Feature-Induced Manifold Disambiguation for Multi-View Partial Multi-label Learning

Jing-Han Wu[1,2]
[1]School of Computer Science and Engineering, Southeast University, Nanjing 210096, China
[2]Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, China
wujh915@seu.edu.cn

Xuan Wu, Qing-Guo Chen, Yao Hu
YouKu Cognitive and Intelligent Lab, Alibaba Group, Hangzhou, China
wx193834@alibaba-inc.com
qingguo.cqg@alibaba-inc.com
yaoohu@alibaba-inc.com

Min-Ling Zhang[1,2*]
[1]School of Computer Science and Engineering, Southeast University, Nanjing 210096, China
[2]Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, China
zhangml@seu.edu.cn

ABSTRACT

In conventional multi-label learning framework, each example is assumed to be represented by a single feature vector and associated with multiple valid labels simultaneously. Nonetheless, real-world objects usually exhibit complicated properties which can have multi-view feature representation as well as false positive labeling. Accordingly, the problem of *multi-view partial multi-label learning* (MVPML) is studied in this paper, where each example is assumed to be presented by multiple feature vectors while associated with multiple candidate labels which are only partially valid. To learn from MVPML examples, a novel approach named Fiman is proposed which makes use of multi-view feature representation to tackle the noisy labeling information. Firstly, an aggregate manifold structure over training examples is generated by adaptively fusing affinity information conveyed by feature vectors of different views. Then, candidate labels of each training example are disambiguated by preserving the feature-induced manifold structure in label space. Finally, the resulting predictive models are learned by fitting modeling outputs with the disambiguated labels. Extensive experiments on a number of real-world data sets show that Fiman achieves highly competitive performance against state-of-the-art approaches in solving the MVPML problem.

CCS CONCEPTS

• **Computing methodologies** → **Supervised learning**; **Machine learning algorithms**.

KEYWORDS

Multi-label learning, partial label learning, multi-view, disambiguation

*Corresponding author

KDD'20, August 22–27, 2020, San Diego, CA

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

Figure 1: An examplar multi-view partial multi-label object. The movie trailer can be represented from different views such as *video frame*, *subtitle* and *audio*. Furthermore, among the seven candidate labels annotated by crowdsourced labelers, only four of them are valid ones including *Elsa*, *Anna*, *Walt Disney* and *Crowd*.

ACM Reference Format:
Jing-Han Wu[1,2], Xuan Wu, Qing-Guo Chen, Yao Hu, and Min-Ling Zhang[1,2]. 2020. Feature-Induced Manifold Disambiguation for Multi-View Partial Multi-label Learning. In *KDD'20: ACM SIGKDD Conference on Knowledge Discovery and Data Mining, August 22–27, 2020, San Diego, CA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

In recent years, multi-label learning has attracted significant research attentions in modeling objects with rich semantics [10, 32, 35]. Generally, multi-label learning takes the basic assumption that each example is represented by a single feature vector in the input space and associated with multiple valid labels simultaneously in the output space. Nonetheless, in real-world learning tasks the objects to be modeled usually exhibit complicated properties. As shown in Figure 1, a movie trailer can have multi-view representation such as *video frame*, *subtitle* and *audio* and a number of candidate annotations given by crowdsourced labelers, among which only *Elsa*, *Anna*, *Walt Disney* and *Crowd* are valid ones.

To account for the need of learning under these circumstances, the problem of *multi-view partial multi-label learning* (MVPML)

is investigated in this paper. Formally speaking, let $\mathcal{X} = \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \cdots \times \mathbb{R}^{d_U}$ denote the input space consisting of U views with d_u ($1 \le u \le U$) being the dimensionality of the u-th view. Furthermore, let $\mathcal{Y} = \{\lambda_1, \lambda_2, \ldots, \lambda_q\}$ denote the label space consisting of q possible class labels. Let $\mathcal{D} = \{(\boldsymbol{x}_i, Y_i) \mid 1 \le i \le m\}$ be the MVPML training set, where $\boldsymbol{x}_i = [\boldsymbol{x}_i^1, \boldsymbol{x}_i^2, \cdots, \boldsymbol{x}_i^U] \in \mathcal{X}$ is the $\left(\sum_{u=1}^{U} d_u\right)$-dimensional multi-view instance and $Y_i \subseteq \mathcal{Y}$ is the set of candidate labels associated with $\boldsymbol{x}_i$. Here, MVPML takes the key assumption that the ground-truth labels $\widetilde{Y}_i \subseteq \mathcal{Y}$ for $\boldsymbol{x}_i$ is concealed within in its candidate label set (i.e. $\widetilde{Y}_i \subseteq Y_i$) and thus not directly accessible to the learning algorithm. Accordingly, the task of MVPML is to learn a multi-label classification model $h : \mathcal{X} \to 2^{\mathcal{Y}}$ from $\mathcal{D}$ which is capable of predicting the proper label set for unseen multi-view instance.

To deal with the MVPML problem, one intuitive solution is to resort to its degenerated versions. On one hand, by simply treating all candidate labels as ground-truth ones, the MVPML problem can be degenerated into the *multi-view multi-label learning* (MVML) counterpart with noise-free labeling information [14, 15, 18, 23, 26, 29, 30, 36, 37]. On the other hand, by directly concatenating all feature vectors w.r.t. different views into a single feature vector, the MVPML problem can be degenerated into the *partial multi-label learning* (PML) counterpart with single-view representation [8, 11, 16, 22, 25, 28]. Although it is feasible to invoke off-the-shelf MVML or PML techniques to solve the MVPML problem in degenerated manner, it is highly desirable to develop tailored learning techniques such that the inherent characteristics of MVPML, i.e. *multi-view* and *noisy labeling*, can be taken into full consideration to yield strong generalization performance.

In this paper, a novel approach named FIMAN, i.e. *Feature-Induced MANifold disambiguation for multi-view partial multi-label learning* is proposed to learning from MVPML examples. Firstly, the affinity information conveyed by feature vectors of different views are adaptively fused into an aggregate manifold structure over training examples. After that, FIMAN aims to disambiguate the candidate label set of each training example by preserving the feature-induced manifold structure in label space. Finally, the resulting multi-label classification model is induced by fitting its modeling outputs with the disambiguated labels. The promising performance of FIMAN against state-of-the-art approaches is clearly validated based on extensive comparative studies.

The rest of this paper is organized as follows. Section 2 briefly reviews existing works related to MVPML. Section 3 introduces technical details of the proposed FIMAN approach. Section 4 reports experimental results over the real-world data sets. Finally, Section 5 concludes.

2 RELATED WORKS

As discussed in Section 1, MVPML is closely related to the two learning frameworks MVML and PML which generalize conventional multi-label learning framework [10, 32, 35] in either the input space or the output space.

MVML deals with the problem where each example is represented by multi-view feature vectors and associated with multiple valid class labels simultaneously. Due to the multi-view nature of MVML examples, most works focus on identifying informative shared subspace across different views to facilitate model induction. For multi-label image classification, a low-dimensional shared subspace can be learned by enforcing low-rank constraint [14] or consistency regularization [15, 37]. By adapting the popular co-training framework [1, 34], reliable labeling information communication across different views can be achieved via diversity maximization [29] or confidence-rated filtering [26]. Based on the measurement of multi-view correlations, the shared subspace can be learned based on Hilbert-Schmidt Independence Criterion (HSIC) [30] or matrix factorization [36]. In addition to the shared subspace, view-specific information can also be extracted and jointly utilized for model induction [18, 23].

