文章编号：1003-0077(2016)05-0176-11

# 一种半监督的中文垃圾微博过滤方法

姚子瑜，屠守中，黄民烈，朱小燕

（清华大学 计算机科学与技术系，北京 100084）

**摘 要**：微博作为目前国内外最活跃的信息分享平台之一，其中却充斥着大量的垃圾内容。因此，如何从给定话题的微博数据中，过滤掉与话题不相关的垃圾微博、保留话题相关微博，成为迫切需要解决的问题。该文提出了一种半监督的中文微博过滤方法，基于朴素贝叶斯分类模型和最大期望算法，实现了利用少量标注数据的垃圾微博过滤算法，其优势是仅仅利用少量标注数据就可以获得较为理想的过滤性能。分别对十个话题 140 000 余条新浪微博数据进行过滤，该文提出的模型准确度和 F 值优于朴素贝叶斯和支持向量机模型。

**关键词**：垃圾微博过滤；半监督学习；EM 算法；朴素贝叶斯

**中图分类号**：TP　　**文献标识码**：

## A Semi-supervised Method for Filtering Chinese Spam Tweets

YAO Ziyu, TU Shouzhong, HUANG Minlie , ZHU Xiaoyan

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)

**Abstract**: Microblogging sites are one of the most popular information sharing platforms today. However, among the large amount of posted published every day, spam texts are seen everywhere: users utilize spam posts to advertise, broadcast, boast their own products, and defame their competitors. Therefore, filtering spam tweets is a critical and fundamental problem. In this paper, we propose a semi-supervised algorithm based on Expectation Maximization and Naive Bayesian Classifier (EM-NB), which is able to filter spam tweets effectively using only a small amount of labeled data. The experimental results on more than 140 thousand tweets from Sina Weibo show that our method achieves higher accuracy and F-score than baselines.

**Key words**: spam tweet; naive bayesian classifier; expectation maximization; semi-supervised learning

## 1 引言

微博(Microblog)是一种基于用户关系的短文本信息分享平台。根据文献[1]统计显示，截止到 2014 年 8 月，推特上注册用户数达到近十亿，月活跃用户达 2.71 亿。微博已经成为互联网用户获取和传递信息的重要平台。微博中的博文或推文(Tweets)，涵盖了多个话题，涉及经济、政治、科技、娱乐等多个领域。用户在微博上获取的信息很大一部分来源于微博的话题搜索功能。从“微博搜索”功能搜索下来的话题微博，具有优良的话题实时性和实效性，为用户提供了大量具有高度价值的相关话题信息。然而，其中夹杂着不少包含广告信息的微博(如图 1)。

事实上，这些带有广告信息的微博，极大影响了微博用户体验。文献[2]指出，微博的低门槛、易操作以及传播影响力大等因素，使得“微博广告”的衍生成为一种必然。然而，过度泛滥的微博广告不仅降低了用户的微博体验，更影响了整个微博生态圈的发展。例如，不少用户的微博常常被陌生账号“@”、转发，同时伴有含广告信息及链接的评论；买卖“僵尸粉”成为商家牟利的常用手段；一些不正规的、含有敏感词汇的广告，也逐渐在各大微博门户网站上传播，影响了微博产业的健康发展。

参考新浪微博发布的《垃圾营销信息管理规定

收稿日期：2015-09-21　定稿日期：2016-03-20
基金项目：国家自然科学基金(61332007,61272227)


**图 1　iphone6 话题下垃圾微博示例**

征求意见稿》[3]，本文定义“垃圾微博”为：通过信息流或微博公共区域（如话题搜索页）发布的，带有售卖链接、软性植入推广或有奖营销活动宣传的微博。我们可以将垃圾微博划分为两类：

（1）广告推广类

此类微博一般有两种表现形式：①以“关键词堆积”的形式提及热搜话题，以提高广告自身的检索命中率，但其内容与话题无关。例如，搜索话题“理财产品”，可以检索到下面的微博：

“＃那些年，我们一起追过的女孩＃＃乌俄地缘局势＃＃普京讲话＃＃聪明理财＃＃理财产品＃＃最火理财产品＃＃如何理财”。

这条微博通过堆积关键词如“那些年，我们一起追过的女孩”、“普京讲话”等均为时间段内的热搜词，提高了广告被检索到的概率，但微博内容本身与检索话题无关，是商家实现微博营销的一种常用手段。②微博话题本身就是一种销售产品，或自身不是销售产品、但具有潜在产业链的情况下，微博文本虽然与话题相关，但是提供的均为销售信息。例如，搜索话题“iphone5”，可以检索到：

“iPhone5S 情侣彩绘 iphone4s 手机壳 iphone5 手机壳卡通苹果 5S 外壳 32 包邮”。

由于“iphone5”本身就是一种销售产品，且它含有很广泛的产业链（手机壳，手机贴膜等），因此，搜索“iphone5”话题时，往往会含有大量的广告微博。

（2）有奖营销类

此类微博主要表现为，存在大量的“活动宣传”词汇，如“中奖”、“抽奖”、“好礼”等；含有＃（Hashtag）符号，尤其在微博的“微话题”中最常见。例如，搜索话题“环保”，可以检索到诸如“＃到国美购海尔健康环保＃ 521 期待人品大爆发，大奖小奖统统搬回我的家！”和“＃环保＃祝小编天天开心，也祝贵博粉丝多到爆，越来越红火！”的微博。这两条微博均为微博上的“微话题”活动，以一对＃符号开头，正文含有“大奖”“小奖”等词，表现出“期望在活动中赢取大奖”的正向情感。

从新浪微博随机爬取的话题微博数据显示，手机产品的微博中，垃圾微博占比高达 70％！因此，从相关话题的微博中过滤垃圾微博、筛选出非垃圾微博，逐渐得到了学术界和工业界的关注。TREC 从 2011 年新增了微博任务开始，一直将给定话题的实时微博信息筛选作为其任务之一。

