Chinese Word Classification Based On Statistics

Zhao Shi-wan, Xia Ying, Ma Shao-ping, Wang Yu, Su Zhong
State Key Laboratory of Intelligent Technology and System
Dept. of Computer Science and Technology, Tsinghua University
Beijing 100084

Abstract: Chinese words classification based on statistics plays an important role in Natural Language Processing, such as speech recognition, intelligent Chinese input method, and so on. We first do statistics and calculation work on the large-scale corpus text, and then use the average mutual information as the global cost function for clustering all Chinese words into a predefined number of classes with a hybrid top-down splitting and bottom-up merging approach. The result of classification is encouraging and can be used in the class-based language model.

Keywords: Chinese words classification, Mutual information, Class-based language model

基于统计的中文词分类*

赵石硕 夏莹 马少平 王昱 苏中
（智能技术与系统国家重点实验室，清华大学计算机系，北京 100084）

摘要： 基于统计的中文词分类在语音识别、汉字智能输入等自然语言处理领域有着重要的应用。本文以平均互信息作为评价函数，对大规模的语料库进行统计，采用自顶向下分裂和从下而上合并相结合的方法进行自动分类，得到较好的分类结果，可以在基于类的语言模型中作为词类使用。

关键词： 中文词分类， 互信息， 基于类的语言模型

一、引言

基于统计的中文词分类在自然语言处理领域有着重要的应用。机器自动生成的词类可以取代文法的词类；在分类基础上建立的基于类的语言模型可以应用于语音识别、OCR、汉字智能输入等许多领域。众所周知，基于词的语言模型在自然语言处理的许多方面取得了巨大的成功。然而，基于词的语言模型也存在着许多的问题，如参数空间庞大，训练数据不足，数据稀疏等。词的分类可以在一定程度上解决上述问题。在计算语言学方面的应用中，不管是采用统计的方法，还是采用文法的方法，对词类进行处理都比对单个的词进行处理时问题的复杂度要小的多。我们用基于类的语言模型取代基于词的语言模型，可以减小模型的参数空间，减少系统对存储空间的要求。从而可以在小型的系统如掌上电脑、移动电话上建立基于类的语言模型，实现智能输入等。

词的分类是建立基于类的语言模型的基础。

无论是针对中文，还是别的语言，人们对词的分类算法已经做了许多的研究。

Brown 等人[1]提出了两个词的自动分类算法。在他们实现的两个分类算法中，都是利用平均互信息作为评价函数。*算法I.* (1) 首先将每一个词都当成一个单独的类，然后计算所有相邻类的互信息；(2) 将互信息损失最少的两个类合并；(3) 经过 V−C 次合并得到 C 个类；(4) 在得到 C 个类以后，对词汇表中的每一个词，将它移到一个使得平均互信息最大的类中，重复该步骤直到互信息不再增加为止。然而，他们认为，当词汇表的大小超过 5,000 时，这个算法是不可行的。*算法II.* 对一个大的词汇表，(1) 将 C 个频度最高的词作为 C 个单独的类；(2) 将未被分配的词中频度最高的一个词作为第 C+1 类，然后将这 C+1 个类中互信息损失最少的两个类合并；(3) 经过 V−C 步后，词汇表中的 V 个词被分成 C 个类。用这个方法，一个有 260,741 个英文单词的词表被分成了 1,000 类。

* 国家重点基础研究（G1998030509）、自然科学基金和 863 高技术项目

0-7803-5995-X/00/$10.00 ©2000 IEEE

Chang 和 Chen[2]在他们的论文中，以混乱度作为全局最优评价函数，提出了一个模拟退火的词分类算法：(1) 初始化：将每个词随机分配到一个类中，类的总数是事先定义好的。(2) 移动：随机的选取一个词，将该词重新分配到一个随机选取的类中。 (3) 接受或者返回：如果混乱度的改变在控制的范围之内〔在一定的范围内增加或减少〕，则接受新的分配，否则，撤销刚才 2 的操作。(4) 循环：重复上述两个步骤，直到混乱度收敛为止。该算法试图找出一个全局最优的分类方案，但是在训练集比较大的时候，算法的时间复杂度太大。

