文本自动分类中特征权重算法的改进研究

徐凤亚　罗振声
（清华大学计算语言学研究室，北京 100084）

摘　要　文章研究并改进了文本自动分类中的特征权重算法。传统的特征权重算法着重于考虑频率和反文档频率等因素，而未考虑特征的类间、类内分布与低频高权信息。该文重点研究了特征的类间、类内分布，以及低频高权特征对分类的影响，并在此基础上提出了低频高权特征集的构造方法及特征权重的新算法，同时将该算法推广到多层次分类体系。实验证明该算法能有效提高分类的精确度，而且在多级分类中也能取得很好的效果。

关键词　特征项　权重算法　分布信息　低频高权特征　文本分类

文章编号 1002-8331-(2005)01-0181-04　**文献标识码** A　**中图分类号** TP301.6

An Improved Approach to Term Weighting in Automated Text Classification

Xu Fengya　Luo Zhensheng
(The Institute of Computational Linguistics, Tsinghua University, Beijing 100084)

Abstract: This article aims to improve the algorithm of term weighting in automated text classification.Traditional algorithms only consider about TF(Term Frequency), IDF(Inverse Document Frequency)and so on, and do not consider DI(Distribution Information) among and inside classes and LFHW(Low Frequency but High Weight) terms.This article mainly researches about the impassion of DI and LFHW terms on classification, the construction of LFHW term sets and new approaches to term weighting.These new approaches are also applied to the hierarchical classification system. The comparison of experimental results proves that these new approaches can not only improve the precision of classification, but also have a good performance in hierarchical classification.

Keywords: term, weighting algorithm, DI, LFHW Terms, text classification

1　引言

随着信息技术的发展，网络信息量的猛增对电子文本的管理和检索提出了更高的要求，事先对文本进行有效分类成为必不可少的一环[1]。目前存在着很多不同的分类方法，如布尔模型、纯粹贝叶斯法、*SVM* 支持向量机法、*KNN* 最近邻法、神经网络法、决策树法以及模糊分类法等。

在很多分类方法中通常都要使用向量模型，用特征项（T_1, T_2,...,T_n）及相应权值 W_{T_i} 来代表特征信息，并用这些特征信息评价未知文本与标准类文本的相似程度。由于分类方法都建立在特征项频率统计和权重计算的基础上，因此特征权重算法的优劣将直接影响到分类的精确度。

目前最常用的是基于 *TF-IDF* 的传统特征权重算法。该文研究了传统算法的优点和不足，并结合特征项的类间、类内分布，以及低频高权特征项对分类的影响，提出了一种新的特征权重算法。实验证明该算法能大大提高文本分类的精确度，而且在多级分类中也有较好的效果。

2　传统的特征权重算法

传统的特征权重算法主要考虑特征项的频率信息 *TF* 以及反文档频率信息 *IDF*[2]。此外，还有对特征项的长度及出现位置[3]作加权处理的研究。

2.1　特征项的频率信息

特征项频率 *TF*(Term Frequency)是指特征项在文档中出现的次数。特征项可以是字、词、短语，也可以是经过语义概念词典进行语义归并或概念特征提取后的语义单元。不同类别的文档，在某些特征项的出现频率上有很大差异，因此频率信息是文本分类的重要参考之一。在最初的文本自动分类中，文档向量就是用 *TF* 来构造的。

2.2　反文档频率信息

单纯使用 *TF* 往往会导致一个问题，就是文档中大量出现的禁用词会干扰特征权重的计算。禁用词在所有文档中出现的频率都比较高，对文档意义的贡献度却很小。为了处理禁用词，有的系统采用了禁用词过滤的办法。这样做需要依赖于一个专家构造的禁用词词典。不过禁用词的界定本身就是一个主观性很强的判断，而且词典在扩充和修改上都需要一定程度的人工干预，因此更为合理的处理禁用词以及一些接近禁用词的高频词的办法是使用反文档频率。

反文档频率 *IDF*(Inverse Document Frequency)是特征项在文档集分布情况的量化。*IDF* 常用的计算方法为：

$$idf(T_k)=\log(N/n_k+L) \quad (1)$$

其中，L 的取值通过实验来确定。N 为文档集中的总文档数，n_k 为出现特征项 T_k 的文档数。

IDF 算法的核心思想是，在大多数文档中都出现的特征项不如只在小部分文档中出现的特征项重要。*IDF* 算法能够弱化一些在大多数文档中都出现的高频特征项的重要度，同时增强一些在小部分文档中出现的低频特征项的重要度。

