·人工智能及识别技术· 文章编号:1000-3428(2019)06-0242-07 文献标志码:A 中图分类号:TP393

# 基于流形正则化极限学习机的文本分类算法研究

庞皓明,冀俊忠,刘金铎,姚 垚

(北京工业大学 多媒体与智能软件技术北京市重点实验室,北京 100124)

**摘 要:** 基于极限学习机的文本分类方法在对输入的文本特征进行随机映射时,会呈现一种非线性的几何结构,利用最小二乘法无法对其进行求解,影响文本的分类性能。为此,引入一种新的流形正则化思想,提出基于极限学习机的改进算法。利用拉普拉斯特征映射保持输入文本特征的几何结构。基于样本的类别信息对样本点之间的距离进行修正,优先选择类别相同的样本点,以改善分类性能。在 Reuters 和 20newsgroup 数据集上的实验结果表明,与正则化极限学习机算法、AdaBELM 算法等相比,该算法分类性能较好,*F*1-*measure* 值可达 91.42%。

**关键词:** 文本分类;监督学习;正则化极限学习机;流形正则化;特征映射

**中文引用格式:** 庞皓明,冀俊忠,刘金铎,等. 基于流形正则化极限学习机的文本分类算法研究[J]. 计算机工程,2019,45(6):242-248.

**英文引用格式:** PANG Haoming, JI Junzhong, LIU Jinduo, et al. Research on text classification algorithm based on manifold regularization extreme learning machine [J]. Computer Engineering, 2019, 45(6): 242-248.

# Research on Text Classification Algorithm Based on Manifold Regularization Extreme Learning Machine

PANG Haoming, JI Junzhong, LIU Jinduo, YAO Yao

(Beijing Key Laboratory of Multimedia and Intelligent Software Technology, Beijing University of Technology, Beijing 100124, China)

**【Abstract】** In the text classification process, the Extreme Learning Machine (ELM) randomly maps the input text features and presents a nonlinear geometric structure. As a result, the least square method cannot solve such nonlinear structures and thus affects the text classification performance. To solve this problem, this paper introduces a new manifold regularization and presents an improved algorithm based on extreme machine learning. The Laplace feature mapping is used to preserve the geometry of input text features. The distance between sample points is modified based on the category information of the sample, and the sample points with the same category are selected first to improve the classification performance. Experimental results on the datasets of Reuters and 20newsgroup show that, compared with the Regularization Extreme Learning Machine (RELM), AdaBELM and other algorithms, the proposed algorithm has better classification performance, and the *F*1-*measure* can reach 91.42%.

**【Key words】** text classification; supervise learning; Regularization Extreme Learning Machine (RELM); manifold regularization; feature mapping

**DOI:** 10.19678/j.issn.1000-3428.0050777

## 0 概述

目前,很多机器学习方法被应用于文本分类问题中,例如,朴素贝叶斯[1]、决策树[2]、支持向量机[3]和 K 近邻[4]等。随着技术的发展,深度玻尔兹曼机[5]、循环卷积神经网络[6]、长短期记忆神经网络[7]和深度卷积神经网络[8]等深度学习方法也逐渐应用到文本分类问题中。

文献[9]提出一种单隐层的前馈神经网络学习算法,即极限学习机(Extreme Learning Machine, ELM),其已被应用到许多不同的领域[10-12]。文献[10]将正则化极限学习机(Regularization Extreme Learning Machine, RELM)应用到文本分类问题中,该方法通过在 ELM 的线性方程组中构建正则项以解决 ELM 计算稳定性不足和过拟合问题[13]。文献[14]将一种新的特征选择方法与 ELM 相结合来对文本进行分类。文献[15]将集成学习和 ELM 相结合提出一种新的文本分类模型 AdaBELM,解决了过

**基金项目:** 国家自然科学基金(61375059,61672065)。

**作者简介:** 庞皓明(1993—),男,硕士研究生,主研方向为机器学习、自然语言处理;冀俊忠(通信作者),教授、博士生导师;刘金铎、姚 垚,博士研究生。

**收稿日期:** 2018-03-14 **修回日期:** 2018-05-07 **E-mail:** jjz01@bjut.edu.cn

拟合的问题。虽然 ELM 在文本分类问题上取得了良好的分类结果,但是其在映射的过程中会将输入的文本特征随机映射到 ELM 的特征空间中,呈现一种非线性的几何结构。由于通用的最小二乘法无法对非线性结构进行恢复和求解,因此会影响文本的分类性能。

2000 年,关于流形学习的研究取得重要进展[16-18],至此,流形学习方法受到机器学习、模式识别、数据挖掘等领域研究者的广泛关注。流形学习不仅能将高维输入数据点映射到一个全局低维坐标系中,还能保留邻接点之间的关系,使数据保持原有的几何结构。同时,该模型还具有平移、旋转不变的特性。已有学者将流形正则化思想和 ELM 相结合应用于监督学习中[19-20],但其在应用过程中存在一定的缺陷。在有监督流形学习中,样本的类别信息可对样本点间的距离进行修正,帮助类别信息融入邻接图的构造中,但是文献[19-20]所使用的流形学习方法,仅通过欧式距离对样本点之间的距离进行度量,忽视了类别信息对分类性能的提升作用。

综上所述,在文本分类过程中,ELM 对文本特征的随机映射影响了文本的分类性能。流形学习可以使数据在新的特征空间中保持原有的几何结构,使原特征空间中距离很近的样本在新的投影空间中相互接近[21-22]。因此,可以将流形正则化思想引入极限学习机相中,以保持文本特征的几何结构。另一方面,监督学习中样本的类别信息有助于提高分类性能,而已有的流形学习与极限学习机相结合的方法没有考虑样本的类别信息。因此,本文将一种新的流形正则化思想引入到极限学习机中,提出基于流形正则化极限学习机的文本分类方法(Mainfold Regularization Extreme Learning Machine on Text classification,MRELMT)。

