基于 Winnow 算法的文本过滤*

赵林　夏迎炬　黄萱菁　吴立德
复旦大学计算机科学与工程系
zhaolinfd@yahoo.com　yingjuxia@yahoo.com.cn　xjhuang@fudan.edu.cn　ldwu@fudan.edu.cn

摘　要：本文提出了一种在自适应文本过滤中将 Winnow 分类器和基于向量空间模型（VSM）的分类器相结合的算法。在处理文本流时，只有被两个分类器都过滤出的文本才被判定为相关文本。文中详细描述了在我们的过滤系统中所使用的 Winnow 算法以及所进行的一系列证实其有效性的实验。结果显示 Winnow 分类器的采用在 2002 年度的 TREC（文本检索会议）过滤任务中取得了显著的性能提高。

关键词：Winnow，文本过滤，向量空间模型

Text Filtering Based on Winnow Algorithm

Zhao Lin　Xia Yingjv　Huang Xuanjing　Wu Lide
Department of Computer Science and Engineering, Fudan University
zhaolinfd@yahoo.com　yingjuxia@yahoo.com.cn　xjhuang@fudan.edu.cn　ldwu@fudan.edu.cn

Abstract: This paper explores the combination of Winnow with Vector Space Model (VSM) based classifier in adaptive text filtering. When processing a document stream, only the documents retrieved by both classifiers are judged relevant. The Winnow algorithm used in our system is described in detail and a set of experiments is carried out to verify its effect. The results show that adoption of Winnow has obtained significant improvement in TREC (Text REtrieval Conference) 2002 filtering track.

Keywords: Winnow, Text filtering, VSM

1 引言

文本过滤的任务是从大量文本流中寻找那些符合用户兴趣的文本。给定主题的简要描述和一些训练文本，系统在经过学习后可以去寻找那些和该主题相关的文本。当文本流输入时，

* 本项目受国家自然科学基金项目（60103014）和 863 计划（2001AA114120，2002AA142090）资助

用户可以向系统提供被检出文档的反馈信息，用以自适应地修改过滤模板，从而使过滤系统随着时间的推移逐步提高它的准确率。

过滤的主要问题涉及到有监督的机器学习领域，解决的通常方法是设计一个好的文本分类器，例如贝叶斯分类器，神经网络，K 近邻分类器，Rocchio 分类器，决策树和支持向量机等[12, 15]都是较常用的方法。近年来，在 Winnow[6]和感知器[2, 10]等简单权重更新算法上出现了很多研究，理论上可以证明，当应用于线性可分的二元分类问题时，这种算法能够在有限步内找到一组权重将训练集分成两类[16]，而且在许多实际的分类问题上都显示了很好的效果。

我们知道，文本过滤并不是一个线性可分的问题而且文本向量的维数通常都很高，所以目前的方法都不是很有效。但根据理论证据，将各自独立的不同分类器进行适当结合，在性能上可以比单个分类器有所提高[5, 12]。因此我们决定引入 Winnow 分类器，将其和向量空间模型分类器相结合，这样理论上能够较好的解决过滤问题，在随后的实验中结果也是令人满意的。

本文由以下几个部分组成。第二节详细描述了在我们的自适应文本过滤系统中所使用的 Winnow 算法，第三节给出了实验结果，第四节进行总结。

2 文本过滤中使用的 WINNOW 算法

2.1 语料和数据

我们使用的语料是在 TREC2002 的过滤任务中所使用的 Reuter 语料集。其中训练集和测试集各包括 83,650 和 723,141 篇文档。总共有 100 个主题，每个主题被提供三篇正例文本用于训练。此外每个主题还包括两个数据：描述域和叙述域，它们提供了对该主题的描述信息和简要说明并对相关和无关文档做了定义，被我们用来训练该主题的 Winnow 分类器。例如某个主题的域说明如下，其中<title>为主题名，<desc>和<narr>分别为该主题的描述域和叙述域：

<title> Economic espionage
<desc> Description:
What is being done to counter economic espionage internationally?
<narr> Narrative:
Documents which identify economic espionage cases and provide action(s) taken to reprimand offenders or terminate their behavior are relevant. Economic espionage would encompass commercial, technical, industrial or corporate types of espionage. Documents about military or political espionage would be irrelevant.