一个有效的分类特征项应该既能体现所属类别的内容，又能将该类别同其它类别相区分。所以，在实际应用中，*TF* 与

IDF 通常是联合使用的[4]。*TF* 与 *IDF* 的联合公式如下(其中 i 表示类别号):

$$weight_{TF\_IDF}(T_{ik})=tf(T_{ik})\times idf(T_k) \tag{2}$$

在很多情况下还需要将向量归一化,*TF–IDF* 的归一化计算公式如下(其中 s 表示类别 i 中特征项的总个数):

$$weight_{TF\_IDF}(T_{ik})=\frac{tf(T_{ik})\times\log(N/n_k+L)}{\sqrt{\sum_{k=1}^{s}(tf(T_{ik}))^2[\log(N/n_k+L)]^2}} \tag{3}$$

2.3 特征项的长度信息

特征项的长度也可以作为一种权重衡量因素。自动分词后的统计结果表明:文档中出现的单字词数量通常最多,但包含的信息量较少;多字词数量通常最少,但包含的信息量较多,重要度也较高。一般说来,较长的特征项能表达较为专指的概念,如"世界乒乓球锦标赛"专指"体育",因此要给这样的多字词较高的权重。

2.4 特征项的位置信息

一般来说,出现在文章标题和首段中的词表达文章主题的能力比正文中的其它词要强。所以,在分类中通常还需要对文档的标题和首段作加权处理。如文献[5]结合了词在文档的位置,突出了标题的重要性。Patent 系统[6]也充分考虑了位置因素,只采用标题、文摘和文章的前 20 行以及部分重要章节进行频率统计。另外,目前网络上广泛使用的文档格式为 HTML,与普通的文本文件相比,HTML 文档中有明显的标识符,结构信息更加明显,对象属性更为丰富。因此,在对这类文档进行分类时,也应充分考虑其特点,对标题和特征信息较多的文本赋予较高的权重。

3 传统的特征权重算法的不足

传统的特征权重算法存在明显的不足。因为 *TF–IDF* 是将文档集作为整体来考虑的,特别是其中 *IDF* 的计算,并没有考虑到特征项在类间和类内的分布情况。如果某一特征项在某个类别大量出现,而在其它类别出现很少,这样的特征项的分类能力显然是很强的。但这在 *TF–IDF* 算法中是无法体现的。

另一方面,同样是集中分布于某一类别的不同特征项,类内分布相对均匀的特征项的权重应该比分布不均匀的要高。因为如果某一特征项只在某个类别的一两篇文档中大量出现,而在类内的其它文档中出现得很少,那么不排除这一两篇文档是该类别中特例的情况,因此这样的特征项不具备代表性,权重相对较低。对于这种情况,传统的 *TF–IDF* 算法也不能很好地处理。

这里通过一个很小的文档集来说明上述问题。假设有三个类别,每个类别各 5 篇文档,只考虑三个特征项 T_1、T_2 和 T_3。

表 1 为特征项在各篇文档中出现的频率。表 2 为传统 TF–IDF 算法的权值计算结果。其中 *tf* 为特征项在相应类别中出现的频率,N_k 为类别中出现该特征项的文档数,最终在此基础上根据公式(3)(其中 L 取 0.1)得到归一化的 TF–IDF 权值。

表 1 各个特征项的出现频率

特征项 \ 文档	类 1					类 2					类 3				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
T_1	4	3	2	5	6	1	1	1	1	1	1	1	1	1	1
T_2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
T_3	1	1	1	1	1	3	1	3	1	1	2	2	2	2	2

表 2 特征权重计算结果(TF–IDF)

特征项 / 类别	T_1		T_2		T_3	
	$tf(N_k)$	TF–IDF	$tf(N_k)$	TF–IDF	$tf(N_k)$	TF–IDF
类 1	20(5)	0.873	10(5)	0.436	5(5)	0.218
类 2	5(5)	0.348	10(5)	0.696	9(5)	0.627
类 3	5(5)	0.333	10(5)	0.667	10(5)	0.667