## 1　相关工作

### 1.1　极限学习机

ELM 作为一种单隐层的前馈神经网络,其输入层和隐藏层之间的权值矩阵,以及隐藏层节点的偏置是随机产生的。在训练过程中,只需要设置隐藏层节点的个数,就可以通过最小二乘法计算出隐藏层和输出层之间的权值矩阵。

给定 $N$ 个不同的训练样本 $(\boldsymbol{x}_i, \boldsymbol{y}_i)$,$\boldsymbol{x}_i = [x_{i1}, x_{i2}, \cdots, x_{in}]^{\mathrm{T}} \in \mathbb{R}^n$,$\boldsymbol{y}_i = [y_{i1}, y_{i2}, \cdots, y_{im}]^{\mathrm{T}} \in \mathbb{R}^m$,$i = 1, 2, \cdots, N$,隐藏层节点个数为 $L$,激活函数为 $g(x)$,则 ELM 模型可表示为:

$$\sum_j^L \boldsymbol{\beta}_j g(\boldsymbol{x}_i) = \sum_j^L \boldsymbol{\beta}_j g(\boldsymbol{w}_j \boldsymbol{x}_i + b_j) = \boldsymbol{y}_i \tag{1}$$

其中,$\boldsymbol{w}_j = [w_{j1}, w_{j2}, \cdots, w_{jn}]^{\mathrm{T}}$,$j = 1, 2, \cdots, L$,$\boldsymbol{w}_j$ 表示输入层和第 $j$ 个隐藏层节点之间的权值向量,$b_j$ 表示第 $j$ 个隐藏层节点的偏置,$\boldsymbol{\beta}_j = [\beta_{j1}, \beta_{j2}, \cdots, \beta_{jn}]^{\mathrm{T}}$ 表示第 $j$ 个隐藏层节点与输出层之间的权值向量。ELM 也可以通过矩阵的形式表示为:

$$\boldsymbol{H\beta} = \boldsymbol{Y} \tag{2}$$

其中,$\boldsymbol{H}$ 是隐藏层的输出矩阵,$\boldsymbol{\beta}$ 是隐藏层和输出层之间的权值矩阵,具体如下:

$$\boldsymbol{H} = \begin{bmatrix} g(\boldsymbol{w}_1, b_1, \boldsymbol{x}_1) & g(\boldsymbol{w}_2, b_2, \boldsymbol{x}_1) & \cdots & g(\boldsymbol{w}_n, b_n, \boldsymbol{x}_1) \\ g(\boldsymbol{w}_1, b_1, \boldsymbol{x}_2) & g(\boldsymbol{w}_2, b_2, \boldsymbol{x}_2) & \cdots & g(\boldsymbol{w}_n, b_n, \boldsymbol{x}_2) \\ \vdots & \vdots & & \vdots \\ g(\boldsymbol{w}_1, b_1, \boldsymbol{x}_L) & g(\boldsymbol{w}_2, b_2, \boldsymbol{x}_L) & \cdots & g(\boldsymbol{w}_n, b_n, \boldsymbol{x}_L) \end{bmatrix}_{N \times L} \tag{3}$$

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1^{\mathrm{T}} \\ \boldsymbol{\beta}_2^{\mathrm{T}} \\ \vdots \\ \boldsymbol{\beta}_L^{\mathrm{T}} \end{bmatrix}_{L \times M} \tag{4}$$

极限学习机为了提高预测的准确性,将训练误差(经验风险)和输出矩阵的二次范式(结构风险)同时最小化,获得隐藏层和输出层之间的权值矩阵,可以得到如下公式:

$$\min: \| \boldsymbol{H\beta} - \boldsymbol{Y} \|_2^2 + \frac{C}{2} \| \boldsymbol{\beta} \|_2^2 \tag{5}$$

其中,$C$ 是正则化因子,$\| \cdot \|_2^2$ 表示二范数。通过最小二乘法计算得到隐藏层和输出层之间的权值矩阵:

$$\boldsymbol{\beta} = \left( \frac{1}{C} + \boldsymbol{H}^{\mathrm{T}} \boldsymbol{H} \right)^{-1} \boldsymbol{H}^{\mathrm{T}} \boldsymbol{Y} \tag{6}$$

ELM 隐藏层和输出层之间的权值通过最小二乘法直接计算得到,算法的整个学习过程一次完成,无需迭代,因此该模型具有较快的学习速度。但是,输入层和隐藏层之间的权值矩阵和隐藏层节点的偏置是通过随机初始化得到的,在计算过程中会使文本特征从输入层随机地映射到隐藏层。因此输入的文本特征在 ELM 特征空间中的分布具有一定的随机性,这会使得输入的文本特征出现某种非线性的几何结构,仅使用最小二乘法无法对其几何结构进行求解和恢复,进而影响文本的分类性能。

### 1.2　流形学习

拉普拉斯映射是流形学习中的一种映射方法,数据在经过拉普拉斯特征映射之后,依然能保持原数据的特征。具体过程描述如下:

首先计算每个样本点 $Z_m$ 和其他样本点之间的欧氏距离,使用 k 近邻方法找到与样本点距离最近的 $k$ 个样本点作为 $Z_m$ 的邻居节点。根据式(7)计算样本点之间的权重 $w_{mn}$,得到样本点之间的矩阵 $\boldsymbol{W}$。

$$w_{mn} = \begin{cases} 1, & e(Z_m, Z_n) = 1 \\ 0, & \text{其他} \end{cases} \tag{7}$$

