doi:10.3772/j.issn.2095-915x.2019.02.004

面向出版社富媒体知识的文本分类研究

刘琼昕[1,2] 宋祥[2,3] 王鹏[2,3]

1. 北京市海量语言信息处理与云计算应用工程技术研究中心 北京 100081；
2. 北京理工大学 计算机学院 北京 100081；
3. 中国科学技术信息研究所 富媒体数字出版内容组织与知识服务重点实验室

开放科学（资源服务）标识码（OSID）

摘要：大数据环境下，出版行业面临着富媒体数据带来的跨媒体数据组织和海量历史数据的挑战。为了形成有效的知识组织，针对富媒体出版社的文本数据具有数据量巨大、标签分层级的特点，本论文使用截断奇异值分解进行降维，应用线性分类核支持向量机模型，并且设计了多层级分类方法，对富媒体文本进行文本分类。实验表明，在富媒体出版社的文本数据下，本文方法取得了较好的文本分类结果。在 150 维的文本特征下，区域分类的第二级分类效果最好，其中准确率达到 0.98，召回率达到 0.76，F1 指标达到 0.87。

关键词：富媒体；文本分类；支持向量机；降准

中图分类号：TP391 G35

Research on the Processing of Rich Media Knowledge for Publishers

LIU Qiongxin[1,2] SONG Xiang[2,3] WANG Peng[2,3]

1. Beijing Engineering Applications Research Center on High Volume Language Information Processing and Cloud Computing, Beijing 100081, China;
2. School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China;
3. The Key Laboratory of Rich-media Knowledge Organization and Service of Digital Publishing Content Institute of Scientific & Technical Information of China

Abstract: The publishing industry faces the challenge of cross-media data organization and massive historical data brought by rich media data in big data area. The text data for rich media publishing houses has the characteristics of huge data and hierar-

基金项目：富媒体数字出版内容组织与知识服务重点实验室开放基金项目（ZD2018-07/02）：“富媒体数字出版内容的知识挖掘及发现技术研究”。

作者简介：刘琼昕（1972-），博士，副教授，研究方向：自然语言处理、人工智能、知识管理，E-mail：summer@bit.edu.cn；宋祥（1994-），硕士研究生，研究方向：自然语言处理、机器学习；王鹏（1995-），硕士研究生，研究方向：自然语言处理、机器学习。

chical labels. In order to form an effective knowledge organization, this paper uses TSVD to reduce dimensionality, applies LinearSVM model, and designs Multi-level classification method for text classification of rich media texts. Experiments show that under the texts of rich media, our method has achieved good results. Under the 150-dimensional text feature, the second-level effect of regional classification is the best, with the accuracy rate reaching 0.98, the recall rate reaching 0.76, and the F1 index reaching 0.87.

Keywords: Rich media; text classification; SVM; reduce dimension

引言

随着互联网技术的飞速发展，各行各业都受到新兴技术的影响，开始不断普及各种新媒体。对于出版社领域，传统的纸质载体的垄断地位被打破，来源于互联网的富媒体大数据带来了巨大的挑战和机遇[1]。在当前富媒体大数据环境下，出版社领域不仅面临着多模态数据、跨媒体数据组织的挑战，同时还面临着海量历史数据、信息多样性、信息的快速生成以及用户实时服务的要求[2]。对富媒体资源进行必要的加工、分析富媒体数据，从而形成有效的知识组织是实现工作效率提高的捷径。

论文以富媒体数字出版社的知识为研究对象，针对数字出版社领域的富媒体知识的特征，应用机器学习等领域的相关知识，对富媒体文本进行文本分类，实现了富媒体知识的整理和分析。

1 相关背景知识

国外学者最早在二十世纪五十年代末开始对文本分类进行研究。最早是通过半人工的方式实现。常用的手段是通过定义一些规则，然后依据制定的规则进行文本分类。随着网络的发展，信息飞速增长，文本分类的重要度逐渐增大，学者对其的探索也不断加深，半人工的方式被淘汰。

