博士学位论文

基于核方法的语义角色标注研究

Kernel-based Semantic Role Labeling

车 万 翔

哈尔滨工业大学

2008 年 12 月

国内图书分类号：TP391.2
国际图书分类号：681.324

工学博士学位论文

基于核方法的语义角色标注研究

博 士 研究生：车 万 翔
导　　师：李　生　教 授
副　导　师：刘　挺　教 授
申 请 学 位：工学博士
学 科、专 业：计算机应用技术
所 在 单 位：计算机科学与技术学院
答 辩 日 期：2008 年 12 月
授予学位单位：哈尔滨工业大学

Domestic Classified Index: TP391.2
U.D.C.: 681.324

Dissertation for the Doctoral Degree in Engineering

Kernel-based Semantic Role Labeling

Candidate:	Wanxiang Che
Supervisor:	Professor Sheng Li
Co Supervisor:	Professor Ting Liu
Academic Degree Applied for:	Doctor of Engineering
Specialty:	Computer Application Technology
Affiliation:	School of Computer Science and Technology
Date of Defence:	December, 2008
Degree-Conferring-Institution:	Harbin Institute of Technology

摘 要

全自动的语义分析一直是自然语言理解的主要目标之一。通过深层语义分析，可以将自然语言转化为形式语言，从而使计算机能够与人类无障碍的沟通。为达此目的，人们已经进行了多年的努力，然而由于这一问题过于复杂，目前取得的效果并不理想。浅层语义分析是对深层语义分析的一种简化，它只标注与句子中谓词有关的成分的语义角色，如施事、受事、时间和地点等。其能够对问答系统、信息抽取和机器翻译等应用产生推动作用。语义角色标注是浅层语义分析的一种实现方式，具有定义清晰，便于评价的优点，近年来受到越来越多学者的关注。

目前主流的语义角色标注研究集中于使用各种统计机器学习技术，利用多种语言学特征，进行语义角色的识别和分类。近年的研究表明，影响语义角色标注系统性能的首要因素并非机器学习模型，而是使用的特征。因此，若想提高系统的性能，细致的特征工程工作是必不可少的。然而，随着越来越多特征的加入，特征之间的相互影响越来越严重，使得系统性能增长的趋势逐渐趋缓，并达到一个上限。为此必须寻找新的方法以解决这一问题。

基于核的方法通过对已有特征进行组合或者分解，将低维特征空间映射到高维特征空间，从而将在低维空间不容易区分的问题在高维空间加以解决，是一种可行的解决方案。

本文正是利用核方法这一优点，将其应用于语义角色标注这一问题中。除了使用已有的核方法外，还提出了多种新的核方法。

首先，我们构造了一个语义角色标注基线系统，该系统使用特征向量表示待分类对象，并在特征向量之上，使用基于多项式核的方法，自动的对特征进行组合。实验结果表明，当使用二次多项式核对特征进行两两组合时，该系统是目前已知的最好的基于单句法分析器的语义角色标注系统之一。

接着，我们针对基线系统中，特征向量很难恰当的表示结构化的特征这一问题，使用了卷积树核自动将较大的结构特征进行分解，并能够在多项式时间内进行核函数的计算。然而，通常的卷积树核混淆了语义角色标注中不同的特征，因此，我们提出了混合卷积树核融合多种树核，来对不同种类的特征分别进行建模，最终获得优于标准卷积树核的性能。然后将混合卷积树核与多项式核进行融合，得到的复合核取得了比单独使用两种核都好的结果。

但是，标准卷积树核要求两棵子树之间必须是精确匹配的，而不考虑结构相似，语义角色相同的情况。因此，我们提出了新的句法驱动卷积树核，在核函数的设计过程中，融入了语言学知识，容许结构和节点的近似匹配，最终取得了较标准卷积树更好的性能。最后同样与多项式核进行融合，并取得了更好的性能。

最后，我们使用基于核的方法，实现了一个了目前最好的中文语义角色标注系统。主要贡献在于提出了更适用于中文的新特征，同时首次将核方法应用于中文语义角色标注中，得到了与英文相同的性能趋势，从而也证明了我们提出的核方法的有效性。

关键词　语义角色标注; 多项式核; 卷积树核; 混合卷积树核; 句法驱动的卷积树核

Abstract

Automatic semantic parsing has always been one of the main goals of natural language understanding. Through deep semantic parsing, the natural language can be translated into the form language, so that computers can communicate with human beings freely. To that end, it has been going on for years of effort. However, because this issue is too complex, the results are not very idea now. Shallow semantic parsing is a simplified deep semantic parsing. It only labels predicate related constituents with semantic roles in a sentence, such as agent, patient, time, place, and so on. The technique can promote many applications, such as question and answering, information extraction, and machine translation. Semantic role labeling is a way to achieve shallow semantic parsing. It has many advantages, such as clear definition and easy to evaluate. More and more researchers have paid much attention on it in recent years.

At present, the mainstream studies of semantic role labeling focus on the use of a variety of statistical machine learning techniques, the use of all kinds of linguistics features, and to identify and classify semantic roles. In recent years, studies have shown that machine learning models is not the primary factor to effect the semantic role labeling performance, but the use of the features. Therefore, in order to improve the system performance, detailed features engineering work is essential. However, as more and more features have been added, the interaction among features has become more and more serious. It makes the growth trend of system performance gradually slowing down and reaching an upper bound. So we must find new ways to solve this problem.

Through the combination or decomposition of features, kernel-based methods can map low-dimensional feature space into higher-dimensional feature space. Thereby, it makes the problem which is not easy to distinguish in low-dimensional feature space becoming addressed in high-dimensional feature space.

We make use of the advantages of this method and apply to the semantic role labeling task. In addition to using existing kernel-based methods, we propose a variety of new methods.

At first, we construct a baseline semantic role labeling system, which uses a fea-

ture vector to represent a classification object and uses a polynomial kernel to combine features automatically. The evaluation results show that the system is one of the state-of-the-art systems, which base on single syntactic parser.

Then, for our baseline system, it has the problem of difficultly representing structure features. We use the convolution tree kernel to decompose these larger structure features and compute the kernel function in polynomial time. However, the traditional convolution tree kernel confuses the different features used in semantic role labeling. Therefore, we provide hybrid convolution tree kernel to make fusion different convolution tree kernels, which can model different features with different kernels. The eventuation results show that the novel method is better than the traditional convolution tree kernel. At last, we combine the hybrid convolution tree kernel and the polynomial kernel into a composite kernel. The composite kernel outperforms either of the two individual kernels.

However, the standard convolution tree kernel requires exact matching between two sub-trees, without taking into account the similar structures which have the same semantic roles. Therefore, we propose a new grammar-driven convolution tree kernel. In the design process of the kernel, we integrate linguistic knowledge, and allow the node and the structure approximate matching. The new kernel outperforms the standard convolution tree kernel. Finally, combined with the same polynomial kernel, the system achieve a better performance.

At last, we use the methods described above to achieve the best Chinese semantic role labeling system. The main contribution is that we propose more new Chinese oriented features. At the same time, we use the above three kernel-based methods in Chinese semantic role labeling. The final performance trend is consistent with the English one, which also proves that our kernel-based methods are effective.

Keywords Semantic Role Labeling; Polynomial Kernel; Convolution Tree Kernel; Hybrid Convolution Tree Kernel; Grammar-driven Convolution Tree Kernel

目 录

Contents

第 1 章 绪论

1.1 课题背景及意义

1.1.1 课题背景

语言是信息的重要载体，为使计算机具有理解、处理和生成自然语言的能力，必须使计算机能够分析自然语言。语言分析一般分为三个层次：句法、语义、语用。句法分析关心的是词语如何排列形成正确的句子，并决定每个词语在句子中充当的结构角色。句法分析问题早已引起人们的广泛关注，并取得了积极的进展。所谓语义分析，指的是将自然语言句子转化为反映这个句子意义 (即句义) 的某种形式化表示。即将人类能够理解的自然语言转化为计算机能够理解的形式语言，做到人与机器的互相沟通。而语用分析则研究影响语言行为 (如招呼、劝说) 的标准和支配轮流发言的规则，目前在自然语言处理领域还鲜有研究。

对句子进行深入的语义分析，一直是从事自然语言处理研究的学者们追求的主要目标。例如对于句子：“ 张三吃了苹果 ” 和 “ 苹果被张三吃了 ”，虽然它们的表述形式不同，但含义相同，表示成语义的形式同为：“ 吃 (张三，苹果) ”。更确切地，语义分析指的是根据句子的句法结构和句中每个实词的词义推导出能够反映这个句子意义 (即句义) 的某种形式化表示。多年来，国内外自然语言处理界的学者们一直在探索有效的自动语义分析方法，期间经过了许多的挫折，也取得了一定的进展。

在最早期的自然语言理解研究领域，用自然语言回答特殊领域的问题是一个非常关键的任务[1] ，虽然句法分析是此任务的主要模块，但是语义理解对于找到答案也非常重要。语义分析在 20 世纪 70 年代受到越来越多学者的重视，包括使用自然语言进行数据库检索[2] ，以及小故事理解[3] 等等。

这一时期的研究主要集中在语义理解，知识表示和推理等复杂的问题上，开发出来的系统能够对特殊的句子或者小故事进行有趣的语义理解、推理等。然而，它们需要相当大量的和应用有关的知识工程的工作，因此相当脆弱，并且不容易扩展到新文本和新的应用领域。结果是开发出的系统能够对叙述性的故事进行相当深的理解，但是只能局限于几个小故事里[4] 。

由于上面的方法需要获取大量的知识，因此知识工程的进展成为了瓶颈，因此在 20 世纪 80 年代人们将注意力主要集中到了知识工程上。

为了避开困难的语义理解，自然语言研究学者开始将注意力集中在简单，但是实用的任务上。尤其是到了 90 年代，随着统计学习方法的发展，人们在一些简单的应用上取得了很大的进展，例如语音识别，词性标注，句法分析等等[5, 6] 。并且，产生了越来越多基于互联网的应用，这也要求我们要处理各种通用的语言现象[7] 。因此，现在许多自然语言处理的研究专注到信息检索领域[8, 9] ，而非人工智能领域的自然语言理解方向了。

近年，出现了一些使用统计方法来获取语义信息的研究。如基于语料库的词义消歧 (Word Sense Disambiguation ，简称 WSD) 技术[10, 11] ，它虽然触及了语义问题，但是这只是在理解单个词的层次上，而不是进行整个句子的理解。关于信息抽取的研究也触及了一些语义理解，然而现有方法使用相当低水平的字符、句法等模式抽取方法来获取特定的信息[12–14] 。

随着计算机处理能力的提高以及统计机器学习等理论的发展，浅层语义分析 (Shallow Semantic Parsing) 逐渐被研究人员所重视，它可以看作是一种通用的信息抽取技术，抽取的信息不再限定于某个类别 (如人名、地名等) ，而是抽取句子中相对通用的语义信息，如：某一动词的施事、受事等和领域无关的语义信息。本文的研究重点 — 语义角色标注(Semantic Role Labeling ，简称 SRL)[15, 16] ，是浅层语义分析的一种实现方式，其具有问题定义清晰，便于人工标注和评测等优点，同时又具有非常广泛的应用前景。

1.1.2 课题意义

自动的深层语义分析一直是人们所追求的目标，它是进行深层句义理解，乃至篇章理解的唯一途径。若不进行语义分析，我们虽然可以提高自然语言处理系统的速度，短期内也给自然语言处理带来了一些进步，但是并不能真正实现人类使用机器对语言进行正确理解的梦想。语义分析从提出到现在，虽然其经过了几多的兴衰，但是人们始终没有放弃这方面的努力。

语义角色标注，正是这种努力的一种尝试，其很可能成为通往深层语义分析的大道。语义角色标注具有任务明确，分析结果便于利用等优点，它符合现阶段人们对语义分析的理解，同时利用现有的软、硬件技术能够实现。因此我们说，现在进行的语义角色标注研究，既不会像早期的研究者一样目标过于远大，不易实现，也不会像后来的研究者一样只专注短期目标，过于实用。

从自然语言处理技术本身来看，现阶段使用机器学习的方法进行语言分

析是一个较为热门，且取得了不错效果的方向。而语义角色标注给人们提供了一个很好的测试机器学习技术的平台。它综合利用了分词、词性标注等底层的语言信息，以及高层的句法分析，命名实体识别等信息，人们从这些信息中可以挖掘各种特征，再利用各种机器学习算法，做到自动的语义角色标注。因此，我们可以说，只要处理好了语义角色标注问题，其它类似需要机器学习技术以及多种语言学特征的自然语言处理问题，如关系抽取等，都可以迎刃而解。

当然，语义角色标注毕竟只是自然语言理解的底层技术，最终要靠实际的应用体现其价值。在许多高层次的研究和应用上，语义角色标注都大有用武之地。如果语义角色标注问题得到有效的解决，将对包括信息抽取[17, 18] 、自动问答[19, 20] 、机器翻译[21] 、信息检索[22] 、自动文摘[23] 等在内的许多研究和应用产生巨大的帮助。

1.2 研究现状及分析

本节首先介绍什么是语义角色标注，然后介绍目前用于语义角色标注的主要资源，接着重点介绍目前语义角色标注常用的方法，最后介绍如何评价语义角色标注的性能以及国际上组织的一些评测。

1.2.1 语义角色标注定义

语义角色标注是浅层语义分析的一种实现方式，其具有问题定义清晰，便于人工标注和评测等优点，因此目人们更多的将注意力集中于语义角色标注上。

该方法不对整个句子进行详细的语义分析，而只是标注句子中某些短语为给定谓词 (动词、名词、形容词等) 的语义角色，这些短语作为此谓词的框架的一部分被赋予一定的语义含义，例如 “[委员会 Agent][明天 Tmp] 将要 [通过 V][此议案 Passive] 。”，其中，“ 通过 ” 为谓词，“ 委员会 ” 、“ 此议案 ” 和 “ 明天 ” 分别是其施事、受事和动作发生的时间。另外，语义角色标注不考虑时态信息，例如 “ 他要来北京。” 与 “ 他来北京了。”，虽然时态并不相同，但是语义角色表示是相同的，同为：“ 来 (他，北京) ”。同时，语义角色标注也不考虑谓词改变但语义不变的情况，例如 “ 曹雪芹写了《红楼梦》” 与 “《红楼梦》的作者是曹雪芹 ”，虽然它们的语义相同，但是由于谓词不同，语义角色标注的表示结果并不一样，需要根据不同的应用进行更深入的处理。

1.2.2 语义角色标注语料资源

和其它基于有指导机器学习技术的自然语言处理问题一样，要想进行语义角色标注，也需要语料资源的支持。目前，英语的语义角色标注资源较为丰富和成熟，比较知名的包括 FrameNet[24] 、PropBank[25] 和 NomBank[26] 三种。

其中，U.C.Berkeley 开发的 FrameNet 以框架语义为标注的理论基础对英国国家语料库进行标注。它试图描述每个谓词 (动词、部分名词以及形容词) 的语义框架，同时也试图描述这些框架之间的关系。从 2002 年 6 月发布至今，现共标注了约 49,000 句。其中每个句子都标注了目标谓词和其语义角色、该角色句法层面的短语类型 (如 NP ， VP 等) 以及句法功能 (如主语、宾语等) 。FrameNet 现包含 1,462 个目标谓词 (927 个动词， 339 个名词和 175 个形容词) 。图 1-1 是 FrameNet 中表示身体动作的一个语义框架以及对一个句子的标注实例。

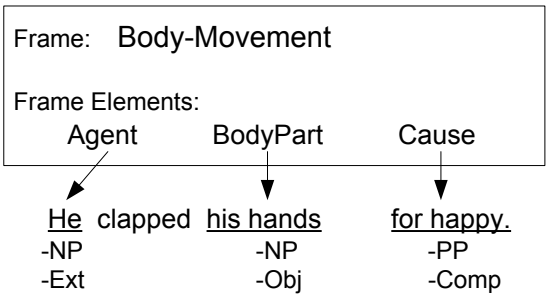


图 1-1 FrameNet 框架以及句子标注示例

Fig. 1-1 FrameNet and Sentence Annotation Example

PropBank 是宾夕法尼亚大学在 Penn TreeBank 句法分析语料库的基础上标注的语义角色标注语料库。与 FrameNet 不同的是，PropBank 只对动词 (不包括系动词) 进行标注，相应的被称作谓语动词。而且只包含 20 多个语义角色。其中核心的语义角色为 Arg0~5 六种， Arg0 通常表示动作的施事，Arg1 通常表示动作的影响等， Arg2~5 根据谓语动词不同会有不同的语义含义。它们的具体含义通常由 PropBank 中的 Frams (框架) 文件给出，例如 “buy” 的一个语义框架如图 1-2 所示。

此文件说明当 “buy” 取 01 号语义，做 “ 购买 ”(“purchase”) 的含义时，Arg0 代表购买者 (“*buyer*”) ， Arg1 代表购买的东西 (“*thing bought*”) 等等。

其余的语义角色为附加语义角色，使用 ArgM 表示，在这些参数后面，还需要跟附加标记来表示这些参数的语义类别，如 ArgM-LOC 表示地点，ArgM-TMP 表示时间等等。表 1-1 列举了 PropBank 定义的 18 个附加标记及其具体含义。图 1-3 是 PropBank 中对一个句子的标注实例。

与 FrameNet 相比，PropBank 基于 Penn TreeBank 手工标注的句法分析结果进行标注，因此标注的结果几乎不受句法分析错误的影响，准确率较高。而且它几乎对 Penn TreeBank 中的每个动词及其语义角色进行了标注，因此覆盖范围更广，可学习性更强。

为了弥补 PropBank 仅以动词作为谓词，存在标注过于粗略的缺点，纽约大学的研究人员开发了NomBank[26] 。与 PropBank 不同的是，NomBank 标注了 Penn TreeBank 中的名词性的谓词及其语义角色。例如：名词短语 “John’s replacement Ben” 和 “Ben’s replacement of John” 中，名词 “replacement” 便是谓词，Ben 是 Arg0 ，表示替代者；John 是 Arg1 表示被替代者。另外 NomBank

RoleFrame buy.01 “purchase”:
Roles:
Arg0: *buyer*
Arg1: *thing bought*
Arg2: *seller*
Arg3: *price paid*
Arg4: *benefactive*

图 1-2 “buy” 的语义框架示例

Fig. 1-2 A Role Frame Sample of “buy”