垃圾微博的过滤工作，可以转化为微博的文本分类工作。然而，微博文本具有内容短小、用语不规范、大量社会化内容等特点，使得传统的长文档的分类方法不再适用。同时，在给定话题下，垃圾微博与非垃圾微博往往具有“区分度低”、“主题接近”的情况，也给垃圾微博过滤带来了极大的挑战。此外，由于人工标注数据获取代价高，往往需要耗费大量人力物力，而无标注数据易于获取、数量大，因此，在实

现文本分类时，我们需要尽可能地减少人工标注数据量，充分利用无标注数据。这为垃圾微博过滤问题带来了更大的挑战。

本文针对给定话题的垃圾微博过滤问题，提出了基于朴素贝叶斯分类器(Naive Bayesian Classifier)和最大期望(Expectation Maximization)算法的半监督中文垃圾微博过滤模型。本文提出的方法，将垃圾微博过滤问题转化为二分类问题，即将微博分为“垃圾微博”和“非垃圾微博”两类；同时，使用半监督的学习方法，仅需要人工标注少量的样本作为训练数据，充分利用未标注数据，迭代地扩充了分类器中的特征，自动地生成新的分类器，极大缓解了人工标注样本的困难。另外，考虑到同一话题下，“垃圾微博”与“非垃圾微博”在主题和词分布上的接近，本文模型使用的训练数据均为同一个话题下的正负样本，从而挖掘出更细粒度、更具代表性的特征词汇用于分类。最终，本文提出的方法在十个微博话题、十四万余条新浪微博文本的测试数据上，对每个话题分别进行五组不同训练集标注量的实验，实现了具有较高准确度和F值的微博过滤工作，且效果优于经典的朴素贝叶斯分类模型和支持向量机模型。

## 2 相关工作

近年来，国内外学者针对虚假评论做了大量的研究。研究人员针对虚假评论的研究主要分为两个方面：对虚假评论文本内容的研究和对评论发布者的特征研究。Jindal和Liu[4-5]将评论站点上的虚假评论分成三种类型：对特定产品的不真实的评论、对品牌的评论以及不带情感信息的评论。对于第二和第三种类型的虚假评论，他们标注了一部分数据集，用有监督学习的方法来识别虚假评论。实验结果发现这类虚假评论比较容易识别。对于第一种类型的虚假评论，他们假设重复的评论都是虚假评论，将虚假情感识别问题转化成重复评论识别任务。但是，直接使用重复评论来当作虚假评论是不合适的。Li等人[6]提出使用LDA主题模型识别旅店评论中的虚假评论，在800条旅店评论测试集上获得了非常高的准确率，但是这种有监督的方法需要较多人工标注，并不适用于实际应用。针对这个问题，文献[7]提出了一种检测欺诈性评论的半监督方法。在只标注正样本的情况下，作者首先获得可靠的负样本，后基于支持向量机模型和LDA主题模型对测试文本进行分类。另一个角度，研究人员主要考虑从评论发布者的特征属性来判断该发布者是否是评论造假者(Opinion Spammer)。Lim等人[8]提出使用用户的行为特征来识别评论造假者，但没有考虑对应评论的文本内容特征。Wang等人[9]结合评论文本和评论发布者，提出了用户可信度(trustiness of reviewers)、评论真实性(honesty of reviews)和商店可靠性(reliability of stores)三个概念，用图模型阐述了三个概念之间的联系。

在文本分类技术方面，国外学者积极致力于寻找新的分类方法，减少文本分类对标注数据的依赖性。一方面，不少学者致力于以标注“特征”作为标注样本。Druck等人[10]于2008年提出Generalized Expectation Criteria，用标注的特征样本，学习未标注特征的类别，实现文本分类；文献[11]基于Generalized Expectation Criteria，完成了自动的特征抽取，其训练出的情感分类器在实验中效果优于用标注文本训练出来的分类器。另一方面，国内外学者提出了半监督的学习方法，试图减少样本标注量。文献[12]基于经典的朴素贝叶斯分类器与最大期望算法提出了半监督的文本分类方法，在20Newsgroups[13]的不同话题上做了30组实验，每组实验均只标注正样本，获得了较高的F值。文献[14]同样基于朴素贝叶斯分类器，但是，作者提出了一种利用未标注样本中词汇边际概率作为约束的朴素贝叶斯分类器。作者利用大量未标注数据表现出来的词汇边际概率特征，避免了标注样本量少、标注数据稀疏带来的问题。Settles在文献[15]中同时考虑了标注文本和标注特征，并在训练朴素贝叶斯分类器过程中，对标注特征加大权重。

由于微博文本相对标准数据集而言，缺少语言规范性，文本也更加短小，常规的文本分类方法很难在微博文本上得到很好的应用。此外，研究人员在评价分类模型效果时，往往采用准确度(Accuracy)作为评价标注。然而，在垃圾微博过滤的问题中，我们更关注“垃圾微博”的分类效果，希望寻找一种能够精确而高效地将“垃圾微博”从微博文本中过滤出来的方法，即一种F值较高的过滤方法。这是因为，在一般情况下，由于给定话题下的“垃圾微博”占整个话题微博的比例较小(约10%—20%)，即使分类器将所有微博均判定为“非垃圾微博”，也能达到较高的准确度，但是却难以达到良好的F值。本文提出的模型，将朴素贝叶斯分类模型应用在实际生活中，用于解决同一话题下的微博分类问题，并且结合最大期望算法迭代地挖掘未标注文本中的新的特

征，有效地降低了人工标注样本的成本，达到较高的 F 值，且易于推广到处理多个话题的实际应用中。

## 3　半监督的垃圾微博过滤算法

在实际生活中，每天的微博话题不计其数，且每个话题下的数据有各自的特点，需要分别标注训练样本。因此，采用传统的有监督的垃圾微博过滤方法，需要昂贵的大量数据标注。本文提出了一种半监督的垃圾微博过滤方法，在少量标注样本的情况下，充分利用大量的未标注数据，能够获得较高的 F 值和准确度，大大降低了微博过滤的数据标注规模，相比有监督的方法，更适用于实际应用。