从表 1 中可以看出:T_1 在各类中出现的 *tf* 差别很大,分类能力应该最强;而 T_2 在各篇文档中的 *tf* 都相同,对分类基本没有信息贡献,因此其分类能力应该最弱。但从表 2 的结果来看,T_2 的权值却非常高。这是因为根据 *TF–IDF* 算法的定义,特征项的权重由 *TF* 和 *IDF* 决定。当文档集中包含特征项 T_1、T_2 和 T_3 的文档数相同时,这些特征项的 *IDF* 相同,特征项的权重由其 *TF* 唯一确定。所以导致表 2 得到了一个极不合理的结果,几乎没有分类能力的被赋予了很高的权值。由此可见,在没有考虑到特征项在类间和类内分布的比例情况[7]时,单纯使用 *TF–IDF* 算法会导致很大误差。

此外,为了提高运行效率,往往还需要对文档向量进行压缩处理,仅保留权值较高的特征项,从而形成维数较低的文档向量。这样一来,低频的词条就很有可能会被删除。但是低频的词条中会包含一些出现次数很少而重要度却很高的专指概念。传统的 *TF–IDF* 算法未加任何处理,忽略了这些重要低频高权特征项的分类作用。

4 特征权重算法的改进

针对传统权重算法的不足,该文提出了改进算法,即在传统 *TF–IDF* 的基础上,结合分布信息 *DI*(Distribution Information)和低频高权特征 *LFHW*(Low Frequency but High Weight)信息进行联合加权。

4.1 使用 DI 的改进算法

特征项的分布信息,又称为特征项频率分布的离散度。离散度可分为类间离散度 DI_{ac}(Distribution Information Among Classes)与类内离散度 DI_{ic}(Distribution Information Inside a Class),分别表示特征项在类间与类内文档间的分布差异。

4.1.1 特征项的类间离散度

DI_{ac} 的计算公式如下:

$$\frac{\sqrt{\left[\sum_{i=1}^{m}(tf_i(T_k)-\overline{tf(T_k)})^2\right]\Big/(m-1)}}{tf(T_k)} \tag{4}$$

其中,$tf_i(T_k)$ 表示 T_k 在第 i 类中的出现频度,m 为类别总数。$\overline{tf(T_k)}$ 为 T_k 在各类中出现频度的平均值,计算公式如下:

$$\overline{tf(T_k)}=\frac{1}{m}\sum_{i=1}^{m}tf_i(T_k) \tag{5}$$

4.1.2 特征项的类内离散度

DI_{ic} 的计算公式如下:

$$\frac{\sqrt{\left[\sum_{j=1}^{n}(tf_j(T_k)-\overline{tf'(T_k)})^2\right]\Big/(n-1)}}{tf'(T_k)} \tag{6}$$

其中,$tf_j(T_k)$ 表示 T_k 在第 j 篇中的出现频度,n 为类内总文档数。$\overline{tf'(T_k)}$ 为 T_k 在各篇文档中出现频度的平均值,计算公式如下:

$$\overline{tf'(T_k)}=\frac{1}{n}\sum_{j=1}^{n}tf_j(T_k) \tag{7}$$

4.1.3 TF-IDF-DI 算法

由公式(4)~(7)可得,一、当特征项只在一个类别中出现时,其 DI_{ac} 取最大值 1,其分类能力最强;当特征项在每个类别中的 TF 都相同时,其 DI_{ac} 取最小值 0,其分类能力最弱。可见,DI_{ac} 的分布区间为[0,1],同时特征项的 DI_{ac} 与分类能力成正比。二、当特征项只在一篇文档中出现时,DI_{ic} 取最大值 1,其分类能力最弱;当特征项在每篇文档中的 TF 都相同时,DI_{ic} 取最小值 0,其分类能力最强。因此,DI_{ic} 的分布区间也为[0,1],但特征项的与分类能力成反比。所以,在权重计算时可以用 $1-DI_{ic}$ (T_{ik})来表示。

在传统 TF-IDF 算法的基础上结合类别分布信息,即可得到 TF-IDF-DI 的计算公式:

$$weight_{TF\_IDF\_DI}(T_{ik})=\frac{tf(T_{ik})\times\log(N/n_k+L)}{\sqrt{\sum_{k=1}^{s}(tf(T_{ik}))^2[\log(N/n_k+L)]^2}}\times DI_{ac}(T_{ik})\times(1-DI_{ic}(T_{ik})) \quad (8)$$