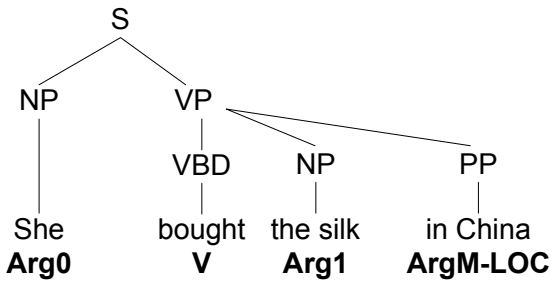


图 1-3 Propbank 中的句子标注示例

Fig. 1-3 PropBank Sentence Annotation Example

容许角色出现相互覆盖的情况，这也是与 PropBank 不同的。

除英语外，许多其它语言也建立了各自的语义角色标注库，例如：SALSA [27] 是基于 FrameNet 标注体系，大量标注的德语语料库； Prague Dependency

表 1-1 PropBank 附加标记列表

Table 1-1 A List of the Secondary Tags in PropBank

语义附加成分的 11 个附加标记	
ADV	adverbial, default tag (附加的，默认标记)
BNE	beneficiary (受益人)
CND	condition (条件)
DIR	direction (方向)
DGR	degree (程度)
EXT	extent (扩展)
FRQ	frequency (频率)
LOC	locative (地点)
MNR	manner (方式)
PRP	purpose or reason (目的或原因)
TMP	temporal (时间)
TPC	topic (主题)
谓语动词作为参数的 4 个附加标记	
CRD	coordinated arguments (并列参数)
PRD	predicate (谓语动词)
PSR	possessor (持有者)
PSE	possessee (被持有)
习惯用动词的 6 个附加标记	
AS	为、是、作、做
AT	在、于
INTO	成、入、进
ONTO	上
TO	到、至
TOWARDS	向、往

Treebank[28] 项目进行了大量的句法和语义标注 (捷克语) ，甚至包括指代消解的标注等。Chinese PropBank[29] 是宾州大学基于 Chinese Penn TreeBank[30] 标注的汉语语义角色标注资源，标注方法参考 English PropBank 。Chinese Nombank 也是宾州大学研制的，其将传统 English Nombank 的标注框架，扩展到对中文名词性谓词的标注。山西大学构建的 Chinese FrameNet (CFN)[31] 是基于框架语义参考英文 FrameNet 构建的中文语义角色标注语料库。

本文以 PropBank (包括英文和中文) 为实验数据，主要原因是其规模较大，覆盖范围广，同时其广为研究人员所采用，便于进行对比实验。当然，我们的语义角色标注方法不局限于 PropBank ，还可以推广到其它语料资源上。

1.2.3 语义角色标注方法

我们主要从标注步骤、标注单元、标注方法、特征构造、标注模型、对多个语义角色标注系统的融合方法、以及中文语义角色标注等七个方面来介绍构造一个语义角色标注系统需要的若干背景知识。

1.2.3.1 标注步骤 通常的语义角色标注分为 4 个步骤：剪枝 (Pruning) 、识别 (Identification) 、分类 (Classification) 和后处理 (Post-processing) 。

其中，剪枝指的是根据启发式规则，删除大部分不可能成为语义角色的标注单元[32] ，这样可以大幅减少待识别实例的个数，提高系统的效率。识别过程一般是对一个标注单元是否是语义角色加以判别，并保留识别成语义角色的标注单元，待下一步进一步分类究竟属于哪个语义角色类，这样也可以减少进入分类判别的实例的个数，加快处理速度。最后根据语义角色之间的一些固有约束进行后处理[33] 。这些约束通常包括，一个谓语动词不能有重复的核心语义角色并且语义角色不存在相互重叠或嵌套等等。

当然，并非所有的系统都包括以上 4 个步骤，特别是前两个步骤，其主要目的是提高处理效率，但随之带来的是召回率的下降，即损失了一些本应是语义角色的标注单元。因此，在某些系统中，去除了剪枝步骤。还有些系统合并了识别和分类步骤[34] ，直接对语义角色进行分类，也就是将非语义角色的标注单元也看成是一类。

1.2.3.2 标注单元 在上面提到的 4 个步骤中，每一步的处理对象都是标注单元，这些基本的处理单元可以是句法成分 (Constituent) 、短语 (Phrase) 、词 (Word) 或者依存关系 (Dependency Relation) 等等。

在图 1-3 的短语结构句法分析树中，每一个非终结节点，如 S ， NP ， VP

等，都是句法成分。一般认为每个语义角色是与某一句法成分相对应的。也就是说一个语义角色必然对应着一个句法成分，反之未必。如在图 1-3 的例子中， Arg0 对应一个 NP ， ArgM-LOC 对应一个 PP 等等。因此，现在多数的语义角色标注系统通常都是以句法成分为基本标注单元的[35] 。而且这种策略，在短语结构句法分析比较成熟的语言 (如英文等) 上表现得较好，有人甚至认为这种句法分析对语义角色标注是必须的[36] 。然而，在其它语言上，很难自动的获得这种深层句法分析的结果，而且现有的句法分析系统，在通用领域表现欠佳。为此有人试图将语义角色标注建立在浅层句法分析的基础之上[37] ，毕竟浅层句法分析的鲁棒性要好于深层句法分析。因为通过浅层句法分析只能获得非嵌套短语的信息，而不能获得全部的句法分析结果，也就是不能获得句法成分的分析结果，但是我们一般认为一个非嵌套的短语属于同一语义角色，因此产生了使用短语作为语义角色标注的基本元的系统[38] 。词是比短语更细的语言单位，有些语义角色标注系统也使用词作为标注的基本单位，然而效果并不如基于短语的和基于句法成分的理想[39] 。以上的方法都是建立在短语结构句法分析方法基础之上的， Hacioglu 使用依存句法分析结果进行语义角色标注[40] ，也取得了可以与基于短语结构句法分析的相比较的结果。我们可以直接使用依存句法分析器获得依存句法分析的结果，也可以转化短语结构句法分析的结果为依存句法分析结果。与基于短语结构句法分析的方法相比，基于依存句法分析不但可以利用短语之间的依存特征，而且只需要学习和预测与谓词有依存关系的短语为某种语义角色即可，因此也加快了标注的速度。

1.2.3.3 标注方法 语义角色的识别和分类步骤尤为重要，它们可以看做是分类问题。也就是说，人们可以逐一判断一个标注单元是否是某一动词的语义角色，更进一步的，可以预测其属于何种具体的语义角色。

最初人们使用基于规则的方法来解决分类问题，但是，此方法需要专家构筑大规模的知识库，这不但需要有专业技能的专家，也需要付出大量劳动。同时，随着知识库的增加，矛盾和冲突的规则也随之产生。为了克服知识库方法的缺点，人们后来使用机器学习的方法来解决此问题。该方法的优点是不需要有专业技能的专家书写知识库，只需要有一定专业知识的人对任意一种语言现象作出适当的分类即可。然后以此为训练数据，再使用各种学习方法构造性能卓越的分类器。该方法通常称为有指导学习 (Supervised Learning) 方法。虽然它能够较好的解决一些已有大量正确标注语料库的自然语言处理问题，但是通常，我们获得这种语料库的代价也是昂贵的。为此，人们试图使

用未标注的语料库直接进行学习，这种方法被称作无指导学习 (Unsupervised Learning)[41] 。或者只借助少量标注语料，利用大量未标注语料的半指导学习 (Semi-supervised Learning) 。然而无论是无指导学习，还是半指导学习，其理论都不甚完备，效果也不如有指导学习方法。因此，人们目前的主要精力还是集中在有指导学习方法上。

使用机器学习方法的语义角色标注遇到的情况也是类似，到目前为止，大多数工作集中在使用有指导学习方法进行语义角色标注，其间也有一些学者试图使用无指导方法进行语义角色标注[42, 43] ，但是性能也都没有基于有指导的方法高。

有指导机器学习方法：采用机器学习方法的优劣，对于语义角色标注系统的性能有重要的影响。因此，我们需要考察不同学习方法的优缺点，以便选择适合语义角色标注的方法。

下面给出有指导学习方法的一个形式化定义：人们通常使用特征向量表示一个实例，是实例的一种数值化的表示方式。也就是说，一个实例被转化为特征向量 $\vec{x}$ ，其中 x_i 为 m 维特征向量 $\vec{x}$ 的第 i 个元素，$\vec{x} \subseteq \Re^m$ 。机器学习算法的目的就是对于给定的一组训练数据$(\vec{x}^1, y^1), (\vec{x}^2, y^2), \ldots, (\vec{x}^n, y^n)$ ， n 是训练样本的个数，学习一个分类函数 f ，使得对于给定新的特征向量， f 能够将其正确的分类，即 $f(\vec{x}') = y'$ 。其中对于二元分类问题 $y^i \in \{-1, +1\}$ ，多元分类问题 $y^i \in \{1, 2, \ldots, k\}$ ， k 是输出类别个数。

一般有指导的机器学习包括预处理、人工标注、训练和测试等步骤，图 1-4 展示了一个基于文本的有指导机器学习过程。

目前，语义角色标注主要用到的有指导机器学习方法有生成模型和判别模型两种：

1、生成模型： 生成模型是最早用于语义角色标注的分类模型[44] ，其假设观察值 (词、短语、词性等) 是由一系列的隐藏值 (语义角色) 生成的，当使它们的联合概率最大时，即 $r^* = \arg\max_r P(w, r)$ ，其中 w 是词序列， r 是语义角色序列，这时可以获得最好的语义角色标注结果 r^* 。Thompso 等人[45] 又对其进行了完善，使其更具通用性。该模型的最大优点是训练速度快，最终性能对训练语料的依赖性不强。但是其对数据较差的描述能力和较强的特征独立性假设也是其性能不尽如人意的主要原因。自然语言处理中经常采用的 Naive Bayes[46] 分类方法也是一种联合概率模型，其同样具有上述的优缺点。

2、判别模型： 判别模型直接估计分类的最终的优化目标 — 条件概率。这一过程通常是通过迭代的方法估计一些优化的组合系数来完成的。判别模

型一般包括：线形插值、支持向量机、感知器、最大熵等等。下面逐一进行简要介绍：

线性插值 (Linear Interpolation)：Gildea 等人改进了其先前的工作，采用线形插值方法[16] 平滑概率模型，其中的平滑参数通过训练语料统计获得。虽然该方法算法较为简单，其获得性能也没有后面介绍的支持向量机 (Supported Vector Machines， SVM) 等方法高，但是作者使用的语义角色标注特征多为后继研究者所借鉴。

支持向量机 (Supported Vector Machines， SVM)：支持向量机是基于Vapnik[47] 提出的统计学习原理构建的一种线形分类器。之后成功的应用于文本分类[48] 等自然语言处理领域。支持向量机的基本思想是使构成的超平面分割训练数据能够获得最大的间隔 (Large Margin)。由于支持向量机理论的完备以及其较好应用效果，因此经常被用作分类器处理各种自然语言处理问题。最典型的是 Pradhan 等人的工作[49]，他们使用支持向量机获得了较好的语义角色标注效果。然而，支持向量机也并非完美，其存在一些固有的缺点，训练效率低就是其最主要的问题。另外，支持向量机设计的初衷是处理二

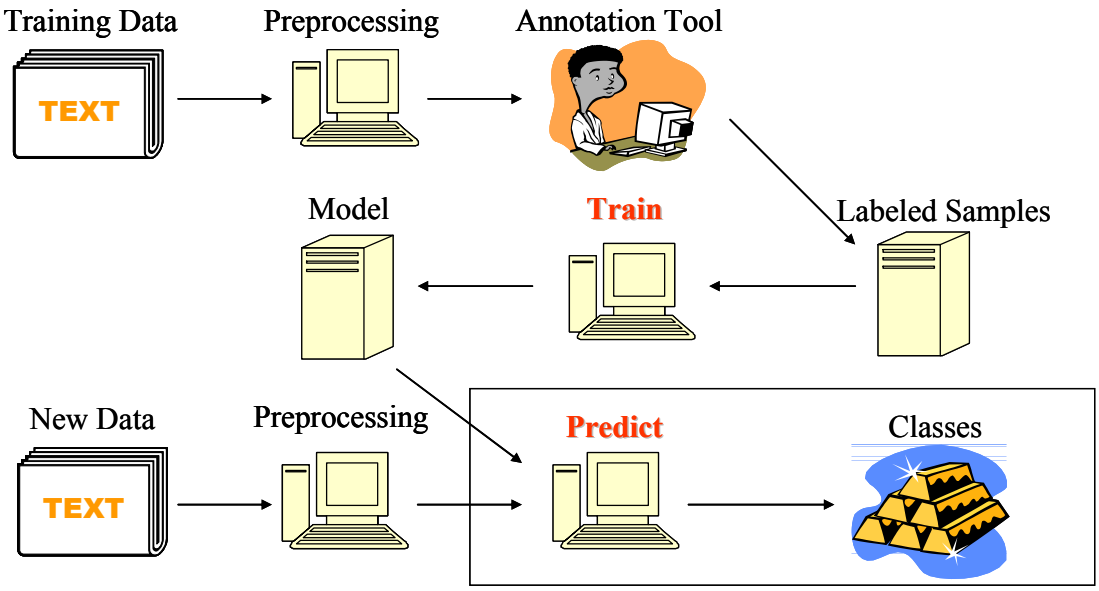


图 1-4 有指导机器学习一般过程

Fig. 1-4 General Process of the Supervised Machine Learning

元分类问题，目前对于其如何处理多元分类问题，还没有一个统一的结论，而且目前的处理方法往往效率低下。支持向量机的另一个缺点就是其对于输出结果不能从概率上进行解释，也就是不能准确地给出各个输出结果的概率分布，这就给一些利用概率结果的后处理应用带来了麻烦。

感知器 (Perceptron)：感知器分类器最早由 Rosenblatt[50] 提出，其又被称作错误驱动的方法，基本思想是对于权值向量 ω 和一个新的训练数据 x^i，如果权值向量 ω 对应的超平面不能将 x^i 正确地分开，就可利用 x^i 来修正 ω。可对训练数据反复迭代这一过程，直至所有的训练数据都能正确分开 (如果是线性可分)。因此，它也是一种线性分类器，能够快速处理数据线性可分问题。然而，对于线性不可分问题则无能为力了，特别是待处理的实际问题多为线性不可分的。因此感知器在经过了一段时间的发展后，沉寂了将近 20 年，直到后来人们想出各种使感知器能够处理非线性问题时它才又焕发了活力。人们想到的方法不外乎两种：1、使用多层感知器，即人工神经网络 (Artifical Neural Network)；2、使用核 (Kernel) 方法，将低维不可分问题映射到高维空间，变成线形可分问题。而人工神经网络的方法由于其要求较大的数据量，对学习结果不可控制，而且还很难加入人们的先验知识，因此在对自然语言处理等复杂问题的处理上，往往显得力不从心，于是逐渐淡出人们的视野。而核方法克服了人工神经网络方法的缺点，逐渐为人们所重视。后面将详细介绍核方法。这里我们介绍的感知器方法是被称作基于投票的感知器算法 (Voted Perceptron)[51]，它是对原始感知器方法的一种改进，其中融入了支持向量机中最大边缘 (Large Margin) 的思想。Carreras 等人[52] 使用该方法进行语义角色标注，获得了与支持向量机差别不大的性能，而且训练速度要较支持向量机快很多。

SNoW：SNoW (Sparse Network of Winnows)[53]，即稀疏 Winnow 网络，它是对原始 Winnow 算法的一种改进，可以高速处理大数据量，多类别问题。其提出者，UIUC 大学 Roth 等人，开发并共享了 SNoW 学习工具包*，并与其合作者 Koomen 等[54] 将该学习方法成功的应用于语义角色标注。

Boosting：Boosting[55] 是另一类基于判别式的分类方法，其基本思想是组合多个弱 (Weak) 分类器 (只要比随机分类好)，Schapire 等人证明组合这些弱分类器可以形成一个强分类器。Màrquez 等人[56] 以及Surdeanu 等人[57] 使用 AdaBoost (Boosting 思想的一种实现) 算法进行语义角色标注，效果也很好。

最大熵模型 (Maximum Entropy Models)：最大熵模型又被称作 Logistic 模

*http://l2r.cs.uiuc.edu/∼danr/snow.html

型、指数 (Exponential) 模型、Log-linear 模型等。它的基本思想是为所有已知的因素建立模型，而把所有未知的因素排除在外[58] 。也就是说要找到这样一个概率分布，它满足所有已知的事实，且不受任何未知的因素的影响。最大熵分类器已经成功应用于信息抽取，句法分析等多个自然语言处理领域。目前最大熵模型是语义角色标注中应用最为广泛的模型之一[34, 35, 59, 60] ，究其原因，主要是因为最大熵模型能够较为准确地给出每个输出角色的概率值，并且方便的处理多类问题，另外一个不可忽视的原因就是最大熵模型较之支持向量机有更快的训练速度。然而，最大熵模型也并非完美，首先是其不便于使用复杂的结构特征，其次不支持特征自动组合，需要手工组合特征的缺点也限制了其性能的进一步提高。

决策树模型 (Decision Tree)：决策树[61] 也是一种重要的学习方法，其学习结果是呈树状的规则，因此学习结果容易观察，而且便于修改。它适合处理实例可被块状分割的问题。曾被 Chen 等人[62] 以及Ponzetto 等人[63] 应用于语义角色标注。但是决策树学习方法对于处理高维问题效果并不理想。而近年出现的随机森林算法[64] 是对决策树算法的一种改进，Nielsen 等人将其应用于语义角色标注任务[65] 取得了较好的效果。

多分类器融合 (Multi-classifier Fusion)：对多个分类器输出的结果进行融合也是分类器发展的一个方向，这种方法往往获得比其中任何一个分类器好的效果，因此经常被应用于各种评测之中。Ngai 等[66] 、以及Tsai 等[67] 就利用多分类器融合的方法参加了 SENSEVAL-3 和 CoNLL2005 语义角色标注的比赛。

基于核 (Kernel) 的学习方法 (Kernel based Methods)：如在介绍判别模型中所述，支持向量机，感知器等学习算法只能处理线性可分问题，这不利于在实际问题中的应用。现在，人们通常将在低维空间线性不可分问题映射到高维空间，在高维空间变成线性可分问题。而这种映射往往是通过计算核函数的方法隐含进行。人们将此类方法称作基于核 (Kernel) 的学习方法。核函数的计算能很好的融入支持向量机、感知器等学习算法，使它们能够较好的处理线性不可分问题。

当然，核的方法也不是一劳永逸的，人们必须根据实际问题，设计和使用不同的核函数，其过程也是较为繁琐和需要技巧的，但毕竟这给了我们一条利用高效学习算法的长处、避免其短处的途径。现在，研究者们又将核的思想升华，使其能够帮助人们更好的处理结构化的数据，发掘更多的结构化特征。最早由 Haussler[68] 以及 Watkins[69] 提出使用动态规划的方法高效计算结构化