Gao 和 Chen[3]提出了一个自顶向下的二叉树分裂的方法，他们利用词的上下文的方向性，同时得到两个分类二叉树。McMahon[4]在他的论文中，提出了一个类似退火的分类算法。李涓子[6]在她的博士论文中，提出了一种聚类的算法。她认为聚类过程主要由三个部分组成：聚类时词分布的描述方法，聚类采用的控制策略以及控制聚类过程的目标函数。她在聚类时采用自顶向下的方法，词的分布信息用的是词的二元同现关系，利用信息论中的熵作为聚类时的目标函数。

上面描述的自顶向下分裂的方法和从下而上合并的方法，两者具有一定的互补性。在自顶向下的方法中，上层的失误在下层是无法纠正的，而且下层的分类结果精确度较低。因此，在本文中，我们采用自顶向下分裂和从下而上合并相结合的方法。我们使用平均互信息作为分类的全局评估函数，分类过程分为两个步骤，首先，我们采用合并的方法将词表中的一些词聚在一起，形成一些小的词类。在第二个阶段，我们把第一步得到的词类作为一个单独的词来加以考虑，然后采用自顶向下的方法，对整个词表进行分类。在实际的工作中，我们首先对大规模语料文本进行了统计和计算工作，得到词的一元和二元信息，在这个基础上，我们进行了词的分类。然后建立了一个基于类的二元 Markov 语言模型，实现了一个智能拼音输入法。

我们对实现的系统进行了一系列实验，实验结果是令人满意的。

本文第二节介绍了我们采用的分类算法，第三节给出了分类的结果及其在基于类的语言模型中的应用，第四节给出了我们的一些结论。

二、中文词分类算法

2.1 互信息的计算公式 词分类算法的实现跟采用的评价函数密切相关。本文采用平均互信息作为全局评价函数对汉字进行分类。根据信息学原理，熵的定义如下：

$$H(X) \equiv \sum_{x \in X} p(x) \log \frac{1}{p(x)}$$

其中 X 是一个离散的随机变量，其概率分布为 $p(x)$，$x \in X$。熵 $H(X)$ 是一个描述随机变量 X 的不确定性的统计量，一个随机变量的熵越大，它的不确定性也就越大。我们通过上面的公式导出两个随机变量之间的互信息公式。

$$M(X,Y) \equiv H(X) - H(X \mid Y)$$

从上面的公式中我们可以看出，在已知 Y 的情况下，随机变量 X 的不确定度程度会减小，而两者之间的互信息表明了这个减少的程度。在自然语言中，词类 $\{C_1, C_2, C_3, \cdots\cdots C_N\}$ 的分布显然也满足随机分布，我们同样可以得到词类的互信息计算公式如下：

$$M_a(f) = \sum_{C_i} \sum_{C_j} P(C_i, C_j) \times \log \frac{P(C_i, C_j)}{P(C_i)P(C_j)}$$

其中 $P(C_i)$，$P(C_j)$ 及 $P(C_i, C_j)$ 分别表示词类 C_i、C_j 出现的概率及它们之间的同现概率。在本论文中，我们对 200 兆字的语料进行了统计，在统计的基础上，我们用下面的公式来进行计算：

$$P(C_i) = \frac{\sum_{w \in C_i} N_w}{N_{total}}$$

$$P(C_i, C_j) = \frac{\sum_{w_1 \in C_i} \sum_{w_2 \in C_j} N(w_1 w_2)}{N_{total}}$$

其中 N_w 表示词 w 在统计的语料库中出现的次数，而 N_{total} 是所有词在整个语料库中出现的次数。$N(w_1 w_2)$ 表示词对 $w_1 w_2$ 在语料库中出现的次数。

