联合嵌入式多标签分类算法

刘慧婷 [1, 2] 冷新杨 [1, 2] 王利利 [1, 2] 赵 鹏 [1, 2]

摘 要 现有的一些多标签分类算法, 因多标签数据含有高维的特征或标签信息而变得不可行. 为了解决这一问题, 提出基于去噪自编码器和矩阵分解的联合嵌入多标签分类算法 Deep AE-MF. 该算法包括两部分: 特征嵌入部分使用去噪自编码器对特征空间学习得到非线性表示, 标签嵌入部分则是利用矩阵分解直接学习到标签空间对应的潜在表示与解码矩阵. Deep AE-MF 将特征嵌入和标签嵌入的两个阶段进行联合, 共同学习一个潜在空间用于模型预测, 进而得到一个有效的多标签分类模型. 为了进一步提升模型性能, 在 Deep AE-MF 方法中对标签间的负相关信息加以利用. 通过在不同数据集上进行实验证明了提出 Deep AE-MF 方法的有效性和鲁棒性.

关键词 多标签分类, 矩阵分解, 去噪自编码器, 标签嵌入

引用格式 刘慧婷, 冷新杨, 王利利, 赵鹏. 联合嵌入式多标签分类算法. 自动化学报, 2019, **45**(10): 1969−1982

DOI 10.16383/j.aas.c180087

A Joint Embedded Multi-label Classification Algorithm

LIU Hui-Ting[1, 2] LENG Xin-Yang[1, 2] WANG Li-Li[1, 2] ZHAO Peng[1, 2]

Abstract Some existing classification algorithms become infeasible anymore, because most multi-label data contains high-dimensional features or label information. To solve this problem, a joint embedded multi-label learning classification algorithm named Deep AE-MF is proposed in this paper, which is based on denoising auto-encoder and matrix factorization. The algorithm includes two parts: the feature embedding part uses denoising auto-encoder to obtain the nonlinear representation of feature space learning, and the label embedding part directly learns the potential representation and decoding matrix of the corresponding label space using matrix factorization. In order to get an effective classification model, Deep AE-MF combines the two phases of feature embedding and label embedding to learn a potential space for model prediction. To further improve the performance of the model, the negative correlation between tags is exploited in Deep AE-MF. Experiments on different datasets show the effectiveness and robustness of the proposed Deep AE-MF method.

Key words Multi-label classification, matrix factorization, denoising auto-encoder, label embedding

Citation Liu Hui-Ting, Leng Xin-Yang, Wang Li-Li, Zhao Peng. A joint embedded multi-label classification algorithm. *Acta Automatica Sinica*, 2019, **45**(10): 1969−1982

单个对象含有多个标签注释的学习与挖掘是众多领域中经常遇到和研究的问题[1−7]. 例如: 在文本分类中, 每个文档可能会被赋予几个预先定义的主题; 在生物学领域中, 每个基因可能会同时含有几种不同的功能片段、如新陈代谢功能、转录功能和蛋白质合成功能等; 在场景图片分类中, 每张场景图片从不同的角度分析会有不同含义, 如人物、沙滩和天空等. 这些问题都有一个共同特点, 即单个实例同时含有多个标签, 或被同时分为多个类别, 被称为多标签问题.

近年来, 多标签学习问题被众多学者关注与研究, 提出了一系列的多标签学习算法. 例如, 基于二元相关 (Binary relevance, BR) 的方法[5] 将多标签学习问题分割成多个独立的二元分类问题, 即为每一个标签训练一个分类器; 基于标签排序的方法[8−9] 将标签成对比较进行排序, 把多标签学习转化为标签排序问题: 基于算法改编的方法[10−13] 将单标签学习的算法进行改进使之适用于多标签学习. 随着深度学习技术[14] 的发展, 文本表示能力的进一步提高, 基于表示学习的多标签算法[15] 被提出; 此外, 基于树的方法[16−17] 和基于嵌入的方法[18−28] 被提出用于提高多标签分类性能和减少面对高维数据时产生的昂贵的时间开销.

现存的多标签分类算法可分为两大类: 问题转化法 (Problem transformation methods, PTM) 和

收稿日期 2018-02-05 录用日期 2018-05-18
Manuscript received February 5, 2018; accepted May 18, 2018
国家自然科学基金 (61202227, 61602004) 资助
Supported by National Natural Science Foundation of China (61202227, 61602004)
本文责任编委 张敏灵
Recommended by Associate Editor ZHANG Min-Ling
1. 安徽大学计算智能与信号处理教育部重点实验室 合肥 230601 2. 安徽大学计算机科学与技术学院 合肥 230601
1. Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, Anhui University, Hefei 230601 2. School of Computer Science and Technology, Anhui University, Hefei 230601

算法改编法 (Algorithm adaption methods, AAM). PTM 在解决多标签分类问题时, 算法具有简单, 易于理解与实现等优点. 但 PTM 缺点也比较明显, 如基于二元相关的方法忽略了标签间关系: 考虑到标签间的高阶相关性的标签幂集法 (Label powerset)[29] 因含有指数级的标签空间, 不仅会导致训练时间复杂度过高, 而且还可能存在标签类别的不平衡性等问题; 基于链式标签法的算法[9, 30], 性能完全依赖于链式标签排序, 但最优排序未知. AAM 通过改进已有的机器学习算法来解决多标签学习问题, 如基于 SVM 改进的 RankSVM 算法[10−11]、基于 kNN 改进的 IMLLA 算法[12] 及基于朴素贝叶斯改进的 NBML 算法[13]; 这些改进型的算法避免了为每个标签单独学习而忽略了标签间的关系, 当遇到具有高维特性的多标签数据时, 不仅需要较大的时间消耗, 性能还会有所损失.

主成分分析算法 (Principal component analysis, PCA)[31]、线性判别分析 (Linear discriminant analysis, LDA)[32] 及局部线性嵌入 (Locally linear embedding, LLE)[33] 等各种嵌入技术常被用于多标签分类任务. 矩阵分解技术在低维嵌入过程中可同时得到低维嵌入表示 C 及解码矩阵 D, 相比于使用 PCA 及 LDA 需要两个独立步骤 (编码与解码), 降低了误差.

为了解决 PTM 与 AAM 面临的问题, 考虑到矩阵分解技术的优势, 本文提出基于去噪自编码器 (Stack denoising autoencoder, SDAE) 和矩阵分解的联合嵌入学习算法 Deep AE-MF, 该算法不但能够得到一个具有深层语义的文本表示, 还能在降低时间复杂度的同时探索标签间的关系. 它能够将 SDAE 对特征学习到的深层语义低维表示和矩阵分解得到的标签低维表示联合在一起共同学习, 得到一个高效的多标签分类模型. 与 BR 型算法对比, Deep AE-MF 在学习时能够利用矩阵分解技术对标签间的关系进行间接探索; 与 AAM 型算法相比较, Deep AE-MF 使用 SDAE 技术对特征进行了非线性学习, 得到了深层语义的文本表示; 与特征/标签嵌入类算法相比, Deep AE-MF 整合了 SDAE 和矩阵分解两种技术对特征与标签同时进行联合嵌入学习, 使得模型的预测与嵌入两个学习阶段同时进行.

在 Deep AE-MF 算法中, 特征部分的学习利用 SDAE 能够对浅层特征挖掘出深层语义的特性得到一个深层语义的低维表示; 标签部分的学习则使用矩阵分解技术得到低维嵌入表示 C 及解码矩阵 D, 避免了采用不合适的编码函数的风险同时, 解决了编码与解码需要单独学习的代价; 最后将特征与标签部分学习得到的低维表示联系在一起, 使得特征嵌入和标签嵌入能够共同学习并得到一个共享潜在子空间用于模型的学习和预测阶段. 本文主要贡献点如下:

1) Deep AE-MF 方法紧密耦合了 SDAE 和矩阵分解, 是一种新的基于深度学习和矩阵分解联合学习的多标签分类算法.

2) 特征嵌入过程使用 SDAE 能够有效地学习到数据浅层特征对应的深层语义表示.

3) Deep AE-MF 方法将特征嵌入和标签嵌入进行联合学习, 能够为特征与标签空间找到一个有效的潜在共享子空间, 提高多标签分类算法的泛化性能.

4) 实验部分通过在 6 个常用的数据集上对比 10 种多标签分类算法, 证明了提出的 Deep AE-MF 方法的有效性.

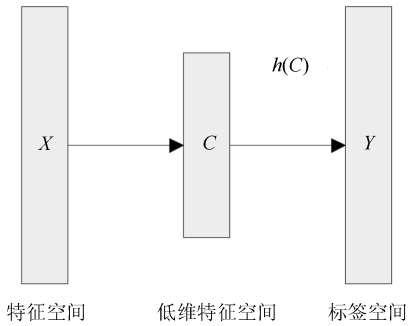
1 相关工作

随着互联网技术的普及, 信息呈现指数式的爆炸增长, 使得数据信息具有高维、无序及冗余等特点. 例如, 在一个网络社区中对某张图片进行标记注解时, 其标记可能需要从百万候选标记中选择. 这些高维数据的出现, 使现有的基于 PTM 和 AAM 的多标签分类方法变的不可行[18], 因为这些方法面对高维数据问题时, 所需要的时间代价是不可负担的.

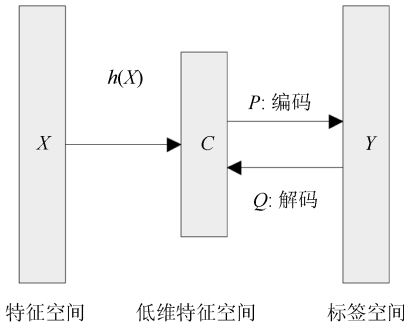
为了解决上述问题, 文献 [34] 提出利用标签之间存在的依赖关系对标签构造一棵树结构用于多标签分类; 文献 [35] 采用部分标签代替全体标签的方法. 这些方法只是在一定程度上缓解了高维多标签学习存在的时间花销过高的问题.