### 3.1　符号定义

在文本分类中，假设：

- 文本类别 $C=\{c_1,c_2,\cdots,c_{|C|}\}$。在过滤工作中，我们只考虑“垃圾微博”和“非垃圾微博”两类，即 $C=\{c_1,c_2\}$，其中，$c_1$、$c_2$ 分别表示“垃圾微博”和“非垃圾微博”；
- 训练文本集 $D_l=\{d_1,d_2,\cdots,d_l\}$，其中 $d_i(i=1,2,\cdots,l)$ 是每一则标注的微博文本，文本 $d_i$ 对应标注类别为 $c_{d_i}\in\{c_1,c_2\}$；
- 未标注文本集 $D_u=\{d_{l+1},d_{l+2},\cdots,d_n\}$，其中 $d_j(j=l+1,l+2,\cdots,n)$ 是每一则未标注文本集微博文本；
- 词汇集 $W=\{w_1,w_2,\cdots,w_{|W|}\}$，其中 $w_m(m=1,2,\cdots,|W|)$ 是每一个词汇集中的词。

本文解决的问题，就是在已知少量标注训练集 $D_l$ 和大量未标注文本集 $D_u$ 的情况下，估计待分类文本集中，每一则文本属于各个类别的概率。

### 3.2　算法概览

算法步骤如图 2 所示。首先，在标注样本下，训练初始的朴素贝叶斯分类器 Classifier-Init（步骤 2），计算类别下出现词的概率参数 $P(w_m|c_k)$ 如公式（1）所示，记为 $P^{(0)}(w_m|c_k)$ 如式（1）所示。

$$P^{(0)}(w_m\mid c_k)=\frac{1+\sum_{i=1}^{l}N(w_m,d_i)1(c_k\mid d_i)}{|W|+\sum_{s=1}^{|W|}\sum_{i=1}^{l}N(w_s,d_i)1(c_k\mid d_i)} \tag{1}$$

$$1(c_k\mid d_i)=\begin{cases}1 & \text{if} \quad c_{d_i}=c_k\\ 0 & \text{if} \quad c_{d_i}\neq c_k\end{cases} \tag{2}$$

其中，$N(w_m,d_i)$ 表示文本 $d_i$ 中出现词 $w_m$ 的次数，$1(c_k|d_i)$ 表示文本 $d_i$ 是否属于类别 $c_k$。对于“垃圾微博”，其 $1(c_1|d_i)=1$，$1(c_2|d_i)=0$；对于“非垃圾微博”，$1(c_1|d_i)=0$，$1(c_2|d_i)=1$。考虑到零频次词汇的干扰，对公式（1）进行拉普拉斯平滑（Laplace Smooth），即加 1 平滑。

| 垃圾微博过滤模型 Spam-Weibo |
|---|
| 输入：训练文本集，未标注文本集，待分类文本集 |
| 输出：待分类文本集中的垃圾微博集合 Spam 和非垃圾微博集合 NonSpam |
| 1：Spam=∅，NonSpam=∅；<br>2：以 $D_l$ 为训练样本，在朴素贝叶斯分类模型下，训练初始分类器 Classifier-Init，计算 $P^{(0)}(w_m\|c_k)$ 如公式（1）（2）；<br>3：执行 EM-NB($P^{(0)}(w_m\|c_k)$, $D_u$) 算法，训练新的拓展分类器 Classifier-EM；<br>4：基于 Classifier-EM 模型，对待分类集中的文本进行分类，如果 $P(c_1\|d_i)>0.5$，则 Spam=Spam∪$\{d_i\}$；否则，NonSpam=NonSpam∪$\{d_i\}$；<br>5：返回 Spam 和 NonSpam。 |

图 2　算法总体流程

步骤 3，在未标注文本集下，执行 $EM\text{-}NB(P^{(0)}(w_m|c_k),D_u)$，即通过迭代地用未标注文本 $D_u$ 对初始训练结果 $P^{(0)}(w_m|c_k)$ 进行扩展，得到新的分类器 Classifier-EM；

步骤 4-5 对待分类文本集中的文本进行分类，当其属于“垃圾微博”类别的概率大于其属于“非垃圾微博”类别的概率时，判断该文本为垃圾微博文本。最后，返回过滤后的类别结果，算法结束。

### 3.3　EM-NB 算法

基于朴素贝叶斯模型和最大期望算法的 EM-NB 算法如图 3 所示。算法中涉及迭代的步骤，以上角标 $t$ 表示迭代次数。算法由两步完成。

1. Expectation 步骤（E-Step）：

根据此时的参数 $P^{(t)}(w_m|c_k)$ 及式（3），重新计算未标注文本集中每一则文本分别属于“垃圾微博”和“非垃圾微博”的概率：$P^{(t)}(c_1|d_j)$ 和 $P^{(t)}(c_2|d_j)$。注意，$P^{(t)}(c_k|d_j)$ 是一个范围在 0-1 的概率值，表示文本的概率类别。

$$P^{(t)}(c_k\mid d_j)=\frac{P^{(t)}(c_k)\prod_{h=1}^{|d_j|}P^{(t)}(w_{d_j,h}\mid c_k)}{\sum_{r=1}^{2}P^{(t)}(c_r)\prod_{h=1}^{|d_j|}P^{(t)}(w_{d_j,h}\mid c_r)}\quad k=1,2 \tag{3}$$

其中，$w_{d_j,h}$ 表示文本 $d_j$ 第 $h$ 个位置处的词，$|d_j|$ 表示文本 $d_j$ 的词数。

2. Maximization 步骤（M-Step）：

重新估计分类器的参数。首先，根据式（4）计算

类别的先验概率$P^{(t+1)}(c_k)$：

$$P^{(t+1)}(c_k)=\frac{\sum_{i=1}^{l}1(c_k\mid d_i)+\sum_{j=l+1}^{n}P^{(t)}(c_k\mid d_j)}{n}$$
$$k=1,2 \tag{4}$$

