 1992. On the convergence of the coordinate descent method for convex differentiable minimization. *Journal of Optimization Theory and Applications* 72, 1 (1992), 7–35.

Laura Miozzi, Rosario Michael Piro, Fabio Rosa, Ugo Ala, Lorenzo Silengo, Ferdinando Di Cunto, and Paolo Provero. 2008. Functional annotation and identification of candidate disease genes by computational analysis of normal tissue gene expression data. *PLoS One* 3 (2008).

Christopher C. Paige and Michael A. Saunders. 1982. LSQR: An Algorithm for Sparse Linear Equations and Sparse Least Squares. *ACM Trans. Math. Softw.* 8, 1 (1982), 43–71. DOI:http://dx.doi.org/10.1145/355984.355989

John P. Pestian, Christopher Brew, Pawel Matykiewicz, D. J. Hovermale, Neil Johnson, K. Bretonnel Cohen, and Wodzislaw Duch. 2007. A shared task involving multi-label classification of clinical free text. In *Proceedings of the Workshop on BioNLP*. 97–104.

R.M. Piro and F. Di Cunto. 2012. Computational approaches to disease-gene prediction: rationale, classification and successes. *FEBS J.* 279 (2012), 678–696.

John Shawe-Taylor and Nello Cristianini. 2004. *Kernel Methods for Pattern Analysis*. Cambridge University Press.

Vikas Sindhwani and David S. Rosenberg. 2008. An rkhs for multi-view learning and manifold co-regularization. In *ICML*. 976–983.

U. Martin Singh-Blom, Nagarajan Natarajan, Ambuj Tewari, John O. Woods, Inderjit S. Dhillon, and Edward M. Marcotte. 2013. Prediction and validation of gene-disease associations using methods inspired by social network analyses. *PLoS One* 8 (2013).

Alex J. Smola and Risi Kondor. 2003. Kernels and Regularization on Graphs. In *COLT*. 144–158.

Le Song, Animashree Anandkumar, Bo Dai, and Bo Xie. 2014. Nonparametric Estimation of Multi-View Latent Variable Models. In *ICML*. 640–648.

Karthik Sridharan and Sham M. Kakade. 2008. An Information Theoretic Framework for Multi-view Learning. In *COLT*. 403–414.

Liang Sun, Shuiwang Ji, and Jieping Ye. 2008. Hypergraph spectral learning for multi-label classification. In *KDD*. 668–676.

Liang Sun, Shuiwang Ji, and Jieping Ye. 2009. A least squares formulation for a class of generalized eigenvalue problems in machine learning. In *ICML*. 977–984. DOI:http://dx.doi.org/10.1145/1553374.1553499

Paul Tseng. 2001. Convergence of a Block Coordinate Descent Method for Nondifferentiable Minimization. *Journal of Optimization Theory and Applications* 109, 3 (2001), 475–494.

Grigorios Tsoumakas and Ioannis Katakis. 2007. Multi-Label Classification: An Overview. *International Journal of Data Warehousing and Mining* 3, 3 (2007), 1–13.

Jingdong Wang, Yinghai Zhao, Xiuqing Wu, and Xian-Sheng Hua. 2011. A transductive multi-label learning approach for video concept detection. *Pattern Recognition* 44, 10-11 (2011), 2274–2286. DOI:http://dx.doi.org/10.1016/j.patcog.2010.07.015

Le Wu and Min-Ling Zhang. 2013. Multi-Label Classification with Unlabeled Data: An Inductive Approach. In *ACML*. 197–212. http://jmlr.org/proceedings/papers/v29/Wu13.html

Hongxia Yang and Jingrui He. 2014. Learning with Dual Heterogeneity: A Nonparametric Bayes Model. In *KDD*. 582–590.

Pei Yang and Wei Gao. 2013. Multi-View Discriminant Transfer Learning. In *IJCAI*. 1848–1854.

Pei Yang, Jingrui He, and Jia-Yu Pan. 2015. Learning Complex Rare Categories with Dual Heterogeneity. In *SDM*. 523–531.

Pei Yang, Jingrui He, Hongxia Yang, and Haoda Fu. 2014. Learning from Label and Feature Heterogeneity. In *ICDM*. 1079–1084.

Hsiang-Fu Yu, Prateek Jain, Purushottam Kar, and Inderjit S. Dhillon. 2014. Large-scale Multi-label Learning with Missing Labels. In *ICML*. 593–601.

Jintao Zhang and Jun Huan. 2012. Inductive multi-task learning with multiple view data. In *KDD*. 543–551.

Min-Ling Zhang and Zhi-Hua Zhou. 2014. A Review on Multi-Label Learning Algorithms. *IEEE Trans. Knowl. Data Eng.* 26, 8 (2014), 1819–1837.

Min-Ling Zhang. 2011. LIFT: Multi-Label Learning with Label-Specific Features. In *IJCAI*. 1609–1614.

Min-Ling Zhang and Kun Zhang. 2010. Multi-label learning by exploiting label dependency. In *KDD*. 999–1008.

Min-Ling Zhang and Zhi-Hua Zhou. 2007. ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognition* (2007), 2038–2048.

Xiaoyu Zhang, Jian Cheng, Changsheng Xu, Hanqing Lu, and Songde Ma. 2009. Multi-view multi-label active learning for image classification. In *ICME*. 258–261.

Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf. 2004. Learning with Local and Global Consistency. In *NIPS*.

Dengyong Zhou, Jiayuan Huang, and Bernhard Scholkopf. 2007. Learning with Hypergraphs: Clustering, Classification, and Embedding. In *NIPS*. 1601–1608.

Jiayu Zhou, Jianhui Chen, and Jieping Ye. 2011. Clustered Multi-Task Learning Via Alternating Structure Optimization. In *NIPS*. 702–710.