为了更加有效地缓解高维数据带来的时间花费过大而影响算法性能的问题, 维度约减/嵌入技术被用于多标签学习. 维度约减/嵌入的方法大致可分为两类: 基于特征的维度约减 (Feature space dimension reduction, FSDR)[19−23] 和基于标签的维度约减 (Label space dimension reduction, LSDR)[24−28]. 如图 1 (a) 所示, FSDR 首先将高维的特征空间 X 转化至低维潜在空间 C, 接着在 C 与 Y 之间学习到一个映射 $h(C)$; 对未知标签实例进行预测时, 先将其对应的高维特征转化为低维表示, 再利用映射 $h(C)$ 得到最终的预测. 文献 [19] 在 BR 算法框架上对每个独立的二元分类问题学习时, 使用无监督线性方式将原始特征空间转化为低维潜在空间, 有效地减少了时间开销. 文献 [20] 指出在 BR 框架下, 每个学习问题的输入都是相同的, 可通过维度约减学习得到一个共享的子空间, 不仅可以减少时间开销, 还解决了当标签高维时参数过多的问题. 文献 [19−20] 均是采用线性方式对高维特征进行低维嵌入学习, 但是在现实世界中, 数据的高维表示与低维表示之间的关系, 大都是非线性的. 文献 [21] 则

是利用非线性的核函数对原始特征进行非线性转化, 文献 [22−23] 利用深度学习能够挖掘提取特征之



(a) 特征嵌入
(a) Feature embedding



(b) 标签嵌入
(b) Label embedding

图 1 基于嵌入方法的两种模型图

Fig. 1 Illustration of models based on embedding method

间深层关系的特性, 进一步对非线性关系探索, 但由于只考虑到特征信息限制了模型的性能. LSDR 型嵌入学习则是针对高维的标签向量而提出的一种嵌入技术, 如图 1 (b) 所示, 该方法先对高维的标签空间 Y 进行编码至低维潜在空间 C, 接着再进行学习由 X 与 C 之间的映射 $h(X)$; 当对未知标签实例进行预测时, 首先由映射 $h(X)$ 得到低维标签表示的预测结果, 然后再使用 Q (解码) 得到最终的预测结果. 文献 [24] 首次采用基于标签嵌入的方法进行多标签分类, 它通过对稀疏的原始标签进行探索, 提出使用线性技术 CS (Compress senseing) 将原始标签空间转化为低维标签空间 C, 再利用 CoSaMP 重构方法将 C 解码为原始标签空间表示. 作为对文献 [24] 的进一步研究, 文献 [18] 将标签重构过程和分类模型预测过程进行联合优化, 提出基于贝叶斯框架的 BLM-CS 方法用于进行多标签分类. 虽然文献 [18, 24] 能够有效地减少面对高维数据时的多标签学习的时间开销, 但是在对低维标签空间 C 转化时未考虑标签之间存在的关系, 限制了其模型的性能. 于是文献 [25] 提出 PLST 方法在原始标签空间与低维标签空间重构解码过程中使用 PCA 降维技术, 在对高维标签向量进行约减的同时探索到标签间的关系. 文献 [26] 指出 PLST 方法在进行维度约减时只是单纯地考虑标签信息却没有使用相关的特征信息, 因此在 PLST 方法基础上提出 CPLST 方法, 在原始标签空间重构的过程中引入相关的特征信息, 进一步提高模型对未标记数据预测的准确率. 文献 [27] 提出 FaIE 方法, 在对原始标签矩阵空间 Y 转化时使用矩阵分解技术得到低维空间表示 C 和解码过程 Q; 与 PLST 和 CPLST 方法使用的显式编码相比, 减少不恰当使用编码函数的风险. 文献 [28] 也是基于标签嵌入型的方法. LSDR 与 FSDR 均是以找到一个合适且有效的低维表示空间为目标, 因此又可统一被称为基于嵌入的方法.

由上述介绍可知, 现有基于嵌入的方法中, 少数工作利用核函数 (如: 多项式核、高斯核等) 进行非线性方式转化; 更少有工作在转化时同时利用特征与标签信息. 因此, 本文提出 Deep AE-MF 方法, 将模型的预测与学习过程紧密耦合共同优化学习, 在联合嵌入学习过程中, 不仅能够将特征的非线性表示用于标签嵌入学习过程中, 还能对标签间的关系进行探索并加以利用, 从而得到一个高效的多标签分类模型.

2 Deep AE-MF 算法分析与设计

2.1 基础定义

给定一个含有 N 个样本的数据集 $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{N} = \{X, Y\}$, 其中 $X = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N]^{\mathrm{T}} \in \mathbf{R}^{N\times d}$, $Y \in \mathbf{R}^{N\times K}$, X 指数据集的特征空间, Y 指数据集的标签空间, N 是数据集中样本的个数, d 是特征向量的维度, K 是标签向量的维度. 对于实例对 $(\boldsymbol{x}_i, \boldsymbol{y}_i)$, $\boldsymbol{x}_i$ 是数据集中第 i 个实例对应的特征向量, $\boldsymbol{y}_i$ 则是数据集中第 i 个实例对应的标签向量, 当其含有第 j 个标签时有 $y_{ij} = 1$; 否则, 有 $y_{ij} = -1$. 在模型训练学习输入时统一用 $(\boldsymbol{x}_{\mathrm{tr}}, \boldsymbol{y}_{\mathrm{tr}})$ 表示训练实例对, 测试输入时统一用 $(\boldsymbol{x}_{\mathrm{test}}, \boldsymbol{y}_{\mathrm{test}})$ 表示测试实例对, $\boldsymbol{x}_{\mathrm{tr}}$ 和 $\boldsymbol{y}_{\mathrm{tr}}$ 分别指训练与测试时使用的实例对应的特征向量, $\boldsymbol{x}_{\mathrm{test}}$ 和 $\boldsymbol{y}_{\mathrm{test}}$ 分别指训练与测试时使用的实例对应的标签向量.

定义 1. 多标签分类是指利用给定数据集 $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{N}$, 学习到一个映射 F: $X \to Y$, 当给定一个测试实例 $(\boldsymbol{x}_{\mathrm{test}}, \boldsymbol{y}_{\mathrm{test}})$, 输入 $\boldsymbol{x}_{\mathrm{test}}$ 由映射 F 可正确地预测出 $\boldsymbol{y}_{\mathrm{test}}$.

定义 2. 在多标签分类学习中, 若在标签 i 出现的实例中, 总是有标签 j 出现或标签 j 几乎都不出现, 这种标签之间的共现或非共现现象被认为标签间具有相关性; 前者被称作标签间正相关性, 后者则

是标签间负相关性. 二者形式化定义分别如下:

$$Pos(j,t)_{j\neq t}=\sum_{i=1}^{N}I(y_{ij}=y_{it}) \tag{1}$$

$$Neg(j,t)_{j\neq t}=\sum_{i=1}^{N}I(y_{ij}\neq y_{it}) \tag{2}$$

式 (1) 统计的是任意两个不同的标签在数据集中的共现次数, 次数越大则认为二者具有更强的正相关性 (即二者具有更强的 "正向依赖" 关系); 式 (2) 统计的是任意两个不同的标签在数据集中的非共现次数, 次数越大则认为二者具有更强的负相关性 (即二者具有更强的 "负向依赖" 关系).

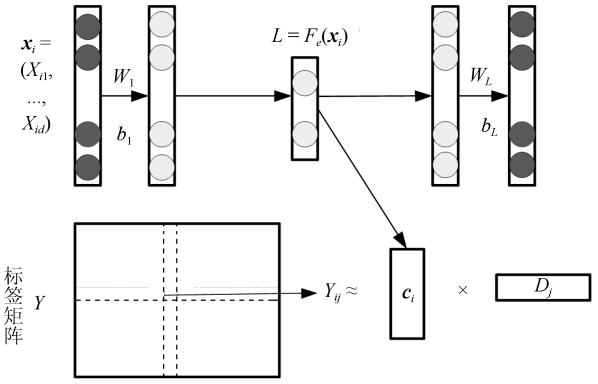


图 2 Deep AE-MF 算法模型图

Fig. 2 The model of algorithm deep AE-MF

2.2 Deep AE-MF 方法

在标签嵌入方法的思想上结合深度学习, 提出一种多标签分类方法 Deep AE-MF. 如图 2 所示, Deep AE-MF 是基于 SDAE 和矩阵分解的联合嵌入学习模型. 由文献 [36] 可知使用 SDAE 对原始特征矩阵 X 探索与学习, 可得到一个具有深层语义的低维表示 L (即 $F_e(X)$); 矩阵分解则是对标签矩阵 Y 直接分解学习得到 Y 在低维空间的潜在表示 C 及其解码矩阵 D (即 $Y=CD^{\mathrm{T}}$). 在训练过程中, 将训练实例对 $(\boldsymbol{x}_{\mathrm{tr}},\boldsymbol{y}_{\mathrm{tr}})$ 输入到 Deep AE-MF 模型中, 由 SDAE 和矩阵分解分别得到对应的特征与标签低维空间表示, 再利用 CCA (Canonical correlation analysis) 技术将两者对应的低维空间表示耦合在一起, 使二者对应的低维潜在表示具有最大相关性, 即更小的差异性, 以此为模型学习到合适的潜在低维空间 C. 于是, Deep AE-MF 方法的目标函数如下所示:

$$\begin{aligned}\Omega=\min_{\Theta}\lambda\Phi_1(X)+\beta\Phi_2(Y,C,D)+\\ \alpha\Phi_3(F_e,C)+\gamma\Phi_4(\Theta)\end{aligned} \tag{3}$$

其中, Φ_1 为特征嵌入学习损失, Φ_2 是标签嵌入学习损失, Φ_3 是指 X 与 Y 联合嵌入学习共同子空间损失, Φ_4 则是模型中参数 Θ 的正则化, α、β、λ、γ 则是用于平衡各种损失的超参数.

当 Deep AE-MF 模型学习完成后, 能够对任意输入预测其对应的标签. 即在 Deep AE-MF 模型中输入测试实例 $\boldsymbol{x}_{\mathrm{test}}$ 后, 首先 $\boldsymbol{x}_{\mathrm{test}}$ 通过 SDAE 中 F_e 编码转化为低维空间表示, 接着再利用矩阵 D 进行解码得到最终预测结果 $\boldsymbol{y}_{\mathrm{test}}$ (即 $\boldsymbol{y}_{\mathrm{test}}=DF_e(\boldsymbol{x}_{\mathrm{test}})$). 下面将按照特征嵌入、标签嵌入及联合嵌入三部分详细介绍 Deep AE-MF 模型.