数据的 Kernel 函数，这又被称作卷积核 (Convolution Kernel) 。

之后，越来越多的学者利用相似的思想于自己的研究工作，如句法分析[70]，文本分类[71] 等等。特别是 Moschitti 通过计算句法分析树的核函数进行语义角色标注[72] 。

1.2.3.4 **特征构造** 经自然语言处理学者多年研究发现，对系统最终性能影响最大的因素往往不是学习算法，而是算法所采用的特征。也就是说，对于使用相同特征的不同算法，系统的性能往往差异不大。在现有的各种语义角色标注系统中也表现了类似规律[35] 。因此，对于我们来说，寻找有效的特征是较构造有效的算法更为关键的问题。

目前，由 Gildea 等人[16] 在其语义角色标注系统中使用的语言学特征往往被当作各个系统的基本特征所使用，列举如下：

- 句法成分相关特征

1. **短语类型**

2. **句法成分核心词：**Collins 在其博士论文[73] 附录中描述了一些识别一个句法成分核心词的规则

3. **句法成分核心词的词性**

- 谓词相关特征

1. **谓语动词原形**

2. **语态**

3. **子类框架：**谓语动词所在 VP 的子类框架，如在图 1-3 中，子类框架为 VP→VBD,NP,PP

4. **谓语动词的词性**

- 谓语动词 - 句法成分关系特征

1. **路径：**句法树中，从句法成分到谓语动词之间的句法路径

2. **位置：**句法成分和谓语动词之间的位置关系

在此基础之上，人们又开发出了各种新的、有效的特征，如 Xue 等人增加了句法框架 (Syntactic frame)[32] 、动词类别[74] 等特征。

另外，对这些特征进行组合形成新的特征也是有效提高系统性能的一种途径。例如我们一般有这样的直觉，即一般 Arg0 角色位于谓语动词“前”且谓语动词是主动语态或者位于谓语动词“后”且谓语动词是被动语态。于是**位置**特征与**语态**特征的组合形成的新特征就是一个有效的特征。

关于本文所使用的特征以及特征的详细说明，我们将在后面章节中介

绍。

1.2.3.5 **标注模型** 根据是否使用全局的角色信息，可以将标注模型分为两大类，即局部模型和全局模型。

目前，多数基于句法成分的语义角色标注系统[35]，都是使用局部模型，即直接使用分类器，对各个句法成分的语义角色类别进行分类。而多数基于短语的语义角色标注系统[37]，大多采用全局模型。与局部模型不同，全局模型最显著特点是还考虑标注结果互相之间的关系。

局部模型相对简单，这里不进行赘述。我们重点讨论全局模型。根据使用全局语义角色信息的方式不同，全局模型可以分为两种：1) 在后处理步骤考虑语义角色全局信息；2) 在分类的过程中考虑语义角色全局信息。

我们先来看第一种在后处理步骤考虑语义角色全局信息的方法，这种方法相对比较简单。比如利用语义角色之间的约束关系，使用贪心的策略保留满足约束的语义角色。还有稍复杂些的系统，使用基于整数线性规划的方法[36]，可以保证在满足约束的条件下，达到语义角色标注结果全局最优。

第二种方法是在分类的过程中考虑语义角色全局信息，这种方法往往采用类似词性标注等序列标注的方法，具体做法包括：

最大熵马尔科夫模型 (Maximum Entropy Markov Models)[75] 是利用最大熵模型进行序列标注的一种方法，此方法使用比马尔科夫模型更多的上下文信息，因此在命名实体识别、词性标注等方面取得了较马尔科夫模型更好的效果。Blunsom[76] 将其用于了语义角色标注。但是最大熵马尔科夫模型会出现标注偏置 (Label Bias) 问题而影响最终的性能。因此，人们使用条件随机域 (Conditional Random Fields， CRF)[77] 方法对此进行改进。并有学者将其应用于语义角色标注[78–80]。

Jiang 等人[81] 不但利用了自然顺序语义角色的标注信息，还利用了树结构顺序的标注信息，其研究表明，利用树结构顺序的信息会取得更好的结果。

1.2.3.6 **多系统融合** 以上介绍的方法都局限于单一语义角色标注系统，然而由于句法分析的错误，语义分析的性能等原因，独立的语义角色标注系统很难获得满意的分析结果，因此有学者试图融合多种语义角色标注的结果，并取得了较为满意的标注结果，CoNLL-2005 评测的前 4 名，均采用了多系统融合的策略[35]。具体的融合方法包括：

(1) 基于规则的方法[56]

(2) 基于整数线性规划的方法[36]

(3) 基于重排序的方法[82]

(4) 基于机器学习的方法[83]

(5) 基于组块分析的方法[84]

当然，以上的各种融合方法，都或多或少的提高了系统的整体性能，当然也造成了系统构造复杂，分析效率不高等问题，还有待进一步解决。

1.2.3.7 **中文语义角色标注** 中文语义角色标注的工作开展较晚，研究得也不是很充分。最早进行研究的是 Sun 等[85]，由于在当时还没有中文方面的专门语料，所以他们只是人工标注了包含个别动词的一些语料，然后在这些语料上进行研究。虽不成系统，但是毕竟是一个有意义的开端。后来，伴随着中文 PropBank 的构建，Xue 等[74] 开始了比较系统的中文语义角色标注的工作，并得出了一些很有意思的结论。但是总体来讲，由于中文和英文的结构相似，目前中文语义角色标注的工作还是在沿着英文的道路在前进。

1.2.4 语义角色标注评测

我们首先介绍一下语义角色标注的评价方法。接着介绍一些国际评测的情况。

1.2.4.1 **语义角色标注评测方法** 在语义角色标注的识别阶段，通常采用信息检索中经常使用的准确率 (Precision) 、召回率 (Recall) 和 $F_{\beta=1}$ 来评价系统的性能。它们的定义分别为：

$$Precision = \frac{\text{正确标注为语义角色的个数}}{\text{分类器预测为语义角色总数}}$$

$$Recall = \frac{\text{正确标注为语义角色的个数}}{\text{测试数据中语义角色总数}}$$

$$F_{\beta=1} = \frac{2 * Precesion * Recall}{Precesion + Recall}$$

而在语义角色分类阶段，由于此时输入的都是正确的语义角色，我们需要做的仅是将其划分成不同的类别。因此仅使用分类的精确率 (Accuracy) 就可以衡量分类的效果。最终整体考虑识别和分类的结果时使用各个类别和整体的准确率、召回率和 $F_{\beta=1}$ 来衡量。

然而，仅通过比较两个系统的 $F_{\beta=1}$ 的差距并不能确切的说明两个系统的

优劣。例如，假设一共有 10 个评测数据，即使两个系统的性能相差 10% 之巨，也不能说明其中一个系统绝对比另一个好，因为它很可能仅仅比另一个多标注正确一个评测数据，而这也可能随机扰动的结果。由此我们可以看到，在比较两个系统性能的时候，单纯比较 $F_{\beta=1}$ 是不行的，还要考查测试数据的规模。也就是说，如果数据规模很小，则只有在两个系统 $F_{\beta=1}$ 差距很大的情况下，才能判断一个系统明显比另一个系统好；反之，如果数据规模很大，即使两个系统 $F_{\beta=1}$ 差距不大，也能判断一个系统明显比另一个系统好。这就是一个典型的显著性检验问题，在语义角色标注中，通常使用 χ^2 显著性检验来比较两个系统的性能。

与一般的检索问题一样，语义角色标注的最终结果也有四种情况，如表 1-2 所示，即正确的正例、正确的反例、第一类错误 (错误的正例) 、以及第二类错误 (错误的反例) 。这四种情况可以看成四个混淆集，它们的数量分别为 a 、b 、c 和 d ，则 χ^2 的计算公式为：

$$\chi^2 = \frac{(ab-cd)^2(a+b+c+d)}{(a+b)(a+c)(a+d)(c+d)}$$

表 1-2 语义角色标注系统的一般结果

Table 1-2 The General Results of a Semantic Role Labeling System

	标注为正例	标注为反例
标准正例	a	c
标准反例	d	b

χ^2 公式兼顾了两个系统性能的差距以及测试数据的规模。χ^2 的值越大，说明两个系统的差距越显著。一般通过查找 χ^2 表，获得在 3 自由度 * 下当 χ^2 的值大于某一数值时，两个系统的不同以 $1-p$ 的概率是显著的。例如当 $p=0.05$ 时，查表知当 $\chi^2 > 7.815$ 时，两个系统有 95% 的概率显著不同。

1.2.4.2 语义角色标注国际评测 对于语义角色标注，国际上于 2004 年至今经举行过多次评测，分别为 Senseval-3† 以及 CoNLL (Conference on Computational Linguistics Learning) 会议主办的SRL Share Task 2004[37] 、 2005[35] ，SemEval-

*n 个混淆集，$n-1$ 个自由度，在语义角色标注问题中，有 4 个混淆集

†http://www.cs.unt.edu/∼rada/senseval/senseval3/workshop.html

2007*，以及 CoNLL-2008[86]。其中 Senseval-3 和 SemE-2007 是以 FrameNet 为训练和测试语料。而 CoNLL 的历次语义角色标注评测则是以 PropBank 和 NomBank 为语料库。在这些评测中，CoNLL 系列评测以其参加队伍之多，影响之广泛而引起了人们的普遍关注。

2004 年，共有来自美国、西班牙、韩国等 10 家单位参加了 CoNLL 评测。但其并不提供人工标注的句法分析结果，取而代之的是使用自动标注的 Chunk 结果，以及自动标注的命名实体结果等。这些单位使用了 SVM， Winnow，最大熵，Perceptron 等多种统计学习方法。其中来自 Colorado University 的 Hacioglu 等人[38]，采取以短语为标注单元，语义角色识别和分类分步进行的策略，使用 SVM 算法在不使用全局特征的条件下，获得了最好的标注结果，测试集合的 $F_{\beta=1}$ 达到了 69.49%。

2005 年 CoNLL 继续举行 SRL Shared Task。此次评测较之 2004 年的评测参赛队伍翻了一番，包括我实验室在内，提交最终结果的单位达到了 19 个。与 2004 年的评测主要有四点不同：

(1) 此次提供足够大的训练语料，这为评测系统的性能随着语料库的规模变化而变化提供了方便；

(2) 提供了完全句法分析结果，但是句法分析结果并非手工标注而是自动标注的结果，因此在句法分析树上，不含有空节点以及功能短语标记等；

(3) 为了评测在新的领域中系统的性能，此次评测提供的测试数据不单从 Penn TreeBank 中抽取，还使用了其它领域的数据；

(4) 此次评测不但包括封闭测试 (只能使用主办方提供的数据，而不能使用其它的数据)，还包括开放测试 (不但可以使用主办方提供的数据，还可以使用任何外部数据，如 WordNet， VerbNet 等等)。

由此可见，此次评测更面向实际的语义角色标注系统，因此也广为后继研究人员所借鉴。来自 UIUC 的 Koomen 等人[54] 使用 SNoW 分类器，综合多种深层句法分析的输出结果，加上使用整数线性规划 (Integer Linear Programming) 的后处理方法，取得了最好的成绩，测试集合的 $F_{\beta=1}$ 达到了 79.44%。这也代表了当今最好的语义角色标注系统效果。

2008 年，CoNLL 评测再次将语义角色标注作为其主要的评测内容，与上两次基于短语结构句法分析的语义角色标注不同，本次评测以依存句法分析为基础，除了考查语义角色标注的性能外，还需要考查句法分析系统的性能。

*http://nlp.cs.swarthmore.edu/semeval/tasks/index.shtml

这次评测引起了空前的关注，共有 52 只队伍报名参赛，但是由于系统的复杂性，最终共有包括我实验室在内的19 个队伍提交了最终的结果。来自瑞典的 Lund 大学获得的评测的第一名，他们主要采用基于重排序的全局模型。而我们采用级联方式的系统，复杂度远远低于他们，但是最终也获得了较好的性能，并获得了第二名的成绩。

以上国际评测的成功举办，为语义角色标注研究提供了统一的数据集、测试方法和评价标准，极大地推动了这一研究领域的发展。

1.3 本文主要研究内容

语义角色标注是对语义分析的一个有益的尝试，它以词性标注、句法分析等其它自然语言处理技术为基础，使用统计学习的技术进行分析。但由于语义角色标注是一个新兴的研究课题，亟待解决的问题还有很多，如：标注步骤的制定、标注单元的选取以及标注模型的使用等各个方面。但是通过对现有语义角色标注系统的分析，我们认为目前提高语义角色标注系统性能最主要的方法就是开发更合适的机器学习算法以便利用更丰富和有效的语言学特征。

因此，本文首先研究常用的多项式核在语义角色标注上的应用，该方法以特征向量为处理对象，因此很难表示结构化的特征，存在数据稀疏的问题，为此有人使用卷积树核对结构化特征进行建模并取得了不错的效果。但是，一般的卷积树核并不适合语义角色标注问题，它混淆了不同种的特征。我们提出的混合卷积树核很好的解决了这个问题。然而，无论是卷积树核还是混合卷积树核，都要求子结构必须严格匹配，这给发现语义相同的近似子结构造成了困难，因此我们提出了基于句法的近似匹配机制 — 句法驱动的混合卷积树核，并进一步提高了语义角色标注系统的性能。以上的这些基于核方法的工作，最终还在中文语义角色标注问题上得到了验证。另一方面，对于核方法这一通用机器学习算法的研究，相信对其它类似的自然语言处理问题也具有一定的借鉴作用。本文的具体组织如下：

第 1 章，介绍了本文课题的研究背景、意义以及现状，概述了本文的主要研究内容；

第 2 章，构造了一个使用特征向量的多项式核语义角色标注系统，并与常用的基于最大熵系统进行了对比，实验证明基于多项式核的系统能够进行更好的特征组合，其性能也优于基于最大熵的系统，并超过 CoNLL-2005 中使

用单义句法分析结果的最好的系统；

第 3 章，首先介绍了卷积树和如何更好的对结构化信息更好的建模，针对卷积树核在语义角色标注上存在的问题，我们提出了一个混合卷积树核，它能够更好的对不同种的特征进行建模，最终与多项式核进行融合，取得了更好的效果；

第 4 章，针对卷积树核子结构匹配过于严格的问题，我们提出了句法驱动的卷积树核，它容许子结构在符合语言学知识条件下的近似匹配，进一步减弱了数据稀疏问题，并进一步提高了语义角色标注系统的性能；

第 5 章，以上的实验均基于英文语义角色标注问题，为了验证我们的方法不受语言的限制，我们构造了一个基于特征的中文语义角色标注基线系统，接着我们针对中文的特点，提出了新的特征。并在中文语料库上验证了各种基于核的方法的性能。

第 2 章　基于二次多项式核的语义角色标注

2.1 引言

基于特征向量的方法是目前最通用的语义角色标注方法，其具有特征构造灵活，效率和准确率较高的优点。因此，本章首先构造了一个使用特征向量方法构造的语义角色标注系统，并重点介绍如何使用多项式核来更好的利用特征向量，并与不使用多项式核的方法进行了对比。实验显示使用多项式核能够对特征进行组合，特别是二次多项式核，能够对特征进行两两组合，获得了最好的实验结果。最终该系统性能优于目前的最好系统。在以后的章节，我们将以此系统作为基线系统，在其上进行各种改进，以进一步提高语义角色标注系统的性能。

2.2 基于二次多项式核的语义角色标注系统

本节将介绍我们构造的一个典型的语义角色标注系统，该系统基于短语结构句法分析结果，以句法成分作为标注单元，包括语义角色标注的全部 4 个步骤，提取多种语言学特征，并使用基于分类的策略。该系统在 CoNLL 2005 语义角色标注评测中进行了验证，使用基于二次多项式核的分类器时，其性能优于本次评测最好的系统。下面详细介绍其使用的各种策略：

2.2.1 基于句法成分的标注单元

通过对以往研究的分析发现，由于可以利用更多的特征，基于句法成分的语义角色标注的性能，要较之基于短语的系统高十个百分点左右[35, 37] 。因此我们的系统以短语结构句法分析结果中的句法成分作为标注单元，所谓句法成分，即一棵句法分析树中每个非终结 (词) 节点，如图 2-1(a) 所示。每个语义角色几乎都有句法成分与其对应，因此该方法便于语义角色边界的确定，并在实验中获得了较高的性能。

2.2.2 四个标注步骤

该系统使用了语义角色标注系统中全部的 4 个步骤，即：剪枝、识别、分

类和后处理。下面详细介绍每一个步骤的方法：

1、剪枝：根据启发式规则，删除大部分不可能成为语义角色的句法成分，这样可以大幅减少输入到分类器的实例的个数，提高训练和测试的效率。本文参考 Xue 等人[32] 的做法，采用的具体剪枝过程为：

第 1 步： 对于输入的句子及其对应的句法分析树，设置谓词的词性为当前节点，其全部的兄弟节点作为候选语义角色。如果某个兄弟节点是 PP (介词短语) ，则该节点的子节点也作为候选语义角色；

第 2 步： 重新设置当前节点的父节点为新的当前节点，重复第 1 步的过程，直到到达根节点为止。

对于图 2-1(a) 中所有的句法成分，经过剪枝后的候选语义角色如图 2-1(b) 所示。注意，其中加圈的是经过每个步骤处理后保留下来的句法成分，以下表示相同。

2、识别：逐一判别每个经过剪枝的句法成分是否为语义角色，并保留识别成语义角色的标注单元[87] 。这是一个二元分类过程，每个经过剪枝步骤而被保留的句法成分都被分为 Arg 或者 NULL 两类。可以使用二元分类器完成此任务。

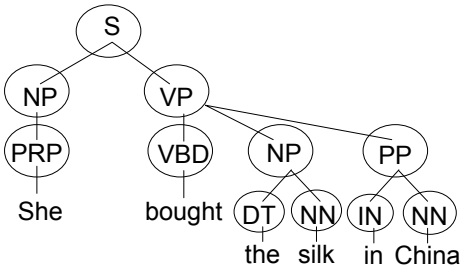
如图 2-1(c) 是识别为语义角色的句法成分，其中 IN 和 NN 被识别为 NULL 而予以舍弃。