其中,$e(Z_m, Z_n) = 1$ 表示样本 $Z_m$ 和 $Z_n$ 为邻居。通过计算拉普拉斯算子的广义特征向量,求解广义特征值问题 $\boldsymbol{Lf} = \boldsymbol{LDf}$,其中 $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$,$D_{mn} = \sum w_{mn}$。

拉普拉斯映射方法虽然在映射过程中可以保持原有数据的几何结构,但是在计算样本点之间的距离时忽略了样本的类别信息。流形假设认为,如果 2 个样本属于同一类别,则它们在实际空间中的距离要比不属于同一类别的 2 个样本的距离要更近一些[21-22]。因此,当样本间的距离相同时,该方法无法选出属于同一类别的样本点,只能随机选取。

## 2 算法介绍

本文将一种新的流形正则化思想引入极限学习机中,通过流形正则化极限学习机对文本进行分类,以提高分类性能。如图 1 所示,流形正则化极限学习机文本分类模型由 4 个部分组成:

1)预处理:通过预处理操作清除文档数据中无用的信息和噪声等。该部分包括编码转换、移除停用词、大小写转换、英文词干提取、移除低频词、训练集和测试集分割等操作。

2)文本表示:使用空间向量模型(Vector Space Model,VSM)对文本进行表示,将一篇文章表示为特征空间的一个向量。

3)特征提取:采用奇异值分解(Singular Value Decomposition,SVD)的方式对文本特征进行特征提取,将原始的高维文本特征映射到其低维的子空间中。

4)文本分类:使用流形正则化极限学习机文本分类方法对文本进行分类。

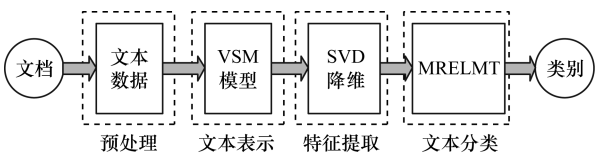


**图 1 基于流形正则化极限学习机的文本分类模型**

### 2.1 流形正则化思想

当计算样本间的距离相同时,现有的流形正则化思想无法选出属于同一类别的样本点。针对这一问题,新的流形正则化思想使用一种有监督的距离度量方法来计算样本点之间的距离。在距离的度量方法中引入样本的类别信息,基于这些类别信息对样本点之间的距离进行修正,进而优先选择类别相同的样本点。

流形正则化主要利用流形假设的思想,如果2 个样本 $Z_m$ 和 $Z_n$ 接近,那么它们的标签 $p_m$ 和 $p_n$ 也应该接近。为实现这一假设,将流形正则化的目标函数定义为:

$$\min:\frac{1}{2}\sum_{mn}s_{mn}\parallel p_m-p_n\parallel_2^2 \tag{8}$$

其中,$s_{mn}$表示样本 $Z_m$ 和 $Z_n$ 之间的相似度。在计算样本点相似度之前,需要使用 k 近邻方法找到与样本点距离最近的 $k$ 个样本点。使用流形学习中有监督的距离度量方法计算样本点之间的距离,即在距离计算时考虑样本类别信息。该方法认为在样本点的所有近邻点中,其同类样本点邻近的概率大于异类邻近的概率,因此在计算 2 个样本点之间的距离时增加修正项,使同类样本点间的距离尽可能小于异类样本点间的距离。此方法不仅可以保持同类样本间的距离,而且还拉大了不同类样本间的距离。有监督的距离度量方法可以保持有监督数据的流形结构,减少对输入特征的随机映射。样本间距离计算过程如下:

$$D(z_m,z_n)=\begin{cases}\sqrt{1-\mathrm{e}^{\frac{-d^2(z_m,z_n)}{\alpha}}}, & p_m=p_n\\ \sqrt{\mathrm{e}^{\frac{-d^2(z_m,z_n)}{\alpha}}}-\sigma, & p_m\neq p_n\end{cases} \tag{9}$$

其中,$\alpha$ 和 $\sigma$ 是指定常数,$D(z_m,z_n)$是结合样本点类别信息之后的距离,$d(z_m,z_n)$是忽略样本点类别信息的欧式距离,即 $d(z_m,z_n)=|z_m-z_n|$。参数 $\alpha$ 用来避免指数函数中的 $d(z_m,z_n)$的快速增加,特别是当其自身比较大的时候,参数 $\alpha$ 的调节作用较大。同时,参数 $\alpha$ 与数据集中的数据密集程度密切相关,一般取所有成对数据点的欧氏距离的平均值。参数 $\sigma(0\leqslant\sigma\leqslant1)$是调节不同类别数据点间距离的常数因子。

在计算样本点间的距离之后,选择 $k$ 个样本作为样本 $z_m$ 的邻居节点,并通过高斯核函数计算样本点与邻居节点之间的相似度 $s_{mn}$。本文只对属于同一类别并且互为邻居的节点进行计算,其他互为邻居的数据不进行计算。相似度计算过程如下:

$$s_{mn}=\begin{cases}\exp\left(\dfrac{-\parallel z_m-z_n\parallel_2^2}{\rho}\right), & e(z_m,z_n)=1,c(z_m)=c(z_n)\\ 0,\text{其他}\end{cases} \tag{10}$$

其中,$c(z_m)=c(z_n)$表示样本 $z_m$ 和 $z_n$ 的类别相同。样本 $z_m$ 和 $z_n$ 之间的相似度越大,则 $s_{mn}$的值也越大。因此得到相似度矩阵 $\boldsymbol{S}$ 如下:

$$\boldsymbol{S}=\begin{bmatrix}s_{11} & s_{12} & \cdots & s_{1N}\\ s_{21} & s_{22} & \cdots & s_{2N}\\ \vdots & \vdots & & \vdots\\ s_{N1} & s_{N2} & \cdots & s_{NN}\end{bmatrix}_{N\times N} \tag{11}$$