目前，文本分类算法已经非常成熟，常见的机器学习方法有 K 邻近算法[3]、贝叶斯方法[4]等，具有较为完备的理论基础。1995 年，Vapnik 提出了 SVM 分类算法[5]，核心思想是确定一个最优的分类超平面，所有样本点到该超平面的间隔最大。这个超平面是一个具有“硬间隔”（hard margin）的划分超平面，即所有样本点都必须被该超平面正确划分。由于真实世界的数据具有噪声，简单的线性分类难以有较好的结果，于是产生了“软间隔”（soft margin），允许一些样本出错。对于非线性可分的训练数据，SVM 利用“核函数”（kernel function），使数据在一个新的高维向量空间上实现线性可分[6]。由于当样本较大时，SVM 变量较多，训练花费时间较多，算法会非常低效，目前学者提出了多种提升速度的求解方法，其中常用的是 Platt 提出的启发式算法——序列最小化（SMO）算法[7]。由于现实任务中文本分类不仅包括二分类问题，还包括多分类问题，学者研究出包括 One-vs-Rest、One-vs-One、

doi:10.3772/j.issn.2095-915x.2019.02.004

DAGSVM 等[8]在内的 SVM 的改进算法，使其可以支持多分类。还有学者研究，SVM 算法还可以应用到回归领域中。

Meng 等人提出了一种两阶段特征选择方法，接连使用特征贡献度和潜在语义索引（LSI）两种特征选择方法，在特征选择结束后，将特征向量输入到支持向量机（SVM）中[9]。实验显示，两阶段特征选择方法在垃圾邮件数据集上的表现较好。Uguz 提出了两种不同的两阶段特征选择方法，使用信息增益，PCA 和遗传算法[10]。首先使用信息增益进行特征选择，然后使用 PCA 或遗传算法实现维度降低。这两种方法的输出分别输入到 KNN 和 C4.5 决策树分类器。这种方法能够在精确率、召回率和 F1 指标等评价指标中得到较高结果。在另一项研究中，Uguz 又提出了新的两阶段特征选择方法，使用卡方特征选择，PCA 和粒子群优化（PSO）。首先使用卡方特征选择方法执行初始降维，然后应用 PCA 或 PSO 方法，最后将这两种方法的结果输入 KNN 和 C4.5 决策树进行分类[11]。Haltas 等人分析了四种流行的启发式搜索算法在文本分类中的效果，分别是遗传算法、PSO、进化搜索和 TABU 搜索[12]。首先使用信息增益的方法进行初始降维。然后使用四种流行的启发式搜索算法实现第二次降维，最后将得到的特征向量输入朴素贝叶斯分类器进行分类。根据 F1 指标，TABU 搜索算法的性能略优于其他三种搜索算法。Wang 等人提出了一种新的文本分类的两阶段特征选择方法，它结合了类别相关度特征选择和 LSI，接连通过类别相关度特征选择和 LSI，实现降维，最后将特征向量输入 SVM 分类器进行分类[13]。

集成方法也常用于文本分类中。Larkey 和 Croft 最早将集成方法应用到文本分类中，他们使用了三种分类器作为集成基分类器，分别是 KNN、相关反馈和贝叶斯分类器，在医疗文本上进行分类[14]。Dong 和 Han 使用了三种不同的朴素贝叶斯变种模型和 SVM 分类器，比较了六种不同的同质集成和一种异质集成的分类器的效果[15]。Fung 等使用动态加权的方式来组合异构集合分类器[16]。Liu 等提出了成对集成的方法，比现在常用的 ECOC 方法有更好的性能[17]。Gangeh 等人应用随机子空间方法去解决文本分类问题，重点讨论了提交给基本分类器的每个随机子空间的集合参数和维数大小[18]。Elghazel 等人提出了一种新的多标签文本分类算法——多标记旋转森林（MLRF），该方法以旋转森林和潜在语义索引为基础[19]。对于情感分析，Kanakaraj 提出了一个集合分类器，将几种分类器的进行组合，实现对基础分类器的改进[20]。Onan 等人对由关键字表示的 Twitter 文本进行评估，使用了四种不同基本分类器和五种不同的集成方法[21]。