其中，$1(c_k\mid d_i)$的含义同式(2)。此时，重新估计分类器的参数值$P^{(t+1)}(w_m\mid c_k)$如式(5)。

$$P^{(t+1)}(w_m\mid c_k)=\frac{1+\sum_{i=1}^{l}N(w_m,d_i)1(c_k\mid d_i)+\sum_{j=l+1}^{n}N(w_m,d_j)P^{(t)}(c_k\mid d_j)}{|W|+\sum_{s=1}^{|W|}\sum_{i=1}^{l}N(w_s,d_i)1(c_k\mid d_i)+\sum_{s=1}^{|W|}\sum_{j=l+1}^{n}N(w_s,d_j)P^{(t)}(c_k\mid d_j)} \tag{5}$$

迭代进行E-Step和M-Step(图3，步骤2-5)直至分类器参数收敛。步骤3中，用KL距离(Kullback-Leibler Divergence)来衡量两次迭代的分类器参数概率分布的相似性，如式(6)所示。当两次迭代的参数分布距离小于阈值$\sigma(\sigma\geqslant 0)$时，认为参数收敛。

$$D_{KL}(P^{(t)}(w_m\mid c_k)\parallel P^{(t-1)}(w_m\mid c_k))=\sum_{w_m\in W}P^{(t)}(w_m\mid c_k)\log\frac{P^{(t)}(w_m\mid c_k)}{P^{(t-1)}(w_m\mid c_k)}k=1,2 \tag{6}$$

EM-NB($P^{(0)}(w_m|c_k)$,$D_u$)算法
输入：初始分类器训练结果$P^{(0)}(w_m|c_k)$，未标注文本集$D_u$，收敛边界$\sigma$
输出：拓展后的分类器 Classifier-EM
1: START EM-NB
2: WHILE $D_{KL}(P^{(t)}(w_m|c_k)||P^{(t-1)}(w_m|c_k))\geq\sigma$:
3: E-Step：根据参数及公式(3)，计算$P^{(t)}(c_k|d_j),d_j\epsilon D_u$;
4: M-Step：计算公式(4)(5)，重新估计参数$P^{(t+1)}(c_k)$和$P^{(t+1)}(w_m|c_k)$;
5: END WHILE
6: RETURN Classifier-EM

图3 EM-NB算法流程

实际实验中，当迭代次数两次时，分类器参数已经收敛，故实验过程中迭代次数为两次。

## 4 实验及分析

### 4.1 实验数据

实验数据来自新浪微博2014年的微博数据，分别按“阿里巴巴”、“iphone5”、“余额宝”、“雾霾”、“环保”、“理财产品”、“NBA”、“自贸区”、“华为”、“油价”十个话题进行微博搜索，按话题随机爬取了如下共145 304条微博数据，如表1所示。每个话题均随机抽取若干条数据作为“标注集”，人工标注后，用于实验中的训练集和测试集数据；剩余微博作为“未标注集”，不做标注。

表1 话题微博实验数据

| 话题 | 微博数/条 | | |
|---|---|---|---|
| | 共 计 | 标注集 | 未标注集 |
| 阿里巴巴 | 9 124 | 1 500 | 7 624 |
| iphone5 | 14 647 | 2 023 | 12 624 |
| 余额宝 | 18 773 | 1 500 | 17 273 |
| 雾霾 | 16 210 | 1 500 | 14 710 |
| 环保 | 18 154 | 1 500 | 16 654 |
| 理财产品 | 23 749 | 1 500 | 22 249 |
| NBA | 14 664 | 1 500 | 13 164 |
| 自贸区 | 12 173 | 1 500 | 10 673 |
| 华为 | 10 192 | 1 500 | 8 692 |
| 油价 | 23 141 | 1 500 | 21 640 |
| 共计 | 160 827 | 15 523 | 145 304 |

不同话题的微博数据，垃圾微博与非垃圾微博的数量比例不同。例如，“iphone5”话题的垃圾微博大约占85%，“阿里巴巴”话题的垃圾微博则仅占20%左右。每个话题的标注集垃圾微博分布如表2所示。

表2 标注集数据

| 话题 | 微博数/条 | | | 垃圾微博近似占比/% |
|---|---|---|---|---|
| | 共 计 | 垃圾微博 | 非垃圾微博 | |
| 阿里巴巴 | 1 500 | 331 | 1 169 | 20 |
| iphone5 | 2 023 | 1 705 | 318 | 85 |
| 余额宝 | 1 500 | 271 | 1 229 | 20 |
| 雾霾 | 1 500 | 239 | 1 261 | 15 |
| 环保 | 1 500 | 261 | 1 329 | 20 |
| 理财产品 | 1 500 | 736 | 764 | 50 |
| NBA | 1 500 | 198 | 1 302 | 15 |
| 自贸区 | 1 500 | 83 | 1 417 | 5 |
| 华为 | 1 500 | 764 | 736 | 50 |
| 油价 | 1 500 | 473 | 1 027 | 30 |

### 4.2　实验步骤

1. 文本预处理

文本预处理分为两个步骤：

（1）微博内容抽取：从爬取下来的原始微博数据中，抽取出微博内容本文，其他数据如微博用户 ID、微博 ID、发布时间等，暂不予以考虑；

（2）微博内容分词：分词工具采用张华平博士提供的 NLPIR（ICTCLAS2014 版本）中文分词工具①。同时，导入用户词典（共 1416 个常用词），防止常见词汇如“淘宝”、“微博”等被错误拆分，影响最终效果。

2. 特征抽取

本文使用的特征为词汇特征、表情符号和数字。

（1）词汇特征：在文本预处理后的分词结果中，去掉停用词，剩余每个词汇作为一个特征；同时，对于微博中的“@用户名”结构，仅保留@符号作为符号特征，删去@后的用户名；

（2）表情符号：删去中英文标点符号如句号（。.）、逗号（，,）、双引号（“”）、顿号（、）、冒号（：:）等，保留符号如 Hashtag（#）【】以及其他少见的符号如＊☆～o 等。

（3）数字处理：在商业广告或宣传中，常常出现商品标价。然而，由于商品标价不尽相同，难以提取作为表征广告微博的特征。因此，特征抽取过程中，将数字如“123”“12.34”等转化为字符串“Number”，即用“Number”字符串表示所有不同数值的数字。