3、分类：使用多类分类器，将识别为语义角色的句法成分，进一步划分为具体的语义角色。

如图 2-1(d) 是分类后的结果，其中识别为语义角色的 3 个句法成分具体的类别分别为 Arg0 、Arg1 和 ArgM-LOC 。

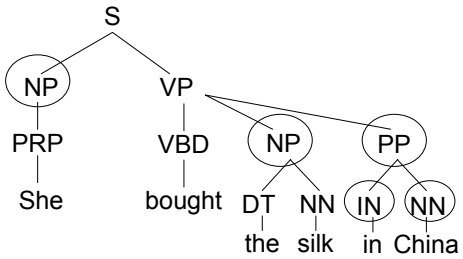
4、后处理：经过分类步骤产生的语义角色标注结果中，会有个别不满足语义角色标注约束的情况，即：重复和嵌套等。在语义角色标注中，一般对于一个谓词，不能够存在两个或两个以上相同的语义角色 (重复) ；另外，若一个字符串被标注为语义角色，则其子串不能同时被标注 (嵌套) 。我们在后处理阶段使用基于规则的方法避免重复和嵌套的情况发生。其方法简单描述为，当重复和嵌套的情况发生时，我们只保留分类阶段输出的概率较大的角色，该方法使用了贪心的策略，即没有考虑全局的最优解，实验结果表明其在不损失太多精度的条件下具有简单可行的优势。

如图 2-1(d) 同时也是后处理步骤后的结果，由于没有发生重复和嵌套的情况，因此并没有修改分类步骤的结果。



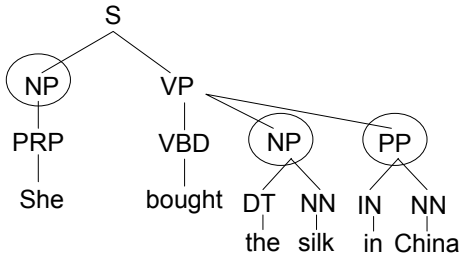
a) 句法分析结果及全部句法成分

a) A Parse Tree and All Constituents



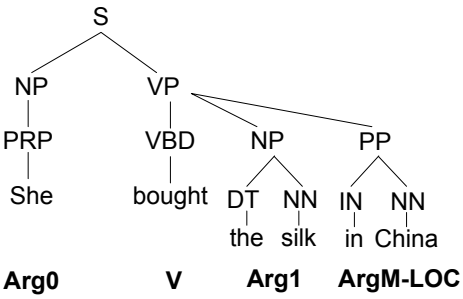
b) 经过剪枝后的句法成分

b) Constituents by Pruning



c) 经过识别后的句法成分

c) Constituents by Identification



d) 经过分类和后处理步骤形成的语义角色标注结果

d) Semantic Roles Labeling Result by Classification and Post-Processing Stages

图 2-1 标注步骤示例

Fig. 2-1 An Example for Labeling Stages

2.2.3 基于多项式核方法的分类器

上面第 2 、3 步骤使用分类器对实例进行二元或者多元的分类，其性能的好坏直接影响整个语义角色标注系统。在此，我们采用一种基于核方法的分类器。与其它分类方法相比，其主要优点是能够将低维线性不可分问题通过对特征进行组合或者分解，映射到高维空间，转化为线性可分问题，同时通过对核函数的计算，隐藏了映射的细节，从而使得时空复杂性降低到可以接受的范围。

核方法一般是和支持向量机等线性分类器配合使用的，它将复杂的分类问题分为两个部分，如图 2-2 所示，分别是与问题无关的线性分类器 (又称为核机器， Kernel Machine) ，以及与问题相关的核函数 (Kernel Function) 。其中核函数的作用是通过对具体分类问题的分析，隐式的将线性不可分问题映射到高维空间，然后使用线性分类器进行分类。

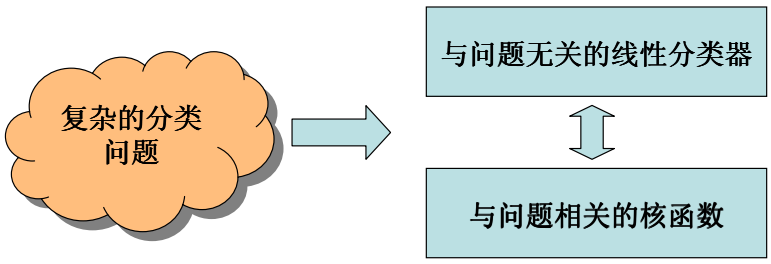


图 2-2 核方法解决分类问题的策略

Fig. 2-2 The Strategy of Kernel Methods

因此本节首先简单介绍支持向量机，特别介绍其如何引入的核方法。

2.2.3.1 支持向量机　对于二元线性可分的数据集，如图 2-3 所示，存在很多的线性分类超平面 (对于二维数据，则是一条直线) 能够将这两类分开。但是直觉上来看，两类数据中间的那个超平面是最好的分类界面。然而，其它的分类方法，如感知器等，仅仅保证能找到其中的一个分类超平面。而支持向量机的目标则是要找到最中间的那个。Vapnik[88] 利用统计学习的 VC 维理论和结构风险最小化 (Structure Risk Minimize, SRM) 原理证明了这种直觉的正确性，这种分类方法是在模型的复杂性 (即对特定训练样本的学习精度) 和学习能力 (即无错误的识别任意样本的能力) 之间寻求最佳折衷，以期获得最好的推广能力。

支持向量机的思想可以表示为寻找一个分类超平面，使其与任何数据的距离达到最大。从该平面到最近的数据点的距离则被定义为分类器的间隔(margin)，而这些点则被称作支持向量 (support vectors，在向量空间中，每个点都可以表示为从原点到该点的向量)。在支持向量机中，分类界面仅与支持向量 (通常数目较少) 有关，因此利用其进行分类的效率较高。如图 2-4 所示，其中黑色填充的实例为支持向量。

形式化的，此分类超平面可以用垂直于该平面的法向量 $\vec{\omega}$ 和截距 b 表示，$\vec{\omega}$ 又被称作权重向量。

则分类器表示为：

$$f(\vec{x}) = \mathtt{sign}(\vec{w}^T\vec{x} + b)$$

值为 +1，则表示正例；为 -1 表示反例。

如图 2-5 所示，假设从点 $\vec{x}$ 到分类超平面的距离表示为 r，其应该垂直于该超平面，并且与其交于点 $\vec{x}'$，则有：

$$\vec{x}' = \vec{x} - yr\frac{\vec{w}}{|\vec{w}|}$$

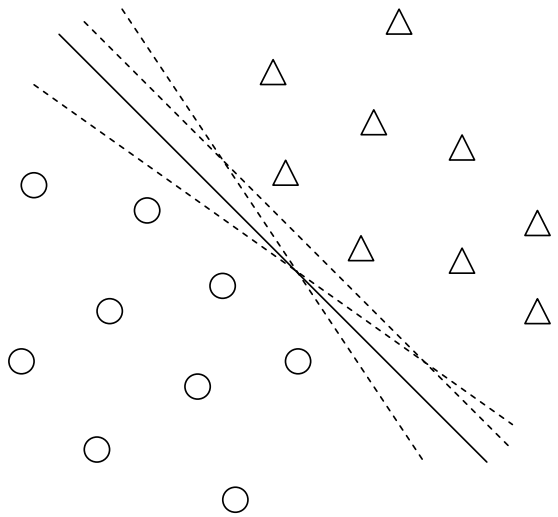


图 2-3 线性分类超平面

Fig. 2-3 Linear Classification Planes

其中 y 用来改变 $\vec{x}$ 在超平面不同侧时的符号，$\frac{\vec{w}}{|\vec{w}|}$ 指明方向。同时，由于 $\vec{x}'$ 在分类超平面上，则应该满足 $\vec{w}^T\vec{x}' + b = 0$ 。进一步的：

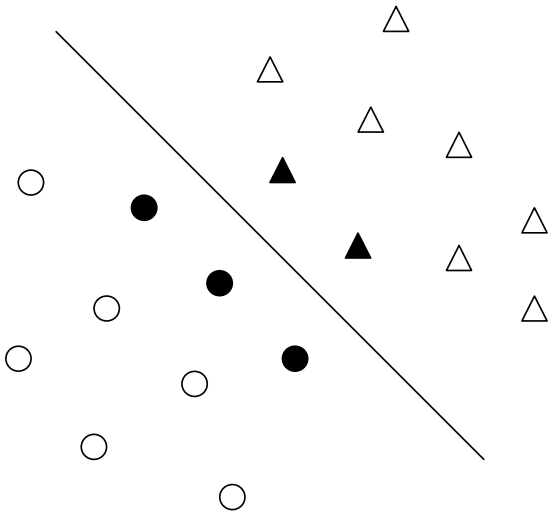


图 2-4 支持向量示例

Fig. 2-4 Sample of Support Vectors

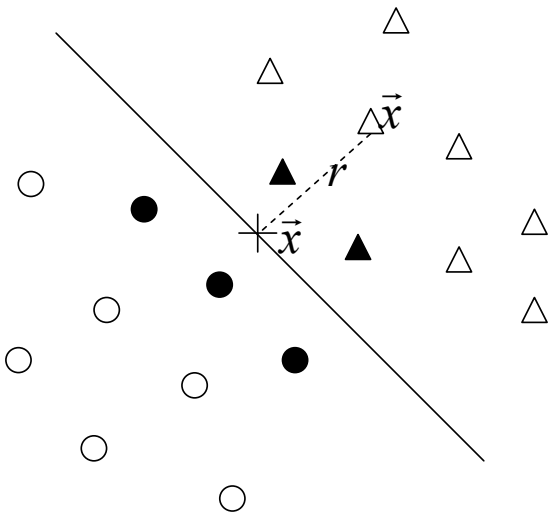


图 2-5 实例到分类超平面的距离

Fig. 2-5 The Distance from an Example to Classification Plane

$$\vec{w}^T(\vec{x} - yr\frac{\vec{w}}{|\vec{w}|}) + b = 0$$

解得：

$$r = y\frac{\vec{w}^T\vec{x} + b}{|\vec{w}|}$$

为了表示方便，我们可以通过改变 $|\vec{w}|$ 的大小，使得对于任意实例 $(\vec{x_i}, y_i)$，以下的不等式成立：

$$y_i(\vec{w}^T\vec{x_i} + b) \geq 1$$

并且至少有一个实例 (支持向量) 使得等式成立。因此最大间隔 $\rho = 2/|\vec{w}|$，其中 2 是为了以后计算方便而引入的常量，我们的目标仍然是使得此间隔最大，也就是寻找最优化的 $\vec{w}$ 和 b，即：

$$\arg\max_{\vec{w},b} \rho = 2/|\vec{w}|$$

s.t. 对于所有的训练实例 $(\vec{x_i}, y_i)$，$y_i(\vec{w}^T\vec{x_i} + b) \geq 1$

最大化 $2/|\vec{w}|$ 等价于最小化 $|\vec{w}|/2$，因此支持向量机最终的标准化形式为：

$$\arg\min_{\vec{w},b} \frac{1}{2}\vec{w}^T\vec{w}$$

s.t. 对于所有的训练实例 $(\vec{x_i}, y_i)$，$y_i(\vec{w}^T\vec{x_i} + b) \geq 1$

这是一个典型的线性约束下的二次优化问题 (Quadratic optimization，QP)，可以使用多种算法对其进行求解，在此不进行赘述。

然而，下面引入拉格朗日乘子的求解方法需要值得我们注意，对上式中的每个不等式约束引入一个拉格朗日乘子 α_i，获得以下拉格朗日方程：

$$L(\vec{w}, b, \vec{\alpha}) = \frac{1}{2}\vec{w}^T\vec{w} - \sum_{i=1}^{N}\alpha_i(y_i(\vec{w}^T x_i + b) - 1)$$

其中，$\vec{\alpha} = (\alpha_1, \ldots, \alpha_n)^T$。对 $L(\vec{w}, b, \vec{\alpha})$ 求偏导，则有下列等式成立：

$$\vec{w} = \sum_{i=1}^{N} \alpha_i y_i \vec{x}_i \tag{2-1}$$

$$0 = \sum_{i=1}^{N} \alpha_i y_i \tag{2-2}$$

将两个等式带入 $L(\vec{w}, b, \vec{\alpha})$ 中获得支持向量机的对偶 (dual) 表示为对下面的公式求最大值：

$$\tilde{L}(\vec{\alpha}) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{i=1}^{N} \alpha_i \alpha_j y_i y_j \vec{x}_i^T \vec{x}_j \tag{2-3}$$

相应的 $\vec{\alpha}$ 约束于：

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \tag{2-4}$$

$$\alpha_i \geq 0 \tag{2-5}$$

这又是一个二次优化问题，同样可以使用多种算法解得最优化的 $\vec{\alpha}$ 。最终解的形式为：

$$\vec{w} = \sum_{i=1}^{N} \alpha_i y_i \vec{x}_i \tag{2-6}$$

$$b = y_k - \vec{w}^T \vec{x}_k \text{ 对于任意的 } \vec{x}_k \text{ ，其相应的 } \alpha_k \neq 0 \text{ 。} \tag{2-7}$$

在此解中，大多数的 α_i 都为 0 。不为 0 的 α_i 指明其对应的 $\vec{x}_i$ 为支持向量。

最终的分类函数为：

$$f(\vec{x}) = \mathtt{sign}(\sum_{i=1}^{N} \alpha_i y_i \vec{x}_i^T \vec{x} + b) \tag{2-8}$$

请注意，在优化以及最终的分类过程中，所有对数据的操作仅需要通过计算数据对之间的点积完成（$\vec{x}$ 与 $\vec{x}_i$，或者 $\vec{x}_i$ 与 $\vec{x}_j$），这点非常重要，我们将在后面详细说明。

2.2.3.2 多项式核 上面介绍了数据线性可分的情况下，支持向量机的构造以及求解过程，然而在现实世界中，尤其是处理自然语言这种复杂的问题时，很多情况是非线性的。如何使用支持向量机这一强有力的分类工具呢？

下面我们来看一个一维的实例，假设数据如图 2-6 所示分布，则使用线性分类器会很容易的将实例分开。而如果数据如图 2-7 所示，其中一类被另一类夹在中间，则不可能使用线性分类器将其分开。为了将如此分布的数据正确的分类，有两种解决方案：一种是使用非线性的分类器，但是这样将不能够利用支持向量机进行分类；另一种方法是将低维线性不可分数据映射到高维空间，变为在高维空间的线性可分问题，然后可以仍然利用支持向量机这一强大的线性分类工具。例如图 2-8 所示，如果我们利用二次函数 $f(x) = x^2$ 将原一维数据映射到二维，则可以利用一个线性分类器将其分割。

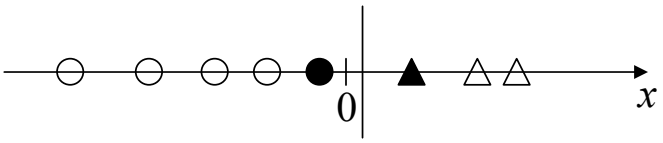


图 2-6 一维数据的线性可分的情况

Fig. 2-6 Linearly Separable Case

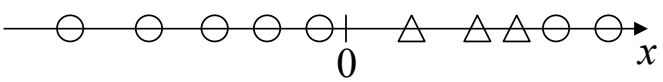


图 2-7 一维数据的线性不可分的情况

Fig. 2-7 Non-linearly Separable Case

支持向量机为这种映射提供了简单有效的途径，被称之为“核 (Kernel)”方法。如线性支持向量机所有对数据的操作仅需要通过计算数据对之间的点积完成。假设 $K(\vec{x}_i, \vec{x}_j) = \vec{x}_i^T \vec{x}_j$，则公式 2-8 转化为：

$$f(\vec{x}) = \mathtt{sign}(\sum_{i=1}^{N} \alpha_i y_i K(\vec{x}_i, \vec{x}_j) + b) \tag{2-9}$$

现在，假设我们通过映射 $\Phi : \vec{x} \mapsto \phi(\vec{x})$ 将数据点 $\vec{x}$ 映射到高维空间，则高维空间上的点积为：$\phi(\vec{x}_i)^T \phi(\vec{x}_j)$，如果这种点积恰好可以通过原始空间的计算简单高效的直接获得，则可以不进行 $\Phi : \vec{x} \mapsto \phi(\vec{x})$ 的映射，而是简单的计算 $K(\vec{x}_i, \vec{x}_j) = \phi(\vec{x}_i)^T \phi(\vec{x}_j)$ 的值，接着带入公式 2-9 获得分类器。核函数 K 相应的就是某些扩展空间上的点积。

例如，对于 2 维向量 $\vec{u} = (u_1, u_2)$ 和 $\vec{v} = (v_1, v_2)$，设二次函数 $K(\vec{u}, \vec{v}) = (1 + \vec{u}^T \vec{v})^2$，我们希望证明其是一个合法的核函数，也就是说 $K(\vec{u}, \vec{v}) = \phi(\vec{u})^T \phi(\vec{v})$。如果 $\phi(\vec{u}) = (1, u_1^2, \sqrt{2}u_1 u_2, u_2^2, \sqrt{2}u_1, \sqrt{2}u_2)$，则有：

$$\begin{aligned} K(\vec{u}, \vec{v}) &= (1 + \vec{u}^T \vec{v})^2 \\ &= 1 + u_1^2 v_1^2 + 2u_1 v_1 u_2 v_2 + u_2^2 v_2^2 + 2u_1 v_1 + 2u_2 v_2 \\ &= (1, u_1^2, \sqrt{2}u_1 u_2, u_2^2, \sqrt{2}u_1, \sqrt{2}u_2)^T (1, v_1^2, \sqrt{2}v_1 v_2, v_2^2, \sqrt{2}v_1, \sqrt{2}v_2) \end{aligned}$$

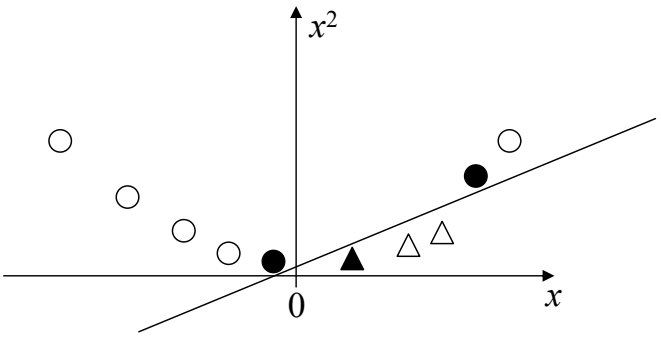


图 2-8 一维线性不可分数据映射到二维空间变为线性可分

Fig. 2-8 Mapping Non-linearly Separable Case in One-dimension to Linearly Separable Case in Two-dimension

$$= \quad \phi(\vec{u})^T \phi(\vec{v})$$

也就是说 $K(\vec{u},\vec{v}) = (1+\vec{u}^T\vec{v})^2$ 是一个合法的核函数，我们可以不用通过显示的将 $\phi(\vec{u})$ 映射为 $(1, u_1^2, \sqrt{2}u_1u_2, u_2^2, \sqrt{2}u_1, \sqrt{2}u_2)$ 而获得 $K(\vec{u},\vec{v})$ 的值。