PML deals with the problem where each example is represented by a single feature vector and associated with multiple candidate labels which are only partially valid. Due to the noisy labeling nature of PML examples, most works focus on estimating the labeling confidence of each candidate label being the ground-truth one to facilitate model induction. Correspondingly, labeling confidence estimation and predictive model induction can be opitimized in an iterative manner via confidence-weighted ranking loss minimization [25], confidence matrix low-rank approximation [16] or correlation matrix factorization [28]. Alternatively, the learning task can be fulfilled by decoupling the labeling confidence estimation and predictive model induction in a two-stage manner via credible label elicitation [8] or soft label discriminative learning [11, 22].

Both MVML and PML can be viewed as degenerated versions of MVPML, which makes the task of learning from MVPML more challenging to be solved. One prior attempt towards MVPML works by equally fusing similarity graph constructed over each view and then disambiguating candidate label set for model induction based on label propagation [3]. Nonetheless, the modeling contribution of each view generally varies to each other and the effectiveness of label propagation heavily relies on the quality of initial labeling confidences which are less robust to be instantiated under noisy labeling scenario.

In the next section, a novel MVPML approach with strong generalization performance is presented, where affinity information conveyed by each view is fused adaptively and candidate label disambiguation is conducted based on manifold structure preservation.

3 THE PROPOSED APPROACH

Following the notations given in Section 1, the task of MVPML is to learn a multi-label predictor $h : \mathcal{X} \mapsto 2^{\mathcal{Y}}$ from the training set $\mathcal{D} = \{(\boldsymbol{x}_i, Y_i) \mid 1 \le i \le m\}$, where $\boldsymbol{x}_i = [\boldsymbol{x}_i^1, \boldsymbol{x}_i^2, \cdots, \boldsymbol{x}_i^U] \in \mathcal{X}$ and $Y_i \subseteq \mathcal{Y}$ corresponds to the set of candidate labels associated with $\boldsymbol{x}_i$. Firstly, FIMAN aims to generate a manifold structure over all training examples by adaptively fusing the affinity information conveyed by different views. Accordingly, let $G^u = (V^u, E^u, \mathbf{W}^u)$ denote the affinity graph w.r.t. the u-th view ($(1 \le u \le U)$). Here, $V^u = \{\boldsymbol{x}_i^u \mid 1 \le i \le m\}$ corresponds to the vertex set consisting of feature vectors w.r.t. the u-th view. Furthermore, $E^u = \{(\boldsymbol{x}_i^u, \boldsymbol{x}_j^u) \mid \boldsymbol{x}_i^u \in k\mathrm{NN}(\boldsymbol{x}_j^u)\}$ corresponds to the edge set where there will be a directed edge from $\boldsymbol{x}_i^u$ to $\boldsymbol{x}_j^u$ iff $\boldsymbol{x}_i^u$ is among the k-nearest neighbors of $\boldsymbol{x}_j^u$ identified in V^u.

For the weight matrix $\mathbf{W}^u = [w_{ij}^u]_{m\times m}$, values of its elements are optimized by solving the following kNN-based minimum error reconstruction problem:

$$\begin{aligned}\min_{\mathbf{W}^u} \sum_{j=1}^{m} \left\| \boldsymbol{x}_j^u - \sum\nolimits_{(\boldsymbol{x}_i^u, \boldsymbol{x}_j^u)\in E^u} w_{ij}^u \cdot \boldsymbol{x}_i^u \right\|^2 \\ \text{s.t.}: \sum\nolimits_{(\boldsymbol{x}_i^u, \boldsymbol{x}_j^u)\in E^u} w_{ij}^u = 1 \ (1 \le j \le m) \\ w_{ij}^u \ge 0 \ \left(\forall\, (\boldsymbol{x}_i^u, \boldsymbol{x}_j^u) \in E^u\right) \\ w_{ij}^u = 0 \ \left(\forall\, (\boldsymbol{x}_i^u, \boldsymbol{x}_j^u) \notin E^u\right)\end{aligned} \tag{1}$$

The resulting problem (1) corresponds to a standard quadratic programming (QP) problem which can be efficiently solved by invoking any off-the-shelf QP toolbox.

To fully exploit the multi-view feature presentation of training examples, FIMAN tries to fuse the affinity information conveyed by different views in an adaptive manner. By taking the basic assumption that different views would have varying contributions for model induction, FIMAN chooses to learn a low-dimensional embedding which aligns the affinity matrix of different views with ensemble manifold regularization [24, 31, 33]. Specifically, let $\mathbf{P} \in \mathbb{R}^{p\times m}$ be the embedding matrix with $p = \min_{1\le u\le U}\{d_u\}$ and $\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_U]^\top$ be the non-negative weight vector characterizing the relative contribution of each view, the embedding matrix $\mathbf{P}$ and weight vector $\boldsymbol{\beta}$ can be determined adaptively by solving the following optimization problem:

$$\begin{aligned}\arg\min_{\mathbf{P},\boldsymbol{\beta}} \sum\nolimits_{u=1}^{U} \beta_u^r \cdot \mathrm{tr}\left(\mathbf{P}\mathbf{L}^u\mathbf{P}^\top\right) \\ \text{s.t.}: \mathbf{P}\mathbf{P}^\top = \mathbf{I} \\ \sum\nolimits_{u=1}^{U} \beta_u = 1, \ \beta_u \ge 0 \ (1 \le u \le U)\end{aligned} \tag{2}$$

Here, r is a non-negative parameter controlling the weight scale which is set to be the number of views (i.e. U) in this paper. In addition, $\mathbf{L}^u$ is the normalized graph Laplacian matrix w.r.t. the u-th view:

$$\mathbf{L}^u = \mathbf{I} - \left(\mathbf{D}^u\right)^{-\frac{1}{2}} \mathbf{W}^u \left(\mathbf{D}^u\right)^{-\frac{1}{2}} \tag{3}$$

where $\mathbf{D}^u$ is the diagonal matrix whose diagonal element corresponds to the column-wise sum of affinity matrix $\mathbf{W}^u$.

The resulting problem (2) corresponds to a nonconvex optimization problem with nonlinear constraints. Thereafter, the (local) optimal solution to this problem can be found based on alternating optimization. Accordingly, let $\mathbf{P}^{(t)}$ and $\boldsymbol{\beta}^{(t)}$ be the current solution at the t-th iteration. By fixing the weight vector β, the embedding matrix $\mathbf{P}$ can be updated by optimizing the following problem:

$$\begin{aligned}\arg\min_{\mathbf{P}} \ \mathrm{tr}\left(\mathbf{P}\mathbf{L}^{(t)}\mathbf{P}^\top\right) \\ \text{s.t.}: \mathbf{P}\mathbf{P}^\top = \mathbf{I}\end{aligned} \tag{4}$$

Here, $\mathbf{L}^{(t)} = \sum_{u=1}^{U} \left(\beta_u^{(t)}\right)^r \cdot \mathbf{L}^u$. It is not difficult to show that the optimal solution to problem (4) can be set as $\mathbf{P}^{(t+1)} = [\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_p]^\top$ where $\boldsymbol{e}_k$ $(1 \le k \le p)$ corresponds to the orthonormal eigenvector of $\mathbf{L}^{(t)}$ w.r.t. the k-th smallest eigenvalue.