3. 模型选择

实验主要比较三个模型的微博过滤效果：

（1）支持向量机 SVM：模型实现采用 LibSVM 工具②，选择在文本分类任务中效果较好的线性核支持向量机（Linear-SVM）。文本特征取用“特征抽取”后的结果，但需要转化为其在文本中的频率。同时注意，由于支持向量机对训练集正负数据比例非常敏感，在实现支持向量机过滤微博时，首先对训练集数据进行比例调整，对比例较大的一类数据进行随机抽样，保证训练集数据中两个类别的比例大约在 1∶1。实验表明，比例调整后，支持向量机的微博过滤效果优于比例失衡情况下的效果。

（2）朴素贝叶斯分类器 NB：文本特征取用“特征抽取”后的结果，利用训练集中“垃圾微博”与“非垃圾微博”的数据，实现一个完全有监督的模型；

（3）本文的中文微博过滤算法 Spam-Weibo：文本特征及训练集处理同朴素贝叶斯分类器，同时，使用话题下的“未标注集”作为算法中所需的未标注文本集；

4. 实验内容

为了验证在少量标注数据集下三个模型的微博过滤效果，实验采用不同数量的标注训练集：分别随机抽取 32、64、128、256、512 条标注微博用于训练分类器，剩余的微博作为测试集，对每个话题分别进行五种训练集数量的实验。例如，对于标注集共 1500 条微博的话题，五组实验中的训练集和测试集数据量如表 3 所示。

注意，每个话题的每组实验，均进行十次实验，最终取十次实验的平均结果作为该组实验的结果。

表 3　实验组数据

| 实验组 | 训练集/条 | 测试集/条 | 共计 |
|---|---|---|---|
| 1 | 32 | 1 468 | 1 500 |
| 2 | 64 | 1 436 | 1 500 |
| 3 | 128 | 1 372 | 1 500 |
| 4 | 256 | 1 244 | 1 500 |
| 5 | 512 | 988 | 1 500 |

### 4.3　评价指标

评价垃圾微博的过滤效果，主要从两个指标入手：准确率和 F 值。

- 准确率（Accuracy）表征了测试数据的整体分类效果，如式（7）所示。

$$\text{Accuracy}=\frac{\text{TP}+\text{TN}}{\text{TP}+\text{FN}+\text{TN}+\text{FP}} \tag{7}$$

- F 值表征了测试数据分类为“垃圾微博”类别的分类效果，如式（8）所示。

$$\text{F}=\frac{2\text{TP}}{2\text{TP}+\text{FN}+\text{FP}} \tag{8}$$

其中，TP（True Positives）表示被正确判断为垃圾微博的垃圾微博数；FN（False Negatives）表示被错误判断为非垃圾微博的垃圾微博数；TN（True Negatives）表示被正确判断为非垃圾微博的非垃圾微博数；FP（False Positives）表示被错误判断为垃圾微博的非垃圾微博数。

以上两个评价指标中，准确率（Accuracy）考虑了垃圾微博与非垃圾微博的整体分类效果。然而，

① http://ictclas.nlpir.org/

② http://www.csie.ntu.edu.tw/～cjlin/libsvm/

准确率的高低容易受到微博数据中垃圾微博与非垃圾微博数量比例的影响。例如，当测试集中80%的文本属于“垃圾微博”类别时，即使将所有的测试集微博均判断为垃圾微博，依旧能够得到80%的准确度。因此，F值显得更加重要。在本文的评价指标中，F值仅考虑了“垃圾微博”的分类效果。理想情况下，希望“垃圾微博”的精确度(Precision)和召回率(Recall)都很高，保证过滤之后的剩余微博中，绝大部分都是“非垃圾微博”，且尽可能少地将非垃圾微博划分为垃圾微博类别。F值很好的衡量了这个效果。

### 4.4 实验结果及分析

实验结果如表4～8及图2～3所示。表4～5显示，当训练集标注量为32或64条微博时，Spam-Weibo的微博过滤F值平均比朴素贝叶斯分类器分别提高了4%和3%，且每个话题下，Spam-Weibo均有最好的F值。此外，Spam-Weibo在大部分话题下同样取得了最高的准确度(90.3%，92.6%)，即，该模型在保证整体过滤效果的前提下，提高了“垃圾微博”过滤的能力。相反，支持向量机只有少量几个话题如“雾霾”、“环保”、“理财产品”取得了较好的分类准确度，整体过滤效果较差。此外，对于话题“阿里巴巴”、“理财产品”、“自贸区”、“华为”四个话题，当训练集仅为32条标注数据时，Spam-Weibo取得的分类效果优于64条训练集数据时朴素贝叶斯分类器的过滤效果，也就是说，在本文提出的Spam-Weibo微博过滤模型下，仅标注32条数据，就能取得优于朴素贝叶斯分类器下两倍标注量的效果。

表6显示，当训练集标注量为128条微博时，Spam-Weibo模型依旧在所有话题下均拥有最高的准确度(93.7%)和F值(85.7%)。“阿里巴巴”、“iphone5”、“环保”、“理财产品”、“NBA”、“自贸区”、“华为”七个话题中，当数据标注量为64条微博时，Spam-Weibo取得的过滤效果优于或近似于其他模型在128条标注量下的过滤效果。

随着训练集标注量的增大，Spam-Weibo模型在256条训练集标注量(表7)下虽然依旧有最高的准确度和F值，但是与NB模型的差距在缩小；当标注量达到512条时(表8)，我们提出的模型相对于朴素贝叶斯分类模型，过滤效果均略低(0.5%)。因此，我们提出的Spam-Weibo模型更适用于在标注数据量较少的情况。同时，需要注意，标注量从256增加到512时，两个模型的准确度都几乎没有变化，F值只增加了1%。实际生活中，新浪微博等每天产生的话题数不计其数，如果每个话题都标注500条数据，这是非常耗费人力的；因此，我们使用的Spam-Weibo模型更具实用性，它为数据标注人员减轻了大量负担。