并非任意的函数 K 都是核函数，其必须满足 Mercer 条件，即 K 必需是连续、对称 * 且半正定 † 。只有这样的 K 才意味着存在一个映射 $\Phi : \vec{x} \mapsto \phi(\vec{x})$ 使得 $K(\vec{u},\vec{v}) = \phi(\vec{u})^T\phi(\vec{v})$ 。如果 K 不满足 Mercer 条件，则相应的二次规划问题可能无解。

在各种核函数中，形如 $K(\vec{u},\vec{v}) = (1+\vec{x}^T\vec{z})^d$ 的多项式核 (Polynormial Kernel) 是最常见的一种。当 $d=1$ 时，其实就是线性核，也就是前面介绍的线性分类情况。当 $d=2$ 时，又被称作二次核，如上面的例子所示。

上面的多项式核实际上是对特征进行了多阶组合 (阶数与多项式次数相同) ，例如二次核，除了保留原始特征外，还对特征进行了两两组合。这在自然语言处理问题中尤其有用，例如对于文本分类问题，往往单独的词不能表示其类别，如 “operating” 和 “system” ，我们需要使用多项式核对其进行组合，描述其共现的情况，即如果 “operating” 和 “system” 共同出现，则较为容易的判断为计算机类别。

2.2.3.3 多元分类 然而，即使引入了基于核的方法，仍然改变不了支持向量机二元分类的特性。为了完成语义角色分类任务，我们需要一个多元分类器。将二元分类器转换为多元分类器一般有两种策略：1) 一对多，即将每一类单独和其它所有类构造一个二元分类器 (共 $|\mathbb{C}|$ 个，其中 $|\mathbb{C}|$ 是类别的个数) ，最终分类结果取得分最大的那个类别；2) 一对一，即每两个类别构造一个分类器，(共 $|\mathbb{C}|(|\mathbb{C}|-1)/2$ 个) ，最终选择被最多的分类器选择的那个类别。由于一对一策略需要构造较多的分类器，而且 Rifkin 和Klautau[89] 已经证明，这两种策略的性能相差不多，因此为了提高系统的效率，我们采用一对多的策略进行多类分类。

2.2.3.4 线性不可分 上面介绍的都是理想的情况下 (数据自身线性可分或者经过映射在高维空间线性可分) ，构造支持向量机分类器以及引入基于核的方法。但是在现实世界中，由于噪声数据等情况的出现，很多实际问题并不能找到一个理想的分类界面，或者即使能够找到，由于其分类间隔过小，分类器

* 对于任意的 $\vec{x}$ 和 $\vec{y} \in X$ ，有 $K(\vec{x},\vec{y}) = K(\vec{y},\vec{x})$ 。

† 由元素 $K_{ij} = K(\vec{x_i},\vec{x_j})$ 定义的矩阵 K 是半正定的，也就是说对于任意的 $c_1,\ldots,c_N \in R$ ，有 $\sum_{ij} c_ic_jK_{ij} \leq 0$ 。

的泛化能力也不够强，如图 2-9 所示，为了获得泛化能力较强的分类器，我们需要忽略一些噪声数据，即在训练时容许一些错分的情况。为此，我们引入松弛因子 ξ_i 对错分的实例进行惩罚。此时支持向量机的优化问题变为：

$$\arg\min_{\vec{w},b,\xi_i\geq 0} \frac{1}{2}\vec{w}^T\vec{w} + C\sum_{i}^{N}\xi_i$$

s.t. 对于所有的训练实例 $(\vec{x_i}, y_i)$ ， $y_i(\vec{w}^T\vec{x_i} + b) \geq 1 - \xi_i$

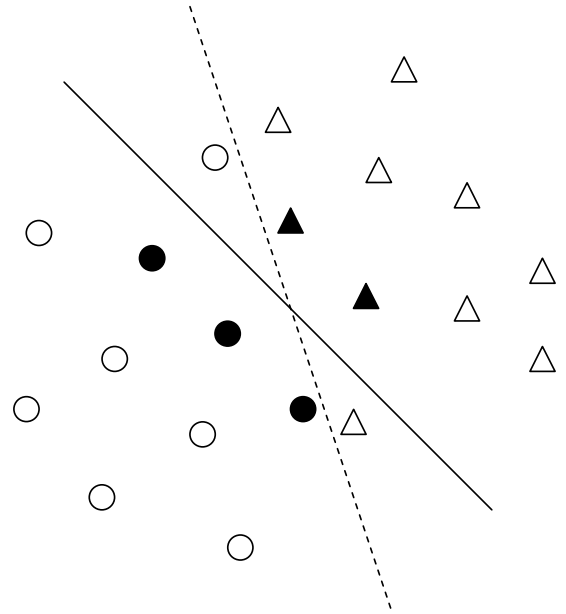


图 2-9 容错的支持向量机

Fig. 2-9 Support Vector Machine with Fault-tolerance

参数 C 是一个正则化项，用来控制分类器的泛化能力。随着 C 变大，对错分的数据惩罚变大，于是分类器对已知数据的拟合越来越好，以至于泛化能力越来越差；相反，随着 C 变小，对错分的数据惩罚变小，于是分类器对已知数据的拟合越来越差，但是泛化能力越来越好；因此，选择一个合适的 C ，对于实际的分类问题非常重要。

最终获得此时支持向量机的对偶表示为对下面的公式求最大值：

$$\tilde{L}(\vec{\alpha}) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{i=1}^{N} \alpha_i \alpha_j y_i y_j \vec{x}_i^T \vec{x}_j \tag{2-10}$$

相应的 $\vec{\alpha}$ 约束于：

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \tag{2-11}$$

$$0 \leq \alpha_i \leq C \tag{2-12}$$

这里松弛因子 ξ_i 及其相应的拉格朗日乘子都消失了。同时该优化问题仍然只需计算数据点之间的点积即可求解，所以基于核的方法仍然适用。

至此，我们已经介绍了基于核的分类器 (尤其是性能最好的支持向量机) 的全部背景知识，下面需要介绍分类器中另外一个重要的部分：特征构造。

2.2.4 语义角色标注中的特征构造

本节详细介绍我们的英文语义角色标注系统中使用的各种特征。我们以图 2-10 为实例，其中谓词为 *bought* ，待考察的句法成分为第一个 “NP” 。

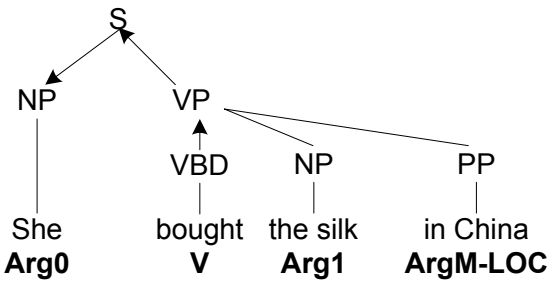


图 2-10 路径特征演示

Fig. 2-10 Path Feature Illustration

- 句法成分相关特征

1. **短语类型：** 句法成分的短语类型。此例为：“NP” 。短语类型对语义

角色往往有较为明显的指示作用，如核心角色往往是 “NP”，而 “PP” 一般是时间或者地点等。

2. **核心词词干：**Collins 在其博士论文[73] 附录中描述了一些识别句法成分核心词的规则。而对于介词短语，由于其核心词一般为介词，而借此自身又不能反映出介词短语的语义 (例如介词 “in”，既可以表示时间，又可以表示地点)，因此我们使用短语中的最后一个名词来代替传统的核心词。最后采用基于规则的方法[90] 对一个成分中的核心词提取词干。此例的第一个 NP 短语中，核心词为：“she”，而 PP 短语的核心词为 “China”。核心词对于语义角色也具有较为明显的指示作用，例如 “she” 一般作施事，而不会作受事等。

3. **核心词的词性：**此例为 “she” 的词性 “PR” (代词)，对于 NP 短语 “the silk”，则为 “NN” (名词)。此特征是对核心词的泛化，能够缓解一定的数据稀疏问题。

4. **最后一个词的词干：**一个成分中最后一个词的词干。此例仍为 “she”，对于 NP 短语 “the silk”，则为 “silk”。

5. **最后一个词的词性：**一个成分中最后一个词的词干。此例仍为 “she” 的词性 “PR”，对于短语 “the silk”，则为 “NN”。

6. **命名实体：**如果一个成分以命名实体结尾，则把命名实体的类型作为特征，类型包括三种：地点 (LOC)、组织 (ORG)、人物 (PER)。此例为 “NULL”，表示非任何命名实体。而 PP 短语 “in China” 则为 “LOC”。一般 “LOC” 直接指明了该句法成分为地点语义角色。

7. **句法成分前一个词：**此例为 “NULL”，表示句法成分前面不存在任何词。

8. **句法成分后一个词：**此例为 “bought”。

9. **句法成分前面第二个词：**此例为 “NULL”，表示句法成分前面第二个词不存在。

10. **句法成分后面第二个词：**此例为 “the”。

- 谓语动词相关特征

1. **谓语动词原形：**此例为 “buy”。之所以取动词原型而不是动词本身，也是为了缓解数据稀疏问题，在此例中，无论 “buy” 进行各种变形，如 “bought”，“buies” 等，对各个句法成分的语义角色影响都不大。

2. **语态：**我们用规则识别谓词是主动语态还是被动语态。当谓词的词性为 VBN，并且前面有 AUX 时，则为被动语态。比例为：“active”，表示主动语态。语态信息对于语义角色标注非常重要。例如对于主动语态，施事往往在动

词之前，而被动语态则正好相反。

3. **子类框架：** 谓语动词所在 VP 的子类框架。此例中，子类框架为“VP→VBD,NP,PP”。子类框架的信息往往也指明了语义角色的信息。

4. **谓语动词的词性：** 此例为“VBD”，表示动词的过去式。这是对谓词特征的一个泛化。

5. **谓语动词的后缀：** 我们使用谓词的最后 3 个字母，此特征对于泛化谓词特征有一定的作用，例如以“ing”结尾的谓词，多表示主动形式等。此例为“ght”。

- 谓语动词 - 句法成分关系特征

1. **路径：** 句法树中，从句法成分到谓语动词之间的句法路径。此例中，如图 2-10 所示，“She”所在 NP 到谓语动词的路径为“VBD↑VP↑S↓NP”，表示从 VBD 经过 VP 向上到达 S，然后向下到达 NP。路径已经被证明是非常有用的特征，因为在谓词与某一句法成分之间，往往具有固定的路径模式。

2. **路径长度：** 一个成分和它的谓词之间的路径长度。此例为“3”。路径越长，一个句法成分称为语义角色的可能性就越小。

3. **位置：** 句法成分和谓语动词之间的位置关系，有两个取值，“前”和“后”。对于“覆盖”的情况，我们根据启发式规则直接将其忽略，因为这种情况下，句法成分不可能成为谓词的角色。此例为“前”。位置特征对于语义角色也具有较强的指示作用，例如施事往往在“前”。

4. **部分路径：** 全部向上的路径，此例为“VBD↑VP↑S”。由于路径特征过于稀疏，所以需要对其进行一定的泛化，部分路径能够在一定程度上达到此目的。

5. **从句层级：** 在一个成分和谓词之间的路径上的从句的数量。此例不包含从句，特征为“0”。从句层级越多，一个句法成分称为语义角色的可能性就越小。

6. **谓词到句法成分的路径上“VP”个数：** 此例为“1”。“VP”越多，一个句法成分称为语义角色的可能性就越小。

2.2.5 局部标注模型

这里，我们使用最常用的局部模型，即仅利用上面所列的局部特征，不考虑各个语义角色类别之间的关系，对各个句法成分的语义角色属性直接分类。虽然这种方法可能会丢失一些全局的信息，但是其具有简单易行的优点，使得我们能够将主要研究精力集中在分类器对语义角色标注系统性能的影响

上，而不用顾及模型的影响等情况。

2.3 对比系统

为了和基于核方法的系统进行对比，我们又构造了两个对比系统，分别为基于规则的方法和基于最大熵分类器的方法，下面分别进行详细的介绍。

2.3.1 基于规则的语义角色标注

基于规则的语义角色标注系统，采用了下面六条简单的规则进行标注：

(1) 将目标动词组块中的 ***not*** 以及 ***n't*** 标注为 ***ArgM-NEG***
(2) 将目标动词组块中的情态动词标注为 ***ArgM-MOD***
(3) 将目标动之前的第一个 ***NP*** 标注为 ***Arg0***
(4) 将目标动之后的第一个 ***NP*** 标注为 ***Arg1***
(5) 将目标动之前的 ***that***, ***which*** 和 ***who*** 标注为 ***R-Arg0***
(6) 如果目标动词是被动，则将 ***Arg0*** 和 ***Arg1*** 交换

2.3.2 基于最大熵分类器的语义角色标注

最大熵分类器是目前语义角色标注系统中最常用的分类器，它适合处理多类分类问题，能够较为准确地给出每个输出类别的概率值，同时具有较快的训练速度，因此我们将其选做对比系统。

最大熵模型是最大熵分类器的理论基础，其基本思想是为所有已知的因素建立模型，而把所有未知的因素排除在外[58] 。也就是说要找到这样一个概率分布，它满足所有已知的事实，且不受任何未知的因素的影响。最大熵分类器已经成功应用于信息抽取，句法分析等多个自然语言处理领域。在预测一个句法成分是否为某一语义角色的过程中会涉及各种因素，假设 $\vec{x}$ 就是一个由这些因素构成的向量，变量 y 的值为语义角色类型。$p(y|\vec{x})$ 是指系统对某个句法成分预测为某一语义角色的概率。这个概率可以用上述思想来估计。最大熵模型要求 $p(y|\vec{x})$ 在满足一定约束的条件下，必须使得下面定义的熵取得最大值：

$$H(p) = -\sum_{\vec{x},y} p(y|\vec{x}) \log p(y|\vec{x})$$

这里的约束条件实际上就是指所有已知的事实，一般可以用以下的方式来表述：

$$f_i(\vec{x},y)=\begin{cases}1 & \text{如果 } (\vec{x},y) \text{ 满足某一条件}\\ 0 & \text{否则}\end{cases},i=1,2,\ldots,n$$

称 $f_i(\vec{x},y)$ 为最大熵模型的特征函数，n 为所有特征的总数。可以看到这些特征描述了向量 $\vec{x}$ 与语义角色 y 之间的联系。概率 $p(y|\vec{x})$ 必须满足上述特征的约束，由此可以定义一个受限的概率分布族为：

$$\mathfrak{F}=\{p(y|\vec{x}):E_p\{f_i\}=E_{\tilde{p}}\{f_i\},1\leq i\leq n\}$$

其中：

$$E_p\{f_i\}=\sum_{\vec{x},y}f_i(\vec{x},y)p(\vec{x})p(y|\vec{x})$$

$$E_{\tilde{p}}\{f_i\}=\sum_{\vec{x},y}f_i(\vec{x},y)\tilde{p}(\vec{x})\tilde{p}(y|\vec{x})$$

现在的问题就是要在受限的概率分布族中找到一个具有最大熵的分布，即：

$$p^*(y|\vec{x})=\arg\max_{p(y|\vec{x})\in\mathfrak{F}}\{-\sum_{\vec{x},y}(\tilde{p}(x)p(y|\vec{x}))\log p(y|\vec{x})\}$$

可以求出上式的解为：

$$p^*(y|\vec{x}) = \frac{\exp(\sum_i\lambda_i f_i(\vec{x},y))}{Z(\vec{x})} \tag{2-13}$$

$$Z(\vec{x}) = \sum_y\exp(\sum_i\lambda_i f_i(\vec{x},y)) \tag{2-14}$$

其中 λ_i 是每个特征的权重。

与支持向量机相比，最大熵分类器天然的支持多类分类，而且通过公式 2-13 可以获得每个类别 y 的相对准确的概率输出。但是，由于最大熵不具有对偶的表示形式，从而也不能够使用核函数，即不能使用基于核的方法。因此其仍然是一种线性分类器，不能很好的应对自然语言处理中常见的非线性可

分问题。

2.4 实验及讨论

2.4.1 数据资源

我们使用 CoNLL SRL Shared Task 2005* 评测提供的数据作为训练、开发和测试集。主要来自基于华尔街日报 (WSJ) 标注的 PropBank 。其中 Section 02-21 为训练集；Section 24 为开发集；测试集除了 Section 23 外，还包括一部分来自 Brown 语料 3 个单元的标注结果。语料库详细的统计规模如表 2-1 所示。

表 2-1 语料库规模
Table 2-1 Corpus Size

	Train	Development	tWSJ	tBrown
Sentences	39,832	1,346	2,416	426
Tokens	950,028	32,853	56,684	7,159
Propositions	90,750	3,248	5,267	804
Arguments	239,858	8,346	14,077	2,177

此次评测提供了两个完全句法分析的结果，分别来自 Collins[73] 和 Charniak[91] 。然而，毕竟自动句法分析效果并不完美，所以不可能每一个角色都能够在句法分析树中找到与之匹配的句法成分。据统计，在训练集中大约 10% 的语义角色找不到与之相匹配的句法成分。表 2-2 中列出了前 3 个最不匹配成分的角色类型。这里，Charniak 句法分析器有 10.08% 的角色不匹配，而 Collins 句法分析器为 11.89% 。因此，我们看到 Charniak 句法分析器在语义角色标注问题上性能优于 Collins 句法分析器，所以我们在后面的实验中采用 Charniak 分析器产生的句法分析结果作为语义角色标注的输入。

另外， Chieu 等人[92] 还提供了命名实体自动识别的结果。Màrquez 等人[93, 94] 还提供了词性标注以及组块分析 (Chunking) 的结果。这些都可以作为特征引入到语义角色标注中来。

*http://www.lsi.upc.edu/∼srlconll/

2.4.2 多项式核分类器的实现

我们使用开源的 SVM-Light[95] 作为支持向量机分类器，其实现了包括多项式核在内的各种常用的核函数，同时容许用户自定义正则化参数 C 。由于 SVM-Light 仅实现了二元分类，为了进行多元语义角色分类，我们使用了一对多的策略。

对于最大熵分类器，我们使用张乐的最大熵模型工具包 *，其实现了带有高斯先验平滑的 L-BFGS 参数估计算法[96] 。

2.4.3 实验结果及讨论

在使用支持向量机分类器的时候，除了需要确定使用的核函数外，还需要优化正则化参数 C 。图 2-11 显示了在使用多项式核函数 ($d = 2$) 的时候，系统性能在开发集上随参数 C 变化的曲线。从图中可以看出，参数 C 对系统最终的性能有非常大的影响。同时，当 C 在取一定值 (此时为 4) 的时候，系统的性能达到峰值。随着 C 值增大，系统的泛化能力降低，性能随之变差；随着 C 减小，系统的对训练数据拟合能力降低，性能也随之变差；

如前所述，多项式核能够将特征进行组合，适合于众多自然语言处理问题。因此我们在此使用多项式核，并考察多项式核的阶数 (参数 d) 对系统性能的影响。

表 2-3 比较了不同阶数多项式核之间的性能，其中线性核即为 $d = 1$ 的多项式核，每种核函数在取得最优性能时的 C 也在表中列出。最大熵分类器中，我们在开发集上获得最优参数。其中，高斯先验值 $g = 2$ ，迭代次数 $i = 1,000$ 。另外，我们还给出了基于规则的系统的性能。最后，我们还给出了 CoNLL-2005 语义角色标注共享任务的一个系统 (Surdeanu and Turmo[57]) ，它最终在所有参加的系统中排第 5 ，但是在使用单一的句法分析器的系统中

*http://homepages.inf.ed.ac.uk/s0450736/maxent~toolkit.html

表 2-2 前 3 个最不匹配角色

Table 2-2 The Top-3 Roles with no Matching Constituents.