2.2 观察和假设

通过和从训练文本中建立的主题模板中的词作比较，我们发现只有占每个主题描述域和叙述域大约36%的单词出现在模板中。这一情况说明主题域并没有被充分利用，因此我们决定根据这两个领域构建一个 Winnow 分类器，并将它和之前使用的 VSM 分类器相结合。由于这两个分类器的来源、形成截然不同，对于许多输入文本会作出不同的相关性判断。从直观上来讲，如果这些文本无关的话，那么这两个分类器的结合将会增加系统的准确率，而以查全率的下降为代价。由于 TREC 对系统性能的评价方式更加注重准确率，所以系统的整体性能会从中得到提高。该假设将会通过后面的实验加以证明。

2.3 Winnow 简介

Winnow是一种乘法式权重更新算法，当其应用于高维问题，特别是存在有许多无关属性，并且目标概念只依赖于特征空间中的一小部分特征时，它表现了良好的效果[3, 11]。同时Winnow算法是一种错误驱动算法，只有当系统输出结果和目标结果不同时，权重才会被调整。

该算法已经被成功的用于很多实际领域，像词性标注，拼写检查，日程安排等等[11, 3, 1, 7]，而且Winnow的多种变体算法也已经出现，例如Balanced Winnow[12]，Regularized Winnow[16]等，它们在一些特定问题中能够提高Winnow的性能。

2.4 Winnow 算法

在我们的过滤系统中，我们采取了基本的Winnow算法，它在以前的研究中已经被证实在文本分类问题上取得很好的效果[8]。

Winnow分类器的训练阶段如下。首先，从每个主题的描述域和叙述域中去掉禁用词。这里禁用词包括虚词以及那些对主题意义无贡献的词，像“relevant”，“irrelevant”等等。随后用剩下的词为每个主题初始化一个Winnow分类器。每个单词的初始权重 w_i 都被设为相等，初始阈值 θ 被设成1。Winnow将根据在文本过滤过程中，自身输出结果和实际结果的比较来调整词的权重。对于任一输入训练文本，我们计算Winnow中单词的加权和。

如果 $w^T x \geq \theta$，那么该文本被看作相关文本，被 Winnow 检索出。

如果 $w^T x < \theta$，该文本则被看作无关文本，不被检索出。

这里$x=(x_1, x_2 \ldots x_n)$是一个二元向量，即如果Winnow中的某个词出现在文本中，x_i就被设置成1；否则被设置成0。我们根据下面的规则来调整各项的权重：

1. 如果某个相关文本没有被Winnow检索出，说明目前设置的权重偏低，于是对任意 $x_i = 1$ 将 w_i 提升为 $1.5w_i$。
2. 如果某个无关文本被Winnow检索出，说明目前设置的权重偏高，于是对任意 $x_i = 1$ 将 w_i 降为 $0.8w_i$。
3. 否则，结果正确，则不必作任何变动。

这样的过程在训练集上不断重复直到所有的训练文本都被遍历过，从而我们就完成了Winnow分类器的权重训练过程，接下来开始为其设置阈值。在此我们并没有为每个主题设置相同阈值，而是对每个主题的三篇相关文档计算$w^T x$，得到它们的平均值，将此作为每个主题的最终阈值。用公式表示如下：

$$\theta = \frac{w^T x_1 + w^T x_2 + w^T x_3}{3}$$

这里θ为阈值，w是Winnow的权重向量，x_1, x_2, x_3分别是三篇相关文档的二元向量表示。

通过以上过程我们就完成了 Winnow 分类器的构造。整个训练阶段的完整流程图如图 1 所示，一方面我们要建立 Winnow 分类器，另一方面要建立向量空间模型分类器，即从正例文本和伪正例文本中根据互信息量公式抽取特征向量，生成初始的过滤模板，并根据初始模板和全部训练样本之间的相似度来选择最优的初始阈值。