最后，实验结果显示，SVM模型在各种训练标注集下，都无法取得较好的效果，这是因为SVM模型对于训练集数据量的大小以及正负样本比例非常敏感。首先，注意到当训练标注量达到512条时，大部分话题已经在SVM模型下取得了较优秀的准确度，其中，话题“NBA”的准确度，比其在256条训练集下增加了约10个百分点；并且，几乎每个话题都在训练标注量翻倍以后，准确度或F值都有5至10个百分点的增加，例如，“环保”，“余额宝”话题。因此，SVM模型对于标注数据的需求远比朴素贝叶斯模型和本文模型要强烈。同时，SVM模型对于训练数据的正负比例异常敏感。实验过程中，我们在进行比对后，发现先对训练集数据进行正负比例调整，再使用SVM模型，效果会比直接使用SVM模型要好的多。并且，在标注量512条微博时，“理财产品”，“华为”等正负比例相对均衡的话题，SVM模型取得的分类效果是很好的，并且随着标注量的增加，其分类准确度能够稳步增长。最后，需要注意的是，在调整SVM模型训练集正负样本比例的过程中，会对原有训练样本进行抽样，因此，即使是在大标注量的情况下，对于比例严重失衡的一些话题来说，抽样过后实际的标注量还是很少的，这是导致SVM模型在某些话题下，分类效果一直不稳定且很差的原因。然而，现实生活中，大部分话题下的垃圾微博比例在10%—20%左右，因此，SVM模型并不能适用于实际的垃圾微博过滤中。

表4 32条训练集数据的过滤效果

| 话题 | Accuracy | | | F值 | | |
|---|---|---|---|---|---|---|
| | SVM | NB | Spam-Weibo | SVM | NB | Spam-Weibo |
| 阿里巴巴 | 0.349 | 0.871 | **0.894** | 0.411 | 0.649 | **0.734** |
| iphone5 | 0.546 | 0.853 | **0.867** | 0.598 | 0.917 | **0.921** |

续表

| 话　题 | Accuracy | | | F　值 | | |
|---|---|---|---|---|---|---|
| | SVM | NB | Spam-Weibo | SVM | NB | Spam-Weibo |
| **余额宝** | 0.371 | 0.918 | **0.919** | 0.369 | 0.717 | **0.729** |
| **雾霾** | 0.831 | **0.883** | 0.875 | 0.549 | 0.575 | **0.612** |
| **环保** | 0.719 | **0.849** | 0.846 | 0.252 | 0.391 | **0.495** |
| **理财产品** | 0.888 | 0.921 | **0.944** | 0.885 | 0.920 | **0.940** |
| NBA | 0.643 | 0.917 | **0.923** | 0.284 | 0.540 | **0.595** |
| **自贸区** | 0.741 | 0.982 | **0.985** | 0.441 | 0.809 | **0.846** |
| **华为** | 0.613 | 0.787 | **0.836** | 0.723 | 0.818 | **0.851** |
| **油价** | 0.572 | 0.925 | **0.941** | 0.596 | 0.844 | **0.874** |
| **Average** | 0.627 | 0.891 | **0.903** | 0.511 | 0.718 | **0.760** |

**表 5　64 条训练集数据的过滤效果**

| 话　题 | Accuracy | | | F　值 | | |
|---|---|---|---|---|---|---|
| | SVM | NB | Spam-Weibo | SVM | NB | Spam-Weibo |
| **阿里巴巴** | 0.381 | 0.888 | **0.907** | 0.364 | 0.722 | **0.771** |
| iphone5 | 0.690 | 0.871 | **0.901** | 0.752 | 0.927 | **0.942** |
| **余额宝** | 0.555 | 0.940 | **0.948** | 0.485 | 0.807 | **0.833** |
| **雾霾** | 0.791 | 0.915 | **0.917** | 0.499 | 0.688 | **0.739** |
| **环保** | 0.679 | **0.881** | 0.871 | 0.248 | 0.576 | **0.619** |
| **理财产品** | 0.893 | 0.942 | **0.952** | 0.891 | 0.940 | **0.949** |
| NBA | 0.605 | 0.938 | **0.944** | 0.310 | 0.713 | **0.755** |
| **自贸区** | 0.908 | **0.985** | **0.985** | 0.643 | 0.843 | **0.851** |
| **华为** | 0.625 | 0.823 | **0.870** | 0.731 | 0.841 | **0.876** |
| **油价** | 0.528 | 0.953 | **0.968** | 0.593 | 0.930 | **0.949** |
| **Average** | 0.665 | 0.914 | **0.926** | 0.552 | 0.799 | **0.829** |

**表 6　128 条训练集数据的过滤效果**

| 话　题 | Accuracy | | | F　值 | | |
|---|---|---|---|---|---|---|
| | SVM | NB | Spam-Weibo | SVM | NB | Spam-Weibo |
| **阿里巴巴** | 0.395 | 0.906 | **0.927** | 0.398 | 0.784 | **0.832** |
| iphone5 | 0.557 | 0.895 | **0.909** | 0.594 | 0.939 | **0.946** |
| **余额宝** | 0.486 | 0.955 | **0.956** | 0.476 | 0.871 | **0.874** |
| **雾霾** | 0.888 | **0.937** | 0.930 | 0.588 | 0.797 | **0.802** |
| **环保** | 0.788 | 0.893 | **0.894** | 0.303 | 0.617 | **0.669** |
| **理财产品** | 0.911 | **0.953** | **0.953** | 0.901 | **0.951** | **0.951** |
| NBA | 0.715 | 0.945 | **0.947** | 0.373 | 0.754 | **0.769** |
| **自贸区** | 0.952 | 0.985 | **0.987** | 0.714 | 0.846 | **0.869** |
| **华为** | 0.713 | 0.853 | **0.881** | 0.773 | 0.865 | **0.886** |
| **油价** | 0.533 | 0.978 | **0.982** | 0.591 | 0.966 | **0.970** |
| **Average** | 0.694 | 0.930 | **0.937** | 0.571 | 0.839 | **0.857** |