相似度矩阵 $\boldsymbol{S}$ 是一个对称矩阵。经过变换,流形正则化思想的目标函数可以表示为:

$$\min:\ \mathrm{tr}(\boldsymbol{Y}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{Y}) \tag{12}$$

其中,$\boldsymbol{Y}=[\boldsymbol{y}_1\quad\boldsymbol{y}_2\quad\cdots\quad\boldsymbol{y}_N]^{\mathrm{T}}$ 是样本点的输出矩阵,$\boldsymbol{L}$ 是经过计算得到的拉普拉斯矩阵,$\boldsymbol{L}=\boldsymbol{D}-\boldsymbol{S}$,$\boldsymbol{D}$ 是一个 $N\times N$ 大小的对角矩阵,计算如下:

$$D_{mm}=\sum_{i=1}^{n}s_{mn} \tag{13}$$

其中,$D_{mm}$表示矩阵 $\boldsymbol{D}$ 中第 $m$ 行的对角值。

### 2.2　基于流形正则化极限学习机的文本分类模型

基于上述的分析和推导,本文将流形正则化思想引入极限学习机中,以克服其在文本分类问题中出现的问题。首先,对文本数据集进行如下数学描述:文本数据集 $\boldsymbol{\Omega}=\{(\boldsymbol{x}_i,\boldsymbol{y}_i)\}$, $i=1,2,\cdots,N$,有 $N$ 篇文档$(\boldsymbol{x}_i,\boldsymbol{y}_i)$,其中,$\boldsymbol{x}_i=[x_{i1},x_{i2},\cdots,x_{ir}]^{\mathrm{T}}\in\mathbb{R}^r$ 代表文本数据集 $\boldsymbol{\Omega}$ 中的第 $i$ 篇文档,$x_{ir}$代表第 $i$ 篇文档中的第 $r$ 个特征词的 TF-IDF 权值,$r$ 为经过 SVD 处理之后文本数据集 $\boldsymbol{\Omega}$ 中特征词的个数,$\boldsymbol{y}_i=[y_{i1},y_{i2},\cdots,y_{iq}]^{\mathrm{T}}\in\mathbb{R}^q$ 则代表第 $i$ 篇文档所属的类别,$q$ 表示文本数据集 $\boldsymbol{\Omega}$ 中的类别数。此外,需要设置隐藏层节点个数 $o$ 和激活函数 $g(\boldsymbol{x})$,随机设置隐藏层节点参数:权值 $a_j\in\mathbb{R}^n$,偏置 $b_j\in\mathbb{R}^n$,$j=1,2,\cdots,o$。流形正则化极限学习机结构如图 2 所示。

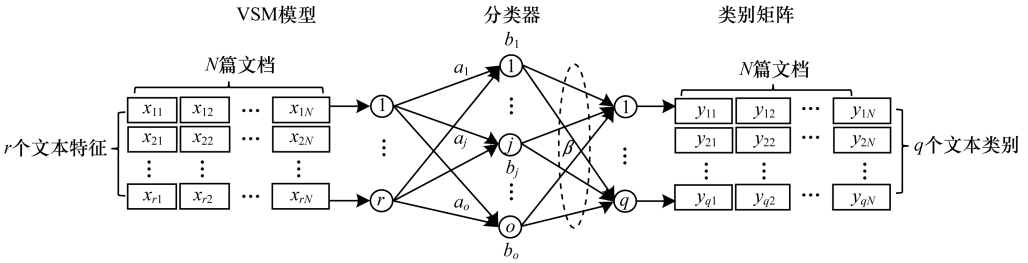


**图 2　流形正则化极限学习机示意图**

为了避免 ELM 在训练过程中,随机设置输入层和隐藏层之间的权值 $a_o$ 以及隐藏层节点的偏置 $b_o$ 而产生非线性的映射,本文将流形正则化思想引入正则化极限学习机中,即将目标函数式(12)引入正则化极限学习机中,使其选出能够保持原有结构的邻接点,得到以下公式:

$$\min:\frac{1}{2}\|\boldsymbol{H\beta}-\boldsymbol{Y}\|_2^2+\frac{C_1}{2}\|\boldsymbol{\beta}\|_2^2+\frac{C_2}{2}\mathrm{tr}(\boldsymbol{Y}^{\mathrm{T}}\boldsymbol{LY}) \tag{14}$$

其中,$C_1$ 为 RELM 的正则化因子,用来平衡极限学习机的经验风险和结构风险,$C_2$ 为流形正则化思想中的正则化因子。通过对目标函数求导并另导数值为 0,得到下式:

$$\boldsymbol{H}^{\mathrm{T}}(\boldsymbol{H\beta}-\boldsymbol{Y})+C_1\boldsymbol{\beta}+C_2\boldsymbol{H}^{\mathrm{T}}\boldsymbol{LH\beta}=0 \tag{15}$$

计算隐藏层和输出层之间的矩阵 $\boldsymbol{\beta}$,结果如下:

$$\boldsymbol{\beta}=(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{H}+C_1\boldsymbol{I}+C_2\boldsymbol{H}^{\mathrm{T}}\boldsymbol{LH})^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{Y} \tag{16}$$