在建立了这两个分类器之后，开始进入过滤阶段。对于输入的文本流，先用Winnow分类器进行相关性判断，对检出的相关文本再计算它和主题模板间的相似度，然后和阈值比较得到过滤结果。过滤出的文本被用于相关反馈，由用户判断是否真正相关，以此来更新主题模板和调整阈值，从而提高系统的性能。同时Winnow的权重在此过程中也可以作动态调整，使用的方法和训练阶段一致。该阶段的体系结构如图2所示。在以上训练和过滤这两个阶段中有关VSM分类器的细节，读者可以参考[4, 13, 14]，这里限于篇幅关系不作详细描述。

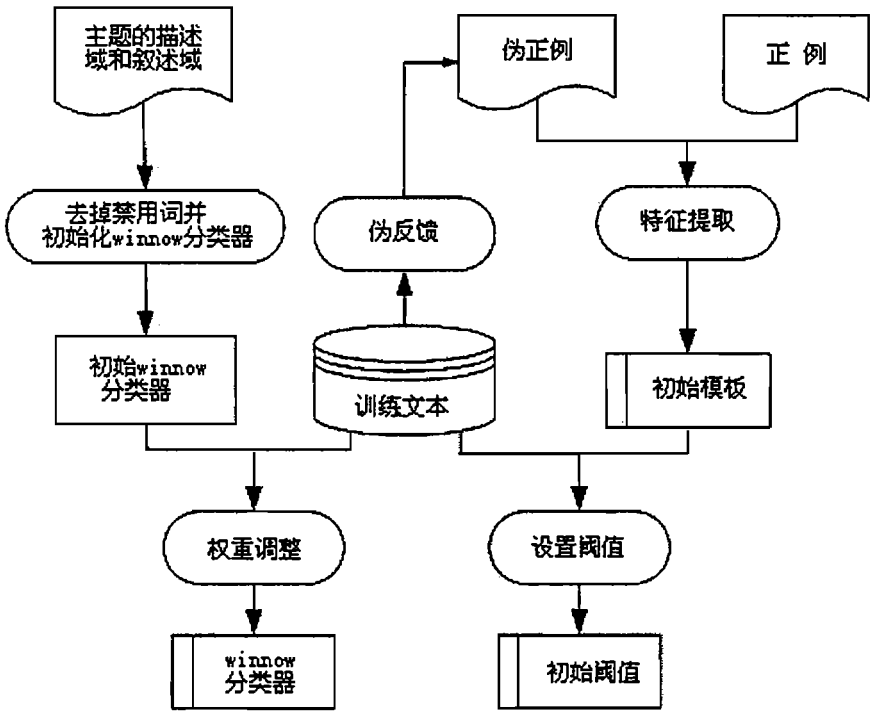

图1. 训练阶段的体系结构

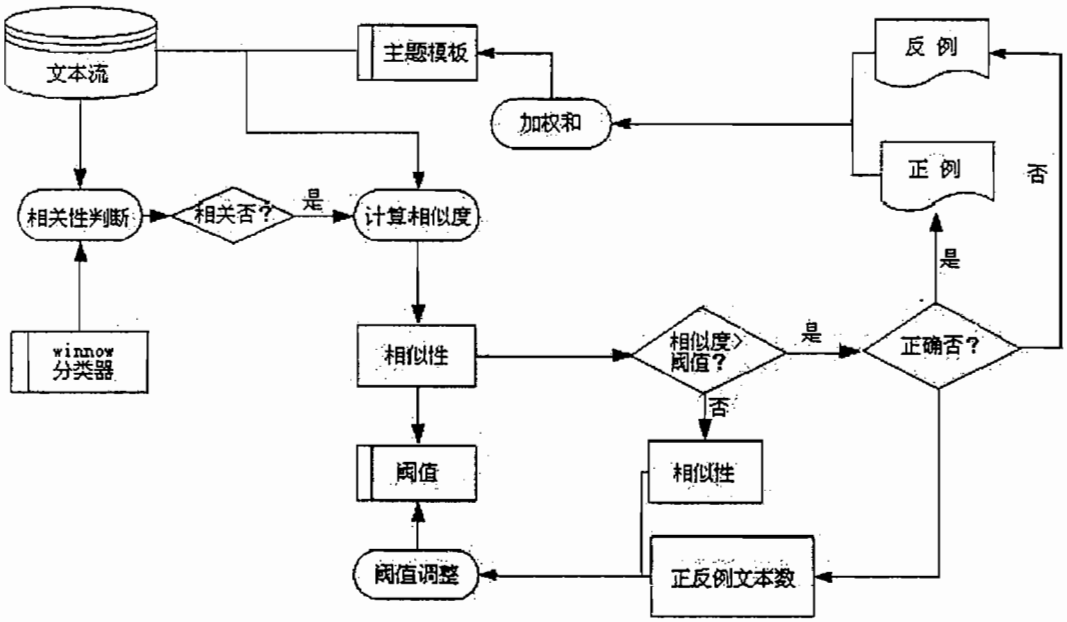

图2. 自适应过滤阶段的体系结构

3 实验结果

为了测试 Winnow 分类器的效果，我们用 TREC 提供的数据做了一系列实验，所用的评测方法来自 TREC2002[9]。表 1 和表 2 分别列出了当我们的过滤系统只用 VSM 分类器和用两种分类器相组合方法时的实验结果。表中所列数据是前 50 个主题的平均性能，表 2 括号中所列数据是相应项对表 1 的增长率。由于在实验过程中，VSM 分类器的各项参数，主要包括特征抽取时互信息量的门限值，生成过滤模板时文本向量的权重以及过滤阈值在作自适应调整时的上升、下降等参数，这些参数取不同值时会在很大程度上影响实验结果的好坏，所以我们对不同取值的参数进行了多次实验比较其结果，表 