2.2.1 特征低维嵌入学习

为了能够将高维特征空间 X 有效地转化至低维嵌入空间 L, 且更好地探索二者间的非线性关系, 使用 SDAE 对特征进行低维嵌入学习. SDAE 是一种以自身输入作为输出的前馈神经网络. 如图 2 上部分所示, SDAE 结构由 5 层网络构成, 以中间层为界, 左边几层称之为编码层 F_e, 右边几层称之为解码层 F_d, 本文取 SDAE 的中间层即 $F_e(X)$ 作为 X 对应的低维潜在空间 L 的表示. 为了避免过拟合, 保证找到有效潜在空间 L, 在对 SDAE 输入时加入高斯噪声 ε, 式 (3) 中 Φ_1 即为对特征低维嵌入学习时 SDAE 产生的损失, 其详细形式如下所示:

$$\Phi_1(X)=\|F_d(F_e(X+\varepsilon))-X\|_F^2 \tag{4}$$

其中, $X\in\mathbf{R}^{N\times d}$ 是对应的未加入噪声的真实输入特征向量, $\varepsilon\in\mathbf{R}^{N\times d}$ 是指通过高斯分布产生的噪声矩阵, 矩阵内的元素值均在 0 与 1 之间, $X+\varepsilon\in\mathbf{R}^{N\times d}$ 指加入高斯噪声后的输入, $F_d(F_e(X+\varepsilon))$ 是 SDAE 的预测输出, $\|\cdot\|_F$ 是指傅里叶标准化 (即矩阵 F-范数). 为了简便, 除非特别说明, 在下面的论述中 $(X+\varepsilon)$ 均用 X 代替.

2.2.2 标签低维嵌入学习

如图 1 (b) 所示, 现有的大多数的标签嵌入学习包括编码 P 与解码 Q 两个独立部分, 通常对于编码部分是基于某种假设得到编码函数 P. 但基于某种假设构造的显式编码函数可能会得到一个不恰当、不准确的低维嵌入转化, 弱化了模型的性能. 为了避免这种风险, 本文对标签空间 Y 进行无假设嵌入学习 — 使用矩阵分解技术直接得到 Y 的低维嵌入表示 C 和对应的解码矩阵 D, 同时隐式地对标签之间的关系进行探索. 为了提高对 Y 重构的能力, Y 与 C、D 之间的差异被期望最小化, 式 (3) 中的 $\Phi_2(Y,C,D)$ 是对 Y 与 C、D 之间差异的描述, 具体形式如下表示:

$$\Phi_2(Y,C,D)=\sum_{i,j}\sum_{y_{ij}\in Pos(\boldsymbol{x}_i)}\left(y_{ij}-\boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{d}_j\right)^2 \tag{5}$$

Y 指标签空间, y_{ij} 是指 Y 中第 i 行第 j 列的元素值, $C = [\boldsymbol{c}_1, \boldsymbol{c}_2, \cdots, \boldsymbol{c}_N]^{\mathrm{T}} \in \mathbf{R}^{N\times s}$ 是对标签嵌入学习时利用矩阵分解得到的 Y 对应的潜在空间表示, $\boldsymbol{c}_i$ 是指潜在空间表示 C 的第 i 列, $D = [\boldsymbol{d}_1, \boldsymbol{d}_2, \cdots, \boldsymbol{d}_K]^{\mathrm{T}} \in \mathbf{R}^{K\times s}$ 则是矩阵分解得到对 C 的解码矩阵, $\boldsymbol{d}_j$ 是指潜在空间表示 D 的第 j 列, $Pos(\boldsymbol{x}_i)$ 是指 $\boldsymbol{x}_i$ 含有的标签集合. 为了简便, 除非特别说明, 在接下来的论述中的形式均为 $C = [\boldsymbol{c}_1, \boldsymbol{c}_2, \cdots, \boldsymbol{c}_N]^{\mathrm{T}} \in \mathbf{R}^{N\times s}$ 和 $D = [\boldsymbol{d}_1, \boldsymbol{d}_2, \cdots, \boldsymbol{d}_K]^{\mathrm{T}} \in \mathbf{R}^{K\times s}$.

2.2.3 特征与标签联合嵌入学习

为了提高对低维嵌入 C 的可预测性, C 的学习过程中应与实例的特征有着更强的相关性[37]. 本文使用 CCA 技术将 X 与 Y 紧密耦合在一起, 并使二者在低维空间具有最大相关性, 以此得到一个共享潜在子空间 C 提高模型的性能. 作为相关跨域数据 (例如, 输入特征数据 X 及其标签数据 Y) 的标准统计技术方法, CCA 在最大化两个域的投影空间的相关性时找到对应的投影 W_1 及 W_2, 即最大化 $corr(W_1^{\mathrm{T}}, W_2^{\mathrm{T}})$. 当使用 DNN (Deep neural network) 取代 CCA 中对应的两个线性投影函数时, 就得到了 DCCA 方法. 该方法能够以梯度下降法学习和更新与 DNN 模型中具有类似目标函数的参数.

在本文中, $\Phi_3(F_e, C)$ 用于衡量特征和标签在低维潜在表示中的差异性, 是标签与特征之间联系的纽带, 对 $\Phi_3(F_e, C)$ 使用 CCA 技术并加入恒等约束[37], $\Phi_3(F_e, C)$ 有着如下的形式:

$$\begin{aligned} &\Phi_3(F_e, C) = \|F_e(X) - C\|_F^2 \\ &\text{s.t.} \quad {F_e}^{\mathrm{T}}(X)F_e(X) = C^{\mathrm{T}}C = I_s \end{aligned} \tag{6}$$

F_e 指 SDAE 中的编码层, $F_e(X) \in \mathbf{R}^{N\times s}$ 是指 X 经过 F_e 得到潜在空间表示, $I_s \in \mathbf{R}^{s\times s}$ 是一个单位矩阵, s 则是潜在空间的维度大小, $C \in \mathbf{R}^{N\times s}$ 是指对标签嵌入时得到的潜在空间表示. 由文献 [38] 可知, 式 (6) 不仅具有的功能等同于标准 CCA 方法的最大化相关性功能, 而且能够使用梯度下降方法有效地对参数更新.

2.3 Deep AE-MF+neg

多标签数据集中有相当一部分比例的样本含有的标签数量少于 2, 因此, 在对 Deep AE-MF 模型进行训练学习时, 由于缺少丰富的标签共现信息 (即标签间的正相关信息) 不能对标签间的正相关信息进行有效探索与利用, 限制了模型的性能; 然而, 这些所含标签数量少于 2 的样本, 却拥有着丰富非共现信息 (即标签间的负相关信息). 为了能够有效地利用标签间的这种负相关信息, 本文在 Deep AE-MF 模型中引入标签负采样策略, 为每个实例采样其对应的负相关标签并用于模型训练学习; 关于采样的具体方案见算法 1. 结合式 (3)～式 (6) Deep AE-MF+neg 的目标函数可表示为:

$$\begin{aligned} \Omega = \min_{\Theta} &\lambda \|F_d(F_e(X)) - X\|_F^2 + \\ &\beta \left\|M^{\mathrm{neg}}(Y - CD^{\mathrm{T}})\right\|_F^2 + \\ &\alpha \|F_e(X) - C\|_F^2 + \gamma\Phi_4(\Theta) \\ \text{s.t.} \quad &{F_e}^{\mathrm{T}}(X)F_e(X) = C^{\mathrm{T}}C = I_s \end{aligned} \tag{7}$$

F_e 和 F_d 分别指 SDAE 中的编码层和解码层, $F_e(X) \in \mathbf{R}^{N\times s}$ 指 X 经过 F_e 得到的低维表示, $C \in \mathbf{R}^{N\times s}$、$D \in \mathbf{R}^{K\times s}$ 分别指对标签嵌入时学习到的潜在空间表示与解码矩阵, 矩阵 $M^{\mathrm{neg}} = [\boldsymbol{m}_1^{\mathrm{neg}}, \boldsymbol{m}_2^{\mathrm{neg}}, \cdots, \boldsymbol{m}_N^{\mathrm{neg}}] \in \mathbf{R}^{N\times K}$ 是由算法 1 生成. 若第 i 个实例含有标签 j 或通过采样到标签 j, 则 $M^{\mathrm{neg}}[i, j] = 1$; 否则, $M^{\mathrm{neg}}[i, j] = 0$, $I_s \in \mathbf{R}^{s\times s}$ 是一个单位矩阵, s 则是低维空间的维度大小, $\Phi_4(\Theta)$ 表示对参数的正则化, 其详细描述见式 (8).

$$\Phi_4(\Theta) = \sum_l (\|W_l\|_2^2 + \|b_l\|_2^2) + \sum_{j=1}^{K} \|\boldsymbol{d}_j\|_2^2 \tag{8}$$

这里的 W_l 和 b_l 分别指 SDAE 中每层的权值矩阵和偏置, $1 \le l \le 5$, $\boldsymbol{d}_j$ 为 D 中的第 j 列. 同理, Deep AE-MF 的目标函数可表示成如下形式:

$$\begin{aligned} \Omega = \min_{\Theta} &\lambda \|F_d(F_e(X)) - X\|_F^2 + \\ &\beta \left\|M(Y - CD^{\mathrm{T}})\right\|_F^2 + \\ &\alpha \|F_e(X) - C\|_F^2 + \gamma\Phi_4(\Theta) \end{aligned} \tag{9}$$

这里的 M 与式 (7) 中 M^{neg} 有所不同, Deep AE-MF 算法在学习时未进行负采样, 矩阵 M 中元素只与数据集的训练实例含有的标签有关, 即当第 i 个实例含有标签 j 时, 有 $M[i, j] = 1$; 否则, $M[i, j] = 0$.

2.4 模型优化

为了能够得到 Deep AE-MF 和 Deep AE-MF+neg 模型, 需要对式 (9) 与式 (7) 进行优化学习. 由式 (3) 可知模型训练时的总损失表示中, $\Phi_1(X)$、$\Phi_2(Y, C, D)$、$\Phi_3(F_e, C)$、$\Phi_4(\Theta)$ 分别是指特征低维嵌入损失, 标签低维嵌入损失, 子空间的学习损失, 参数的正则化项.