By fixing the embedding matrix $\mathbf{P}$, it is also not difficult to show that the weight vector $\boldsymbol{\beta}$ can be updated by the following rule:

$$\beta_u^{(t+1)} = \frac{\left(1/\mathrm{tr}\left(\mathbf{P}^{(t+1)} \cdot \mathbf{L}^u \cdot \mathbf{P}^{(t+1)\,\top}\right)\right)^{\frac{1}{r-1}}}{\sum_{u=1}^{U}\left(1/\mathrm{tr}\left(\mathbf{P}^{(t+1)} \cdot \mathbf{L}^u \cdot \mathbf{P}^{(t+1)\,\top}\right)\right)^{\frac{1}{r-1}}} \tag{5}$$

Let $\boldsymbol{\beta}^*$ be the final weight vector obtained via the alternating optimization procedure, the aggregate manifold structure $\mathbf{W} = [w_{ij}]_{m\times m}$ over all training examples can be specified by fusing the affinity matrix of each view:

$$\mathbf{W} = \sum\nolimits_{u=1}^{U} \beta_u^* \cdot \mathbf{W}^u \tag{6}$$

Then, FIMAN aims to disambiguate the candidate label set of each training example by exploiting the aggregate manifold structure $\mathbf{W}$. Based on the smoothness assumption, affinity relationships among training examples identified in the feature space should be preserved in the label space as well. Let $\mathbf{F} = [\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_m]$ be the $q\times m$ labeling confidence matrix, where $\boldsymbol{f}_j = [f_{j1}, f_{j2}, \ldots, f_{jq}]^\top$ with f_{jl} representing the labeling confidence of λ_l being the ground-truth label of $\boldsymbol{x}_j$. FIMAN imposes manifold structure preserving in the label space by solving the following optimization problem:

$$\begin{aligned}\min_{\mathbf{F}} \sum\nolimits_{j=1}^{m} \left\| \boldsymbol{f}_j - \sum\nolimits_{i=1}^{m} w_{ij} \cdot \boldsymbol{f}_i \right\|^2 \\ \text{s.t.}: \sum\nolimits_{l=1}^{q} f_{jl} = 1 \ (1 \le j \le m) \\ f_{jl} \ge 0 \ (1 \le j \le m, \lambda_l \in Y_j) \\ f_{jl} = 0 \ (1 \le j \le m, \lambda_l \notin Y_j)\end{aligned} \tag{7}$$

Similar to (1), the resulting optimization problem (7) is also a standard QP problem which can also be solved by invoking off-the-shelf QP solver.

For each MVPML training example $(\boldsymbol{x}_i, Y_i)$, the set of ground-truth labels $\widetilde{Y}_i \subseteq Y_i$ are disambiguated from Y_i by thresholding the learned labeling confidence:

$$\widetilde{Y}_i = \{\lambda_l \mid f_{jl} \ge t_d, \ 1 \le l \le q\} \tag{8}$$

Here, $t_d \in [0, 1]$ corresponds to the disambiguation threshold.

Thereafter, the original MVPML training set $\mathcal{D}$ can be transformed into its disambiguated counterpart $\widetilde{D} = \{(\boldsymbol{x}_i, \widetilde{Y}_i) \mid 1 \le i \le m\}$ for subsequent model induction. FIMAN assumes linear models to make prediction on unseen instance, which is capable of achieving reasonable tradeoff between generalization performance and model simplicity.

Let $\boldsymbol{\Omega} = \left[\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \ldots, \boldsymbol{\omega}_q\right] \in \mathbb{R}^{D\times q}$ $\left(D = \sum_{u=1}^{U} d_u\right)$ be the linear predictive model, where $\boldsymbol{\omega}_j = \left[\boldsymbol{\omega}_j^1; \boldsymbol{\omega}_j^2; \ldots; \boldsymbol{\omega}_j^U\right]$ with $\boldsymbol{\omega}_j^u \in \mathbb{R}^{d_u}$ $(1 \le u \le U)$ and $\boldsymbol{\Omega}^u = \left[\boldsymbol{\omega}_1^u, \boldsymbol{\omega}_2^u, \ldots, \boldsymbol{\omega}_q^u\right] \in \mathbb{R}^{d_u\times q}$. Accordingly, let $\mathbf{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m] \in \mathbb{R}^{D\times m}$, $\mathbf{X}^u = \left[\boldsymbol{x}_1^u, \boldsymbol{x}_2^u, \ldots, \boldsymbol{x}_m^u\right] \in \mathbb{R}^{d_u\times m}$ and $\widetilde{\mathbf{Y}} = [\widetilde{\boldsymbol{y}}_1, \widetilde{\boldsymbol{y}}_2, \ldots, \widetilde{\boldsymbol{y}}_m] \in \{0, 1\}^{q\times m}$ where $\widetilde{\boldsymbol{y}}_i = [\widetilde{y}_{i1}, \widetilde{y}_{i2}, \ldots, \widetilde{y}_{iq}]^\top$ with $\widetilde{y}_{il} = 1$ if $\lambda_l \in \widetilde{Y}_i$ and $\widetilde{y}_{il} = 0$ otherwise. Then, the predictive model is induced by solving the following ridge regression problem:

$$\min_{\boldsymbol{\Omega}} \left\| \widetilde{\mathbf{Y}} - \sum\nolimits_{u=1}^{U} \left(\boldsymbol{\Omega}^u\right)^\top \mathbf{X}^u \right\|_F^2 + \eta \cdot \sum\nolimits_{u=1}^{U} \left\|\boldsymbol{\Omega}^u\right\|_F^2 \tag{9}$$

Table 1: The pseudo-code of FIMAN.

Inputs:

$\mathcal{D}$: the MVPML training set $\{(\boldsymbol{x}_i, Y_i) \mid 1 \le i \le m\}$ ($\mathcal{X} = \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \cdots \times \mathbb{R}^{d_U}, \mathcal{Y} = \{\lambda_1, \lambda_2, \ldots, \lambda_q\}, \boldsymbol{x}_i \in \mathcal{X}, Y_i \subseteq \mathcal{Y}$)

k: number of nearest neighbors for affinity graph construction

η: the regularization coefficient for ridge regression

t_d, t_p: the disambiguation threshold t_d and the prediction threshold t_p

$\boldsymbol{x}^*$: the unseen multi-view instance

Outputs:

Y^*: the predicted label set for $\boldsymbol{x}^*$

Process:

for $u = 1$ to U **do**

  Generate the affinity graph $G^u = (V^u, E^u, \mathbf{W}^u)$ w.r.t. the u-th view with $V^u = \{\boldsymbol{x}_i^u \mid 1 \le i \le m\}$, $E^u = \{(\boldsymbol{x}_i^u, \boldsymbol{x}_j^u) \mid \boldsymbol{x}_i^u \in k\text{NN}(\boldsymbol{x}_j^u)\}$ and $\mathbf{W}^u$ obtained by solving optimization problem (1) ;

  Set the normalized graph Lapalacian matrix $\mathbf{L}^u$ according to Eq.(3);

end for

Set $\boldsymbol{\beta}^0 = (1/U) \cdot \mathbf{1}_U$, $p = \min_{1 \le u \le U}\{d_u\}$ and $r = U$;

$t = 0$;

repeat

  Set $\mathbf{L}^{(t)} = \sum_{u=1}^{U} \left(\beta_u^{(t)}\right)^r \cdot \mathbf{L}^u$;

  Update $\mathbf{P}^{(t+1)} = [\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_p]^\top$ where $\boldsymbol{e}_k$ $(1 \le k \le p)$ corresponds to the orthonormal eigenvector of $\mathbf{L}^{(t)}$ w.r.t. the k-th smallest eigenvalue;

  Update $\boldsymbol{\beta}^{(t+1)} = \left[\beta_1^{(t+1)}, \beta_2^{(t+1)}, \ldots, \beta_U^{(t+1)}\right]^\top$ according to Eq.(5);

  $t = t + 1$;

until convergence

Set $\boldsymbol{\beta}^* = \boldsymbol{\beta}^{(t)}$;

Obtain the aggregate manifold structure $\mathbf{W}$ according to Eq.(6);

Obtain the labeling confidence matrix $\mathbf{F} = [\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_m] \in \mathbb{R}^{q \times m}$ by solving optimization problem (7);

Disambiguate candidate label set Y_i into $\widetilde{Y}_i$ $(1 \le i \le m)$ according to Eq.(8);

Induce the predictive model $\Omega = \left[\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \ldots, \boldsymbol{\omega}_q\right] \in \mathbb{R}^{D \times q}$ $\left(D = \sum_{u=1}^{U} d_u\right)$ according to Eq.(10);

Return Y^* according to Eq.(11).