有监督的深度学习网络具有高级抽象能力并有更好的分类准确度。Kalchbrenner 等人使用了一种动态卷积网络，通过动态最大池化层，对线性序列进行全局池化操作，实现对电影评论的情感分类[22]。Johnson 和 Zhang 提出了一种用于文本分类的半监督卷积网络，用于学习小文本区域的特征[23]。Joulin 等提出了一种简单有效的基线分类器，它在准确性与深度学习分类器一样好，但运行速度更快[24]。Conneau 等提出了一种称为 VDCNN 的新架构，直接在字符级别进行操作，并且仅使用小范围的卷积

和池化操作 [25]。Kowsari 等提出了一种新的分层文档分类方法，称为 HDLTex，采用多种深度学习方法来产生分层分类 [26]。Selamat 和 Omatu 提出了新闻网页分类方法（WPCM），它基于 PCA 算法和类别特征，对体育新闻数据集有较好的分类准确率 [27]。

2 模型介绍

文本分类基本流程分为三个阶段，分别是文本预处理、特征工程和分类器，每个阶段需要设计相应的流程。论文应对富媒体文本知识特点，设计了对应的流程，具体参见文本分类流程图（图 1）。

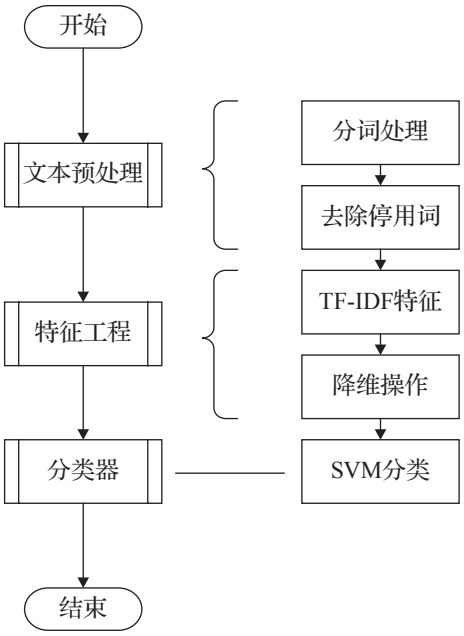

图 1 文本分类流程图

基于富媒体知识的文本分类参见算法 1，其中，模型的输入是全部富媒体文本分类数据。

算法 1：文本分类
输入：文本分类实验数据 *D*。 输出：模型的评价指标：准确率 P、召回率 R、F1 指标。 1. 将实验数据 *D* 切分为训练集 *Train* 和测试集 *Test*。 2. 对训练集 *Train* 进行分词，并去除停用词，得到词矩阵 *M*。 3. 计算词矩阵 *M* 的 TF-IDF 特征，得到训练集 *Train* 的 TF-IDF 特征矩阵 *X*。 4. 使用截断奇异值分解降维，降成 150 维度的特征矩阵 $\overline{X}$。 5. 使用线性核多分类的 SVM 对特征矩阵 $\overline{X}$ 进行训练。 6. 在测试集 *Test* 上测试模型，得到评价指标 P、R、F1。

2.1 文本预处理

在文本预处理阶段，需要对文本进行分词。然后对产生的结果集去除停用词，包括标点符号和类似“这”、“那”、“是”、“等”的一系列词语。初始文本 S=[s_1, s_2, s_3, …, s_n]，其中 s_i 是文本中的句子，$i\in[1,n]$，n 是文本的句子总数。