Args	Cha parser	Col parser	Both
Arg1	5,496	7,273	3,822
ArgM-DIS	1,451	1,482	1,404
Arg0	1,416	2,811	925

是最好的 (与我们用的是相同的分析器) 。只是，他们用了不同的分类器，AdaBoost[55] ，和不同的特征集合。

表 2-3 不同系统之间性能比较

Table 2-3 The performance comparison among different systems

	Rule	Linear ($C = 2$)	Polynomial ($d = 2, C = 4$)	Polynomial ($d = 3, C = 3$)	Maximum Entropy	Surdeanu and Turmo[57]
Devel	36.70	70.03	**75.37**	75.02	75.27	75.17
Test (WSJ)	37.14	73.59	**77.00**	76.21	76.44	76.46
Test (Brown)	43.30	63.30	**65.63**	64.84	65.09	65.42

通过表 2-3 可以看出：

1、当使用多项式核对特征进行组合后 ($d = 2$ 或者 $d = 3$) ，系统的性能对比最大熵以及线性支持向量机显著提高 ($p = 0.05$ 下的 χ^2 检验) 这说明在语义角色标注中，组合特征对分类的性能有较大的帮助。从而也证明了基于多项式核方法的有效性。这往往也符合人们的直觉，例如 “语态” 和 “位置” 特征的组合非常具有指示性，主动语态往往暗示着谓词前面的句法成分为施事，而被动语态则恰好相反。这种例子在语义角色标注中还有很多。当然，我们可以预先提取这种组合信息，但是并非所有的组合信息都如此明显，如果将所

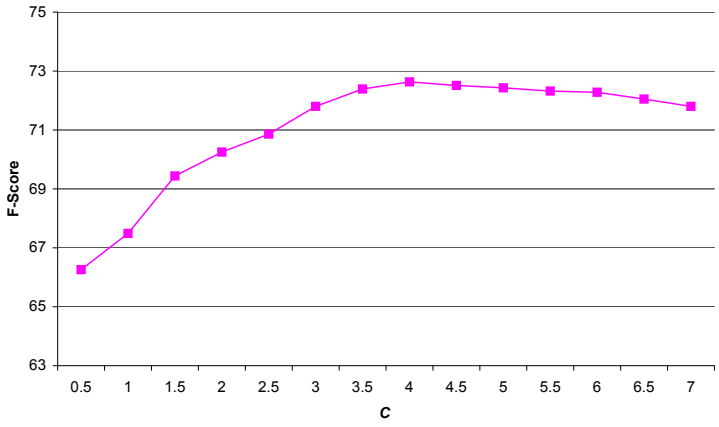


图 2-11 使用多项式核的支持向量机，随参数 C 变化时的性能曲线

Fig. 2-11 The Performance Curve Changing with SVM Parameter C in the Polynomial Kernel

有的特征都进行组合，往往又构成了一个非常庞大的特征空间，不利于计算。因此，通过多项式核的方法，隐式的将特征进行组合，不但能充分挖掘有用的组合特征，还在计算上可行。因此基于多项式核的方法取得了较好的实验效果。

2、同时也要注意，并非组合的特征越多越好，因为只有当 $d=2$ 时，即使用二次多项式核，对特征进行任意的两两组合后，系统的性能达到最大。随着阶数的增加，也就是组合的特征增多，系统的性能反而下降。主要原因是增加了太多的不相关特征。从而也证明，基于二次多项式核的方法更适合于语义角色标注问题。

3、同样不使用组合特征，而直接进行线性分类，最大熵的性能要优于支持向量机。这主要是因为最大熵分类器自身能够支持多类分类，而且能够给每一个输出的类别赋予准确的概率，这对后处理阶段是有利的。

4、使用二次多项式核的方法，在都使用单一句法分析器的条件下，其性能已经超过 CoNLL-2005 评测的最好系统。说明使用多项式核的方法，通过对特征进行适当的组合，系统性能已经能够达到目前最好的水平。

5、通过对 WSJ 和 Brown 测试语料上的结果对比，我们发现所有系统的 WSJ 结果都要较 Brow 结果高 10% 左右，这是一个较为明显的领域移植问题，因为所有系统的训练语料均来自 WSJ 。当然，这也是自然语言处理领域的一个共性问题，如何解决，还有待进一步的研究。

6、基于规则的系统性能明显低于其它基于机器学习方法的系统，这也说明了基于机器学习方法的有效性。但是对比不同测试语料上的性能可以发现，基于规则的方法性能比较稳定，甚至在 Brown 语料库上取得了比 WSJ 还高的性能，可见基于规则的系统不存在领域移植问题。

最后，我们在表 2-4 中列出了基于二次多项式核方法在 WSJ 测试语料库上的详细评测结果。

2.5 本章小结

在本章中，我们构造了一个基于多项式核方法语义角色标注系统。实验结果显示，当使用二次多项式核的时候，该系统性能达到最优，不但超过了线性核以及基于最大熵的方法，而且要优于 CoNLL-2005 评测中使用单一句法分析器的最好系统。这充分体现了基于二次多项式核的方法能够很好的解决线性不可分问题，非常适合于语义角色标注。还有一点需要指出的是，在使用

表 2-4 多项式核详细结果

Table 2-4 The Detail Performance Results of Polynomial Kernel Method

WSJ (Test)	Precision	Recall	$F_{\beta=1}$
Overall	80.78%	73.56%	77.00
Arg0	88.14%	83.61%	85.81
Arg1	79.62%	72.88%	76.10
Arg2	73.67%	65.05%	69.09
Arg3	76.03%	53.18%	62.59
Arg4	78.02%	69.61%	73.58
Arg5	100.00%	40.00%	57.14
ArgM-ADV	59.85%	48.02%	53.29
ArgM-CAU	68.18%	41.10%	51.28
ArgM-DIR	56.60%	35.29%	43.48
ArgM-DIS	76.32%	72.50%	74.36
ArgM-EXT	83.33%	46.88%	60.00
ArgM-LOC	65.31%	52.89%	58.45
ArgM-MNR	58.28%	51.16%	54.49
ArgM-MOD	98.52%	96.37%	97.43
ArgM-NEG	97.79%	96.09%	96.93
ArgM-PNC	43.68%	33.04%	37.62
ArgM-PRD	50.00%	20.00%	28.57
ArgM-REC	0.00%	0.00%	0.00
ArgM-TMP	78.38%	66.70%	72.07
R-Arg0	81.70%	85.71%	83.66
R-Arg1	77.62%	71.15%	74.25
R-Arg2	60.00%	37.50%	46.15
R-Arg3	0.00%	0.00%	0.00
R-Arg4	0.00%	0.00%	0.00
R-ArgM-ADV	0.00%	0.00%	0.00
R-ArgM-CAU	100.00%	25.00%	40.00
R-ArgM-EXT	0.00%	0.00%	0.00
R-ArgM-LOC	83.33%	47.62%	60.61
R-ArgM-MNR	66.67%	33.33%	44.44
R-ArgM-TMP	77.27%	65.38%	70.83

支持向量机分类器进行语义角色标注时，需要注意 C 参数对结果的巨大影响，为了获得较好的性能，必须对 C 参数做适当的调整。

第 3 章　混合卷积树核与二次多项式核相结合

3.1 引言

我们在上一章介绍了基于多项式核的语义角色标注方法，该方法使用特征向量来表示待分类对象，能够高效的对特征进行组合，并且在语义角色标注任务上取得了不错的效果。然而，该方法并不适用于某些结构化的特征，例如路径特征等。因为在基于特征的分类方法中，即使是相似的结构也被表示为不同的特征，因此很容易造成特征的稀疏，对分类效果不利。因此，本章首先介绍一种新的基于核的分类方法–卷积树核，及其在语义角色标注上的应用。然而，一般的卷积树核并不适应语义角色标注问题，因此我们针对语义角色标注这一问题，提出了混合卷积树核，该方法显著提高了语义角色标注系统的性能，同时对于其它类似的自然语言处理问题，也具有较好的借鉴作用。本章首先具体指出了基于多项式核的语义角色标注方法存在的一些问题，接着针对该问题，介绍了基于卷积树核的解决方案，然后提出了对卷积树核的改进方法–混合卷积树核。最后，将混合卷积树核与二次多项式核进行了结合，构造的新的复合核进一步提高了语义角色标注系统的性能。

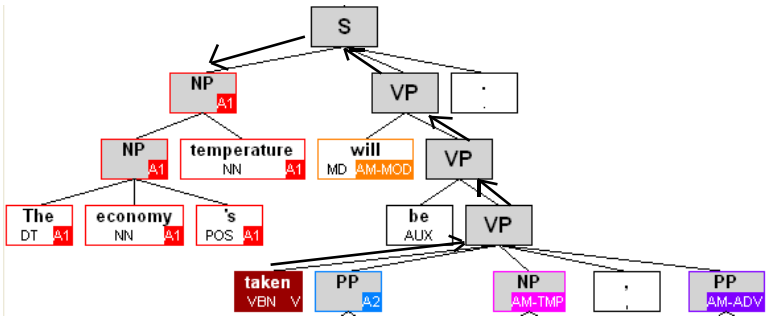
3.2 基于多项式核方法的不足

通过上一章对多项式核方法的介绍，我们可以发现其具有特征构造以及组合灵活，分类性能较高等优点，但是同时我们也要注意该方法的一些明显的不足：

1、特征的构造过程较为繁琐：虽然其不需要特别专业的专家书写细致的规则，但是该方法仍然需要我们对待处理的问题进行深入细致的分析，并给出较为适当的特征模板。这样造成一个明显的问题就是该方法不易扩展，如随着处理语种的变化，特征的提取方法必然需要做出相应的改动等。

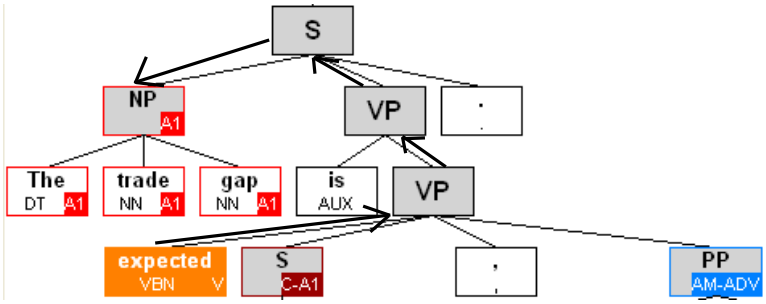
2、不利于表示结构化特征：在基于多项式核方法中，特征一般使用 0 或 1 的二元表示，或者表示为实数。无论用何种方式，对于结构化的特征都是不适用的。例如在语义角色标注中非常重要的路径特征就是典型的结构化特征，当使用二元方式来表示时，即使很相似的两条路径，也会被当作截然不同的

特征。如图 3-1 所示，其中图 3-1(a) 和图 3-1(b) 中，从谓词到句法成分之间的路径分别为 “VBN↑VP↑VP↑VP↑S↓NP” 和 “VBN↑VP↑VP↑S↓NP” 。它们之间仅相差一个 “VP↑”，并且这两个句法成分也具有相同的语义角色。然而，它们的路径特征是不同的，这就会造成一定的数据稀疏问题，从而影响系统的性能。



a) 从谓词 *take* 到论元 Arg1 的路径

a) The Path from Predicate take to Argument Arg1



b) 从谓词 *expect* 到论元 Arg1 的路径

b) The Path from Predicate expect to Argument Arg1

图 3-1 两个路径特征之间的对比

Fig. 3-1 The Comparison between Two Paths

为了克服基于多项式核方法在表示结构化特征时造成的数据稀疏问题，Pradhan 等[49] 试图对路径特征进行了泛化，他们使用了一些启发式的规则，如合并路径中的重复节点，保留部分路径等等。然而，这种基于规则的泛化方法无论是对于新的结构化特征还是新的语种，其可扩展性都不好。另一种泛化方法是将一个大的结构分解为多个子结构，虽然该方法能够克服数据稀疏问题，并且能够适应各种结构以及语种，但是其有可能扩展出数量巨大的子结构 (一般是原来结构规模的指数级) ，在计算上会带来很大的不便。因此，接下来我们按照这一思路，介绍一种可行的解决方案。

3.3 卷积树核

3.3.1 卷积核

在介绍卷积树核之前，我们来看一下什么是卷积核。所谓卷积核 (Convolution Kernel)，是一种通过类似卷积 * 的操作，将较大的结构分解成子结构，然后先计算子结构之间的匹配情况 (求积运算)，再将子结构匹配的结果求和，计算大结构的相似性。Haussler[68] 以及 Watkins[69] 已经证明，这种计算的过程满足核函数成立的条件 (对称以及半正定)，也就是说所构造的相似度函数是一个核函数，并命名为卷积核。同时，他们还给出了一个动态规划的算法，能够在多项式时间内计算卷积核。假如考虑两个字符序列的卷积核，核函数的计算可以通过下面的递归公式实现：

$$k(s,\varepsilon) = 1$$

$$k(sa,t) = k(s,t) + \sum_{j:t_j=a} k(s,t(1:j-1)) = k(s,t) + k'(sa,t)$$

$$k'(sa,tv) = k'(sa,t) + k(s,t)$$

其中，a 是接在字符串 s 后面的一个字符，则字符串 sa 和 t 的共同子串要么在 s 中，要么也是以 a 结束的子串。

例如计算任意两个句子的卷积核，一种方法是将句子分解为连续出现的单词对，然后根据单词对的匹配情况计算卷积核。

例如下面的两个句子：

s = “John loves Mary Smith”

t = “Mary Smith loves John”

所有的单词对的出现情况如表 3-1 所示，其中 1 表示单词对出现，否则为 0，每个单词用首字母代替。

由此表可以看出，两个句子仅有一个单词对相同，因此有 $K(s,t) = 1$。在实际的计算过程中，并非真的将所有的单词对全部列出，然后计算匹配情况，

*卷积的计算过程是先进行求积运算，然后将所有的积求和，即：$(f*g)(t) \equiv \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau$。

而是使用动态规划算法，在更短的时间内计算卷积树。这点和多项式核是一致的。

3.3.2 卷积树核

在树结构上构造的卷积核相应的被称为卷积树核。一棵句法树 T 可以分解为若干子树，并可以由这些子树类型 (不考虑它们的祖先) 个数所构成的向量所表示：

$$\begin{aligned}\Phi(T) &= (\phi_1(T), \phi_2(T), \ldots, \phi_n(T)) \\ &= (\#\text{of sub-trees of type}1, \\ &\quad \#\text{of sub-trees of type}2, \\ &\quad \ldots, \\ &\quad \#\text{of sub-trees of type}n)\end{aligned}$$

因为不同子树的数目是随着树的大小成指数变化的，这样生成一个非常高维的特征空间。因此直接用 $\Phi(T)$ 特征向量来计算是不现实的。为解决这个问题，Collins and Duffy[70] 用一个改进的卷积树核函数来扩展Haussler[68] 和 Watkins[69] 提出的卷积核，这个改进的卷积树核函数能够有效的在高维空间计算两个向量之间的点积。核函数定义如下：

$$\begin{aligned}K(T_1, T_2) &= \langle \Phi(T_1), \Phi(T_2) \rangle \\ &= \textstyle\sum_i (\phi_i(T_1) \cdot \phi_i(T_2)) \\ &= \textstyle\sum_{n_1 \in N_1} \sum_{n_2 \in N_2} \sum_i I_i(n_1) * I_i(n_2)\end{aligned}$$

其中，N_1 和 N_2 分别是树 T_1 和树 T_2 全部节点的集合，指示函数 $I_i(n)$ 的值为 1 ，当且仅当存在一棵以 n 为根节点的类型为 i 的子树，否则值

表 3-1 计算两个句子卷积核的实例

Table 3-1 A sample to compute the convolution kernel between two sentences

	JL	LM	MS	SL	LJ
s	1	1	1	0	0
t	0	0	1	1	1

为 0 。 Collins and Duffy[70] 指出： $K(T_1, T_2)$ 是树结构上的一个卷积核的实例，并可以通过下面的递归定义在 $O(|N_1| \times |N_2|)$ 时间内计算出，其中 $\Delta(n_1, n_2) = \sum_i I_i(n_1) * I_i(n_2)$ ：

(1) 如果 n_1 和 n_2 处的产生式规则不同，则有 $\Delta(n_1, n_2) = 0$ ；

(2) 否则，如果 n_1 和 n_2 子节点相同并且都是叶子节点，则有 $\Delta(n_1, n_2) = \mu$ ；

(3) 否则， $\Delta(n_1, n_2) = \mu \prod_{j=1}^{nc(n_1)} (1 + \Delta(ch(n_1, j), ch(n_2, j)))$ 。

其中 $nc(n_1)$ 是节点 n_1 儿子的个数， $ch(n, j)$ 是节点 n 的第 j^{th} 个儿子， $\mu(0 < \mu < 1)$ 是衰减因子，树的规模越大，则会乘上更多的 μ ，因此可以控制核函数的值不会随着树的规模变大而急剧变大。

Moschitti[72] 提出在语义角色分类中应用卷积树核。他选择包含谓词 - 论元的句法分析树的子结构作为谓词 - 论元特征 (predicate-arguments feature ，PAF) 空间，并在 PAF 特征空间中定义卷积树核。图 3-2 中，给出谓词 buy 和角色 Arg0 的 PAF 核的特征空间。图 3-3 列出在 PAF 特征空间中的全部 15 个子树特征。除了结构特征外， PAF 核还涵盖了许多前面介绍过的特征，如谓词 (VBD→bought) ，核心词 (PRP→she) 等。实际上，除了子结构选择的策略外，PAF 核与 Collins and Deffy[70] 的树核相似。更确切地说， Moschitti[72] 仅仅选择了与谓词和论元相关的结构，并且在该结构之上定义了卷积树核。