$\boldsymbol{H}$ 为隐藏层的输出矩阵,即:

$$\boldsymbol{H}=\begin{bmatrix} g(\boldsymbol{w}_1,b_1,\boldsymbol{x}_1) & g(\boldsymbol{w}_2,b_2,\boldsymbol{x}_1) & \cdots & g(\boldsymbol{w}_o,b_o,\boldsymbol{x}_1)\\ g(\boldsymbol{w}_1,b_1,\boldsymbol{x}_2) & g(\boldsymbol{w}_2,b_2,\boldsymbol{x}_2) & \cdots & g(\boldsymbol{w}_o,b_o,\boldsymbol{x}_2)\\ \vdots & \vdots & & \vdots\\ g(\boldsymbol{w}_1,b_1,\boldsymbol{x}_N) & g(\boldsymbol{w}_2,b_2,\boldsymbol{x}_N) & \cdots & g(\boldsymbol{w}_o,b_o,\boldsymbol{x}_N) \end{bmatrix}_{N\times o} \tag{17}$$

### 2.3　MRELMT 算法

MRELMT 算法的具体过程如下:

**算法**　MRELMT 算法

**输入**　训练样本集 $\boldsymbol{\Omega}=\{(\boldsymbol{x}_i,\boldsymbol{y}_i)\}$, $i=1,2,\cdots,N$, $\boldsymbol{x}_i\in\mathbb{R}^r,\boldsymbol{y}_i\in\mathbb{R}^q$

**输出**　隐藏层节点的输出矩阵 $\boldsymbol{\beta}$

**步骤 1**　设置 MRELMT 隐藏层节点个数 $o$ 和激活函数 $g(\boldsymbol{x})$,随机设置隐藏层节点参数:权值 $a_j\in\mathbb{R}^n$,偏置 $b_j\in\mathbb{R}^n$,$j=1,2,\cdots,o$。

**步骤 2**　将全部数据映射到 MRELMT 的特征空间中。

**步骤 3**　根据式(9)计算样本 $\boldsymbol{x}_i$ 与其他节点之间的距离,并使用 k 近邻方法选择 $k$ 个样本作为 $\boldsymbol{x}_i$ 的邻居节点。

**步骤 4**　根据式(10)计算样本 $\boldsymbol{x}_i$ 和邻居节点之间的相似度,得到相似度矩阵 $\boldsymbol{S}$。

**步骤 5**　根据 $\boldsymbol{L}=\boldsymbol{D}-\boldsymbol{S}$ 计算的拉普拉斯矩阵 $\boldsymbol{L}$。

**步骤 6**　根据式(16)计算隐藏层节点的输出矩阵$\boldsymbol{\beta}$。

MRELMT 算法与极限学习机文本分类算法最大的不同在于其隐藏层和输出层之间的权值矩阵求解方式不同。RELM 是通过最小二乘法直接计算得到,时间复杂度为 $O(1)$,而 MRELMT 算法的通过引入流形正则化思想计算得到。MRELMT 算法在计算权值矩阵 $\boldsymbol{\beta}$ 时,根据式(9)计算每个样本点与其他样本间的距离,并选择出距离最近的 $k$ 个邻居节点。通过式(10)和式(11)计算样本点与邻居节点之间的相似度矩阵,计算相似度矩阵时的时间复杂度为 $O(N^2)$,其中,$N$ 是样本数。然后通过式(13)计算得到 $N\times N$ 大小的对角矩阵 $\boldsymbol{D}$,其时间复杂度为 $O(N^2)$,并通过公式 $\boldsymbol{L}=\boldsymbol{D}-\boldsymbol{S}$ 计算得到样本点的拉普拉斯矩阵 $\boldsymbol{L}$。最后应用最小二乘法计算隐藏层和输出层之间的矩阵 $\boldsymbol{\beta}$,此部分直接通过矩阵的运算完成,无需迭代,故时间复杂度为 $O(1)$。因此,MRELMT 算法的时间复杂度为 $O(N^2)$。与 RELM 算法相比,MRELMT 算法时间复杂度更高,这主要与算法中计算拉普拉斯矩阵有关,且训练样本的数目越多,算法的计算复杂度越高。

## 3　实验结果与分析

### 3.1　实验数据集及评价标准

本文实验使用了 2 个常用的标准英文数据集:20newsgroup 与 Reuters。其中,20newsgroup 数据集是由约 20 000 篇新闻文本组成的数据集,这些文本被分为 20 个不同的组,每个组对应一个类别,每个类别的文档数分布均匀。采用标准的 ModApte 划分方法将

数据集分为训练集和测试集。经过移除停用词、小于5 的低频词,大小写转换,词干还原等预处理操作后,该数据集包含 27 800 个特征词、20 个文本类别、13 192 篇训练集样本和 5 654 篇测试集样本。本文选取前 20% 的特征词,即使用 5 560 个特征词进行实验。Reuters 数据集是由路透社发布的21 578 篇财经新闻组成的文本数据集,这些新闻文本被分为 5 大类,135 小类。由于 Reuters 数据集文档分布不均匀,本文选取其中最常用的 10 个子类别组成的文本数据集。经过与 20newsgroup 数据集相同的预处理操作后,该数据集包含 6 429 个特征词、10 个文本类别、5 271 篇训练集文本、2 252 篇测试集文本。表 1 给出 20newsgroup 和 Reuters 数据集的基本参数。

**表 1 2 个数据集的基本参数**

| 数据集 | 训练样本 | 测试样本 | 特征词 | 类别 |
|---|---|---|---|---|
| Reuters | 5 271 | 2 252 | 6 429 | 10 |
| 20newsgroup | 13 192 | 5 654 | 5 560 | 20 |