Table 2: Characteristics of the multi-view partial multi-label data sets.

Data Set	$\lvert\mathcal{S}\rvert$	$V(\mathcal{S})$	$VDim(\mathcal{S})$	$CL(\mathcal{S})$	$LCard(\mathcal{S})$	Domain	Controlling Parameters
Emotions	593	2	8 / 64	6	1.869	*music*	$p \in \{0.3, 0.5, 0.7\}$, $r \in \{1, 2, 3\}$
Yeast	2,417	2	24 / 79	14	4.237	*biology*	
Pascal	9,963	5	42 / 100 / 196 / 370 / 310	20	1.465	*image*	
EspGame5k	5,192	4	48 / 91 / 519 / 368	268	4.679	*image*	
Mirflickr5k	5,000	5	48 / 93 / 112 / 359 / 318	38	4.711	*image*	
Youku5k	4,988	4	33 / 55 / 503 / 31	114	2.138	*video*	
Youku10k	9,976	4	33 / 55 / 503 / 31	114	2.129	*video*	
Youku15k	14,964	4	33 / 55 / 503 / 31	114	2.131	*video*	

Here, $\eta > 0$ corresponds to the regularization coefficient. Note that for the optimization problem (9), one can consider incorporating the weight vector $\boldsymbol{\beta}^*$ by replacing the term $\sum_{u=1}^{U} (\Omega^u)^\top \mathbf{X}^u$ with

$\sum_{u=1}^{U} \beta_u^* \cdot (\Omega^u)^\top \mathbf{X}^u$. To reduce the risk of overfitting in model induction, FIMAN refrains from overly utilizing $\boldsymbol{\beta}^*$ in this step and chooses to use the vanilla instantiation. Thereafter, the solution to the ridge regression problem (9) corresponds to:

$$\Omega = \left(\mathbf{X}\mathbf{X}^\top + \eta \cdot \mathbf{I}\right)^{-1} \mathbf{X}\widetilde{\mathbf{Y}}^\top \tag{10}$$

Given the unseen multi-view instance $\boldsymbol{x}^* \in \mathcal{X}$, its set of proper labels Y^* are predicted by thresholding the modeling outputs:

$$Y^* = \{\lambda_j \mid \boldsymbol{\omega}_j^\top \cdot \boldsymbol{x}^* \geq t_p,\ 1 \leq j \leq q\} \tag{11}$$

Here, $t_p \in \mathbb{R}$ corresponds to the prediction threshold.

Table 1 summarizes the complete procedure of FIMAN. The affinity graph w.r.t. to each view is generated with kNN-based minimum error reconstruction (Steps 1-4), which are fused adaptively to yield the aggregate manifold structure over all training examples (Steps 5-14). After that, the candidate label set of each MVPML example is disambiguated via manifold structure preservation in the label space (Steps 15-16). Finally, the set of proper labels for unseen instance is returned by querying the predictive model induced by fitting disambiguated labeling information (Steps 17-18).

4 EXPERIMENTS

4.1 Experimental Setup

4.1.1 Data Sets. In this paper, a total of eight real-world data sets have been collected from different domains for thorough experimental studies. For each data set $\mathcal{S}$, the number of examples ($|\mathcal{S}|$), number of views ($V(\mathcal{S})$), dimensionality of each view ($VDim(\mathcal{S})$), number of class labels ($CL(\mathcal{S})$), and average number of ground-truth labels per example (i.e. label cardinality $LCard(\mathcal{S})$) are summarized in Table 2.

In addition, for the `Emotions` [19] data set, the two views of each example correspond to the *rhythm features* and *timbre features* of a piece of music; For the `Yeast` [6] data set, the two views of each example correspond to the *genetic expression* and *phylogenetic profile* of a gene; For the `EspGame5k` [20] data set, the four views of each example correspond to the *DenseHue*, *Gist*, *DenseSift* and *HSV* features of an image; For the *Pascal* [7] and *Mirflickr5k* [12] data sets, in addition to the same four views adopted by `EspGame5k`, the *tag features* are also utilized to represent each example; For the three *Youku* data sets [23], the four views of each example correspond to the *title embedding*, *audio features* [9], *cover picture features* [17] and *video frame features* of a video clip.

In this paper, we follow the widely-used protocol for introducing labeling noise [2–4, 13, 21, 27, 28] to generate MVPML examples with candidate label sets. As shown in Table 2, two controlling parameters p and r are employed to instantiate the generation process. Here, $p \in (0, 1)$ controls the fraction of examples which are partially labeled in the data set (i.e. with false positive labels) and $r \in \mathbb{N}$ controls the number of false positive labels appearing in the candidate label set. Given a multi-view multi-label example $(\boldsymbol{x}, \widetilde{Y})$ with ground-truth label set $\widetilde{Y}$, an MVPML example $(\boldsymbol{x}, Y)$ is generated by randomly adding r false positive labels $\Delta_r \subseteq \mathcal{Y} \setminus \widetilde{Y}$ into $\widetilde{Y}$, i.e. $Y = \widetilde{Y} \bigcup \Delta_r$. For each real-world data set in Table 2, nine different settings for controlling parameters are considered in this paper with $p \in \{0.3, 0.5, 0.7\}$ and $r \in \{1, 2, 3\}$.

4.1.2 Comparing Approaches. In this paper, the performance of FIMAN is compared against five state-of-the-art approaches with parameter configurations suggested in respective literatures:

- GRADIS [3]: An MVPML approach which works by fusing multi-view representation and disambiguating candidate label set based on graph-based label propagation [parameter configuration: $k = 10$, $\alpha = 0.95$, $\eta = 0.1$];
- F2L21F [37]: An MVML approach which works by fusing multi-view representation based on learning a shared subspace via block-row sparse regularization [parameter configuration: grid search for $\lambda_1, \lambda_2 \in \{10^{-2}, 10^{-1}, \ldots, 10^{2}\}$];
- LSAMML [30]: An MVML approach which works by fusing multi-view representation based on learning a shared subspace via Hilbert-Schmidt Independence Criterion [parameter configuration: grid search for $\beta, \gamma \in \{10^{-2}, 10^{-1}, \ldots, 10^{2}\}$];
- PML-LRS [16]: A PML approach which works by disambiguating the candidate label set based on confidence matrix low-rank approximation [parameter configuration: $\eta = 1$, grid search for $\gamma \in \{0.01, 0.1\}$ and $\beta \in \{0.1, 1\}$];
- FPML [28]: A PML approach which works by disambiguating the candidate label set based on correlation matrix factorization [parameter configuration: grid search for $\lambda_1 \in \{0.1, 1\}$, $\lambda_2 \in \{1, 10\}$ and $\lambda_3 \in \{1, 10, 100\}$].