分词是文本分类的关键一步。论文使用 jieba 分词工具进行中文分词 [28]，分词模式使用默认分词，并且使用隐马尔可夫（HMM）模型。分词的第一步是实现对长文本的分割，通过正则表达式匹配字符串，产生短句集合。第二步是通过 jieba 分词库的内置统计字典，对短句集合，构造词语集合，形成一个有向无环图（DAG）。第三步，通过动态规划算法，计算出最大概率路径，得到最佳分词方案。第四步，采用隐马尔可夫（HMM）模型，对连续的单字进行合并，形成词语。最终，经分词后，生成文本 S 的词集 W，结果参见（公式 1）。

$$W=\{w_1^1,w_2^1,w_3^1,\cdots,w_{j1}^1,\cdots,w_1^k,w_2^k,w_3^k,\cdots,w_{jk}^k\} \quad \text{（公式 1）}$$

其中，$w_1^1,w_2^1,w_3^1,\cdots,w_{j1}^1$ 是句子的分词 s_1 集合，

doi:10.3772/j.issn.2095-915x.2019.02.004

$w_1^k, w_2^k, w_3^k, \cdots, w_{jk}^k$是句子 s_k 的分词集合，j_k 是句子 s_k 的分词结果的个数。

分词之后要对词集 W 进行汇总过滤，如果词w_j^i在停用词表 T 中出现，则认为该词语是停用词，去除；若没有出现，则认为该词语是有意义的词语，保留。最终形成过滤后的词集$\overline{W}$，具体参见（公式 2）。

$$\overline{W} = \{w_j^i \mid w_j^i \in W \cap w_j^i \notin T\} \quad （公式 2）$$

其中，T 表示停用词表的词集合。

2.2 特征工程

在特征工程阶段，需要对预处理的词转化为 TF-IDF 特征矩阵。TF-IDF 特征矩阵中每个元素是对应词语的词频（TF）与逆文档频率（IDF）的乘积 [29]。具体参见（公式 3）

$$X = \mathrm{TF} * \mathrm{IDF}\ x_{ij} \in X \cap x_{ij} = tf(w_j^i) * idf(w_j^i) \cap w_j^i \in \overline{W} \quad （公式 3）$$

其中，x_j^i是特征矩阵 X 的第 i 行第 j 列值，$tf(w_j^i)$是词语w_j^i的 TF 值，$idf(w_j^i)$是词语w_j^i的 IDF 值，$tf(w_j^i) * idf(w_j^i)$即是词语w_j^i的 TF-IDF 值。

考虑到富媒体文本中词语数量巨大的特点，TF-IDF 特征矩阵维度会非常巨大，维度数达到千到万级别，如果直接使用，特征矩阵在内存的存取和读取的时间代价太大，会严重影响分类结果和算法效率，甚至无法运行。为了解决这种问题，论文使用截断奇异值分解（TSVD）进行降维，降低特征矩阵的维度。

截断奇异值分解（truncated singular value decomposition，简称 TSVD）是一种矩阵因式分解（factorization）技术。与在数据的协方差矩阵上进行分解 PCA 相似，SVD 分解在数据矩阵上进行，并且不要求输入矩阵必须为方阵。截断奇异值分解，可以把一个矩阵分解为三个矩阵的乘积，并且第二个矩阵中对角线元素是该矩阵的特征值。通过选择最大的 k 个特征值，并取三个矩阵对应列构成新矩阵，连乘后得到的新矩阵的维度减小，即实现了降维 [30]。截断奇异值分解使用前 k 大的奇异值近似代替原矩阵，具体计算公式参见（公式 4）。

$$X \approx X_k = U_k \Sigma_k V_k^T \quad （公式 4）$$

其中，X 表示原特征矩阵，X_k 表示使用 TSVD 产生的降维后的的近似特征矩阵，k 表示降维后的维度，U、Σ、V 分别是特征矩阵的奇异值分解结果，Σ_k 是在 Σ 的对角线元素中截取前 k 个最大的奇异值构成的矩阵，U_k 是 U 截取相应的 k 列构成的矩阵，V_k^T 是 V 截取相应的 k 列构成的矩阵。