采用准确率(Accuracy)、召回率(Recall)、精确率(Precision)和 F1 值(F1-measure) 4 个指标来评估文本分类算法的效果,其具体计算过程如下:

$$Accuracy=\frac{N_{TP}+N_{TN}}{N_{TP}+N_{TN}+N_{FN}+N_{FP}}\times 100\% = \frac{N_{TP}+N_{TN}}{N_P+N_L}\times 100\% \tag{18}$$

$$Precision=\frac{N_{TP}}{N_{TP}+N_{FP}}\times 100\% \tag{19}$$

$$Recall=\frac{N_{TP}}{N_{TP}+N_{FN}}\times 100\% = \frac{N_{TP}}{N_P}\times 100\% \tag{20}$$

$$F1\text{-}measure=\frac{2Recall\cdot Precision}{Precision+Recall}\times 100\% \tag{21}$$

其中,$N$ 表示数据集中的样本个数,$N_P$ 表示正样本个数,$N_L$ 表示负样本个数,$N_{TP}$表示实际为正、预测为正的样本数,$N_{FN}$表示实际为正、预测为负的样本数,$N_{TN}$表示实际为负、预测为负的样本数,$N_{FP}$表示实际为负、预测为正的样本数,且 $N_P=N_{TP}+N_{FN}$,$N_L=N_{TN}+N_{FP}$。

### 3.2 实验环境及参数

本文实验在以下环境配置下进行:Intel(R)Core(TM)i5-4590CPU@3.30 GHz,12 GB 内存,Windows7 64 位操作系统,Matlab2010 软件平台。

首先确定 MRELMT 算法的隐藏层节点个数,将其取值范围设定为 1 000 ~ 11 000,步长为 500。在实验过程中,每次只改变隐藏层节点的个数,经过多次试验,确定隐藏层结点个数为 9 000。图 3 给出 MRELMT 算法在 Reuters 和 20newsgroup 数据集上 *F1-measure* 随着隐藏层节点个数变化的折线图。从图 3 可以看出,MRELMT 算法的 *F1-measure* 随着隐藏层节点个数的增加而不断提高,达到峰值后继续增加节点个数会出现过拟合现象。在 20newsgroup 数据集上,当隐藏层节点达到 4 000 时,*F1-measure* 的变化趋势逐渐平缓,其分类效果趋于稳定,当节点数达 9 000 时,分类效果最佳。在 Reuters 数据集上,当隐藏层节点数较少时,MRELMT 算法的分类效果较差,当节点数达到 5 000 时,分类效果显著提升,之后随着隐藏层节点个数的增加,分类性能持续上升。

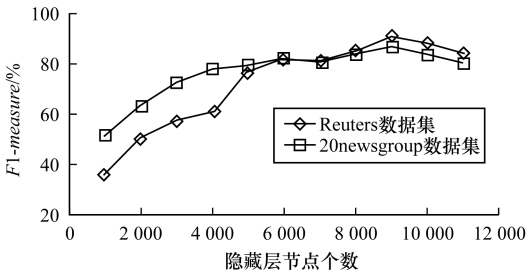


**图 3 MRELMT 算法在 2 个数据集上的 *F1-measure* 值**

然后,确定正则项因子 $C_1$ 和 $C_2$ 的值,其取值范围是 $\{10^{-4},10^{-3},\cdots,10^{3},10^{4}\}$,经过多次实验,取 $C_1=1\ 000$,$C_2=0.01$。在实验过程中发现,正则化参数 $C_1$ 对分类性能的影响较大。选取不同的 $C_1$,算法的分类性能会发生较大变化。当 $C_1$ 取值偏大时,MRELMT 算法的分类性能相对较好。当 $C_1$ 值为 $10^3$ 或 $10^4$ 时,分类效果最好,因此固定 $C_1$ 值,对 $C_2$ 值进行选取。在实验中,$C_2$ 对分类性能的影响没有 $C_1$ 明显。当 $C_2$ 值为 0.01 时,分类性能最佳。将本文算法与 RELM 算法进行对比实验,其参数的设置使用与 MRELMT 算法相同的策略,RELM 的正则化因子取0.01,隐藏层节点个数为 8 000。表 2 给出 MRELMT 算法的训练参数。

**表 2 MRELMT 算法的训练参数**

| 参数 | 取值 |
|---|---|
| 隐藏层节点个数 | 9 000 |
| 正则化因子 $C_1$ | 1 000 |
| 正则化因子 $C_2$ | 0.01 |

### 3.3 结果分析

为了验证 MRELMT 算法的有效性,将其与 RELM 算法[17]在 Reuters 和 20newsgroup 数据集上进行对比。此外,将 MRELMT 算法与 AdaBELM 算法[15]、ELM-OAA[14]和 DBN + Softmax(2)[23]进行比较,以进一步评价多隐层极限学习机在文本分类问题中的分类性能。

#### 3.3.1 流形正则化的效果

表 3 给出本文 MRELMT 算法和 RELM 算法在 2 个数据集上的分类性能评价指标。从表 3 可以看出,在 Reuters 和 20newsgroup 数据集上 MRELMT 算法的 *Accuracy*、*Precision*、*Recall*、*F1-measure* 明显高于 RELM 算法,表明 MRELMT 算法具有更好的分类性能和分类稳定性。这是因为 MRELM 算法在极限学习机中引入流形正则化思想,使文本的分类性能明显提升。由此证明了流形正则化可以使输入的文本特征在经过映射之后保持原有的特征结构。新的流形正则化思想在构建拉普拉斯矩阵的过程中

优先选择类别相同的样本点,在样本间距离相同时,MRELMT 算法会优先选择类别相同的临近点,进一步提升了分类性能。

**表 3　2 种算法在不同数据集上的分类性能**　%

| 数据集 | 算法 | *Accuracy* | *Precision* | *Recall* | *F1-measure* |
|---|---|---|---|---|---|
| Reuters | RELM | 85.43 | 92.23 | 86.37 | 89.23 |
| | MRELMT | 90.39 | 93.12 | 89.78 | 91.42 |
| 20newsgroup | RELM | 84.26 | 85.28 | 84.63 | 84.34 |
| | MRELMT | 86.91 | 86.59 | 85.47 | 86.03 |