To the best of our knowledge, the GRADIS approach is the only prior work which can directly learn from MVPML examples. For F2L21F and LSAMML, both approaches learn from MVPML examples by degenerating them into MVML ones via ignoring the existence of false positive labels. For PML-LRS and FPML, both approaches learn from MVPML examples by degenerating them into PML examples via concatenating the feature representation of different views. As shown in Table 1, parameters for the proposed FIMAN approach are set as $k = 10$, $\eta = 1$, $t_d = 0.4$ and $t_p = 0.6$.

For performance evaluation of the resulting multi-label classification model, six popular multi-label metrics are used including *hamming loss*, *one-error*, *coverage*, *ranking loss*, *average precision* and *micro-averaging AUC*. Due to page limit, detailed definitions on these metrics can be found in recent survey literatures [10, 32, 35]. Generally, for the first four metrics, the *smaller* the metric value the better the performance. For the other two metrics, the *larger* the metric value the better the performance. The values of all evaluation metrics are normalized into [0,1].

4.2 Experimental Results

Under each setting of controlling parameters p and r, ten-fold cross-validation is performed on the generated MVPML data set where the mean metric value as well as standard deviation are recorded. Tables 3 and 4 report the detailed experimental results of each comparing approach in terms of *ranking loss* and *micro-averaging AUC* respectively ($p \in \{0.3, 07\}, r \in \{1, 2, 3\}$), where the best performance on each data set is shown in boldface.[1]

[1]Due to page limit, complete experimental results in terms of each evaluation metric under all controlling parameters are not reported here.

Table 3: Predictive performance of each comparing approach (mean±std. deviation) on the benchmark MVPML data sets in terms of *ranking loss*, where the best performance (the smaller the better) on each data set is shown in boldface.

Comparing Approach	Controlling Parameters	Data Set							
		Emotions	Yeast	Pascal	EspGame5k	Mirflickr5k	Youku5k	Youku10k	Youku15k
FIMAN		**.173±.021**	**.179±.011**	**.092±.003**	**.225±.007**	**.138±.007**	**.068±.005**	**.053±.002**	**.045±.017**
GRADIS		.184±.025	.183±.008	.165±.006	.234±.010	.148±.007	.097±.005	.089±.003	.086±.004
F2L21F	$p = 0.3$	.232±.032	.362±.015	.097±.003	.234±.007	.151±.010	.099±.006	.060±.002	.047±.002
LSAMML	$r = 1$	.273±.043	.505±.019	.146±.005	.474±.017	.193±.011	.126±.008	.087±.004	.503±.030
PML-LRS		.218±.030	.214±.011	.331±.009	.254±.007	.225±.009	.094±.005	.084±.002	.082±.006
FPML		.209±.032	.211±.011	.300±.011	.289±.011	.193±.008	.209±.015	.195±.006	.189±.009
FIMAN		**.178±.022**	**.184±.012**	**.095±.003**	**.230±.007**	**.140±.007**	**.069±.005**	**.054±.003**	**.046±.002**
GRADIS		.196±.025	.189±.010	.186±.005	.253±.006	.167±.009	.119±.004	.115±.004	.115±.002
F2L21F	$p = 0.3$	.238±.027	.364±.014	**.095±.004**	.238±.008	.153±.010	.103±.006	.063±.003	.049±.002
LSAMML	$r = 2$	.232±.036	.504±.020	.151±.005	.490±.020	.195±.010	.128±.006	.094±.002	.491±.017
PML-LRS		.228±.030	.213±.011	.331±.008	.259±.010	.224±.009	.092±.006	.086±.002	.080±.008
FPML		.225±.029	.211±.011	.300±.010	.289±.012	.193±.008	.211±.009	.194±.007	.187±.008
FIMAN		**.177±.025**	**.173±.011**	**.095±.004**	**.233±.006**	**.142±.008**	**.071±.004**	**.055±.003**	**.046±.002**
GRADIS		.180±.041	.177±.013	.267±.008	.265±.005	.176±.008	.134±.008	.128±.003	.130±.003
F2L21F	$p = 0.3$	.240±.031	.364±.014	.097±.004	.243±.006	.155±.009	.109±.006	.066±.002	.051±.002
LSAMML	$r = 3$	.216±.048	.496±.028	.153±.004	.489±.020	.196±.010	.134±.007	.099±.003	.498±.029
PML-LRS		.227±.030	.214±.011	.336±.010	.265±.008	.223±.009	.096±.004	.085±.002	.081±.006
FPML		.231±.029	.211±.011	.302±.011	.289±.010	.194±.008	.205±.009	.191±.006	.183±.007
FIMAN		**.183±.026**	**.186±.011**	**.097±.004**	**.230±.007**	**.140±.007**	**.070±.003**	**.055±.002**	**.047±.002**
GRADIS		.187±.032	.195±.008	.242±.005	.272±.008	.190±.007	.157±.005	.154±.002	.155±.002
F2L21F	$p = 0.7$	.233±.019	.366±.014	**.097±.004**	.241±.007	.154±.010	.103±.005	.064±.002	.049±.002
LSAMML	$r = 1$	.311±.033	.503±.019	.154±.005	.470±.026	.195±.012	.129±.006	.095±.002	.497±.022
PML-LRS		.217±.026	.213±.011	.331±.010	.259±.008	.224±.009	.093±.005	.088±.003	.083±.006
FPML		.214±.025	.213±.011	.300±.011	.288±.011	.194±.008	.212±.012	.198±.009	.186±.008
FIMAN		**.193±.013**	**.187±.014**	**.107±.006**	**.234±.007**	**.143±.007**	**.077±.004**	**.058±.002**	**.051±.003**
GRADIS		.297±.034	.244±.012	.291±.007	.285±.008	.227±.006	.213±.008	.211±.004	.213±.004
F2L21F	$p = 0.7$	.253±.028	.370±.014	.108±.005	.245±.007	.160±.008	.115±.006	.069±.002	.053±.003
LSAMML	$r = 2$	.331±.054	.502±.027	.166±.006	.510±.015	.201±.011	.136±.007	.108±.003	.503±.025
PML-LRS		.250±.034	.214±.011	.336±.009	.265±.009	.223±.009	.098±.005	.087±.003	.085±.007
FPML		.241±.032	.212±.012	.302±.010	.289±.010	.195±.008	.213±.008	.196±.008	.280±.005
FIMAN		**.195±.027**	**.187±.012**	**.111±.005**	**.244±.009**	**.145±.008**	**.082±.004**	**.062±.003**	**.057±.002**
GRADIS		.274±.004	.196±.011	.322±.011	.336±.008	.250±.009	.251±.005	.250±.004	.247±.003
F2L21F	$p = 0.7$	.285±.036	.370±.014	.115±.004	.256±.009	.165±.010	.126±.006	.075±.002	**.057±.003**
LSAMML	$r = 3$	.289±.043	.498±.021	.175±.005	.483±.024	.204±.011	.147±.007	.126±.005	.492±.017
PML-LRS		.316±.029	.213±.011	.326±.008	.272±.010	.216±.010	.103±.006	.089±.002	.087±.007
FPML		.247±.038	.212±.011	.301±.009	.289±.012	.193±.008	.216±.009	.193±.009	.274±.006

As a favorable strategy for comparing multiple approaches over a number of data sets, *Friedman test* [5] is employed here for statistical significance analysis. Table 5 summarizes the Friedman statistics F_F over all evaluation metrics along with the critical value at 0.05 significance level. As shown in Table 5, across the 72 benchmark cases (8 real-world data sets × 9 settings controlling parameters), the null hypothesis of "equal" performance among comparing approaches is clearly rejected as the F_F statistic is greater than the critical value 2.2394.