2.3 分类器

分类器是文本分类的核心。富媒体文本分类是一个多分类多层级分类问题。对于多分类，由于富媒体数据量巨大，论文选择使用支持向量机（SVM）模型 [5]，考虑到运行速度和运行效果，对于 SVM 模型，使用线性分类核。对于数据集较小的任务，使用线性分类核的分类效果没有使用复杂核函数的分类效果要好；但是对于数据集巨大的任务，使用复杂核函数的分类器会产生映射到维度巨大的高维空间，模型花费时间难以接受，甚至无法计算的问题。而对于线性分类核，由于数据量巨大，分类器灵活，分类效果不仅不会下降，甚至比复杂核函数的分类器效果更好。实验证明多分类采用线性分类核也可以产生较好的正确率，同时花费时间也在可接受的范围内。

富媒体文本分类还是一个多层级分类。实

验采用的方法是对每个层级使用 SVM 进行分类。关键在于如何确定每个层级标签对应的样例。以中图分类为例，假设在第一层级分类的时候，选出中图分类中所有层级号为 1 的类目，得到“哲学教育与普及”、“社会科学总论”等。接着与下级标签匹配，如“C0”、“C01”、“C18”、“C289”等下级标签都归类到“C”标签，即得到标签为“C”的的全部类目。最后在文本的第一层级分类的时候，使用全部标注为“C”标签的数据作为一个类目，进行训练。具体参见（公式 5）。

$$M^i=S^i(X_k, L^i) \qquad \text{（公式 5）}$$

其中，i 是层级数，表示第 i 层级，M^i 表示第 i 层级的模型，S^i 表示第 i 层级的 SVM 模型，X_k 是输入的特征矩阵，L^i 是第 i 层级的标签集合，$L^i=\{l_1^i, l_2^i, l_3^i,\ldots, l_j^i\}$，$l_j^i$ 表示第 i 层级的标签，一共有 j 个。

富媒体文本分类包括两部分，第一部分是训练模型，第二部分是使用训练好的模型进行预测。训练算法参见算法 2，其中，训练文件 *Train* 包含样例的标签和数据，标签文件包含全部标签层级信息，分类项目 *Classficaton* 是具体的分类类别，需在区域分类、学科分类、中途分类和行业分类四类中指定之一，停用词文件 *Stopword* 即停用词表，层级 *Layer* 是指需要进行某层级的文本分类。

算法 2：文本分类训练算法
输入：有标注的训练文件 *Train*，皮书文本的标签文件 *Label*，分类项目 *Classificaton*，停用词文件 *Stopword*，层级 *Layer*。 输出：训练模型 *Model*。 1. 从训练文件 *Train* 中提取出 *Train_label*，*Train_txt*。 2. 从标签文件 *Label* 中提取 *Classificaton* 类、*Label* 层级的标签信息，保存为 *Label_dict*。 3. 调用 jieba 分词工具，通过停用词文件 *Stopword* 去除停用词，得到训练词矩阵 *Train_word*。 4. 使用 TF_IDF 特征，把 *Train_word* 表示为特征矩阵 *x*。 5. 调用 Numpy 库，对特征矩阵进行降维到 150，得到新特征矩阵$\overline{x}$。 6. 使用 sklearn 库把 *Label_dict* 中的标签表示为 *multilaber* 形式的标签 *y*。 7. 调用 sklearn 库中 LinearSVM 模型对特征矩阵$\overline{x}$和标签 *y* 进行训练。 8. 把训练模型 *Model* 保存。

测试算法参见算法 3，其中，是测试文件，分类项目是具体的分类类别，需在区域分类、学科分类、中途分类和行业分类四类中指定之一，停用词文件即停用词表，训练模型是训练结束后保存的模型，皮书文本的标签文件是测试文件的真实标签，层级指定需要在某个层级进行测试。