表7 256条训练集数据的过滤效果

| 话 题 | Accuracy | | | F 值 | | |
|---|---|---|---|---|---|---|
| | SVM | NB | Spam-Weibo | SVM | NB | Spam-Weibo |
| **阿里巴巴** | 0.577 | 0.928 | **0.936** | 0.494 | 0.831 | **0.852** |
| iphone5 | 0.697 | 0.905 | **0.912** | 0.773 | 0.945 | **0.948** |
| **余额宝** | 0.483 | 0.969 | **0.972** | 0.451 | 0.911 | **0.918** |
| **雾霾** | 0.891 | **0.949** | 0.936 | 0.634 | **0.851** | 0.827 |
| **环保** | 0.850 | **0.909** | 0.899 | 0.287 | **0.703** | 0.700 |
| **理财产品** | 0.906 | **0.959** | **0.959** | 0.896 | 0.957 | **0.957** |
| NBA | 0.735 | 0.954 | **0.955** | 0.468 | 0.805 | **0.811** |
| **自贸区** | 0.982 | **0.989** | **0.989** | 0.829 | **0.892** | 0.887 |
| **华为** | 0.861 | 0.874 | **0.892** | 0.864 | 0.883 | **0.896** |
| **油价** | 0.579 | 0.982 | **0.983** | 0.606 | 0.972 | **0.973** |
| **Average** | 0.756 | 0.942 | **0.943** | 0.630 | 0.875 | **0.877** |

表8 512条训练集数据的过滤效果

| 话 题 | Accuracy | | | F 值 | | |
|---|---|---|---|---|---|---|
| | SVM | NB | Spam-Weibo | SVM | NB | Spam-Weibo |
| **阿里巴巴** | 0.519 | 0.939 | **0.945** | 0.502 | 0.862 | **0.877** |
| iphone5 | 0.637 | **0.917** | 0.915 | 0.718 | **0.952** | 0.950 |
| **余额宝** | 0.872 | 0.969 | **0.972** | 0.762 | 0.911 | **0.917** |
| **雾霾** | 0.904 | **0.949** | 0.931 | 0.654 | **0.851** | 0.815 |
| **环保** | 0.866 | **0.914** | 0.896 | 0.409 | **0.728** | 0.709 |
| **理财产品** | 0.914 | 0.961 | **0.962** | 0.904 | **0.960** | **0.960** |
| NBA | 0.833 | **0.955** | 0.954 | 0.576 | **0.813** | 0.811 |
| **自贸区** | 0.930 | **0.990** | 0.989 | 0.722 | **0.902** | 0.896 |
| **华为** | 0.885 | 0.887 | **0.896** | 0.881 | 0.894 | **0.900** |
| **油价** | 0.959 | **0.988** | 0.985 | 0.936 | **0.982** | 0.976 |
| **Average** | 0.832 | **0.947** | 0.944 | 0.706 | **0.886** | 0.881 |