以 Deep AE-MF 为例进行优化, Deep AE-MF 模型包括 SDAE 和矩阵分解两个部分, 对于 SDAE 部分的参数优化, 与现有的 DNN 模型优化方法一致使用梯度下降法; 而对于矩阵分解部分的参数优化则采用坐标上升法. 从图 2 和式 (9) 可以看出, $\Phi_1(X)$ 和 $\Phi_3(F_e, C)$ 的梯度用于 SDAE 部分参数优

化, $\Phi_2(Y,C,D)$ 和 $\Phi_3(F_e,C)$ 的梯度则是用于矩阵分解部分的优化, $\Phi_4(\Theta)$ 对应的是正则化项, 在两部分参数更新时会有选择的使用到其对应的梯度.

对于矩阵分解部分的参数 $\boldsymbol{c}_i$ 和 $\boldsymbol{d}_j$ 进行优化, 首先要给定 SDAE 中的参数值, 然后根据式 (9) 分别计算出参数 $\boldsymbol{c}_i$ 和 $\boldsymbol{d}_j$ 的梯度值, $\boldsymbol{c}_i$ 和 $\boldsymbol{d}_j$ 的更新如下所示:

$$\begin{aligned}\boldsymbol{c}_i =& \boldsymbol{c}_i - \eta(D^{\mathrm{T}}M_iD + \alpha I_s)^{-1}\cdot \\ & (D^{\mathrm{T}}M_iY_i + \alpha F_e(\boldsymbol{x}_i))\end{aligned} \tag{10}$$

$$\boldsymbol{d}_j = \boldsymbol{d}_j - \eta(C^{\mathrm{T}}M_jC + \gamma I_s)^{-1}C^{\mathrm{T}}M_jY_j \tag{11}$$

其中, $M_i = \mathrm{diag}\{m_{i1}, m_{i2}, \cdots, m_{iK}\} \in \mathbf{R}^{K\times K}$ 与 $M_j = \mathrm{diag}\{m_{1j}, m_{2j}, \cdots, m_{Nj}\} \in \mathbf{R}^{N\times N}$ 分别是由矩阵 M 中的第 i 行、第 j 列生成的对角矩阵, 矩阵 $M = [\boldsymbol{m}_1, \boldsymbol{m}_2, \cdots, \boldsymbol{m}_N]^{\mathrm{T}} \in \mathbf{R}^{N\times K}$, $C = [\boldsymbol{c}_1, \boldsymbol{c}_2, \cdots, \boldsymbol{c}_N]^{\mathrm{T}} \in \mathbf{R}^{N\times s}$, $D = [\boldsymbol{d}_1, \boldsymbol{d}_2, \cdots, \boldsymbol{d}_K]^{\mathrm{T}} \in \mathbf{R}^{K\times s}$, $Y_i = (Y_{i1}, Y_{i2}, \cdots, Y_{iK})^{\mathrm{T}} \in \mathbf{R}^{K\times 1}$ 是由标签矩阵 Y 的第 i 行组成.

关于 SDAE 中参数 W_l 和 b_l 更新, 首先固定 C 和 D 的当前值, 接着使用反向传播学习算法对 SDAE 中的每层参数进行更新, 每层参数更新如下所示:

$$\begin{aligned}W_l =& W_l - \eta\Bigg(\alpha\sum_{i=1}^{N}\frac{\partial F_e(\boldsymbol{x}_i)}{\partial W_l}(F_e(\boldsymbol{x}_i) - \boldsymbol{c}_i) + \\ & \lambda\sum_{i=1}^{N}\frac{\partial F_d(\boldsymbol{x}_i)}{\partial W_l}(F_d(\boldsymbol{x}_i) - \boldsymbol{x}_i) + \gamma W_l\Bigg)\end{aligned} \tag{12}$$

$$\begin{aligned}b_l =& b_l - \eta\Bigg(\alpha\sum_{i=1}^{N}\frac{\partial F_e(\boldsymbol{x}_i)}{\partial b_l}(F_e(\boldsymbol{x}_i) - \boldsymbol{c}_i) + \\ & \lambda\sum_{i=1}^{N}\frac{\partial F_d(\boldsymbol{x}_i)}{\partial b_l}(F_d(\boldsymbol{x}_i) - \boldsymbol{x}_i) + \gamma b_l\Bigg)\end{aligned} \tag{13}$$

其中, F_e 是指 SDAE 中的编码层, F_d 则是指 SDAE 的解码层, η 是参数更新时的学习速率.

利用上述的方式对相应的参数进行更新优化, 可以学习到 Deep AE-MF 模型, 对于 Deep AE-MF+neg 进行优化学习时, 只需将式 (10) 与 (11) 中对应的 M_i 和 M_j 替换为 M_i^{neg} 和 M_j^{neg} 即可. Deep AE-MF+neg 的学习过程伪代码见算法 1 与算法 2. 算法 1 描述的是对标签进行负采样生成采样矩阵的具体过程, 它利用由式 (2) 得到的负相关性矩阵 Neg 对实例含有的标签采样对应负标签, 采样个数随机生成. 算法 2 描述的是 Deep AE-MF+neg 的学习过程, 它的输入是特征空间 X 与标签空间 Y 及相关的超参数值. 首先, 在训练之前初始化模型所需的权值矩阵 (步骤 1); 接着, 由算法 1 生成标签采样矩阵 M^{neg} (步骤 2); 然后, 由输入参数与前两步的生成结果组成所需目标函数 (即式 (7)), 并按照式 (10) ∼ 式 (13) 对目标函数式 (7) 中的参数进行迭代更新, 直至目标函数值不再变化或变化小于一定阈值 (收敛) 或达到最大迭代次数 (步骤 3). 当模型学习完成后, 对于任意一个的测试实例 $\boldsymbol{x}_{\mathrm{test}}$, 可由 $\boldsymbol{y}_{\mathrm{test}} = DF_e(\boldsymbol{x}_{\mathrm{test}})$ 的方式得到对应标签预测值.

算法 1. 标签的负采样过程

输入. 标签矩阵 Y, 标签数量 K, 样本实例数量 N.

输出. 标签负相关性矩阵 M^{neg}.

步骤 1. 由式 (2) 和 Y 计算得到矩阵 $Neg \in \mathbf{R}^{K\times K}$.

步骤 2. 初始化一个零矩阵 $M^{\mathrm{neg}} \in \mathbf{R}^{N\times K}$.

步骤 3. 利用 Y 与 Neg 进行采样, 得到 M^{neg}

for $n = 0$ to N do
  $temp\_Neg = Neg$
  for $k = 0$ to K do
    if $Y[n,k] == 1$ then
      $M^{\mathrm{neg}}[n,k] = 1$
      随机生成采样个数 S //$S \in [1, \lceil\frac{K}{5}\rceil]$
      for $s = 1$ to S do
        $j = \max\limits_{j}\ temp\_Neg[k,j]$
        $temp\_Neg[k,j] = 0$
        $M^{\mathrm{neg}}[n,j] = 1$
      end for
    end if
  end for end for

算法 2. Deep AE-MF+neg 学习过程

输入. 特征矩阵 X, 标签矩阵 Y, 超参数 λ、α、β、γ 及潜在空间大小 s.

输出. F_e, F_d, D.

步骤 1. 随机初始化 F_e, F_d, C, D, 高斯分布产生一个噪声矩阵 ε.

步骤 2. 由算法 1 得到矩阵 M^{neg}.

步骤 3. 重复步骤 3 直至目标函数收敛 (即函数值不再变化或变化小于一定的阈值) 或达到最大迭代次数.

步骤 3.1. 按照式 (7) 计算出总损失,

$$\begin{aligned}\Omega = \min_{\Theta}\lambda\left\|F_d(F_e(X)) - X\right\|_F^2 + \\ \beta\left\|M^{\mathrm{neg}}(Y - CD^{\mathrm{T}})\right\|_F^2 +\end{aligned}$$

$$\alpha \left\| F_e(X) - C \right\|_F^2 + \gamma \Phi_4(\Theta)$$

步骤 3.2. 按照式 (10) 与式 (11) 更新矩阵分解中的参数 C 与 D

$$\boldsymbol{c}_i = \boldsymbol{c}_i - \eta(D^{\mathrm{T}} M_i D + \alpha I_s)^{-1}(D^{\mathrm{T}} M_i Y_i + \alpha F_e(\boldsymbol{x}_i))$$
$$\boldsymbol{d}_j = \boldsymbol{d}_j - \eta(C^{\mathrm{T}} M_j C + \gamma I_s)^{-1} C^{\mathrm{T}} M_j Y_j$$

步骤 3.3. 按照式 (12) 与式 (13) 更新 F_e, F_d

$$W_l = W_l - \eta\left(\alpha \sum_{i=1}^{N} \frac{\partial F_e(\boldsymbol{x}_i)}{\partial W_l}(F_e(\boldsymbol{x}_i) - \boldsymbol{c}_i) + \lambda \sum_{i=1}^{N} \frac{\partial F_d(\boldsymbol{x}_i)}{\partial W_l}(F_d(\boldsymbol{x}_i) - \boldsymbol{x}_i) + \gamma W_l\right)$$

$$b_l = b_l - \eta\left(\alpha \sum_{i=1}^{N} \frac{\partial F_e(\boldsymbol{x}_i)}{\partial b_l}(F_e(\boldsymbol{x}_i) - \boldsymbol{c}_i) + \lambda \sum_{i=1}^{N} \frac{\partial F_d(\boldsymbol{x}_i)}{\partial b_l}(F_d(\boldsymbol{x}_i) - \boldsymbol{x}_i) + \gamma b_l\right)$$

3 实验

3.1 数据集描述与预处理

为了验证本文提出的 Deep AE-MF 和 Deep AE-MF+neg 方法的性能, 选取了 6 个多标签数据集进行实验测试, 分别为 enron、ohsumed[1]、movieLens[2]、Delicious、EURLex-4K[3] 和 TJ[4], 其中前 5 个是英文类型的多标签数据集, 最后一个则是中文类型数据集. 由于 enron、ohsumed、movieLens 和 TJ 这 4 个数据集是原始字符数据, 为了能够用于实验, 需要进一步对这些数据进行处理, 对于英文类型的数据集进行处理时, 删除数据集中的停用词、词频出现少于 20 词的单词及一些非字符符号等, 每个实例的特征向量表示在这里使用 8 000 维的词袋进行表示; 而对于中文数据集的处理, 步骤与处理英文大体相同, 但由于中文字词之间不像英文有空格作为分割, 在预处理之前, 我们首先要进行分词, 分词采用通用的中文分词工具 ANSJ [5]. 数据集更详细的描述见表 1 和表 2, 由于 EURLex-4K 和 Delicious 数据集是非原始字符数据, 故在表 2 中无须再介绍两者的有关字符信息.