3.3.2　多种算法分类结果的对比

为了进一步验证 MRELMT 算法的分类性能,将本文算法与其他 4 种算法在 Reuters 和 20newsgroup 数据集上进行对比。图 4 给出 5 种算法在 Reuters 数据集上的 *Accuracy*、*Precision*、*Recall*、*F1-measure*。从图 4 可以看出,在 Reuters 数据集上 MRELMT 算法的准确率略低于 DBN + Softmax(2)算法,其召回率略低于 ELM-OAA 算法,但是其精确率和 *F1-measure* 取得了较好的结果。

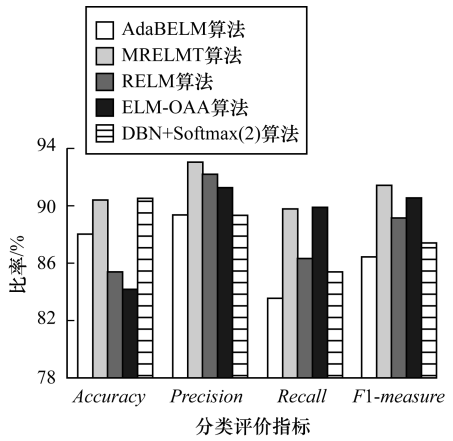


**图 4　5 种算法在 Reuters 数据集上的分类性能**

图 5 给出 5 种算法在 20newsgroup 数据集上分类的 *Accuracy*、*Precision*、*Recall*、*F1-measure*。从图 5 可以看出,在 20newsgroup 数据集上 MRELMT 算法的准确率和召回率都取得较优结果,但是其精确率和 *F1-measure* 低于 DBN + Softmax(2)算法。

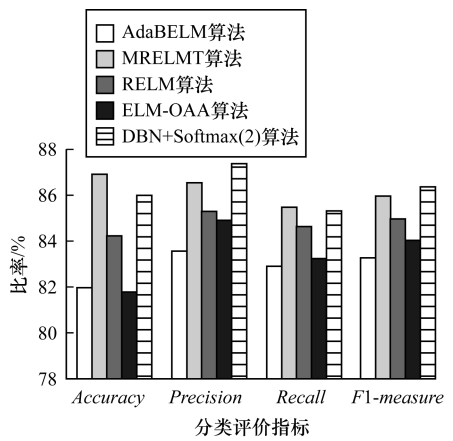


**图 5　5 种算法在 20newsgroup 数据集上的分类性能**

从上述结果可以看出,MRELMT 算法在 Reuters 和 20newsgroup 2 个数据集上的分类性能要优于 AdaBELM、ELM-OAA 和 RELM 等其他使用了极限学习机的算法。*F1-measure* 作为一个结合了精确率和召回率的综合评价标准,可全面反映算法的性能。在 *F1-measure* 指标上,MRELMT 算法在 2 个数据集上都要优于 AdaBELM 算法、ELM-OAA 算法和 RELM 算法,进一步说明在极限学习机中引入流形正则化思想的有效性。其主要原因是 MRELMT 算法可以在一定程度上改善输入文本特征随机映射的问题,使输入的文本特征在极限学习机的特征空间中依然保持原有的几何结构。在引入流形正则化思想后,MRELMT 算法在构建拉普拉斯矩阵的过程中,可通过文本的类别信息对样本间的距离进行修正,以提升算法的文本分类性能。然而,MRELMT 算法在 Reuters 和 20newsgroup 2 个数据集上的某些分类性能指标要低于 DBN + Softmax(2)算法,其主要原因在于 DBN + Softmax(2)算法具有深层网络结构,特征提取能力较强,能够获取更多的文本语义特征,其对文本的分类性能有一定的影响。

## 4　结束语

本文提出一种基于流形正则化极限学习机的文本分类算法。通过引入流形正则化思想,保持输入文本特征的几何结构,并基于文本的类别信息对样本点间的距离进行修正。实验结果表明,与其他极限学习机的文本分类算法相比,该算法的分类性能有一定的提升。下一步将引入多隐层结构的极限学习机对文本进行分类,以改善特征提取能力,获得更高层的文本特征,进而提高文本分类的性能。

## 参考文献

[1] JIANG Liangxiao, WANG Dianhong, CAI Zhihua. Discriminatively weighted naive bayes and its application in text classification [J]. International Journal on Artificial Intelligence Tools, 2012, 21(1): 1-19.

[2] KENEKAYORO P, BUCKLEY K, THELWALL M. Automatic classification of academic Web page types [J]. Scientometrics, 2014, 101(2): 1015-1026.

[3] LILLEBERG J, ZHU Yun, ZHANG Yanqing. Support vector machines and word2vec for text classification with semantic features [C]//Proceedings of the 14th International Conference on Cognitive Informatics and Cognitive Computing. Washington D. C., USA: IEEE Press, 2015: 136-140.

[4] 杨帅华,张清华. 粗糙集近似集的 KNN 文本分类算法研究[J]. 小型微型计算机系统,2017,38(10):2192-2196.

[5] SRIVASTAVA N, SALAKHUTDINOV R R, HINTON G E. Modeling documents with deep boltzmann machines[EB/OL]. [2018-02-25]. https://arxiv.org/ftp/arxiv/papers/1309/1309.6865.pdf.

[6] LAI Siwei,XU Liheng,LIU Kang,et al. Recurrent convolutional neural networks for text classification[C]//Proceedings of the 29th AAAI Conference on Artificial Intelligence. Palo Alto, USA:Association for the Advance of Artificial Intelligence, 2015:2267-2273.