Consequently, *Bonferroni-Dunn test* [5] is employed as the post-hoc test to show the relative performance among the comparing approaches. Here, FIMAN is treated as the control approach where the difference of average rank (over all data sets) between FIMAN and one comparing approach is calibrated with the *critical difference* (CD). If the average rank difference is greater than one CD (CD=0.6983 with $n = 6$ and $N = 72$), the performance between FIMAN and one comparing approach is deemed to be different.

Accordingly, Figure 2 illustrates the CD diagram [5] on each evaluation metric with FIMAN being the control approach. For each

Table 4: Predictive performance of each comparing approach (mean±std. deviation) on the benchmark MVPML data sets in terms of *micro-averaging AUC*, where the best performance (the larger the better) on each data set is shown in boldface.

Comparing Approach	Controlling Parameters	Data Set							
		Emotions	Yeast	Pascal	EspGame5k	Mirflickr5k	Youku5k	Youku10k	Youku15k
FIMAN		**.850±.021**	**.823±.012**	**.909±.004**	**.770±.007**	**.851±.006**	**.936±.004**	**.950±.002**	**.958±.002**
GRADIS		.836±.022	.819±.007	.824±.005	.759±.009	.847±.005	.894±.005	.902±.003	.905±.004
F2L21F	$p = 0.3$	.787±.029	.647±.015	.907±.003	.760±.008	.838±.007	.909±.006	.946±.002	**.958±.002**
LSAMML	$r = 1$	.736±.059	.495±.018	.856±.004	.523±.019	.785±.008	.883±.007	.915±.004	.497±.032
PML-LRS		.799±.025	.778±.010	.670±.006	.751±.008	.768±.010	.915±.005	.925±.003	.928±.005
FPML		.803±.020	.784±.010	.710±.010	.704±.010	.793±.007	.795±.017	.808±.005	.815±.009
FIMAN		**.830±.017**	**.816±.013**	**.908±.003**	**.765±.007**	**.838±.007**	**.935±.004**	**.949±.003**	**.957±.002**
GRADIS		.803±.020	.806±.010	.778±.006	.733±.007	.828±.007	.851±.004	.856±.005	.858±.002
F2L21F	$p = 0.3$	.771±.025	.645±.015	.906±.003	.755±.008	.834±.007	.904±.005	.943±.003	**.957±.002**
LSAMML	$r = 2$	.769±.030	.497±.016	.851±.005	.506±.018	.781±.008	.881±.006	.909±.002	.510±.017
PML-LRS		.788±.028	.776±.010	.672±.006	.744±.010	.770±.010	.916±.006	.923±.002	.929±.005
FPML		.789±.022	.784±.010	.710±.010	.704±.010	.794±.007	.794±.010	.809±.007	.818±.008
FIMAN		**.827±.020**	**.816±.012**	**.904±.003**	**.761±.007**	**.834±.008**	**.933±.004**	**.947±.003**	**.955±.002**
GRADIS		.790±.042	.805±.013	.692±.007	.712±.008	.811±.007	.827±.009	.827±.003	.824±.004
F2L21F	$p = 0.3$	.762±.023	.645±.015	.901±.004	.749±.006	.832±.006	.898±.006	.940±.003	.954±.002
LSAMML	$r = 3$	.673±.053	.503±.028	.845±.004	.517±.017	.778±.007	.874±.007	.904±.003	.503±.029
PML-LRS		.783±.024	.773±.010	.670±.007	.737±.008	.771±.010	.913±.005	.924±.002	.929±.005
FPML		.794±.042	.783±.010	.710±.010	.704±.009	.794±.007	.802±.010	.812±.006	.821±.008
FIMAN		**.822±.017**	**.815±.013**	**.902±.004**	**.765±.006**	**.839±.006**	**.933±.003**	**.948±.002**	**.956±.001**
GRADIS		.805±.023	.803±.007	.752±.004	.728±.007	.821±.005	.836±.005	.838±.002	.837±.002
F2L21F	$p = 0.7$	.771±.026	.643±.015	.901±.004	.753±.008	.834±.007	.904±.005	.942±.003	**.956±.002**
LSAMML	$r = 1$	.699±.045	.494±.019	.848±.005	.526±.030	.781±.009	.880±.006	.911±.005	.506±.023
PML-LRS		.791±.025	.775±.010	.674±.006	.744±.008	.770±.009	.916±.005	.921±.003	.927±.006
FPML		.798±.020	.783±.010	.710±.010	.704±.010	.794±.007	.793±.013	.805±.008	.818±.007
FIMAN		**.811±.015**	**.814±.014**	**891±.006**	**.754±.009**	**.835±.007**	**.927±.003**	**.944±.002**	**.950±.002**
GRADIS		.716±.029	.772±.012	.684±.002	.680±.004	.780±.005	.777±.007	.776±.004	.772±.003
F2L21F	$p = 0.7$	.743±.022	.639±.015	**.891±.005**	.741±.008	.826±.006	.892±.006	.936±.002	.949±.002
LSAMML	$r = 2$	.663±.056	.497±.026	.833±.005	.508±.020	.772±.008	.873±.007	.893±.003	.497±.027
PML-LRS		.743±.026	.769±.009	.673±.007	.732±.010	.772±.008	.911±.006	.922±.003	.925±.006
FPML		.777±.024	.783±.010	.711±.010	.702±.010	.794±.007	.792±.012	.807±.008	.729±.005
FIMAN		**.810±.021**	**.815±.013**	**.885±.005**	**.749±.009**	**.832±.007**	**.922±.003**	**.941±.003**	**.951±.002**
GRADIS		.681±.031	.744±.011	.652±.009	.662±.007	.756±.006	.723±.004	.725±.005	.724±.003
F2L21F	$p = 0.7$	.684±.021	.639±.013	.882±.004	.734±.010	.819±.007	.881±.005	.931±.003	.948±.003
LSAMML	$r = 3$	.584±.029	.505±.020	.822±.005	.513±.021	.764±.009	.861±.008	.876±.004	.510±.018
PML-LRS		.665±.024	.762±.009	.675±.007	.728±.008	.774±.010	.906±.006	.921±.003	.923±.006
FPML		.758±.032	.783±.010	.710±.009	.703±.011	.793±.007	.790±.010	.810±.009	.735±.006

comparing approach, its average rank is marked along the axis with lower (better) ranks to the right. A thick line is used to interconnect the control approach and one comparing approach if their average rank difference is within one CD. Based on the experimental results, it is impressive to observe that:

- As shown in Figure 2, no comparing approach has significantly outperformed FIMAN in terms of each evaluation metric. Furthermore, FIMAN achieves lowest average rank on five out of six evaluation metrics.
- Compared to the MVPML approach GRADIS, FIMAN achieves comparable performance in terms of *one-error* while achieves superior performance in terms of the other evaluation metrics.
- Compared to the two degenerated MVML approaches F2L21F and LSAMML, FIMAN achieves superior performance in terms of each evaluation metric. Furthermore, compared to the two degenerated PML approaches PML-LRS and FPML, FIMAN achieves comparable performance to both of them in

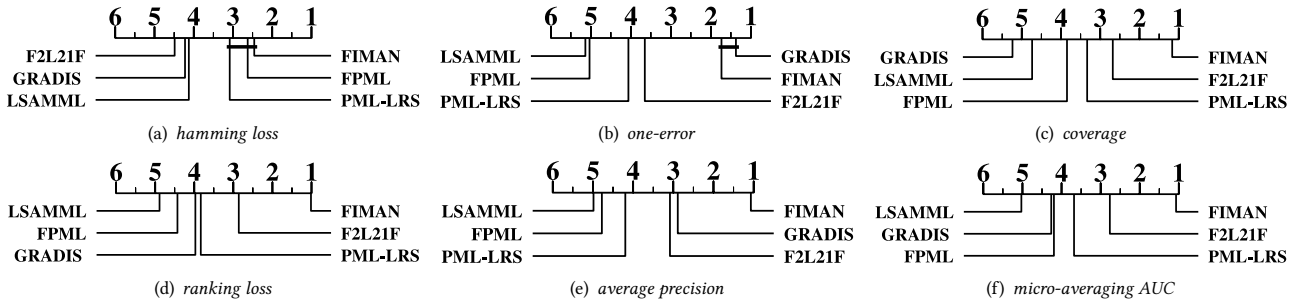


Figure 2: Comparison of Fiman (control algorithm) against other comparing approaches with the *Bonferroni-Dunn test*. Approaches not connected with Fiman in the CD diagram are considered to have significantly different performance from the control algorithm (CD=0.6983 at 0.05 significance level).