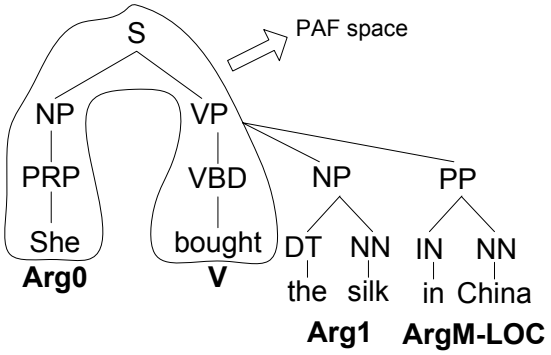


图 3-2 谓词论元特征空间

Fig. 3-2 Predicate Argument Feature space

3.4 用于语义角色标注的混合卷积树核

我们注意到 PAF 特征空间实际由两种特征构成：一种是所谓的句法分析树的路径特征，另一种是所谓的句法成分结构特征。这两种特征空间表示了不同的信息。路径特征捕获一个谓词与它的论元之间的信息，而句法成分结构特征则捕获一个论元的句法结构信息。分别获得这两种特征似乎更合理，因为它们在不同方面对语义角色标注有贡献。并且我们还可以很容易的用不同的权重的线性合并这两种信息。基于以上的考虑，我们计划用两个卷积树核分别捕获两个特征，然后将它们合并成一个混合卷积树核以用于语义角色标注。图 3-4 说明了这两个特征空间，实线圈起的部分是路径特征空间，虚线圈起的部分是句法成分结构特征空间。我们分别将它们命名成路径核和句法成分结构核。形式上，路径核是一个覆盖最小子结构的树核，其中最小子结构包括谓词、句法成分子树根节点和它们的共有祖先节点；句法成分结构核是覆盖一个句法成分的树核。

根据已定义的两个卷积树核，路径核 (K_{path}) 和句法成分结构核 (K_{cs})，我们定义一个新的混合卷积树核来合并这两个单独的核。依照文献[97]，核函数集合在线性合并运算下是封闭的。这意味着如果 K_{path} 和 K_{cs} 都是合法的

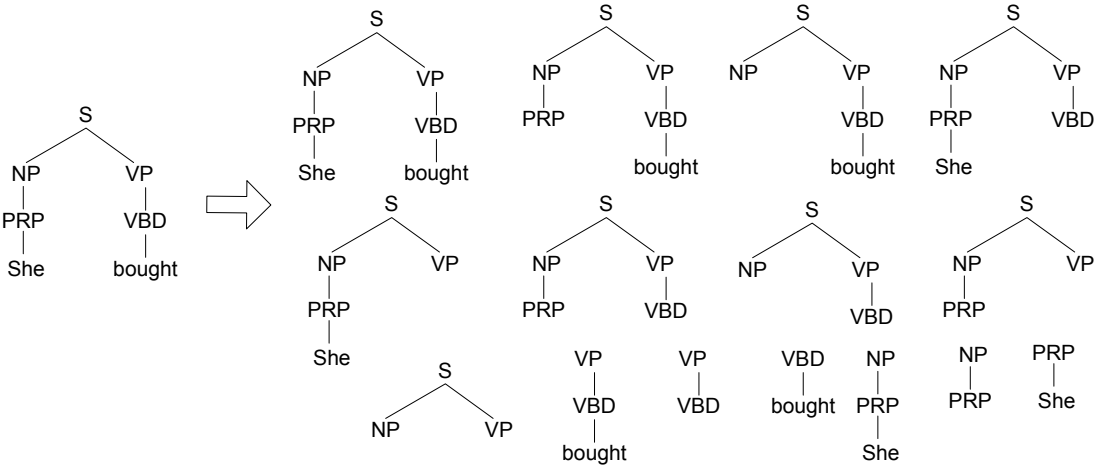


图 3-3 PAF 特征空间扩展的 15 个子树

Fig. 3-3 All 15 Sub-trees Extended from a PAF Space

核函数，则下面的 K_{hybrid} 也是合法的核函数。

$$K_{hybrid}(T_1, T_2) = \lambda K_{path}(T_1, T_2) + (1 - \lambda) K_{cs}(T_1, T_2) \tag{3-1}$$

其中 $0 \leq \lambda \leq 1$，T_1 和 T_2 是两颗句法树。

因为句法分析树大小不是常量，我们分别将 $K_{path}(T_1, T_2)$ 和 $K_{cs}(T_1, T_2)$ 除以 $\sqrt{K_{path}(T_1, T_1) \cdot K_{path}(T_2, T_2)}$ 和 $\sqrt{K_{cs}(T_1, T_1) \cdot K_{cs}(T_2, T_2)}$ 来实现规一化。

与从 PAF 核捕获的特征空间不同，混合卷积树核的新特征空间包括两个独立的部分。图 3-5 说明了新特征空间，其中虚线上方列出了包括 6 棵子树的路径特征空间，虚线下方列出了包括 3 棵子树的句法成分结构特征空间。很明显，这与图 3-3 中列出的 PAF 核的 15 棵子树是不同的。

图 3-6 说明了 PAF 核与我们的混合卷积树核的不同。在 PAF 核中，当分别考察以 NP 和 PRP 为根的句法成分所担当的语义角色时，它们的树结构是相同的，如图 3-6(a) 。然而，这两个成分对句子中的谓词 buy 扮演着不同的角色，不应该被看成是相同的。图 3-6(b) 显示使用混合卷积树核计算的例子。

图 3-7 突出了在图 3-6(b) 中的两个卷积树核之间不同的子树。以 NP 为根 (图 3-6(b)(1)) 比以 PRP 为根 (图 3-6(b)(2)) 多了 4 个路径核特征 (图 3-7(a)) ，少了 2 个句法成分结构的特征 (图 3-7(b)) 。因此，这两个树是有区别的。

另一方面，句法结构特征空间在传统的 PAF 特征空间中占了大部分。在对 CoNLL-2005 共享任务语料库的统计显示句法成分结构大小平均是路径大小的二倍。因此句法成分在 PAF 核计算中扮演重要的角色，如图 3-8 。这里，

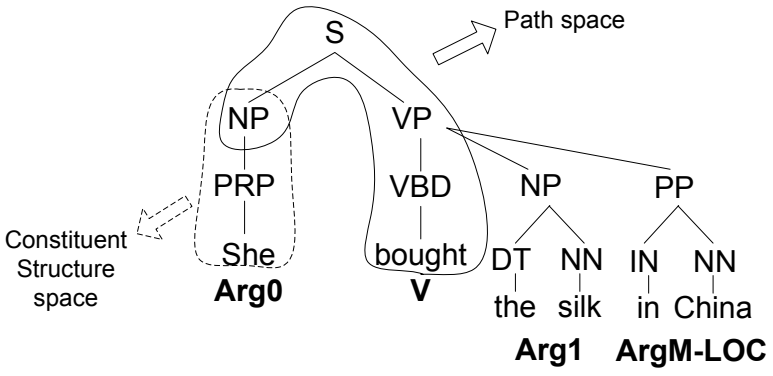


图 3-4 路径和句法成分结构特征空间

Fig. 3-4 Path and Constituent Structure feature spaces

go 是一个谓词，AM-PNC 是一个长的子句。本章实验部分结果显示单独使用句法成分结构核性能不好。因为句法成分结构特征在 PAF 核中占统治地位，

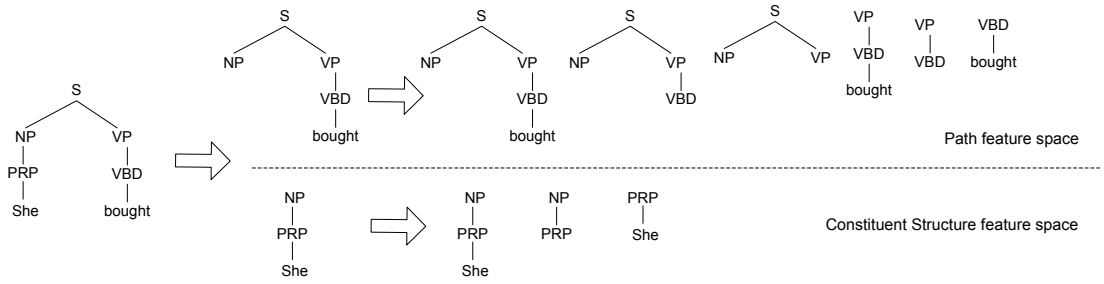
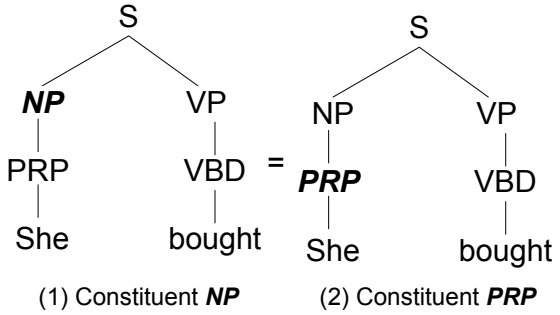


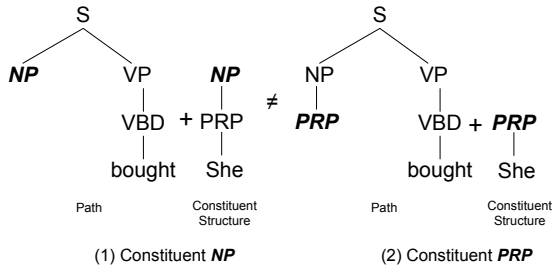
图 3-5 路径特征空间以及句法成分结构特征空间

Fig. 3-5 The Path Feature Space and the Constituent Structure Feature Space



a) PAF 核

a) The PAF kernel



b) 混合卷积树核

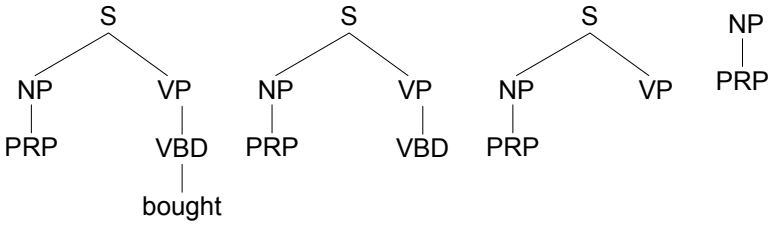
b) The Hybrid Convolution Tree Kernel

图 3-6 PAF 核与混合卷积树核之间的比较

Fig. 3-6 Comparison between the PAF and the Hybrid Convolution Tree Kernels

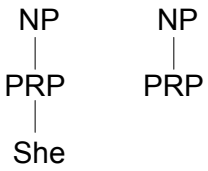
所以 PAF 核的性能也不好。例如，假设最终的 PAF 核值为 0.9(经过规一化)，句法成分结构特征可能贡献出 0.6，路径特征献 0.3。因此，路径特征对最终 PAF 核的贡献不明显。相反，在我们的混合卷积树核中没有这样的问题，因为我们在路径核和句法成分结构核在结合前就被规一化了。这能平衡句法成分结构特征和路径特征的贡献，因此解决了 PAF 的问题。我们也可以调整 K_{path} 和 K_{cs} 的权重而达到最佳性能。仍然用上面的例子，当我们分别对这两种特征进行考虑和规一化，两者都贡献 0.45(假定结合时权重相等)。因此，句法成分结构核的值相应的减小，路径特征的贡献增加。

最后，在 PAF 核中，如果路径和句法成分结构中存在相同的子树，这样就混淆了这两个子树。例如，假设有一个 “VP→VBD” 子树既出现在路径的一个实例中，又出现在句法成分结构的实例中，这两个实例将对 PAF 核贡献为 1。然而，因为它们属于两个不同的特征空间，不应该被认为是相同的。我们的混合卷积树核能够很好的克服这个问题。换句话说，我们的混合卷积树核利用不同的核函数来对不同的语言对象进行建模，这些对象描述目标语言现象的不同特征。从一个机器学习的观点上看，我们的混合卷积树核没有生成与学习目标相关性不大的子结构。



a)　额外增加的路径特征

a) Additional Path features



b)　忽略的句法成分结构特征

b) Ignored Constituent Structure features

图 3-7 使用混合卷积树核获得的不同特征空间

Fig. 3-7 The Different Feature Space with the Hybrid Convolution Tree Kernel

3.5 混合卷积树核与二次多项式核的结合

最后，我们期待通过融合我们的混合卷积树核和二次多项式核，进一步提高语义角色标注系统的性能。

为了充分利用句法成分的信息和标准的扁平特征，我们使用复合核来结合混合卷积树核 (K_{hybrid}) 与二次多项式核的基于特征的方法 (K_{poly})：

$$K_{comp} = \gamma K_{hybrid} + (1-\gamma)K_{poly} \tag{3-2}$$

其中，$0 \leq \gamma \leq 1$ 为组合系数。

与混合卷积树核类似，这种核函数之间的线性组合结果仍为合法的核函数。

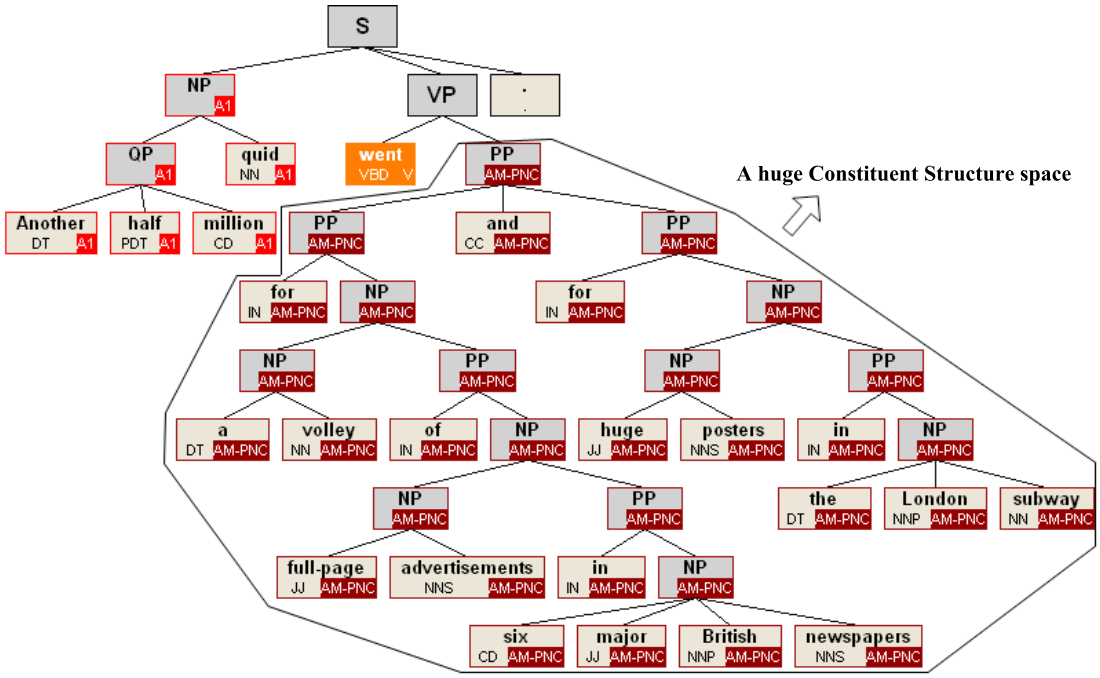


图 3-8 一个语义角色标注的实例

Fig. 3-8 An example of SRL

3.6 相关工作

Mochitti 等[98] 也注意到他们提出的 PAF 核不利于进行语义角色标注，尤其是角色的识别。他们提出了一个改进的 PAF (MPAF, iMproved PAF) 核来进行语义角色标注。在 MPAF 核中，一个句法成分的根节点附加一个 “-B” 符号。因此，对于图 3-6(a) 中的节点 “NP” 和 “PRP”，分别变成了 “NP-B” 和 “PRP-B”。从而，新的核能够区分路径特征与句法成分结构特征的边界线。与我们的混合卷积树核相比，如图 3-9 所示，MPAF 仍有 3 个共同的特征，然而 MPAF 不能捕获包含 “S” 的结构 “S→NP VP”，而 “S→NP VP” 是路径特征的一部分并对语义角色标注来说是个重要的信息。另一方面，尽管 MPAF 方法考虑了路径特征与句法成分结构特征的边界线，MPAF 仍然将它们看成一个整体结构。因此，与 PAF 核相同，它仍然强调句法成分相关的特征，不能分别灵活的考虑路径特征和句法成分结构特征不同的贡献。我们后面的实验也显示了 MPAF 核性能不如我们的混合卷积树核。

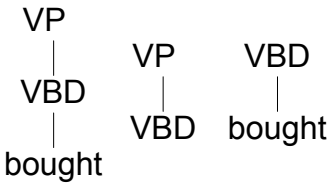


图 3-9 MPAF 核的共同的特征

Fig. 3-9 The Common Features for the MPAF Kernel

最后，据我们所知，目前所有基于卷积树核的工作和以及树核方法在自然语言处理中的应用只利用了一个树核来解决它们所面对的问题。尽管 Zhang 等[99] 比较了不同的树核空间用于关系提取，他们也没有考虑合并一些卷积树核。

3.7 实验及讨论

3.7.1 分类器的实现

在此，我们对 SVM-Light-TK* 进行了改进，即使之能够融入多个卷积树核，以实现混合卷积树核。SVM-Light-TK 以 SVM-Light† 这一流行的支持向量机实现为基础，使之可以嵌入卷积树核函数。

3.7.2 实验结果及讨论

为了加快调整参数以便选择出一组最佳的参数集合的过程，在下面的实验中，我们只用 CoNLL-2005 语义角色标注共享任务语料库中的 WSJ section 02-05 作为训练集和在 CoNLL-2005 开发集上调整好的参数。最终，我们报告的性能则采用了全部训练集和调整好的参数。同 Moschitti[72] 一样，我们在卷积树核的计算中设置树核的衰退因子 $\mu = 0.4$ 。

在 CoNLL-2005 开发集中改变 λ (在公式 (3-1) 中混合卷积树核的权重) 的性能曲线显示在图 3-10 中。这里，我们在 SVM-Light 中用 SVM 默认的设置参数。