由以上公式结合表 1 和表 2 的数据可以计算出相应的 DI_{ac}、DI_{ic}(表 3)以及特征权重(表 4)。

表 3 特征项的类间及类内的分布离散度

特征项 \ 分布离散度	DI_{ac}	DI_{ic}		
	(3 类间)	类 1	类 2	类 3
T_1	0.866	0.605	1	1
T_2	0	1	1	1
T_3	0.331	1	0.711	1

表 4 特征权重计算结果(TF-IDF-DI)

特征项 \ 类别	类 1	类 2	类 3
T_1	0.457	0.301	0.288
T_2	0	0	0
T_3	0.072	0.147	0.221

从表 3 和表 4 的结果来看,T_1 在类 1 分布不均匀,因此权值 W_{T_1} 得到削弱;T_2 在各类的 TF 都相同,对分类没有贡献,因此 W_{T_2} 为 0;T_3 在各个类别的 TF 相近,因此 W_{T_3} 较低。这些都体现了 DI 算法的基本思想。

可见,有效利用特征项的 DI_{ac} 及 DI_{ic} 能够弥补传统权重算法忽视频率分布信息的不足,从而提高文本自动分类的精确度。

4.2 低频高权特征 LFHW

由 TF-IDF-DI 算法可知,同时具备 TF 较高(不包括高频禁用词)、IDF 较高、DI_{ac} 较大以及 DI_{ic} 较小等特点的特征项具有较高的分类能力。此外,一些只在某一类别的少量文档中出现,TF 较低但 IDF 和 DI_{ac} 较高的特征项,通过 TF-IDF-DI 计算也获得了较高的权重,称之为低频高权特征项。然而,这些特征项与其它特征项差别较大,在进行向量相似度计算时不能简单地与其它特征项同等计算。下面举例说明低频高权特征的特殊之处。

在表 5 中,文档 D_1 中高频特征项的出现频率与类别向量 C_{model} 十分相似,而低频特征项差异较大;文档 D_2 正好相反。然而,不管是传统的 TF-IDF 算法还是改进的 TF-IDF-DI 算法,在计算文档向量时都将低频特征项与高频特征项同等处理,所以每个低频特征项都作为向量中的一个维进行计算。由于 C_{model} 的数据来源于大量的训练文档,出现在 C_{model} 中的低频特征项不可能在每篇文档中都出现,所以在一般文档中,这些维除了少数几个具有较高的值以外,其它维的值都为 0;而在类别向量中,这些维在运算之后都变得比较平均。因此在计算待测文档向量时,统计低频特征项出现的联合频率比统计单个频率更加合理。D_1 与 D_2 的低频特征项的联合频率都为 0.04,而 D_1 的高频特征项与 C_{model} 的相似度却比 D_2 的高频特征项与 C_{model} 的相似度要大。显然,文档 D_1 更接近类别向量。

表 5 部分特征项的出现频率

特征项 \ 文档		C_{model}	D_1	D_2
高频特征	T_1	0.1	0.1	0.04
	T_2	0.15	0.14	0.1
	T_3	0.1	0.1	0.16
	T_4	0.05	0.06	0.2
	T_5	0.2	0.2	0.1
低频特征	T_6	0.01	0	0.01
	T_7	0.003	0	0.004
	T_8	0.02	0.03	0.02
	T_9	0.007	0.01	0.006

在一篇文档中可能仅有几个低频高权特征同时出现,然而这些低频高权特征往往是对某个类别的专指概念,因而它们的分类作用不容忽视。显然,具有相同类别特性的低频高权特征在一篇文档中出现得越多,文档的类别就越明确。低频高权特征出现的随意性与其特殊重要性决定了在分类中不能将其简单地与其它特征项同等进行向量空间模型运算。因此系统可以构造低频高权特征集,将所有的低频高权特征作为一个特征项,与其它特征项共同构造空间向量模型,进行文档——类别的相似度计算。

构造低频高权特征集的步骤如下:一、统计特征项在某一类别文档内的出现频率,得到频率 $tf_i(T_k)<l$ 的特征项;二、计算特征项的权值 w_{ik},得到 $w_{ik}>t$ 的特征项。符合这两项的集合即为低频高权特征集,阈值 l 和 t 由实验确定。由此,系统可为每个类别构造一个低频高权特征集。