1 中的数据即为达到了最优性能的一组实验结果，而表 2 数据则是在 VSM 方法的基础上，保持参数不变，加入了 Winnow 分类器所达到的效果，因此这两张表中结果的差异将完全取决于 Winnow 的分类效果。

通过对这两张表的比较，我们可以看出结果证实了我们先前的假设，即组合方法会以查全率的损失为代价增加系统的准确率，最终使系统的整体性能得到提高——T11F 和 T11SU 指标具有显著提升，而这一点是来自于 Winnow 分类器的贡献，从而证实了它的有效性。

表 1. 基于 VSM 分类器的实验结果

	Recall	Precision	T11F	T11SU
R101～R150	0.3159	0.2668	0.2204	0.2287

表 2. 基于组合方法的实验结果

	Recall	Precision	T11F	T11SU
R101～R150	0.2240	0.4511	0.3189 (44.7%)	0.3657 (59.9%)

我们还做了一些实验来对不同大小的测试集作系统测试。图 3 和图 4 显示了采用组合算法和 VSM 算法下的实验结果。纵轴分别代表不同评价指标 T11F 和 T11SU，横轴代表测试集的大小，从 100,000 篇文档到 723,141（测试文本总数）篇文档。从图中可以看出，组合了 Winnow 后的曲线相对于只用 VSM 有显著上升，充分显示了 Winnow 分类器的优势。