[7] ZHOU Peng, QI Zhenyu, ZHENG Suncong, et al. Text classification improved by integrating bidirectional LSTM with two-dimensional max pooling[EB/OL]. [2018-02-25]. https://arxiv. org/pdf/1611. 06639. pdf.

[8] CONNEAU A, SCHWENK H, BARRAULT L, et al. Very deep convolutional networks for text classification [EB/OL]. [2018-02-25]. https://arxiv. org/pdf/1606. 01781. pdf.

[9] HUANG Guangbin, ZHU Qinyu, SIEW C K. Extreme learning machine: theory and applications [J]. Neurocomputing,2006,70(1/2/3):489-501.

[10] GURPINAR F, KAYA H, DIBEKLIOGLU H, et al. Kernel ELM and CNN based facial age estimation[C]// Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops. Washington D. C., USA:IEEE Press,2016:80-86.

[11] BAI Peng,LIU Huaping,SUN Fuchun,et al. Robotic grasp stability analysis using extreme learning machine[C]//Proceedings of ELM'16. Berlin,Germany:Springer,2016:37-51.

[12] YANG Chenguang,HUANG Kunxia,CHENG Hong,et al. Haptic identification by ELM-controlled uncertain manipulator[J]. IEEE Transactions on Systems, Man, and Cybernetics:Systems,2017,47(8):2398-2409.

[13] ZHENG Wenbin, QIAN Yuntao, LU Huijuan. Text categorization based on regularization extreme learning machine[J]. Neural Computing and Applications,2013, 22(3/4):447-456.

[14] ROUL R K,NANDA A,PATEL V,et al. Extreme learning machines in the field of text classification[C]//Proceedings of the 16th IEEE/ACIS International Conference on Software Engineering,Artificial Intelligence,Networking and Parallel/ Distributed Computing. Washington D. C.,USA:IEEE Press, 2015:1-7.

[15] FENG Xiaoyue, LIANG Yanchun, SHI Xiaohu, et al. Overfitting reduction of text classification based on AdaBELM[J]. Entropy,2017,19(7):1-13.

[16] SEUNG H S, LEE D D. The manifold ways of perception[J]. Science,2000,290(5500):2268-2269.

[17] TENENBAUM J B,DE SILVA V,LANGFORD J C. A global geometric framework for nonlinear dimensionality reduction[J]. Science,2000,290(5500):2319-2323.

[18] ROWEIS S T, SAUL L K. Nonlinear dimensionality reduction by locally linear embedding [J]. Science, 2000,290(5500):2323-2326.

[19] 徐嘉明,张卫强,杨登舟,等. 基于流形正则化极限学习机的语种识别系统[J]. 自动化学报,2015,41(9): 1680-1685.

[20] 李冬辉,闫振林,姚乐乐,等. 基于改进流形正则化极限学习机的短期电力负荷预测[J]. 高电压技术, 2016,42(7):2092-2099.

[21] TOMAR V S, ROSE R C. Manifold regularized deep neural networks[C]//Proceedings of the 15th Annual Conference of the International Speech Communication Association. Grenoble, France: International Speech Communication Association,2014:348-352.

[22] GUAN Naiyang,TAO Dacheng,LUO Zhigang,et al. Manifold regularized discriminative nonnegative matrix factorization with fast gradient descent [J]. IEEE Transactions on Image Processing,2011,20(7):2030-2048.

[23] JIANG Mingyang, LIANG Yanchun, FENG Xiaoyue, et al. Text classification based on deep belief network and softmax regression [J]. Neural Computing and Applications,2018,29(1):61-70.

编辑 樊丽娜

(上接第 241 页)

[7] LING Guang,LYU M R,KING R. Ratings meet reviews,a combined approach to recommend [C]//Proceedings of the 8th ACM Conference on Recom-mender Systems. New York,USA:ACM Press,2014:105-112.

[8] CHEN Zhiyuan,LIU Bing. Lifelong machine learning[M]. San Rafael,USA:Morgan and Claypool Publishers,2016.

[9] HU Guangneng,DAI Xinyu,SONG Yunya,et al. A synthetic approach for recommendation: combining ratings, social relations, and reviews [C]//Proceedings of the 24th International Conference on Artificial Intelligence. New York, USA:ACM Press,2015:1756-1762.

[10] THRUN S. Is learning the n-th thing any easier than learning the first[C]//Proceedings of the 8th International Conference on Neural Information Processing Systems. Cambridge, USA:MIT Press,1996:640-646.

[11] FEI Geli,WANG Shuai,LIU Bing. Learning cumulatively to become more knowledgeable[C]//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA: ACM Press,2016:1565-1574.

[12] SHU Lei, LIU Bin, XU Bin, et al. Lifelong-rl: lifelong relaxation labeling for separating entities and aspects in opinion targets [EB/OL]. [2018-03-24]. https:// www. aclweb. org/anthology/D16-1022.

[13] CHEN Zhiyuan,LIU Bing. Topic modeling using topics from many domains, lifelong learning and big data [C]// Proceedings of International Conference on Machine Learning. New York,USA:ACM Press,2014:703-711.

[14] KAPOOR A, HORVITZ E. Principles of lifelong learning for predictive user modeling [C]//Proceedings of International Conference on User Modeling. Berlin, Germany:Springer,2007:37-46.

[15] LIU Qian,LIU Bing,ZHANG Yuanlin,et al. Improving opinion aspect extraction using semantic similarity and aspect associations [EB/OL]. [2018-03-24]. https:// aaai. org/ocs/index. php/AAAI/AAAI16/paper/view/11973/12051.

[16] BLEI D M,NG A Y,JORDAN M I. Latent dirichlet allocation[J]. Journal of Machine Learning Research, 2003,3:993-1022.

编辑 樊丽娜