2.2 分类算法的实现　算法的流程分为两个阶段，第一阶段，采用合并的方法。（1）我们将词汇表V里的每一个词都当成一个单独的类；（2）对任意的两个类C_i和C_j，计算$M(C_i,C_j)$；（3）将互信息损失最少的两个类合并成一个类；（4）经过$V-C_1$次合并得到C_1个类。如图1所示：

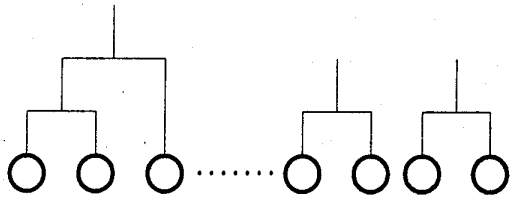

图1. 对词汇表V进行合并得到C1个类

第二阶段，将前面得到的C_1个类当成C_1个单独的词，我们给每一个词引进一个整数结构[5]，让每个结构代表一个词。这样就得到一棵二叉树，整个词汇表是根节点，整数的每一位将词汇表分成两棵子树，0代表左子树，1代表右子树。开始的时候，所有的整数都是随机生成的。分类的过程是处理每个结构的第一位，然后依次是第二位，第三位等等。每一位的处理流程可以分为以下的六个步骤。（1）评估：计算当前分布下的类的平均互信息；（2）试探性移动：将一个词移动到它所在子树相对的子树；（3）重新评估：计算在新的分布下，互信息的增加值；(4)恢复：撤销刚才的词的移动，恢复原来的分布；（5）按上面的步骤，遍历整个词汇表，找到使互信息增加最大的词，进行移动；（6）循环上述步骤，直到互信息不再增加为止。下图给出了对第二层进行分类时的情形。

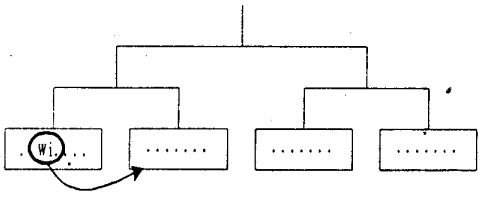

图2. 对第二层进行分类时的情形

三、实验结果及分析

3.1 对语料库的统计　在实验中，我们总共使用了200兆字的语料，包括95、96年的人民日报和90、91、92年的新华社报导。

在进行统计前，我们先对语料进行分词，我们采用最大长度匹配的分词方法。分词后，我们把不同的字母（A-Z），数字（0-9）和其他同类的符号分别归为一个标记。主要是考虑到：首先它们有着一致的同现搭配范围；其次它们对一确定的标记有一致的同现信息；把它们合并为同一个标记，使得该标记与其他标记的同现概率更可信，还可以有效地减少标记个数和减小参数空间。

如上所述，当词汇表的大小为N时，N个标记的统计需要一个$N\times N$的矩阵来存储任意两个标记的同现信息。由于我们要用4个字节来存储同现信息，因此需要$N\times N\times 4$的存储空间。不过，二元同现概率矩阵是一个稀疏矩阵，矩阵中的非零元很少，因而我们可以采用压缩存储的方法。如果用链表的结构来存储数据的话，每个元素要占用额外的4个字节来存储指针，而且，在查找时，要沿着一条指针链查找，时间开销也比较大，因此，链表的结构也是不可取的。我们在统计中采用了可变长度的数组来存储二元同现概率，对于每个标记还需花费4个字节存储数组长度。

3.2 分类结果　在统计完语料库后，我们对频度最高的10,000个词进行了分类。在第一阶段，我们先将这10,000个词合并成5,000个小的词类，然后再将这5,000个词类当成一个整体进行分类，最后得到512个类。表1列出了部分的分类结果。

表1. 基于词的一些分类结果

Class 62：北京 美国 法国 中共 苏联 南非 印度 泰国 加拿大 全体 埃及 西亚 瑞典 波兰 越南 约旦 荷兰 芬兰 奥地利 长春 港澳 秘鲁 八一 三国 印尼 海口 捷克 西德 挪威 议院 缅甸