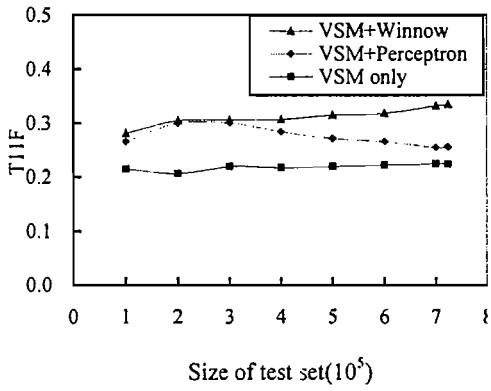

图 3. 对不同大小测试集使用组合算法和只用 VSM 算法时的 T11F 值

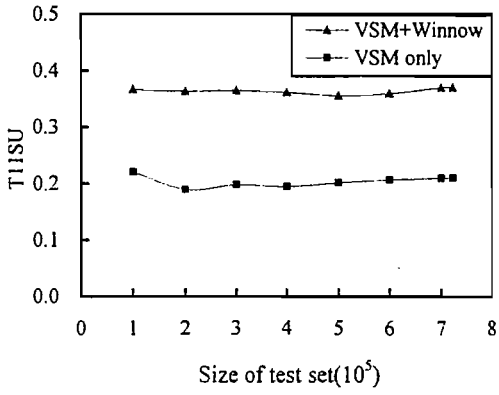

图 4. 对不同大小测试集使用组合算法和只用 VSM 算法时的 T11SU 值

此外，我们还将感知器（Perceptron）算法引入到过滤系统中，试图和 Winnow 算法作性能比较。因为感知器同 Winnow 一样是一种权重更新算法，所不同的是，它采用加法式权重更新策略，而不是 Winnow 所用的乘法式策略。感知器的实验结果表示在图 3 中，如曲线“VSM+Perceptron”所示。从图中可见，Winnow 获得了比感知器更好的性能。Winnow 在整个测试集上的 T11F 值为 0.3343，而感知器则为 0.2551。并且，在感知器的权重训练阶段，我们在训练集上进行了 60 次迭代训练；而对 Winnow 只用了一次，这说明 Winnow 比感知器有更快的收敛速度。所以总体而言，在该实验中，Winnow 比感知器展现了更多的优势，对 Winnow 和感知器的对比感兴趣的读者可以参考[16]。

4 总结

我们实现了将 Winnow 分类器引入自适应文本过滤系统中，通过一系列实验，我们证实了这对于提高系统的性能是一种行之有效的方法，而且该算法的实现也比较简单，不必增加过多复杂工作，且 Winnow 的训练速度也令人满意。在 TREC2002 的过滤任务中该系统取得了较好的成绩，今后我们还将对它作进一步完善，来不断提高我们的文本过滤水平。

参考文献

[1] Avrim Blum. Empirical support for winnow and weighted-majority algorithm: results on a calendar scheduling domain. In *ML-95*, pages 64-72, 1995.

[2] N Cancedda et al. Kernel methods for document filtering. *Report at TREC-11*, 2002.

[3] A. R. Golding and D. Roth. A Winnow based approach to context-sensitive spelling correction. *Machine Learning*, 34(1-3), pages 107-130, 1999.

[4] Xuanjing Huang et al. A text filtering system based on vector space model. *Journal of Software*, 13(4). 2002.

[5] D. A. Hull, J. O. Pedersen, and H. Shutze. Method combination for document filtering. *In Proceedings of SIGIR*, pages 279-298, 1996.

[6] N. Littlestone. Learning quickly when irrelevant attributes abound: A new linear threshold algorithm. *Machine Learning*, 2: 285-318. 1988.

[7] C. Mesterharm. A multi-class linear learning algorithm related to winnow. *In NIPS-12*, pages 519-525. *MIT Press*, 2000.

[8] M. Pazanni. A framework for collaborative, content-based and demographic filtering. *AI Review*, 13(5-6), 1999.

[9] S. Robertson, J. Callen. Guidelines for the TREC 2002 filtering track. 2002.

[10] S E Robertson, S Walker, H Zaragoza. Microsoft Cambridge at TREC-11: Filtering track. *Report at TREC-11*, 2002.

[11] D. Roth, D. Zelenko. Part of speech tagging using a network of linear separators. *COLING ACL*, 1998.

[12] F. Sebastiani. Machine learning in automated text categorisation: a survey. *Technical report, Istituto di Elaborazione dell'Informazione, C.N.R., Pisa, Italy*, 1999.

[13] Lide Wu et al. FDU at TREC-10: Filtering, Q&A, Web and Video tasks. *Notebook for the tenth Text Retrieval Conference*. 2001.

[14] Lide Wu et al. FDU at TREC 2002: Filtering, Q&A, Web and Video tasks. *Notebook for the eleventh Text Retrieval Conference*. 2002.

[15] Y Yang, X Liu. A Re-examination of text categorization methods. In *proc. SIGIR-99*, pages 42-49. 1999.

[16] T Zhang. Regularized winnow methods. *In Advances in Neural Information Processing Systems 13*, pages 703-709, 2001.

附中文参考文献：

[4] 黄萱菁、夏迎炬、吴立德，基于向量空间模型的文本过滤系统，软件学报，13 卷 4 期，2002.