Table 5: Friedman statistics F_F in terms of each evaluation metric as well as the critical value at 0.05 significance level (# comparing approaches $n = 6$, # benchmark data sets $N = 72$).

Evaluation metric	F_F	critical value
hamming loss	20.6313	2.2394
one-error	206.5612	
coverage	114.6231	
ranking loss	88.6629	
average precision	113.8453	
micro-averaging AUC	92.0994	

terms of *hamming loss* while achieves superior performance in terms of the other evaluation metrics.

- As shown in Tables 3 and 4, the performance advantage of Fiman over comparing approaches is stable under various settings of controlling parameters p and r.

4.3 Further Analysis

4.3.1 Performance Sensitivity. As shown in Table 1, the proposed Fiman approach involves four parameters, i.e. k (# nearest neighbors for affinity graph construction), η (regularization coefficient in Eq.(9)), t_d (disambiguation threshold in Eq.(8)) and t_p (prediction threshold in Eq.(11)).

Figure 3 gives an illustrative example on how the performance of Fiman changes as the value of each parameter varies (evaluation metric: *hamming loss*; data sets: `Emotions` and `Yeast`). Accordingly, we can have the following observations: 1) The performance of Fiman is relatively stable under varying values of k (Figure 3(a)) and η (Figure 3(b)); 2) The performance of Fiman is relatively stable as t_d takes value in [0.1 0.4] while gradually worsens as t_d grows from 0.5 to 0.9 (Figure 3(c)); 3) The performance of Fiman gradually improves as t_p grows from 0.1 to 0.4 and becomes relatively stable in [0.5, 0.9] (Figure 3(d)). As per the above observations, the parameter configuration of Fiman given in Subsection 4.1.2 (i.e. k=10, η=1, t_d=0.4, t_p=0.6) is employed in this paper.

4.3.2 Ablation Study. As shown in Table 1, Fiman leverages the multi-view representation of each MVPML example by adaptively fusing the affinity graph constructed over each view to yield an aggregate mainfold structure (Steps 1-14). To show the usefulness of the feature-induced manifold structure, a variant of Fiman-c is investigated.

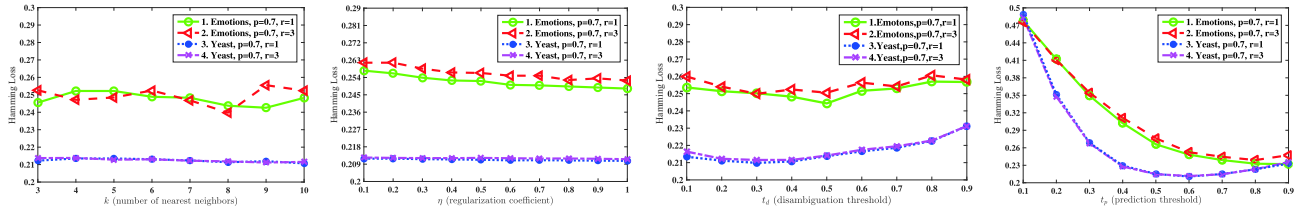
Specifically, Fiman-c instantiates the matrix $\mathbf{W}$ in Step 14 by constructing a weighted kNN graph based on direct concatenation of feature vectors of all views while keeps other learning process unchanged. Accordingly, Table 6 reports the detailed experimental results of Fiman and its variant Fiman-c in terms of each evaluation metric (controlling parameters: $p = 0.5$, $r = 1$). Out of the 48 cases (8 data sets × 6 evaluation metrics), pairwise t-test at 0.05 significance level show that Fiman significantly outperforms Fiman-c in 95.83% cases. These results validate the usefulness of aggregate manifold structure learned by Fiman for solving MVPML problem.

5 CONCLUSION

In this paper, the multi-view partial multi-label learning problem is studied where a novel approach based on feature-induced manifold disambiguation is proposed. To deal with the inherent multi-view and noisy labeling characteristics, affinity information w.r.t. different views are adaptively fused to disambiguate candidate label set by enforcing manifold structure preservation in the label space. Comprehensive comparative studies validate the superiority of the proposed approach in learning from MVPML examples.

REFERENCES

[1] A. Blum and T. Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory*. Madison, WI, 92–100.
[2] C.-H. Chen, V. M. Patel, and R. Chellappa. 2018. Learning from ambiguously labeled face images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40, 7 (2018), 1653–1667.
[3] Z.-S. Chen, X. Wu, Q.-G. Chen, Y. Hu, and M.-L. Zhang. 2020. Multi-view partial multi-label learning with graph-based disambiguation. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*. New York, NY.
[4] T. Cour, B. Sapp, and B. Taskar. 2011. Learning from partial labels. *Journal of Machine Learning Research* 12, May (2011), 1501–1536.
[5] J. Demšar. 2006. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research* 7, Jan (2006), 1–30.
[6] A. Elisseeff and J. Weston. 2002. A kernel method for multi-labelled classification. In *Advances in Neural Information Processing Systems 14*, T. G. Dietterich, S. Becker, and Z. Ghahramani (Eds.). MIT Press, Cambridge, MA, 681–687.



(a) Varying k (η=1, t_d=0.4, t_p=0.6) (b) Varying η (k=10, t_d=0.4, t_p=0.6) (c) Varying t_d (η=1, k=10, t_p=0.6) (d) Varying t_p (η=1, k=10, t_d=0.4)

Figure 3: Performance of FIMAN (in terms of *hamming loss*) changes as the value of each parameter varies on the `Emotions` and `Yeast` data sets. (a) k increases from 3 to 10 with step 1; (b) η increases from 0.1 to 1 with step 0.1; (c) t_d increases from 0.1 to 0.9 with step 0.1; (d) t_p increases from 0.1 to 0.9 with step 0.1.

Table 6: Predictive performance of FIMAN and FIMAN-C in terms of each evaluation metric, where the best performance on each data set is shown in boldface (controlling parameters: $p = 0.5$, $r = 1$; "↓": the smaller the better; "↑": the larger the better).