其中, 平均标记数 $= \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{K}[y_{ij} = +1]$, 标记密度 $= \frac{1}{N \times K}\sum_{i=1}^{N}\sum_{j=1}^{K}[y_{ij} = +1]$, $[\cdot]$ 是指示函数, 当条件为真时, 值为 1; 否则, 值为 0.

[1] http://meka.sourceforge.net/
[2] https://grouplens.org/datasets/movielens/
[3] http://manikvarma.org/downloads/XC/XMLRepository.html
[4] http://tjzhifei.github.io/resource.html
[5] https://github.com/NLPchina/ansjseg

表 1 多标签数据集相关统计

Table 1 Multi-label datasets and associate statistics

数据集	标签数	实例数	特征数	标记密度	平均标记数
enron	53	1 702	8 000	0.0637	3.378
ohsumed	23	13 928	8 000	0.0720	1.663
movieLens	20	10 076	8 000	0.1020	2.043
TJ	9	5 892	8 000	0.2001	1.801
Delicious	983	16 105	500	0.0193	19.03
EURLex-4K	3 993	19 438	5 000	0.0013	5.31

3.2 评价准则

多标记学习框架中, 每个实例可能同时拥有多个类别标签, 因此, 与单标签学习系统相比, 多标签学习系统的评价准则相对会更加复杂些. 到目前为止, 已有多种评价准则被提出并广泛地用于评价多标签学习系统的性能. 现选取以下 5 种评价准则, 即 hamming loss[39]、基于标签的 Macro-F1-label (或称 Macro_F1) 与 Micro-F1-label (或称 Micro_F1)[40]、基于样本实例的 F1 值[41] 及 Precision at top K (P@K)[42], 用于评价多标签学习系统的性能. 在这 5 个评价准则中, 后 4 个的值越大表示模型的性能越好, 最优结果值均为 1; 而第 1 个则是值越小表示模型的性能越好, 最优结果值为 0.

3.3 参数设置

为了验证本文提出的方法 Deep AE-MF 与 Deep AE-MF+neg 的有效性, 将 Deep AE-MF 和 Deep AE-MF+neg 算法与 10 个多标签学习算法, 即 BR[5]、LS_ML[20]、CCA-SVM[20]、CCA-ridge[20]、PLST[25]、CPLST[26]、FaIE[27]、LEML[28]、PD-sparse[43]、和 ML_CSSP[44] 进行实验比较. 对比算法可分为三类: BR 代表的是经典的问题转化算法; LS_ML、CCA-SVM、CCA-ridge 代表的是基于特征嵌入/约减 (FSDR) 型的算法: ML_CSSP、PLST、CPLST、FaIE、LEML、PD-sparse 则是基于标签嵌入/约减 (LSDR) 型的算法, 其中 LEML、PD-sparse 主要是针对极限多标签分类的算法. 对比算法代码全部都是基于 MatLab 实现的, 其中 BR 算法选择 SVM 作为其基分类器. 对 LS_ML、CCA-SVM 和 CCA-ridge 三个方法的参数设置均按照文献 [27] 最好的结果设置. 对于 ML_CSSP、PLST、CPLST、FaIE、LEML、PD-sparse 算法, 超参数按照对应文献中的默认值进行设置. 对于本文提出的 Deep AE-MF 方法, SDAE 使用 5 层的网络结构, 其中学习率选取大小范围是 $\{0.0001, 0.001, 0.01, 0.1\}$, 对于平衡损失的超参数 λ、α、β 及 γ 设置范围则均为 $\{0.001, 0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100, 500,$

1 000}, 潜在空间维度大小选取范围则是 $\{0.1K, 0.2K, \cdots, K\}$, K 表示对应数据集具有的标签个数. 实验结果表明: 当设置损失平衡超参数 $\lambda = 100$, $\beta = 1$, $\alpha = 50$, $\gamma = 0.1$, $s = 0.6K$, 学习率 $\eta = 0.001$ 时, Deep AE-MF 方法具有较好的稳定的性能. 为了能够充分利用标签间的关系, Deep AE-MF+neg 模型考虑了标签间的负相关性 (非共现) 信息, 实验结果显示利用加入的标签间的负相关性 (非共现) 信息能够提升模型性能.

3.4 实验结果与分析

将 Deep AE-MF 和 Deep AE-MF+neg 与其他 10 种常见的多标签算法: BR、CCA-SVM、CCA-ridge、LS_ML、PLST、CPLST、ML_CSSP、FaIE、LEML 和 PD-sparse 进行实验比较. 根据 5 种评价方式, 表 3～表 7 分别列出了本文提出的方法与其他 10 种对比算法在表 1 中数据集上的详细的实验结果, 且对最好的结果进行加粗表示 ('–' 表示缺少实验结果数据).

表 3 显示算法 Deep AE-MF 和 Deep AE-MF+neg 在 6 个数据集中均有 4 个数据集相比于对比算法有着更小的 hamming loss 值, 即有着最好的性能, 而且在这 4 个数据集中有 3 个相对于次优结果的算法分别有着 3 % ～ 10 % 左右的性能提高. 但二者在 ohsumed 和 Delicious 上排在了较差的位置, 与最优结果相比有 1.5 % 左右的差距, 从表 1 的分析可以看出, ohsumed 数据集标签的平均密度相比于其他数据集过小, Delicious 数据集的特征维度偏小. 在 movieLens 数据中, Deep AE-MF+neg 的性能略低于 Deep AE-MF 性能, 原因是 Deep AE-MF+neg 没有像 Deep AE-MF 在预测时偏好于将大部分的标签预测为 −1, 在数据集中标签为 −1 相对 1 所占的比例是非常大的, 故而将标签预测为 −1, 可有效地减少预测错误率 (即得到 hamming loss 值更小). 从 6 个数据集中的综合结果来看, Deep AE-MF 和 Deep AE-MF+neg 是优于其对比算法的.

从表 4 中的结果可以看出: Deep AE-MF 和 Deep AE-MF+neg 这两种方法在 6 个数据集上均取得了最好的结果, 优于所有的对比算法, 表明 SDAE 学习得到的非线性表示有利于分类模型性能的提高. 其中 Deep AE-MF+neg 方法好于 Deep AE-MF 方法, 说明通过利用标签的负相关性 (非共现) 信息可进一步提高模型的性能. 从表 4 中可看出 BR 方法的性能较差, 而基于嵌入方法的性能大都排在中间位置.

从表 5 中的显示的结果看: Deep AE-MF 和 Deep AE-MF+neg 方法在 movieLens、TJ、enron、Delicious 及 EURLex-4K 这 5 个数据集上取得了最好的性能, 且在 Delicious 和

表 2 多标签数据集字符数量统计

Table 2 The number of characters in a multi-label dataset

数据集	含有不同字符数的样本比例					
	50 以内	50～100	100～200	200～400	400～800	800 以上
enron	0.437133	0.287309	0.165100	0.052291	0.014101	0.0440658
ohsumed	0.591008	0.325526	0.082473	0.000992	0	0
movieLens	0.427197	0.558372	0.014431	0	0	0
TJ	0.134589	0.354888	0.339613	0.159708	0.011202	0

表 3 基于hamming loss的性能比较

Table 3 The hamming loss of ten multi-label algorithms with respect to different data sets

算法/数据集	enron	ohsumed	movieLens	TJ	Delicious	EURLex-4K
BR	0.0771	0.1484	0.1992	0.2923	0.0185	0.0032
CCA-SVM	0.1593	0.2148	0.3116	0.3764	–	–
CCA-Ridge	0.1549	0.2140	0.3045	0.3268	–	–
LS_ML	0.1000	0.2119	0.2474	0.2842	–	–
PLST	0.0843	0.1510	0.2186	0.2906	0.0183	0.0037
CPLST	0.0841	0.1512	0.2186	0.2906	0.0182	0.0038
FaIE	0.0841	0.1505	0.2188	0.2882	0.0183	0.0038
ML_CSSP	0.0836	**0.1479**	0.2075	0.2804	**0.0181**	0.0036
Deep AE-MF	0.0518	0.1693	**0.1416**	0.1891	0.0310	0.0013
Deep AE-MF+neg	**0.0509**	0.1630	0.1445	**0.1869**	0.0279	**0.0012**

表 4 基于 Micro-F1-label 的性能比较

Table 4 The Micro-F1-label of ten multi-label algorithms with respect to different data sets

算法/数据集	enron	ohsumed	movieLens	TJ	Delicious	EURLex-4K
BR	0.3451	0.1137	0.3308	0.4281	0.1370	0.1294
CCA-SVM	0.2622	0.1528	0.3058	0.4355	–	–
CCA-Ridge	0.2744	0.1509	0.3074	0.4344	–	–
LS_ML	0.3417	0.1531	0.3633	0.4931	–	–
PLST	0.3638	0.1589	0.3639	0.4781	0.1911	0.1540
CPLST	0.3643	0.1577	0.3642	0.4787	0.1911	0.1534
FaIE	0.3643	0.1593	0.3607	0.4839	0.1911	0.1539
ML_CSSP	0.3606	0.1543	0.3532	0.4850	0.1860	0.1534
Deep AE-MF	0.5475	0.1642	0.3968	0.5421	0.2757	0.4913
Deep AE-MF+neg	**0.5531**	**0.1962**	**0.4122**	**0.5632**	**0.2775**	**0.4936**