算法 3：文本分类测试算法
输入：文本分类测试文件 *Test*，分类项目 *Classificaton*，停用词文件 *Stopword*，训练模型 *Model*，皮书文本的标签文件 *Label*，层级 *Layer*。 输出：模型的评价指标：准确率 P、召回率 R、F1 指标。 1. 从测试文件 *Test* 中提取出 *Test_label*，*Test_txt*。 2. 从标签文件中 *Label* 提取 *Classificaton* 类、*Layer* 层级的标签信息，保存为 *Label_dict*。 3. 调用 jieba 分词工具，通过停用词文件 *Stopword* 去除停用词，得到测试词矩阵 *Test_word*。 4. 加载训练模型 *Model*，输入测试词矩阵 *Test_word*，得到预测结果 *y_test*。 5. 通过真实标签 *Test_label* 和预测结果 *y_test*，计算准确率 P、召回率 R、F1 指标。

doi:10.3772/j.issn.2095-915x.2019.02.004

3 实验与结果分析

3.1 数据集

文本分类数据集是富媒体数字出版社中有关“皮书”的文本信息。“皮书”是近年来由社会科学文献出版社推出的一系列大型图书，由全国的众多权威研究报告组成，目的是分析国内外的社会、经济、政治等领域的现状，预测这些领域在未来的发展状况。文本分类的皮书数据包括四类，分别是中图数据、学科数据、区域数据和行业数据。数据的具体统计信息参见皮书数据说明（表 1）。

表 1 皮书数据说明

类型	数据量（条）	标签数量（条）	层级数
中图数据	1000000	27964	6
学科数据	1000000	3142	5
区域数据	1000000	503	3
行业数据	1000000	1424	4

在皮书数据中，存在许多标签为“0”的数据，代表着数据标签无效。因此，在训练和测试时都需对该标签数据进行特殊处理。在训练中，采取的方式是直接剔除该标签的数据；在测试中，由于分类的标签中“0”占了较大比重，所以单纯的将答案和结果比较容易得出很高的重合率，所以在测试时，不采用把与标签相等的样例作为正确分类的样例的方法，而使用 sklearn 库的进行评估，去除“0”标签。评估指标三项，分别是：准确率（precision）、召回率（recall）和 F1 指标（f1-score）。

3.2 实验结果

实验中第 0 级代表不考虑层次信息，一个标签就是一个类别。降维时特征维数选取 150，效果较好。对于 150 维特征，实验的详细预测结果参见评估结果（表 2~ 表 5）。

表 2 区域分类评估结果

	准确率	召回率	F1 指标
第 0 级	0.31	0.07	0.10
第 1 级	0.97	0.67	0.79
第 2 级	0.98	0.76	0.87
第 3 级	0.32	0.12	0.17

表 3 学科分类评估结果

	准确率	召回率	F1 指标
第 0 级	0.00	0.00	0.00
第 1 级	0.48	0.38	0.36
第 2 级	0.05	0.07	0.06
第 3 级	0.15	0.05	0.07

表 4 中图分类评估结果

	准确率	召回率	F1 指标
第 0 级	0.00	0.00	0.00
第 1 级	0.79	0.64	0.65
第 2 级	0.58	0.37	0.38

表 5 行业分类评估结果

	准确率	召回率	F1 指标
第 0 级	0.08	0.08	0.08
第 1 级	0.64	0.46	0.51
第 2 级	0.28	0.31	0.29
第 3 级	0.33	0.25	0.28

由于为给定测试集，需要在数据集中随机挑选样本作为测试集，论文采用多次抽样进行指标评估。具体方法是每次随机抽取 50 条数据，进行准确率、召回率、F1 指标的计算，将数据放回，多次抽样，将所有结果取平均值得到最