在类别较多时,通常要建立多层级的分类系统。在多层级分类系统中,同样需要对每个大类和小类构造一个低频高权特征集。多层级分类系统在构造多层次特征向量模型的同时构造多层次低频高权特征集。如果将多层级分类模型映射到树结构,那么这棵树代表整个分类系统,树中的各个节点则代表分类体系中相应类别。参考 CCM(Create Class Mode)[8]的作法,多层次的分类系统需要对树的每一层的所有节点自下而上进行处理,具体步骤如下:

(1)若该节点是叶子节点,则统计该节点对应的类别中特征项的 TF 及 DI,然后回到第 1 步分析下一个节点。

(2)若该节点是非叶子节点,假设该节点有 $N_1,N_2,\ldots,N_t$,共 t 个子节点,对应的类别中有 $T_1,T_2,\ldots,T_s$ 共 s 个特征项。提取节点 N_i 对应的类模型 $C_i,i=1,2,\ldots,t$。

①初始化 C_i 为空;

②首先,从 T_1 到 T_s 中提取 TF 低于一定阈值 $tf_i(T_k)<l$ 的特征项,计算这些特征项的权值 w_{ik} 并按大小排列。然后,提取 w_{ik} 最大的 num_{set} 个特征项加入低频高权特征集,并对该特征集的总 TF 进行累加。最后,将集合中的所有特征项作为一个特殊的特征项,统计其 TF 与 DI;

③根据该低频高权特征的 TF 及 DI 对所有特征项的 w_{ik} 重新计算并按大小排列。提取 w_{ik} 最大的 num_{model} 项加入类模型 C_i;

④根据下个节点是否是叶子节点判断下一步转(1)还是转

(2)。如果该节点已经是根节点,则退出。

算法流程如下:

```
1    if IsLeafNode(currentNode)
2        计算 TFIDF(eachTk)& DI(eachTk)
3        continue
4    else
5        Empty(all Ci)
6        提取 tfi(Tk)<l 的 Tk,并按权值大小排序
7        for each(排序后的 Tk)
8            if(Num(TermInLFHWSet)<NumSet)
9                AddToLFHWSet(Tk)
10       计算 TFIDF(LFHWSet)& DI(LFHWSet)
11       计算 TFIDF(otherTk)& DI(otherTk)
12       SortByWeight(allTk)& SelectTerm(Weight>w)
13       for each SortedTk
14           if(Num(TermInClassModel)<NumModel)
15           AddToClassModel(Tk)
16   if IsRoot(CurrentNode)
17       return
18   else
19       goto 1
```

5 实验数据及分析

实验所用的语料来源于《人民日报》1999 年光盘版,选取政治、经济、电脑、体育、教育和法律六大类(其中体育类又分为篮球、足球和排球三个小类),总共约 3000 篇文档。其中 1200 篇文档(每个大类约 200 篇,体育类中三个小类各 60 篇)作为分类的训练集,余下的 1800 篇文档作为测试集。

实验采用目前常用的向量空间模型 *VSM*(Vector Space Model)[9]的夹角余弦公式来计算向量间的相似度。假设类别标准向量为 C,待分类文档向量为 d,两者的相似度可以用这两个向量间的夹角余弦值来度量,计算公式如下:

$$sim(C,d)=cost(C,d)=\frac{\sum_{k=1}^{n} W_{C_k}\times W_{d_k}}{\sqrt{\sum_{k=1}^{n}(W_{C_k})^2}\times\sqrt{\sum_{k=1}^{n}(W_{d_k})^2}} \qquad (9)$$

实验测试了四种算法,即传统的 *TF*、*TF–IDF* 算法、结合 *DI* 的新算法以及结合 *DI* 和 *LFHW* 的新算法。首先计算各个特征项的 w_{ik},然后计算各个类别向量及测试集中待测文档的向量。单层的分类系统需要计算六个大类的类别向量,再通过相似度计算得到待分类文档的所属类别。多层的分类系统需要计算六个大类和三个体育小类的类别向量,然后先与大类向量比较得到各个文档所属的大类类别,如果属于体育类,再与小类向量比较得到其所属的小类类别,所以多层的分类系统共需要九个类别的测试数据。在实验中,采用了几种不同数量的测试集测试各种特征加权算法的分类精确度。分类精确度的定义如下:

$$precision=\frac{\text{正确分到该类别的文档数}}{\text{分到某个类别中的文档数}} \qquad (10)$$

各个类别的分类精确度的平均值为整个分类的精确度。

通过表 6 可以得出:在所有的算法中,传统的 *TF* 算法的分类精确度最差,*TF–IDF* 算法要优于 *TF* 算法,但略差于其它算法。在 *TF–IDF* 算法基础上考虑 *DI* 和 *LFHW* 因素算法的分类精确度最高。同时,测试的文章数越多,分类精确度就越高。从表 6 的多层分类的精确度来看,应用新算法后的分类精确度也明显高于传统算法。因此,改进的加权算法同样适用于多层次分类体系。

表 6 各种不同加权算法的分类精确度

加权算法 / 测试文章数		分类精确度 A1	A2	A3	A4
单层	30*6	72.6	80.1	80.4	86.3
	60*6	75.4	84.3	84.7	87.9
	90*6	75.9	84.5	85.3	87.6
	120*6	80.3	86.6	89.6	89.7
	150*6	80.7	86.9	89.3	90.2
	300*6	82.1	89.3	90.8	91.5
多层	30*9	73.3	78.3	81.2	85.3
	60*9	75.5	83.1	84.7	86.3
	90*9	76.9	85.5	86.1	88.6

注:A1:传统 *TF* 算法 A2:传统的 *TF–IDF* 算法
A3:结合 *DI* 的新算法 A4:结合 *DI*、*LFHW* 的新算法

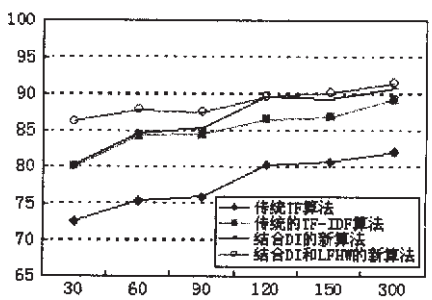

图 1 表 6 中的单层分类部分

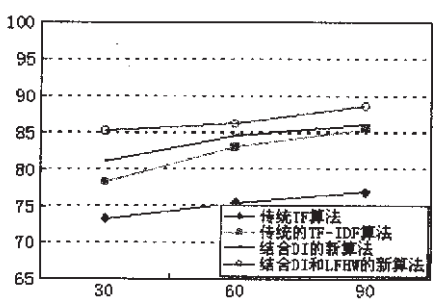

图 2 表 6 中的多层分类部分

6 结束语

特征加权算法的选择对文本自动分类系统的精确度有很大影响。该文研究了传统的加权算法,并在分析其不足的基础上,提出了一种考虑类间类内分布信息与低频高权特征的特征加权改进算法。实验证明,改进的加权算法与传统算法相比,在分类的精确度上有更好的表现。(收稿日期:2004 年 7 月)

参考文献

1.李晓黎,刘继敏,史忠植.概念推理网及其在文本分类中的应用[J].计算机研究与发展,2000,37(9):1032~1038

2.James Auen.Natural Language Understanding[M].The Benjamin/Cummings Publishing Company,1991–05

3.Apte C,Damerau F J,Weiss S M.Automated Learning of Decision Rules for Text Categorization[J].ACM Trans On Inform Syst,12(3):233~251

4.Salton G,Buckley B.Term–weighting Approaches in Automatic Text

(下转 220 页)