表 5 基于 Macro-F1-label 的性能比较

Table 5 The Macro-F1-label of ten multi-label algorithms with respect to different data sets

算法/数据集	enron	ohsumed	movieLens	TJ	Delicious	EURLex-4K
BR	0.0923	0.0656	0.2066	0.4146	0.0338	0.0371
CCA-SVM	0.1045	**0.1150**	0.2572	0.4282	–	–
CCA-Ridge	0.1019	0.1134	0.2556	0.4488	–	–
LS_ML	0.1158	0.1141	0.2971	0.4832	–	–
PLST	0.1149	0.0884	0.2742	0.4717	0.0460	0.0507
CPLST	0.1149	0.0863	0.2744	0.4725	0.0462	0.0514
FaIE	0.1147	0.0863	0.2609	0.4647	0.0461	0.0506
ML_CSSP	0.1147	0.0793	0.2375	0.4580	0.0437	0.0492
Deep AE-MF	0.1356	0.0960	0.3394	0.5440	0.1316	0.1477
Deep AE-MF+neg	**0.1384**	0.1011	**0.3455**	**0.5629**	**0.1324**	**0.1483**

表 6 基于 F1 的性能比较

Table 6 The F1 of ten multi-label algorithms with respect to different data sets

算法/数据集	enron	ohsumed	movieLens	TJ	Delicious	EURLex-4K
BR	0.2885	0.1046	0.2705	0.4482	0.1280	0.2061
CCA-SVM	0.2758	0.1354	0.2982	0.4191	–	–
CCA-Ridge	0.2937	0.1344	0.2983	0.4360	–	–
LS_ML	0.3510	0.1352	**0.3523**	0.4821	–	–
PLST	0.4029	0.1343	0.3158	0.4753	0.1650	0.2502
CPLST	0.4036	0.1330	0.3164	0.4758	0.1651	0.2503
FaIE	0.4000	0.1327	0.3171	0.4738	0.1650	0.2502
ML_CSSP	0.3814	0.1318	0.2854	0.4799	0.1632	0.2419
Deep AE-MF	0.4491	0.1489	0.3307	0.4677	0.2138	0.4291
Deep AE-MF+neg	**0.4582**	**0.1491**	0.3381	**0.5013**	**0.2310**	**0.4365**

EURLex-4K 上与第 3 名结果有接近 10 % 左右的性能提高; 在 ohsumed 中, 基于特征嵌入的几种方法取得了比较好的结果, 比 Deep AE-MF 方法提高了 1.5 % 左右, 但是在其他数据集上的性能与 Deep AE-MF 和 Deep AE-MF+neg 方法相比要差很多. 所以, 在 6 个数据上进行综合性能的比较, Deep AE-MF 和 Deep AE-MF+neg 方法排在前两位, 采用了基于嵌入方法的算法排在中间位置, BR 最差.

从表 6 中的显示的结果看: Deep AE-MF 和 Deep AE-MF+neg 在 enron、ohsumed、Delicious

表 7 基于 P@K 的性能比较

Table 7 The P@K of six multi-label algorithms with respect to different data sets

数据集	EURLex-4K				Delicious			
度量准则/算法	LEML	PD-sparse	Deep AE-MF	Deep AE-MF+neg	LEML	PD-sparse	Deep AE-MF	Deep AE-MF+neg
P@1	0.6340	0.7643	0.8078	**0.8104**	0.6567	0.5182	0.6633	**0.6754**
P@3	0.5035	0.6037	0.6821	**0.6893**	0.6055	0.4418	0.6095	**0.6123**
P@5	0.4128	0.4972	0.5764	**0.5805**	0.5608	0.5656	0.5764	**0.5834**

及 EURLex-4K 上取得了最好的结果, 但在 movieLens 数据集中 Deep AE-MF 和 Deep AE-MF+neg 则排在次优位. 原因是使用了线性与非线性转化的 LS_ML 方法能够对每个实例含有的标签进行较好的预测, 但与 LS_ML 相比本文的方法也只相差 2% 和 1.5%. 对于数据集 TJ, Deep AE-MF+neg 排在了第一位置, Deep AE-MF 则排在了中间偏低的位置, 原因是这些 LSDR 型算法在标签维度约减的过程都直接或间接的利用了标签关系信息, 与 Deep AE-MF 相比能够找到一个更加有效的潜在低维标签空间. 从综合性能上, Deep AE-MF 和 Deep AE-MF+neg 仍然领先于对比算法, 尤其是 BR 方法和 FSDR 型的方法.

表 7 中的数据是有关 Deep AE-MF 和 Deep AE-MF+neg 与极限多标签分类算法在标签数量较大的数据集中实验结果, 在性能比较时, 采用极限多标签分类常用的度量准则 P@K (Precision at top K). 实验结果显示当取不同的 K 值时, Deep AE-MF 和 Deep AE-MF+neg 均取得了最优的结果, 表明了本文提出的算法能够较好解决标签维度过高的问题, 且有着不错的性能.

表 3 ~ 表 7 中在 5 种评价标准上的实验结果显示, 提出的 Deep AE-MF 和 Deep AE-MF+neg 的方法明显优于其对比算法. 在联合嵌入学习过程中, SDAE 得到的非线性表示 $F_e(X)$, 矩阵分解直接得到的低维标签表示 C 和解码矩阵 D, 有利于学习找到一个泛化能力更好的分类模型. 从表中可以看出 Deep AE-MF+neg 的性能几乎一直优于 Deep AE-MF, 表明在对标签嵌入时利用标签之间的非共现信息可以进一步提高算法的性能.

为了在统计上比较提出的算法与其对比算法在 6 个数据集的实验结果, 采用显著性水平为 5% 的 Student′s t test[45]. 在 Deep AE-MF 与除 Deep AE-MF+neg 外的算法对比检验时, 以 Deep AE-MF 的性能差于或等于其对比算法的性能作为零假设, 以 Deep AE-MF 的性能好于其对比算法性能作为备选假设. 从表 8 中可以看出 Deep AE-MF 与每个对比算法在 6 个数据集上的 P 值, 在 hamming loss 上只有一个是大于 0.05 (即支持原假设); 在 Micro-F1-label 上所有 P 值均小于 0.05, 即均支持 Deep AE-MF 的性能是好于其对比算法; 在 Macro-F1-label 上, 仅有两个 P 值是大于 0.05; 综合分析说明 Deep AE-MF 的性能优于其他算法. 对于 Deep AE-MF 与 Deep AE-MF+neg 性能检验时, 以二者性能相当作为零假设, 从表 8 中可以看出在 3 种评价准则与 6 个数据集上, 18 个 P 值中只有 2 个是大于 0.05 (支持原假设), 因此可认为二者的性能是有显著差异的. 上述 t test 的结果与分析验证本文提出算法的有效性.

3.5 参数分析

3.5.1 超参 α 的敏感性分析

为了验证超参数 α 对 Deep AE-MF 性能的影响, 在 {1, 5, 10, $\cdots$, 1 000} 中选择不同值进行实验. 本文在两个数据集上使用三种评价方式来研究参数 α 对实验性能的影响, 结果如图 3 所示.

从图 3 可以看出, 对于 hamming loss 在 enron 和 TJ 两个数据集上, 随着 α 的增加, 曲线先下降再升高 (即性能先上升再下降); 对于基于标签的 Macro_F1 和 Micro_F1 在 enron 和 TJ 两个数据集上, 随着 α 的增加, 曲线先上升再下降 (即性能先上升再下降).

由图 3 中可以得出 $\alpha = 50$ 附近时, 在 enron 和 TJ 上模型均有着最佳的性能. 通过分析可以认为, 当 $\alpha < 50$ 时, 特征和标签联合嵌入所占比重过小, 使得在对标签探索嵌入时, 过于注重对标签空间 Y 的重构, 在学习标签潜在表示空间 C 时未能充分利用特征信息; 当 $\alpha > 50$ 时, 特征和标签联合嵌入时所占比重过大, 表明标签嵌入时在学习标签潜在表示空间 C 时偏好于使用特征信息, 使模型降低对标签空间 Y 的学习, 导致对 Y 的重构或预测能力下降. 综合对实验结果权衡分析, 选取 $\alpha = 50$ 作为最终取值.

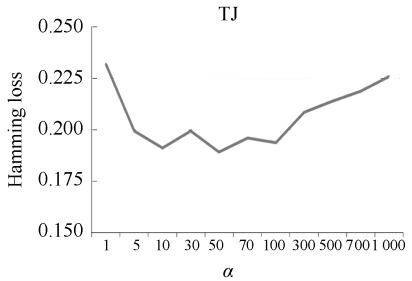
3.5.2 参数 s 的敏感性分析

为找出能够使 Deep AE-MF 性能最佳时的潜在空间维度 s 值, 在 $\{0.1K, 0.2K, \cdots, K\}$ 中选择不同 s 值进行实验, 其中 K 表示数据集标签的个数. 本文在两个数据集上使用三种评价方式来研究参数 s 对实验性能的影响, 结果如图 4 所示.