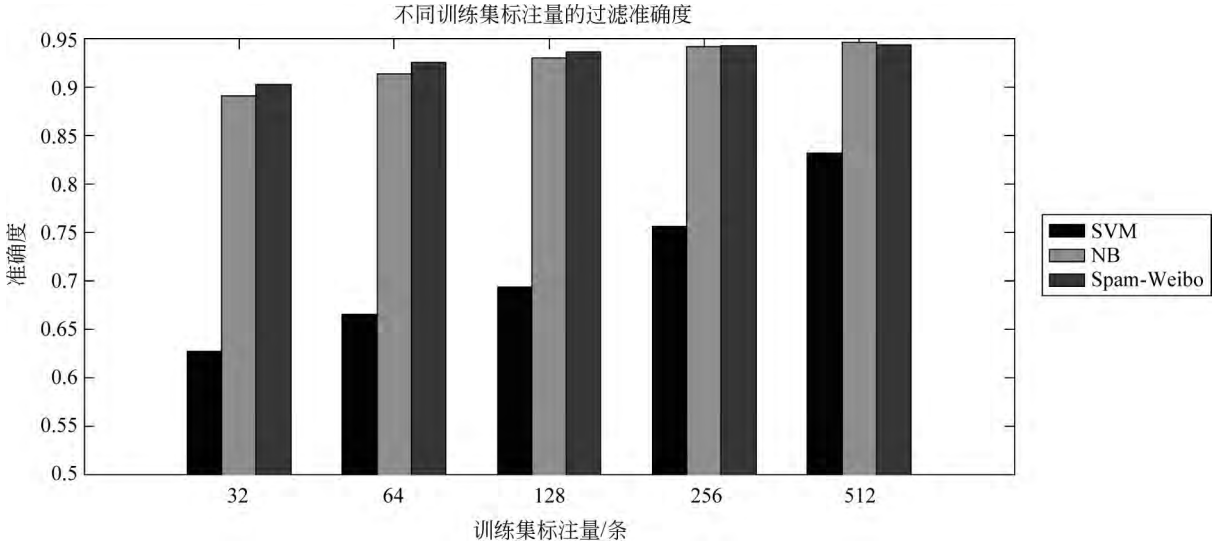


图2 不同训练集标注量的过滤准确度

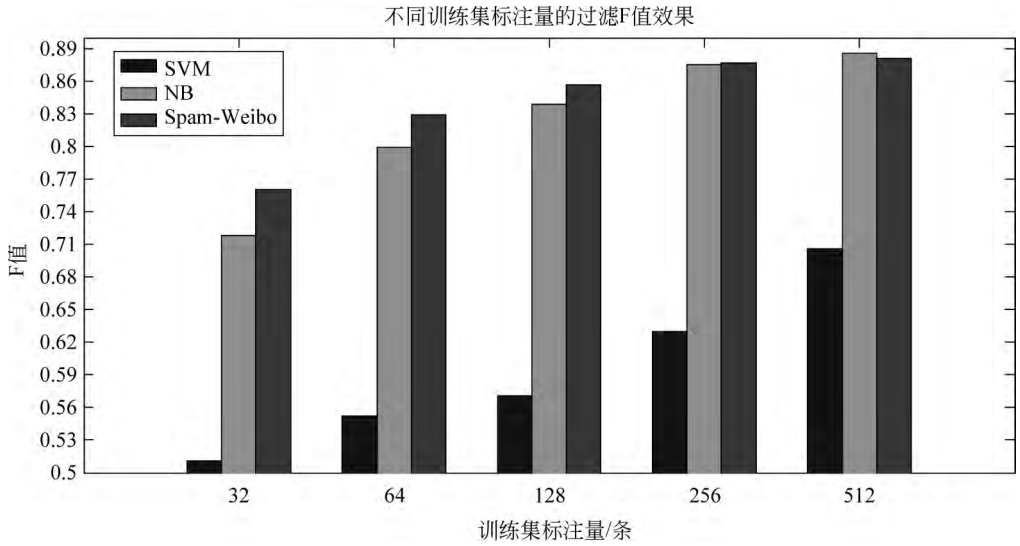


**图 3　不同训练集标注量的过滤 F 值**

## 5　总结

**本文针对给定话题下的垃圾微博过滤问题，提出了基于朴素贝叶斯分类模型和最大期望算法的中文话题微博过滤模型。本文提出的方法以少量标注数据对话题下的微博数据进行分类，极大降低了人工标注数据的成本。在新浪微博的十个话题、十四万余条微博文本下，本文提出的方法在训练集标注量分别为 32、64、128、256 条微博的情况下，准确度和 F 值均高于朴素贝叶斯分类模型（在 32 条标注量时，提高 1%的准确度和 4%的 F 值），且远优于支持向量机模型。同时，在本文提出的模型下，部分话题仅需要用一半的训练集标注量，就能达到比其他模型用成倍训练集更好的效果。**

## 参考文献


[1] By The Numbers：220 Amazing Twitter Statics [OL]. 2014. http://expandedramblings. com/index. php/march-2013-by-the-numbers-a-few-amazing-twitter-stats/#. VCdgtaiSzI0

[2] **陈倩. 微博广告发展现状与传播效果分析[J]. 产业与科技论坛**，2012，11(2)：33-35.

[3] **垃圾营销信息管理规定征求意见稿**[OL]. http://weibo. com/p/1001603697836242954625，2014.

[4] Jindal，Nitin，Bing Liu. Opinion spam and analysis [C]//Proceedings of the 2008 International Conference on Web Search and Data Mining. ACM，2008：219-230.

[5] Jindal N，Liu B. Reviewspam detection[C]//Proceedings of the 16th International Conference on World Wide Web，New York，NY，USA：ACM，2007：1189-1190.

[6] Li Jiwei，Claire Cardie，Sujian Li. Topic Spam：a Topic-Model based approach for spam detection[C]//Proceedings of the ACL，2013.

[7] Ren，Yafeng，Donghong Ji，and Hongbin Zhang. Positive Unlabeled Learning for Deceptive Reviews Detection[C]//Proceedings of the EMNLP，2014.

[8] Lim，Ee-Peng，et al. Detecting product review spammers using rating behaviors[C]//Proceedings of the 19th ACM International Conference on Information and Knowledge Management. ACM，2010：939-948.

[9] Wang Guan，et al. Review graph based online store review spammer detection [C]//Proceedings of Data Mining (ICDM)，2011 IEEE 11th International Conference on. IEEE，2011.

[10] Druck Gregory，Gideon Mann，Andrew McCallum. Learning from labeled features using generalized expectation criteria[C]//Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval. ACM，2008.

[11] YULAN He，Deyu Zhou. Self-training from labeled features for sentiment analysis[C]//Proceedings of Information Processing & Management 2011，47(4)：606-616.

[12] Liu Bing，et al. Partially supervised classification of text documents[C]//ICML，Vol. 2. 2002.

[13] Lang Ken. Newsweeder：Learning to filter netnews [C]//Proceedings of the 12th international conference on machine learning. 1995：331-339.

[14] Lucas，Michael，and Doug Downey. Scaling Semi-supervised Naive Bayes with FeatureMarginals[C]//Proceedings of ACL，2013.

[15] Settles Burr. Closing the loop: Fast, interactive semi-supervised annotation with queries on features and instances[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2011.

**姚子瑜**(1993—),美国俄亥俄州立大学硕士研究生,主要研究领域为自然语言处理、数据挖掘等。

E-mail: yao.470@osu.edu

**黄民烈**(1977—),副教授,主要研究领域为自然语言处理、人工智能等。

E-mail: aihuang@tsinghua.edu.cn

**屠守中**(1983—),博士研究生,主要研究领域为社交网络分析、信息安全、人工智能等。

E-mail: Kart123@163.com

(上接第175页)

[17] Zhou Y, Croft W B. Query performance prediction in web search environments[C]//Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2007: 543-550.

[18] Lang H, Wang B, Jones G, et al. Query performance prediction for information retrieval based on covering topic score[J]. Journal of Computer Science and technology, 2008, 23(4): 590-601.

[19] Cummins R. Predicting query performance directly from score distributions[M]. Information Retrieval Technology. Springer Berlin Heidelberg, 2011: 315-326.

[20] Markovits G, Shtok A, Kurland O, et al. Predicting query performance for fusion-based retrieval[C]//Proceedings of the 21st ACM international conference on information and knowledge management. ACM, 2012: 813-822.

[21] Wu Q, Burges C J C, Svore K M, et al. Adapting boosting for information retrieval measures[J]. Information Retrieval, 2010, 13(3): 254-270.

[22] Microsoft Research. LETOR[EB/OL]. http://research.microsoft.com/en-us/um/beijing/projects/letor/.

**薛源海**(1987—),博士,主要研究领域为信息检索和数据挖掘。

E-mail: xueyuanhai@software.ict.ac.cn

**刘悦**(1971—),博士,副研究员,主要研究领域为信息检索和数据挖掘。

E-mail: liuyue@ict.ac.cn

**俞晓明**(1977—),博士,高级工程师,主要研究领域为信息检索和大数据。

E-mail: yuxiaoming@software.ict.ac.cn