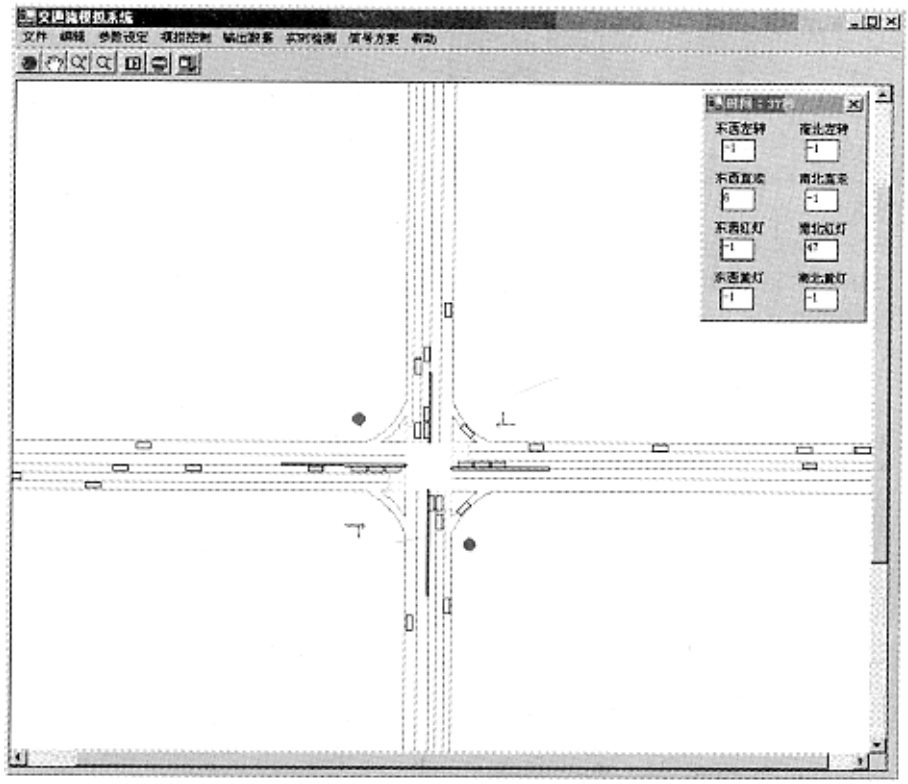

图 8 汕头市金砂汕樟交叉口交通流模拟显示

5.3 模拟系统显示

汕头市金砂汕樟交叉口交通流模拟显示，见图 8。

6 结论

Agent 技术是计算机与人工智能领域的最新研究成果。利用该技术构建交通模拟系统中智能个体，并结合模糊控制理论实现对现实世界的交通流模拟，为解决现代城市所面临的日益纷繁复杂的交通难题提供了一条有效而又经济的途径。文章就个体技术抽象交通模拟系统进行了详细的剖析，并重点讨论了模糊控制技术对交通个体决策方案的建模，最后利用该系统对汕头市金砂路的交通流实例进行模拟。模拟的结果不仅客观地再现了该路口实际采用的交通方案，同时还实现了对实际交通方案的优化，达到了令人满意的效果。由此可见，在城市交通迅猛发展的今天，利用先进的计算机原理和技术去模拟现实世界的交通过程，将在城市道路规划、城市交通流分配、城市物流管理、交通信号灯控制优化等各个相关领域中发挥着越来越大的作用。（收稿日期：2004 年 7 月）

参考文献

1.江斌，黄波，陆锋.GIS 环境下的空间分析和地学视觉化[M].高等教育出版社，2002

2.刘曙光，魏俊民，竺志超.模糊控制技术[M].中国纺织出版社，2001

3.刘运通，石建军，熊辉.交通系统仿真技术[M].人民交通出版社，2002；

4.邹智军.城市道路交通仿真研究[M]

5.QI YANG.A Simulation Laboratory for Evaluation of Dynamic Traffic Management Systems[M]

6.Jean-Claude Thill.Geographic information systems for transportation in perspective[M]

7.张安胜，董敏，林建臻.城市道路交通流仿真算法研究[J].计算机工程，2002；(8)

8.林勇，蔡远利，黄永宣.城市交通系统的微观仿真研究[J].计算机工程与应用，2002；38(20)：13~15

9.杜怡曼，贾顺平.国外城市交通微观模拟系统简介[J].研究与发展

10.商蕾，高孝洪，孙俊.微观交通仿真模型研究[J].交通科学，2003；(2)

（上接 184 页）

Retrieval[J].Information Processing and Management，1998；24(5)：513~523

5.张月杰，姚天顺.基于特征相关性的汉语文本自动分类模型的研究[J].小型微型计算机系统，1998；19(8)：49~55

6.Larkey L S.A Patent Search and Classification System[C].In：proceedings of DL-99，4th ACM Conference on Digital Libraries Berkeley，CA，1999：179~187

7.鲁松，李晓黎，白硕等.文档中词语权重计算方法的改进[J].中文信息学报，2000；14(6)：8~20

8.刘少辉，董明楷等.一种基于向量空间模型的多层次文本分类方法[J].中文信息学报，2001；16(3)：8~26

9.Salton G，Lesk M E.Computer Evaluation of Indexing and Text Processing[J].Association for Computing Machinery，1968；15(1)：8~36