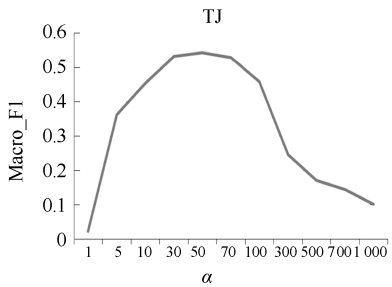
表 8 Student′s t test 结果 P 值 (加粗表示 P 值大于 0.05)

Table 8 P value of Student′s t test results (Bold indicates that P value is greater than 0.05)

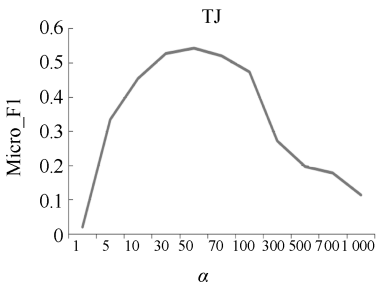
		enron	ohsumed	movieLens	TJ	Delicious	EURLex-4K
	hamming loss						
	BR	1.87E−5	1.02E−3	7.03E−6	2.94E−7	1.32E−5	9.64E−3
	LS_ML	2.93E−5	1.27E−4	5.92E−7	3.28E−7	–	–
	CCA-SVM	3.38E−8	2.04E−6	4.47E−7	4.55E−10	–	–
	CCA-Ridge	5.34E−9	6.01E−6	2.33E−7	3.97E−7	–	–
Deep AE−MF	PLST	2.41E−8	2.91E−3	8.36E−12	3.04E−9	8.04E−6	4.67E−4
	CPLST	2.43E−8	3.04E−3	2.05E−5	1.32E−9	5.01E−6	9.75E−4
	FaIE	3.62E−9	5.83E−4	1.25E−11	3.09E−9	1.61E−5	5.38E−4
	ML_CSSP	9.35E−8	**8.36E−2**	8.18E−7	7.93E−10	3.08E−6	4.29E−3
	Deep AE−MF+neg	1.90E−5	7.39E−4	3.89E−7	2.73E−4	3.21E−3	**1.09E−1**
	Macro-F1-label						
	BR	4.85E−10	3.01E−6	1.73E−7	3.61E−7	2.63E−9	3.12E−9
	LS_ML	4.03E−10	**1.25E−1**	3.26E−7	4.11E−8	–	–
	CCA-SVM	3.19E−8	**5.48E−2**	3.21E−7	3.37E−9	–	–
	CCA-Ridge	6.06E−11	4.84E−4	1.51E−5	3.01E−6	–	–
Deep AE−MF	PLST	1.51E−9	2.23E−3	1.93E−5	6.64E−7	4.38E−8	4.13E−12
	CPLST	1.42E−9	5.19E−3	5.21E−5	1.03E−6	8.21E−9	1.62E−11
	FaIE	1.72E−10	3.99E−2	1.83E−5	5.11E−7	2.26E−7	1.45E−10
	ML_CSSP	1.64E−10	4.12E−4	4.03E−6	3.03E−7	6.63E−9	8.11E−11
	Deep AE−MF+neg	1.61E−5	5.51E−7	**8.11E−2**	3.09E−7	1.18E−3	2.34E−4
	Micro-F1-label						
	BR	1.62E−8	2.82E−5	2.34E−8	5.07E−11	1.35E−8	9.95E−9
	LS_ML	3.90E−7	1.54E−4	2.75E−7	1.31E−10	–	–
	CCA-SVM	2.74E−7	5.75E−4	4.25E−9	6.72E−9	–	–
	CCA-Ridge	2.70E−7	1.84E−4	4.85E−8	1.06E−10	–	–
Deep AE−MF	PLST	5.01E−6	8.47E−3	9.98E−9	2.71E−10	5.21E−8	1.02E−9
	CPLST	7.08E−6	6.36E−3	4.18E−9	4.14E−11	5.08E−8	1.73E−12
	FaIE	1.40E−5	5.86E−3	1.61E−9	1.08E−10	5.35E−9	4.44E−10
	ML_CSSP	6.03E−5	3.01E−4	2.84E−9	6.08E−12	5.86E−7	2.21E−9
	Deep AE−MF+neg	1.2E−2	3.31E−3	8.03E−5	3.45E−8	4.21E−4	2.21E−3



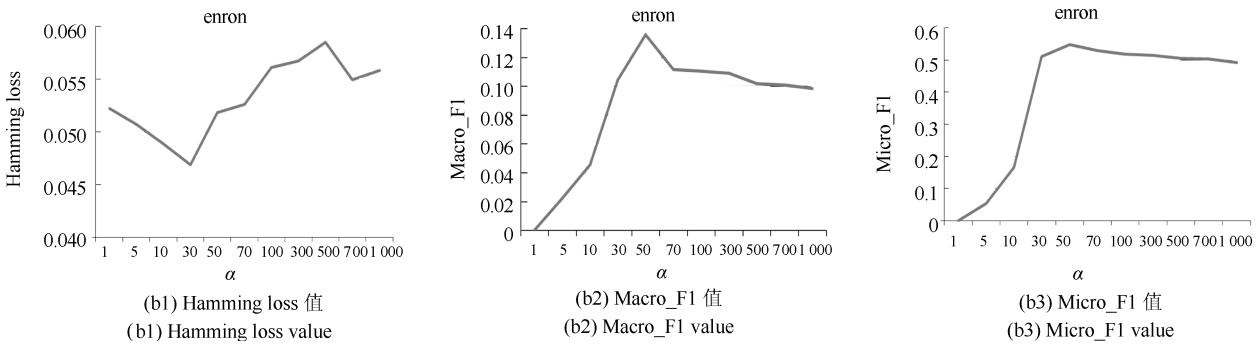
(a1) Hamming loss 值
(a1) Hamming loss value



(a2) Macro_F1 值
(a2) Macro_F1 value



(a3) Micro_F1 值
(a3) Micro_F1 value

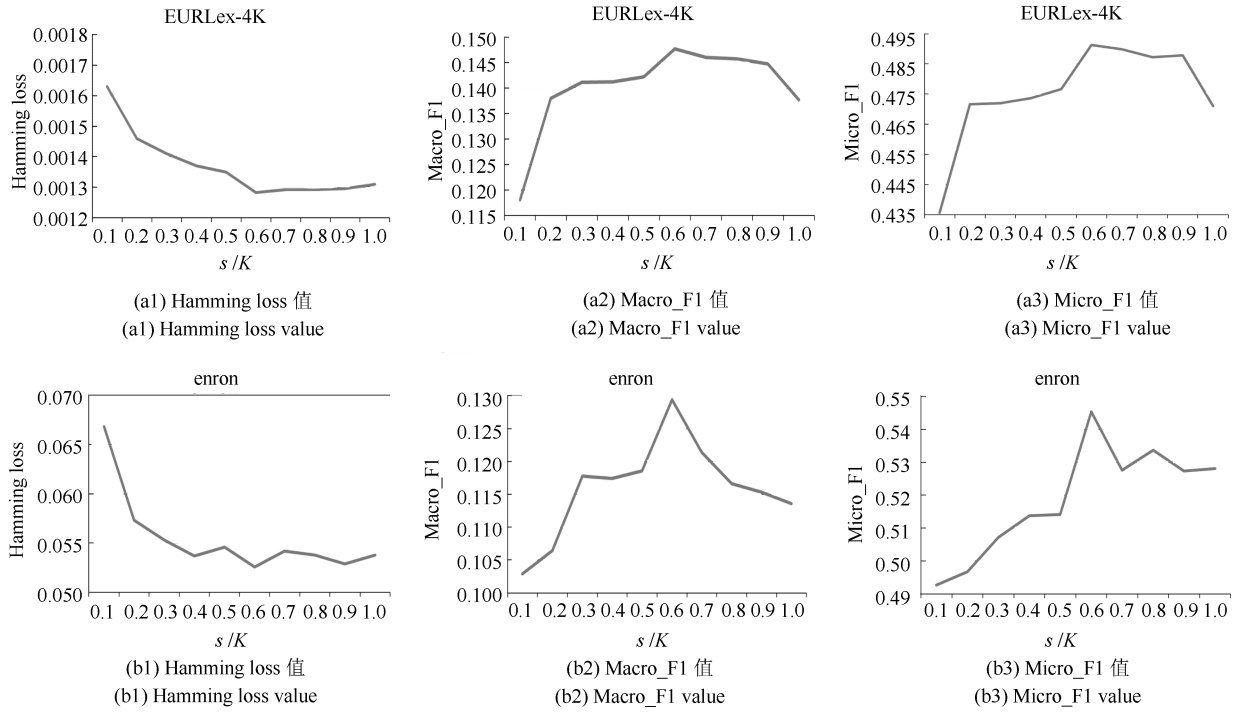


(b1) Hamming loss 值 (b2) Macro_F1 值 (b3) Micro_F1 值

(b1) Hamming loss value (b2) Macro_F1 value (b3) Micro_F1 value

图 3 α 的不同取值对数据集 TJ 和 enron 使用不同度量方式的性能体现

Fig. 3 The performance of Deep AE−MF on data sets TJ and Enronis with respect to different values of α and different metrics



(a1) Hamming loss 值 (a2) Macro_F1 值 (a3) Micro_F1 值

(a1) Hamming loss value (a2) Macro_F1 value (a3) Micro_F1 value

(b1) Hamming loss 值 (b2) Macro_F1 值 (b3) Micro_F1 值

(b1) Hamming loss value (b2) Macro_F1 value (b3) Micro_F1 value

图 4 s/K 的不同取值对数据集 EURLex-4K 和 enron 使用不同度量方式的性能体现

Fig. 4 The performance of Deep AE−MF on data sets EURLex-4K and enron with respect to different values of s/K and different metrics

从图 4 可以看出, 对于 hamming loss 在 EURLex-4K 和 enron 两个数据集上, 随着 s 的增加, 曲线总体先下降再升高 (即性能先上升再下降); 对于基于标签的 Macro_F1 和 Micro_F1 在 EURLex-4K 和 enron 两个数据集上, 随着 s 的增加, 曲线总体先上升再下降 (即性能先上升再下降). 综合衡量图 4 中的实验结果, EURLex-4K 和 enron 在 s 取值为 $0.6K$ 附近时均达到最佳性能, 因此选取 s 为 $0.6K$ 作为最终的取值.

4 结论

本文提出基于 SDAE 和矩阵分解的多标签分类算法 Deep AE−MF 及 Deep AE−MF+neg. Deep AE−MF 算法通过对 SDAE 和 MF 进行耦合得到一个特征嵌入和标签嵌入联合学习框架, 能够有效地对特征非线性关系学习并同时用于标签嵌入学习

中. Deep AE−MF+neg 算法在学习时利用标签之间的负相关 (非共现) 信息特点, 提高标签嵌入学习以此最终提高模型的性能. 实验结果表明, Deep AE−MF 及 Deep AE−MF+neg 优于对比算法, 能够有效地完成相关多标签分类任务.