Comparing Approach	Evaluation Metric	Data Set							
		Emotions	Yeast	Pascal	EspGame5k	Mirflickr5k	Youku5k	Youku10k	Youku15k
FIMAN	*hamm. loss* ↓	**.246±.024**	**.213±.007**	**.072±.003**	**.071±.001**	**.112±.004**	**.018±.001**	**.019±.001**	**.018±.001**
FIMAN-C		.271±.023	.337±.013	.134±.006	.221±.010	.251±.010	.041±.002	.031±.002	.028±.002
FIMAN	*one-error* ↓	**.292±.043**	**.218±.027**	**.287±.013**	**.561±.020**	**.234±.023**	**.354±.026**	.333±.018	**.330±.016**
FIMAN-C		.298±.064	.321±.030	.297±.012	.590±.015	.295±.023	.368±.024	**.324±.020**	.337±.009
FIMAN	*coverage* ↓	**.323±.027**	**.485±.014**	**.141±.005**	**.450±.010**	**.394±.019**	**.135±.007**	**.108±.005**	**.091±.003**
FIMAN-C		.341±.027	.681±.014	.222±.008	.531±.009	.615±.016	.162±.007	.128±.007	.111±.003
FIMAN	*ranking loss* ↓	**.183±.021**	**.183±.012**	**.094±.003**	**.227±.007**	**.140±.007**	**.069±.005**	**.054±.003**	**.046±.001**
FIMAN-C		.201±.028	.372±.016	.139±.005	.287±.006	.295±.009	.072±.003	.063±.003	.054±.002
FIMAN	*avg. prec.* ↑	**.818±.024**	**.756±.013**	**.747±.008**	**.394±.006**	**.651±.012**	**.606±.011**	**.632±.010**	**.640±.009**
FIMAN-C		.804±.034	.609±.016	.722±.008	.350±.004	.528±.014	.597±.010	.625±.010	.635±.007
FIMAN	*mavg. AUC* ↑	**.823±.021**	**.816±.013**	**.905±.004**	**.768±.006**	**.839±.007**	**.934±.004**	**.949±.003**	**.956±.001**
FIMAN-C		.797±.028	.626±.014	**.905±.004**	.707±.007	.705±.011	.920±.004	.943±.003	.947±.002

[7] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. 2010. The Pascal visual object classes (VOC) challenge. *International Journal of Computer Vision* 88, 2 (2010), 303–338.

[8] J.-P. Fang and M.-L. Zhang. 2019. Partial multi-label learning via credible label elicitation. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*. Honolulu, HI, 3518–3525.

[9] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter. 2017. Audio set: An ontology and human-labeled dataset for audio events. In *Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing*. New Orleans, LA, 776–780.

[10] E. Gibaja and S. Ventura. 2015. A tutorial on multilabel learning. *ACM Computing Surveys* 47, 3 (2015), Article 52.

[11] S. He, K. Deng, L. Li, S. Shu, and L. Liu. 2019. Discriminatively relabel for partial multi-label learning. In *Proceedings of the 19th IEEE International Conference on Data Mining*. Beijing, China, 280–288.

[12] M. J. Huiskes and M. S. Lew. 2008. The MIR Flickr retrieval evaluation. In *Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval*. Vancouver, Canada, 39–43.

[13] L. Liu and T. Dietterich. 2012. A conditional multinomial mixture model for superset label learning. In *Advances in Neural Information Processing Systems 25*, P. Bartlett, F. C. N. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger (Eds.). MIT Press, Cambridge, MA, 557–565.

[14] M. Liu, Y. Luo, D. Tao, C. Xu, and Y. Wen. 2015. Low-rank multi-view learning in matrix completion for multi-label image classification. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*. Austin, TX, 2778–2784.

[15] Y. Luo, D. Tao, C. Xu, C. Xu, H. Liu, and Y. Wen. 2013. Multiview vector-valued manifold regularization for multilabel image classification. *IEEE Transactions on Neural Networks and Learning Systems* 24, 5 (2013), 709–722.

[16] L. Sun, S. Feng, T. Wang, C. Lang, and Y. Jin. 2019. Partial multi-label learning by low-rank and sparse decomposition. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*. Honolulu, HI, 5016–5023.

[17] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the 29th IEEE Conference on Computer Vision and Pattern Recognition*. Las Vegas, NV, 2818–2826.

[18] Q. Tan, G. Yu, J. Wang, C. Domeniconi, and X. Zhang. in press. Individually- and commonality-based multiview multilabel learning. *IEEE Transactions on Cybernetics* (in press).

[19] K. Trohidis, G. Tsoumakas, G. Kalliris, and I. Vlahavas. 2008. Multilabel classification of music into emotions. In *Proceedings of the 2008 International Conference on Music Information Retrieval*. Philadelphia, PA, 325–330.

[20] L. von Ahn and L. Dabbish. 2004. Labeling images with a computer game. In *Proceedings of ACM CHI 2004 Conference on Human Factors in Computing Systems*. Vienna, Austria, 319–326.

[21] D.-B. Wang, L. Li, and M.-L. Zhang. 2019. Adaptive graph guided disambiguation for partial label learning. In *Proceedings of the 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Anchorage, AK, 83–91.

[22] H. Wang, W. Liu, Y. Zhao, C. Zhang, T. Hu, and G. Chen. 2019. Discriminative and correlative partial multi-label learning. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. Macau, China, 3691–3697.

[23] X. Wu, Q.-G. Chen, Y. Hu, D. Wang, X. Chang, X. Wang, and M.-L. Zhang. 2019. Multi-view multi-label learning with view-specific information extraction. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. Macau, China, 3884–3890.

[24] T. Xia, D. Tao, T. Mei, and Y. Zhang. 2010. Multiview spectral embedding. *IEEE Transactions on System, Man, and Cybernetics - Part B: Cybernetics* 40, 6 (2010), 1438–1446.

[25] M.-K. Xie and S.-J. Huang. 2018. Partial multi-label learning. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*. New Orleans, LA, 4302–4309.

[26] Y. Xing, G. Yu, C. Domeniconi, J. Wang, and Z. Zhang. 2018. Multi-label co-training. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*. Stockholm, Sweden, 2882–2888.

[27] F. Yu and M.-L. Zhang. 2017. Maximum margin partial label learning. *Machine Learning* 106, 4 (2017), 573–593.

[28] G. Yu, X. Chen, C. Domeniconi, J. Wang, Z. Li, Z. Zhang, and X. Wu. 2018. Feature-induced partial multi-label learning. In *Proceedings of the 18th IEEE International Conference on Data Mining*. Singapore, 1398–1403.

[29] W. Zhan and M.-L. Zhang. 2017. Inductive semi-supervised multi-label learning with co-training. In *Proceedings of the 23rd ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Halifax, Canada, 1305–1314.

[30] C. Zhang, Z. Yu, Q. Hu, P. Zhu, X. Liu, and X. Wang. 2018. Latent semantic aware multi-view multi-label classification. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*. New Orleans, LA, 4414–4421.

[31] L. Zhang, Q. Zhang, L. Zhang, D. Tao, X. Huang, and B. Du. 2015. Ensemble manifold regularized sparse low-rank approximation for multiview feature embedding. *Pattern Recognition* 48, 10 (2015), 3102–3112.

[32] M.-L. Zhang and Z.-H. Zhou. 2014. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering* 26, 8 (2014), 1819–1837.

[33] T. Zhang, D. Tao, X. Li, and J. Yang. 2009. Patch alignment for dimensionality reduction. *IEEE Transactions on Knowledge and Data Engineering* 21, 9 (2009), 1299–1313.

[34] Z.-H. Zhou and M. Li. 2010. Semi-supervised learning by disagreement. *Knowledge and Information Systems* 24, 3 (2010), 415–439.

[35] Z.-H. Zhou and M.-L. Zhang. 2017. Multi-label learning. In *Encyclopedia of Machine Learning and Data Mining*, C. Sammut and G. I. Webb (Eds.). Springer, Berlin, 875–881.

[36] P. Zhu, Q. Hu, Q. Hu, C. Zhang, and Z. Feng. 2018. Multi-view label embedding. *Pattern Recognition* 84 (2018), 126–135.

[37] X. Zhu, X. Li, and S. Zhang. 2016. Block-row sparse multiview multilabel learning for image classification. *IEEE Transactions on Cybernetics* 46, 2 (2016), 450–461.