后的宏平均评价指标。

具体挑选中，需要过滤掉“0”标签。例如测试数据为 50 条，假设其中对于区域分类具有无效标签“0”的数据有 3 条，所以实际测试数据为 47 条，学科分类、区域分类、中图分类和行业分类也是类似的处理。

实验发现，在中图分类、学科分类、区域分类和行业分类四个类别中，区域分类的第二级准确率最高，达到 0.98；区域分类的第二级召回率最高，达到 0.76；区域分类第二级 F1 指标最高，达到 0.87。整体结果中，区域分类的效果最好，原因可能是区域分类中特征包含区域信息较多，且区别度较大，经降维后与第一级、第二级标签相关的特征信息保留最多；学科分类的效果最差，可能降维后特征信息保留较少，特征的区别度不是很大。

对于第 0 级标签（即不考虑层级，所有标签都是单独的类别），四个分类的结果都不好。这是因为富媒体数据标签数量巨大，例如中图分类标签数达到了 27964 条，导致分类器的分类效果不好。从第 1 级开始，随着层级数增加，对每个分类，发现各个评估指标逐渐下降，这是因为随着标签的逐渐细分，对应的分类数据量大幅度降低，如中图分类标签到第 3 级经过降维后就已经没有特征，无法进行评估。

4 结语

在对富媒体知识分析处理中，本文的文本分类方法取得了较好的效果，在 150 维的文本特征下，区域分类的第二级分类效果最好，其中准确率达到 0.98，召回率达到 0.76，F1 指标达到 0.87。文本分类算法的应用可以实现自动化的图书分类，减少了手工进行文献分类所有付出的人力物力；也可运用于出版社领域的信息检索。

应对富媒体数据的特征，在文本分类中，论文设计并实现了相应的处理方案。在特征工程阶段，考虑到富媒体文本中词语数巨大，论文使用截断奇异值分解（TSVD）进行降维，降低特征矩阵的维度；在分类器阶段，由于富媒体文本分类是一个多分类多层级分类，论文对于多分类目标使用线性核支持向量机，对于多层级分类，采用的方法是对每个层级使用 LinearSVM 进行分类。

关于论文描述的文本分类算法，仍然存在着一些问题。例如，当遇到文本存在多标签问题时，论文算法的结果会减少某个文本的分类标签，对于如何解决多标签分类问题，还需要进一步的研究；随着层级的加深，标签的训练样本数大幅减少，训练效果急剧下降，此时可以考虑添加额外的信息，以保证分类效果。根据网络中给出的资料，针对这两个问题，现在已经有很多改进算法，将来的工作是在现有的基础上尝试改进算法，以达到更好的效果。

参考文献

[1] 刘倩 . 富媒体时代传统出版面临的挑战及出路 [J]. 传媒论坛 , 2018, 1(20):159.
[2] 周山丹 . 媒体出版社面临的挑战与机遇 [J]. 编辑之友 , 2005(6):25-26.
[3] Peterson L. K-nearest neighbor[J]. Scholarpedia, 2009, 4(2):1883.
[4] Geman S, Geman D. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images[J]. Readings in Computer Vision, 1987, 20(5-