References

1 Gong Y C, Ke Q F, Isard M, Lazebnik S. A multi-view embedding space for modeling internet images, tags, and their semantics. *International Journal of Computer Vision*, 2014, **106**(2): 210−233

2 Cambria E. Affective computing and sentiment analysis. *IEEE Intelligent Systems*, 2016, **31**(2): 102−107

3 Zhang Chen-Guang, Zhang Yan, Zhang Xia-Huan. Normalized dependence maximization multi-label semi-supervised learning method. *Acta Automatica Sinica*, 2015, **41**(9): 1577−1588
(张晨光, 张燕, 张夏欢. 最大规范化依赖性多标记半监督学习方法. 自动化学报, 2015, **41**(9): 1577−1588)

4 Poria S, Cambria E, Bajpai R, Hussain A. A review of affective computing: from unimodal analysis to multimodal fusion. *Information Fusion*, 2017, **37**: 98−125

5 Boutell M R, Luo J B, Shen X P, Brown C M. Learning multi-label scene classification. *Pattern Recognition*, 2004, **37**(9): 1757−1771

6 Wu Q, Ye Y, Ho S S, Zhou S. Semi-supervised multi-label collective classification ensemble for functional genomics. *BMC Genomics*, 2014, **15**(S9): S17

7 Kazawa H, Izumitani T, Taira H, Maeda E. Maximal margin labeling for multi-topic text categorization. In: Proceedings of the 2005 Advances in Neural Information Processing Systems. Vancouver, Canada: The MIT Press, 2005. 649−656

8 Hüllermeier E, Fürnkranz J, Cheng W W, Brinker K. Label ranking by learning pairwise preferences. *Artificial Intelligence*, 2008, **172**(16−17): 1897−1916

9 Zaragoza J H, Sucar L E, Morales E F, Bielza C, Larrañaga P. Bayesian chain classifiers for multidimensional classification. In: Proceedings of the 22nd International Joint Conference on Artificial Intelligence. Barcelona, Brazil, 2011. 2192−2197

10 Elisseeff A, Weston J. A kernel method for multi-labelled classification. In: Proceedings of the 2002 Advances in Neural Information Processing Systems. Cambridge: MIT, 2002. 681−687

11 Xu J H. An extended one-versus-rest support vector machine for multi-label classification. *Neurocomputing*, 2011, **74**(17): 3114−3124

12 Zhang Min-Ling. An improved multi-label lazy learning approach. *Journal of Computer Research and Development*, 2012, **49**(11): 2271−2282
(张敏灵. 一种新型多标记懒惰学习算法. 计算机研究与发展, 2012, **49**(11): 2271−2282)

13 Zhang M L, Peña J M, Robles V. Feature selection for multi-label naive Bayes classification. *Information Sciences*, 2009, **179**(19): 3218−3229

14 Guo Y, Wu Q Y, Deng C R, Chen J, Tan M K. Double forward propagation for memorized batch normalization. In: Proceedings of the 32nd AAAI Conference on Artiflcial Intelligence. New Orleans, USA: AAAI Press, 2018.

15 Li L, Wang H F. Towards label imbalance in multi-label classification with many labels[Online], available: https://arxiv.org/abs/1604.01304, May 24, 2018.

16 Wu Q Y, Ye Y M, Zhang H J, Chow T W S, Ho S S. MLTREE: a tree-structure-based approach to multilabel learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2015, **26**(3): 430−443

17 Wu Q Y, Tan M K, Song H J, Chen J, Ng M K. MLFOREST: a multi-label tree ensemble method for multi-label classification. *IEEE Transactions on Knowledge and Data Engineering*, 2016, **28**(10): 2665−2680

18 Kapoor A, Jain P, Viswanathan R. Multilabel classification using Bayesian compressed sensing. In: Proceedings of the 2012 Advances in Neural Information Processing Systems. Lake Tahoe, Nevada, USA: NIPS, 2012. 2645−2653

19 Park C H, Lee M. On applying linear discriminant analysis for multi-labeled problems. *Pattern Recognition Letters*, 2008, **29**(7): 878−887

20 Ji S W, Tang L, Yu S P, Ye J P. Extracting shared subspace for multi-label classification. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Las Vegas, USA: ACM, 2008. 381−389

21 Yu S P, Yu K, Tresp V, Kriegel H P. Multi-output regularized feature projection. *IEEE Transactions on Knowledge and Data Engineering*, 2006, **18**(12): 1600−1613

22 Wang J, Yang Y, Mao J H, Huang Z H, Huang C, Xu W. CNN-RNN: a unified framework for multi-label image classification. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, USA: IEEE, 2016. 2285−2294

23 Nam J, Kim J, Mencía E L, Gurevych I, Fürnkranz J. Large-scale multi-label text classification revisiting neural networks. In: Proceedings of the 2014 Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Berlin, Heidelberg, Germany: Springer, 2014. 437−452

24 Hsu D, Kakade S M, Langford J, Zhang T. Multi-label prediction via compressed sensing. In: Proceedings of the 2009 Advances in Neural Information Processing Systems. Vancouver, Canada: MIT Press, 2009. 772−780

25 Tai F, Lin H T. Multilabel classification with principal label space transformation. *Neural Computation*, 2012, **24**(9): 2508−2542

26 Chen Y N, Lin H T. Feature-aware label space dimension reduction for multi-label classification. In: Proceedings of the 2012 Advances in Neural Information Processing Systems. Lake Tahoe, Nevada, US: NIPS, 2012. 1529−1537

27 Lin Z J, Ding G G, Hu M Q, Wang J M. Multi-label classification via feature-aware implicit label space encoding. In: Proceedings of the 31st International Conference on Machine Learning. Beijing, China, 2014. 325−333

28 Yu H F, Jain P, Kar P, Dhillon I. Large-scale multi-label learning with missing labels. In: Proceedings of the 31st International Conference on Machine Learning. Beijing, China: ACM, 2014. 593−601

29 Tsoumakas G, Katakis I. Multi-label classification: an overview. *International Journal of Data Warehousing and Mining*, 2007, **3**(3): 1−13

30 Fu Zhong-Liang. Cost-sensitive ensemble learning algorithm for multi-label classification problems. *Acta Automatica Sinica*, 2014, **40**(6): 1075−1085
(付忠良. 多标签代价敏感分类集成学习算法. 自动化学报, 2014, **40**(6): 1075−1085)

31 Abdi H, Williams L J. Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2010, **2**(4): 433−459

32 Pudil P, Somol P, Haindl M. *Introduction to Statistical Pattern Recognition* (Second Edition). San Diego: Academic Press, 1990. 441−507

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

34 Tsoumakas G, Katakis I, Vlahavas I. Effective and efficient multilabel classification in domains with large number of labels. In: Proceedings of the 2008 ECML/PKDD Workshop on Mining Multidimensional Data. Antwerp, Belgium: Springer, 2008. 53−59

35 Tsoumakas G, Vlahavas I. Random k-labelsets: an ensemble method for multilabel classification. In: Proceedings of the 2008 European Conference on Machine Learning. Berlin, Heidelberg, Germany: Springer, 2007. 406−417

36 Tang Chao-Hui, Zhu Qing-Xin, Hong Chao-Qun, Zhu William. Multi-label feature selection with autoencoders and hypergraph learning. *Acta Automatica Sinica*, 2016, **42**(7): 1014−1021
(唐朝辉, 朱清新, 洪朝群, 祝峰. 基于自编码器及超图学习的多标签特征提取. 自动化学报, 2016, **42**(7): 1014−1021)

37 Zhang Y, Schneider J. Multi-label output codes using canonical correlation analysis. In: Proceedings of the 14th International Conference on Artificial Intelligence and Statistics. Ft. Lauderdale, USA: JMLR, 2011. 873−882

38 Wang W R, Arora R, Livescu K, Bilmes J. On deep multi-view representation learning. In: Proceedings of the 32nd International Conference on Machine Learning. Lille, France: Omni Press, 2015. 1083−1092

39 Schapire R E, Singer Y. BoosTexter: a boosting-based system for text categorization. *Machine Learning*, 2000, **39**(2−3): 135−168

40 Yang Y M. An evaluation of statistical approaches to text categorization. *Information Retrieval*, 1999, **1**(1−2): 69−90

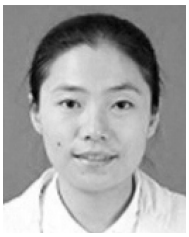
41 Godbole S, Sarawagi S. Discriminative methods for multi-labeled classification. In: Proceedings of the 2004 Pacific-Asia Conference on Knowledge Discovery and Data Mining. Berlin, Heidelberg, Germany: Springer, 2004. 22−30

42 Yen I E H, Huang X R, Ravikumar P, Zhong K, Dhillon I. PD-Sparse: a primal and dual sparse approach to extreme multiclass and multilabel classification. In: Proceedings of the 2016 International Conference on Machine Learning. NY, USA: ACM, 2016. 3069−3077

43 Bhatia K, Jain H, Kar P, Varma M, Jain P. Sparse local embeddings for extreme multi-label classification. In: Proceedings of the 2015 Advances in Neural Information Processing Systems. Montreal, Canada: Cornell University Library, 2015. 730−738

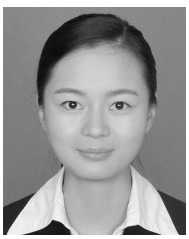
44 Bi W, Kwok J. Efficient multi-label classification with many labels. In: Proceedings of the 2013 International Conference on Machine Learning. Atlanta, GA, USA: ACM, 2013. 405−413

45 Demšar J. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 2006, **7**: 1−30



刘慧婷 安徽大学副教授, 博士. 主要研究方向为机器学习, 数据挖掘. 本文通信作者. E-mail: htliu@ahu.edu.cn
(**LIU Hui-Ting** Ph. D., associate professor at Anhui University. Her research interest covers machine learning and data mining. Corresponding author of this paper.)

冷新杨 安徽大学硕士研究生. 主要研究方向为机器学习, 文本分类.
E-mail: lxy_un@126.com
(**LENG Xin-Yang** Master student at Anhui University. His research interest covers machine learning and text categorization.)



王利利 安徽大学硕士研究生. 主要方向领域为机器学习, 数据挖掘.
E-mail: wll9267@126.com
(**WANG Li-Li** Master student at Anhui University. Her research interest covers machine learning and data mining.)

赵 鹏 安徽大学副教授, 博士. 主要研究方向为机器学习, 智能信息处理.
E-mail: zhaopeng_ad@163.com
(**ZHAO Peng** Ph. D., associate professor at Anhui University. Her research interest covers machine learning and intelligent information processing.)