Class 119：他们 双方 人们 一切 老人 有人 分钟 那些 你们 患者 百姓 顾客 病人 儿子 对方 老师 同学 父母 本人 否则 丈夫 也许 乘客 将来 眼睛

Class 127：其中 现在 有的 如果 但是 尽管 当时 针对 百万 另外 无论 除了 十万 千万 今日 即使 以往 不管 个个 其余 绝对

3.3 基于类的语言模型 在词的二元语言模型中，我们用下面的公式来预测下一个词的出现概率：

$$P(w_2) = P(w_2 \mid w_1)P(w_1)$$

如果我们采用基于类的语言模型，我们采用如下的公式：

$$P(w_2) = P(w_2 \mid C_2)P(C_2 \mid C_1)P(C_1)$$

$$P(w \mid C) = \frac{N_w}{\sum_{wc \in C} N_{wc}}$$

$$P(C_2 \mid C_1) = \frac{\sum_{w1 \in C_1, w2 \in C_2} N_{w1w2}}{\sum_{w1 \in C_1} N_{w1}}$$

人们通常采用混乱度来衡量一个语言模型的好坏，混乱度 PP 在语音识别中是衡量一个语言模型品质的重要尺度。它也被称为模型的平均分歧因子。对于词的和类的二元语言模型，PP 分别为：

$$\exp\left(-\frac{1}{L}\sum_{i=1}^{L}\ln\left(P(w_i \mid w_{i-1})\right)\right)$$

$$\exp\left(-\frac{1}{L}\sum_{i=1}^{L}\ln\left(P(w_i \mid C_i)P(C_i \mid C_{i-1})\right)\right)$$

我们利用前面得到的词类，建立基于类的语言模型。从表 2 中，我们可以看到，基于类的语言模型比基于词的语言模型的混乱度有不同程度的下降。因此，在语音识别等应用领域，我们采用基于类的语言模型，可以改善系统的性能。

表 2. 两种语言模型的混乱度的比较

混乱度	测试集一	测试集二	测试集三	测试集四
基于词的二元模型	256.1	238.3	391.2	437.4
基于类的二元模型	190.8	157.3	322.8	335.1

四、结论

在本文中，我们提出了一个新的分类算法，本算法将自顶向下分裂和从下而上合并两种方法结合起来，综合了两者的优点。在自顶向下的算法中，越是二叉树的底层，分类结果越不精确。而且一旦上层出现错误，后面就没法纠正。因此，我们在采用自顶向下的方法前，先采用合并的方法将词聚成小的词类。在分类的基础上，我们实现了一个基于类的语言模型，并将它和基于词的语言模型进行了比较，取得了比较好的结果。

参考文献

【1】 P. F. Brown and V. J. Della Pietra. Class-based n-gram Models of Natural Language. Computational Linguistics, 18(4): 467-469, 1992.

【2】 C-H Chang, C-D Chen. A Study on Corpus-Based Classification of Chinese Words. 1995

【3】 Jun Gao, XiXian Chen. Probabilistic Word Classification Based on Context-Sensitive Binary Tree Method. 1997

【4】 McMahon, J. Statistical language processing based on self-organising word classification. Ph. D. thesis, The Queen's University of Belfast. 1994.

【5】 McMahon, J. and Smith, F. J. Improving Statistical Language Model Performance with Automatically Generated Word Hierarchies. Computational Linguistics. 1995

【6】 李涓子， 汉语词义排歧方法研究， 博士论文， 清华大学， 1999

【7】 C-H Chang. Word calss discovery for contextual post-processing of Chinese handwriting recognition. In Proc. Of COLING-94, Kyoto, Japan August 1994.

【8】 F. Jelinek and R. Mercer. Classifying Words For Improved Statistical Language Models. IEEE, 1990.

【9】 David J.C. MacKay. Information theory, Inference and Learning Algorithms. *1995~1998*