doi:10.3772/j.issn.2095-915x.2019.02.004

6):25-62.
[5] Chen P H, Lin C J, Schölkopf, Bernhard. A tutorial on v-support vector machines[J]. Applied Stochastic Models in Business & Industry, 2005, 21(2):111-136.
[6] Campbell C. Kernel methods: a survey of current techniques[J]. Neurocomputing, 2002, 48(1-4):63-84.
[7] Shevade S K, Keerthi S S, Bhattacharyya C, et al. Improvements to the SMO algorithm for SVM regression[J]. IEEE Transactions on Neural Networks, 2000, 11(5):1188-11931193.
[8] Joutsijoki H, Juhola M. DAGSVM vs. DAGKNN: An experimental case study with benthic macroinvertebrate dataset[C]. International Conference on Machine Learning & Data Mining. 2012.
[9] Meng J, Lin H, Yu Y. A two-stage feature selection method for text categorization[J]. Computers & Mathematics with Applications, 2011, 62(7):2793-2800.
[10] U?Uz H. A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm[J]. Knowledge-Based Systems, 2011, 24(7):1024-1032.
[11] Harun Uğuz. A hybrid approach for text categorization by using x2 statistic, principal component analysis and particle swarm optimization[J]. Scientific Research and Essays, 2013, 8(37):1818-1828.
[12] Haltas A, Alkan A, Karabulut M. Performance analysis of heuristic search algorithms in text classification[J]. Journal of the Faculty of Engineering and Architecture of Gazi University, 2015, 30(3): 417–427.
[13] Wang F, Li C H, Wang J S, et al. A two-stage feature selection method for text categorization by using category correlation degree and latent semantic indexing[J]. Journal of Shanghai Jiaotong University (Science), 2015, 20(1):44-50.
[14] Larkey L S, Croft W B. Combining Classifiers in Text Categorization[C]. ACM, 1996.
[15] Dong Y S, Han K S. A comparison of several ensemble methods for text categorization[C]. IEEE International Conference on Services Computing. 2004.
[16] Fung G, Yu J, Wang H, et al. A balanced ensemble approach to weighting classifiers for text classification[C]. ICDM 2006 Proceedings of the Sixth International Conference on Data Mining ACM, 2007:869-873.
[17] Liu, Carbonell J, Jin R. A New Pairwise Ensemble Approach for Text Classification[C]. European Conference on Machine Learning, 2003:277-288.
[18] Gangeh M J, Kamel M S, Duin R P W. Random Subspace Method in Text Categorization[C]. International Conference on Pattern Recognition. IEEE Computer Society, 2010: 2049-2052.
[19] Elghazel H, Aussem A, Gharroudi O, et al. Ensemble multi-label text categorization based on rotation forest and latent semantic indexing[J]. Expert Systems with Applications, 2016(57):1-11.
[20] Kanakaraj, Guddeti R M R. Performance analysis of Ensemble methods on Twitter sentiment analysis using NLP techniques[C]. IEEE International Conference on Semantic Computing. IEEE Computer Society, 2015.
[21] Aytuğ Onan, Serdar Korukoğlu, Bulut H. Ensemble of keyword extraction methods and classifiers in text classification[M]. Oxford: Pergamon Press Inc, 2016.
[22] Kalchbrenner N, Grefenstette E, Blunsom P. A Convolutional Neural Network for Modelling Sentences[J]. Eprint Arxiv, 2014, 1.
[23] Johnson R, Zhang T. Semi-supervised Convolutional Neural Networks for Text Categorization via Region Embedding[J]. Advances in Neural Information Processing Systems, 2015(28):919-927.
[24] Joulin A, Grave E, Bojanowski P, et al. Bag of Tricks for Efficient Text Classification[J]. Eprint Arxiv, 2016.
[25] Conneau A, Schwenk H, Barrault Loïc, et al. Very Deep Convolutional Networks for Text Classification[J]. Eprint Arxiv, 2017.
[26] Kowsari K, Brown D E, Heidarysafa M, et al. HDLTex: Hierarchical Deep Learning for Text Classification[J]. IEEE International Conference on Machine Learning and Applications. Eprint Arxiv, 2017.
[27] Selamat A, Omatu S. Web page feature selection and classification using neural networks[J]. Information Sciences, 2004, 158: 69-88.
[28] 张小欢 . 中文分词系统的设计和实现 [D]. 成都 : 电子科技大学 , 2010.
[29] 施聪莺 , 徐朝军 , 杨晓江 . TFIDF 算法研究综述 [J]. 计算机应用 , 2009, 29(b6):167-170.
[30] Fierro R D, Hansen P C. Accuracy of TSVD solutions computed from rank-revealing decompositions[J]. Numerische Mathematik, 1995, 70(4):453-471.