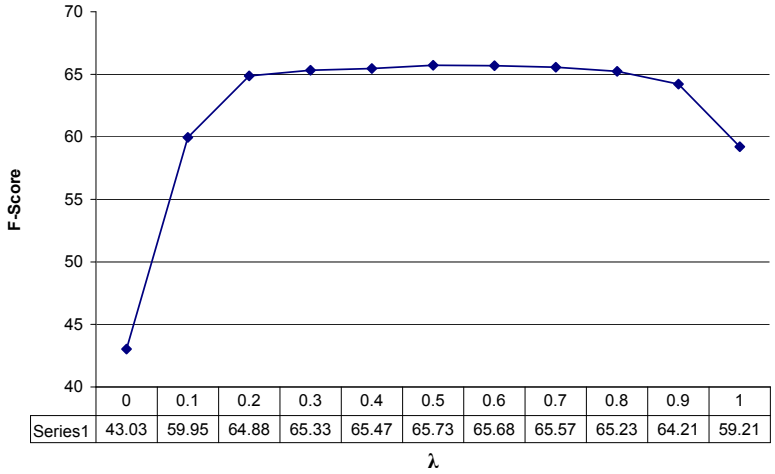


图 3-10 系统性能随着 λ 变化的曲线图

Fig. 3-10 The Performance Changing with λ

* http://dit.unitn.it/∼moschitt/Tree-Kernel.htm
†http://svmlight.joachims.org

图 3-10 显示当 $\lambda = 0.5$ 时，混合卷积树核的性能最好，$\mathrm{F}_{\beta=1} = 65.73$ 。无论是路径核 ($\lambda = 1$ ， $\mathrm{F}_{\beta=1} = 59.21$) ，还是句法成分结构核 ($\lambda = 0, \mathrm{F}_{\beta=1} = 43.03$) 性能都没有混合卷积树核好。说明这两个独立的核相互补充，这也就是我们将 PAF 核分成一个路径核和一个句法成分结构核的原因。另外，路径核性能比句法成分结构核好。通过对参数 λ 的调整，我们能够减少句法成分结构核的影响，使它们总体的贡献最佳。我们设置参数$\lambda = 0.5$ 。两个独立的核 (K_{path} 与 K_{cs}) 分别规一化。与 PAF 核不同，如图 3-8 中 PAF 核中句法成分结构核在最后的核值中起主导作用，而我们的混合卷积树核更强调路径核的贡献。

表 3-2 比较了我们的混合卷积树核， Moschitti 的 PAF 核，MPAF 核，线性核，以及二次多项式核在 CoNLL2005 开发集和测试数据 (WSJ section23 和 Brown 语料) 上的性能。这里， WSJ section 02-05 做训练数据。值得指出的是表 3-2 中列出的参数 C 和 λ 分别是在 CoNLL-2005 开发集上每个独立的方法的最佳参数。

表 3-2 不同系统的性能 ($\mathrm{F}_{\beta=1}$) 比较
Table 3-2 The Performance ($\mathrm{F}_{\beta=1}$) Comparison among Different Systems

	Hybrid ($C = 4.5, \lambda = 0.5$)	PAF ($C = 4$)	MPAF ($C = 4$)	Linear ($C = 2$)	Polynomial ($d = 2, C = 4$)
Devel	68.90	67.03	67.80	69.36	72.63
Test (WSJ)	71.34	69.80	70.61	71.29	74.42
Test (Brown)	60.97	60.11	60.24	60.30	62.24

我们能够从统计上看到混合卷积树核在全部的开发集和测试集都明显 ($p = 0.05$ 下的 χ^2 测试) 优于 PAF 核。另外，虽然 MPAF 核也优于 PAF 核，但它的性能仍然比我们的混合卷积核明显 ($p = 0.05$ 下的 χ^2 测试) 差。说明我们的混合卷积树核在语义角色标注上比 PAF 和 MPAF 核更加有效。另外，第 1 列和 4 列的比较显示只使用混合卷积树核时，其在性能上能够与只用线性核的基于特征的方法相比较。这意味着如果句法结构能够有效的建模，我们的方法能够与那些用大量不同的特征的方法相竞争。

然而，我们的混合卷积树核仍然比用多项式核的基于特征的方法差。原因很简单，因为我们的核只用了句法成分结构的信息，而基于特征的方法用了大量的不同特征，包括词，词性，语态等和它们的组合特征。用二次多项式

核的基于特征的方法获得了最佳性能。这说明通过多项式核实现的特征的二元组合非常有用。因此，我们期待通过组合我们的混合卷积树核和多项式核后性能会更好。即使用复合核 K_{comp} 。

应用于 CoNLL 2005 共享任务的开发集时，性能随着 λ 的变化显示在图 3-11 中。这里，请注意我们在二次多项式核中的 SVM 使用默认的 C 参数。

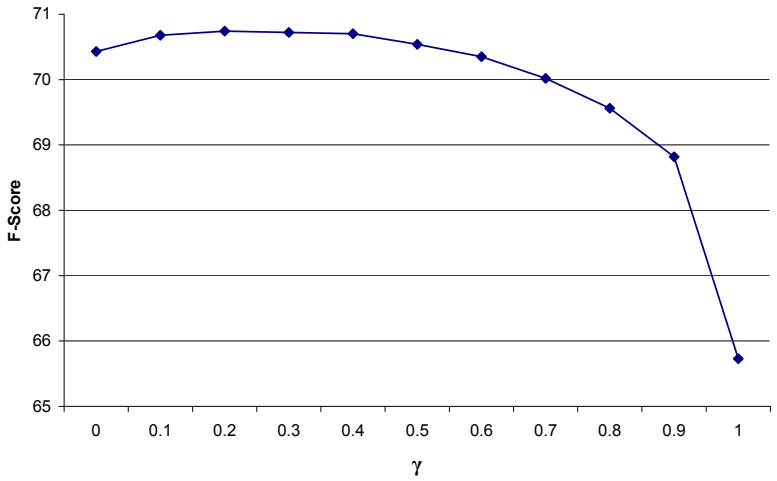


图 3-11 系统性能随着 γ 变化的曲线图

Fig. 3-11 The Performance Changing with γ

我们能够看到当 $\gamma = 0.2$ 时，系统达到最佳性能 $\mathrm{F}_{\beta=1} = \mathbf{70.74}$ 。从统计上看与只用多项式核 ($\gamma = 0$ ， $\mathrm{F}_{\beta=1} = 70.43$) 的基于特征的方法相比有明显的提高，比只用混合卷积树核 (($\gamma = 1, \mathrm{F}_{\beta=1} = 65.73$)) * 高出许多。主要的原因是虽然基于混合卷积树核的方法与标准的基于特征的方法相比能表示更多的句法信息，但是基于特征的方法能获得基于混合卷积树核的方法所不能表示的特征，如语态，命名实体等。这两种方法相互补充。

为了找出复合核的最佳参数，我们再一次的调整 C ，发现当 $C = 4$ 时最佳。

最后，我们用上面的参数设置 (也就是说 $\lambda = 0.5$ ， $\gamma = 0.2$ ，默认的 $e = 0.001$ ，对每个独立的方法使用最佳的 C) 在全部的 CoNLL-2005 语义角色标注共享任务的训练集 (WSJ sections 02-21) 上训练复合核和二次多项式核。表 3-3 对新的复合核 (混合卷积树核 + 二次多项式核) 与上一章介绍的二次多

* 请注意用默认 C 参数得到的所有结果与表 3-2 中的不同，表 3-2 的 C 是最佳的。

项式核进行了对比。结果显示复合核比多项式核性能提高了 0.3% ∼ 0.6%，说明基于树核的方法能够进一步挖掘结构信息，从而进一步提高语义角色标注系统的性能。

表 3-3 复合核与多项式核之间的性能对比

Table 3-3 Performance ($F_{\beta=1}$) Comparison between the Composite Kernel and the Polynomial Kernel

	Composite ($C = 4$)	Polynomial ($d = 2, C = 4$)
Devel	75.66	75.37
Test (WSJ)	77.41	77.00
Test (Brown)	66.21	65.63

3.8 本章小结

在本章中，我们提出了一个混合卷积树核来对语义角色标注中所使用的句法结构信息进行建模。与之前的基于卷积树核的方法不同的是，我们区别了路径和句法成分结构特征空间。在 CoNLL-2005 语义角色标注共享任务的数据集合上实验显示混合卷积树核明显优于之前的 (PAF) 和它的改进版本 (MPAF) 。最终混合卷积树核方法与基于特征的多项式核方法构成的复合核明性能显优于单纯的多项式核方法。以上说明了使用卷积树核对结构化信息建模的有效性。

第 4 章　句法驱动混合卷积树核

4.1 引言

卷积树核要求两棵子树之间必须是精确匹配的，而不考虑任何语言学的知识。这就使得该方法不能处理相似的短语结构 (例如：“buy a car” 和“buy a *red* car”) 以及相似的句法标记 (例如：“*high*/JJ degree/NN” 和“*higher*/JJR degree/NN” 之间的词性变化)。在自然语言中，一些产生式是由另外一些产生式生成的，例如“NP→DT JJ NN” 就是“NP→DT NN” 的一个特例。相同的情况也发生在词性上。然而，标准的卷积树核不能捕获这些语言学知识。为了克服这些问题，本章提出了一种新的句法驱动的卷积树核，在核函数的设计过程中，融入了语言学知识，并在语义角色标注问题中取得了预期的效果。

4.2 句法驱动卷积树核的设计

我们首先介绍自然语言中两种近似匹配情况：句法结构的近似匹配和句法分析树中节点的近似匹配。然后利用这两种近似匹配，设计出句法驱动的卷积树核。

4.2.1 句法驱动的近似子结构匹配

一个标准的卷积树核要求只有完全相同的子结构才是匹配的。这个限制过于严格。例如，在两个句法分析树中，“NP→DT JJ NN” (NP→*a red car*) 和 “NP→DT NN” (NP→*a car*) 对标准卷积树核没有任何贡献，尽管它们有很相似的句法结构属性，从而对一个给定的谓词很可能扮演相同的语义角色。因此，我们提出一种句法驱动的近似匹配机制，通过构造具有可选节点的简化产生式集合，来捕获这类结构上的相似，例如 “NP→DT [JJ] NP” 中的 [JJ] 和 “VP→VB [ADVP] PP” 中的 [ADVP] 都是可选节点。为了方便，我们称 “NP→DT JJ NP” 为原始产生式，“NP→DT [JJ] NP” 为简化产生式。这里，我们定义两个句法驱动标准来确定简化产生式。

(1) 简化产生式必须是合乎文法的。这意味着一个简化的产生式应该是有效的。也就是出现在训练数据所收集到的原始产生式集合中。例如

“NP→DT [JJ] NP” 是有效的，因为 “NP→DT NP” 在原始产生式集合中出现过，而 “VP→[VB ADVP] PP” 是无效的，因为 “VP→PP” 没在原始产生式集合中出现。

(2) 一个简化的产生式必须保留原始产生式的核心子节点和至少有 2 个子节点。这是为了保证简化的产生式保留了原始产生式大部分语义和避免简化产生式的过泛化。核心节点的确定方法与 2.2.4 节介绍的相同，都是使用 Collins 在其博士论文[73] 附录中描述的句法成分核心词识别规则。

在此，原始的产生式集合是从训练语料中提取出来的。通过定义每种短语类型 (如名词短语 NP) 的可选节点，我们能够自动的从原始产生式集合构造一个简化的产生式集合。相同短语类型的产生式 A 和 B 被标记成匹配的，则它们应该在删除 0 或多个可选节点后相等。给出简化的产生式集合，我们能够用下面的公式表示相似子结构匹配机制：

$$M(r_1, r_2) = \sum_{i,j} (I_T(T^i_{r_1}, T^j_{r_2}) \times \lambda_1^{a_i+b_j}) \tag{4-1}$$

这里：

(1) r_1 是一个产生式，代表深度为 1 的一个子树 *，r_2 相同。

(2) $T^i_{r_1}$ 是子树 r_1 通过删除 0 个或多个可选节点得到的第 i 个变种，对于 $T^i_{r_2}$ 同样。只有当使用一个合法的核时，核方法的训练算法收敛。为了保证我们的新核是合法的，则原始子树的所有可能变种都应该考虑。

(3) $I_T(\cdot, \cdot)$ 是一个二元函数当且仅当两个个子树是相同的，否则为 0 。

(4) λ_1 ($0 \leq \lambda_1 \leq 1$) 是惩罚删除的可选节点的权重，a_i 和 b_j 分别表示在子树 $T^i_{r_1}$ 和 $T^j_{r_2}$ 中删除的可选节点的个数。

$M(r_1, r_2)$ 返回了子树 r_1 和 r_2 的相似度，相似度是通过计算它们所有可能的变种的相似度得出的。请注意为了保证新的核是合法的，在公式 (4-1) 中考虑了全部可能的变种。在近似匹配机制下，两个子树在适当的权重下匹配，如果它们在删除 0 个或多个可选择的节点后是相同的。这样，“NP→*a red car*” 和 “NP→*a car*” 在我们设计的新核中是匹配的。因此，伴随着相似子结构匹配机制，我们的方法能够比标准卷积树核使用更多满足语言学约束的相似子结构。

*多层的子树相似匹配能够剃归的完成。我们在后面的章节中讨论。

4.2.2 句法驱动的相似节点匹配

标准的卷积树核只考虑两个 (终止 / 非终止) 节点间的准确匹配。然而，一些相似的词性或短语标记可以表示相似的角色，例如 NN (*dog*) ，NNS (*dogs*) 和 NP (*the dog*) 。为了考虑这样的情况，我们通过引入一些等价节点特征集合来在节点特征间进行相似匹配。如：

- JJ, JJR, JJS
- VB, VBD, VBG, VBN, VBP, VBZ
- NN, NNS, NNP, NNPS, NX*

在同一行的节点特征被认为是相似的，能彼此匹配。与相似子结构匹配机制类似，我们介绍一个新的参数λ_2 ($0 \leq \lambda_2 \leq 1$) 用来加权相似节点匹配。我们叫它节点特征变种 † ，相似节点匹配机制用公式表示成：

$$M(f_1, f_2) = \sum_{i,j} (I_f(f_1^i, f_2^j) \times \lambda_2^{a_i+b_j}) \tag{4-2}$$

这里 f_1 是节点特征，f_1^i 是 f_1 的第 i 个变种，a_i 为 0 当且仅当 f_1^i 和 f_1 是相同的，否则为 1 ，f_2 和 b_j 也类似。函数 $I_f(\cdot,\cdot)$ 为 1 当且仅当两个特征相同，否则为 0 。公式 (4-2) 将全部相同的节点特征变种相加作为节点特征相似度。同公式 (4-1) 一样，为了确保新的核是合法的核，在公式 (4-2) 中考虑了所有可能情况。

4.2.3 句法驱动的卷积树核

相似子结构和节点匹配机制都是句法驱动的，也就是它们保留基本的语言学约束以及原始产生式的语义。给出两个这样的相似匹配机制，我们能够一步步地定义我们的句法驱动卷积树核。首先，我们能够用一个新的特征向量表示一个句法分析树 T ：

$$\Phi'(T) = (\#subtree_1(T), \ldots, \#subtree_n(T))$$

这里 $\#subtree_i(T)$ 是 T 中第 i 种类型子树出现的个数。请注意，与标准卷积树核不同，我们通过引入简化的产生式 (通过相似子结构匹配) ，和节点

*在 Penn TreeBank II 中，短语标记 NX 定义了一类复杂的名词短语 (NP) 。

†在本文中，非终止节点依照句法相似度被分为若干组，一个组中的非终节点被认为是该组中另一个非终节点的变种。

特征变种 (通过相似节点匹配) 来放宽子树出现的条件。换句话说，子树出现的准则被放宽了。例如，在新的核中，一个子树 “NP→DT JJ NP” 分别对它的副本 “NP→DT JJ NP” 和 “NP→DT NP” 贡献 1 和 λ_1 ，而在标准核中只对 “NP→DT JJ NP” 贡献 1 。

然后，我们定义两个句法分析树间的句法驱动核：

$$\begin{aligned} K_G(T_1, T_2) &= \langle \Phi'(T_1), \Phi'(T_2) \rangle \\ &= \sum_i \#subtree_i(T_1) \cdot \#subtree_i(T_2) \\ &= \sum_i ((\sum_{n_1 \in N_1} I'_{subtree_i}(n_1)) \cdot (\sum_{n_2 \in N_2} I'_{subtree_i}(n_2))) \\ &= \sum_{n_1 \in N_1} \sum_{n_2 \in N_2} \Delta'(n_1, n_2) \end{aligned} \tag{4-3}$$

其中， N_1 和 N_2 分别是树 T_1 和 T_2 节点的集合。给出的 a 和 b 分别是全部删除的可选择节点的个数和在节点 n 上的变种节点特征的个数，$I'_{subtree_i}(n)$ 等于 $\lambda_1^a \cdot \lambda_2^b$ ，当且仅当有一个以 n 为根的子树 $subtree_i$ ，否则为 0 。此外，$\Delta'(n_1, n_2)$ 计算以 n_1 和 n_2 为根的共同子树的加权个数，例如：

$$\Delta'(n_1, n_2) = \sum_i I'_{subtree_i}(n_1) \cdot I'_{subtree_i}(n_2) \tag{4-4}$$

显然，$\Delta'(n_1, n_2)$ 能够进一步用下面的递归公式计算出来。

R-A: 如果 n_1 和 n_2 是词性节点，则：

$$\Delta'(n_1, n_2) = \lambda \times M(f_1, f_2) \tag{4-5}$$

其中， f_1 和 f_2 分别是节点 n_1 和 n_2 的特征。$M(f_1, f_2)$ 在公式 (4-2) 中定义过，同标准卷积树核相似，惩罚因子 λ $(0 \leq \lambda \leq 1)$ 使得核不随子树大小发生剧烈变化。

R-B: 否则，如果 n_1 和 n_2 都是相同的非终结节点，通过删除可选节点来生成根为 n_1 和 n_2 且深度为 1 的全部变种子树(分别由 T_{n_1} 和 T_{n_2} 表示) ：

$$\begin{aligned}\Delta'(n_1,n_2) = \lambda \times \sum_{i,j}(I_T(T^i_{n_1},T^j_{n_2}) \times \lambda_1^{a_i+b_j} \\ \times \prod_{k=1}^{nc(n_1,i)}(1+\Delta'(ch(n_1,i,k),ch(n_2,j,k))))\end{aligned} \tag{4-6}$$

其中：

- $T^i_{n_1}$ 和 $T^j_{n_2}$ 分别表示 T_{n_1} 的第 i 个变种子树和 T_{n_2} 的第 j 个变种子树。
- $I_T(\cdot,\cdot)$ 是一个指示函数，当且仅当两个子树相同时为 1 ，其它时为 0 。
- a_i 和 b_j 分别代表子树 $T^i_{n_1}$ 和 $T^j_{n_2}$ 中删除的可选节点的个数。
- $nc(n_1,i)$ 返回 n_1 中第 i 个子树变种 $T^i_{n_1}$ 中全部子节点个数。
- $ch(n_1,i,k)$ 是 n_1 中第 i 个子树变种 $T^i_{n_1}$ 的第 k 个子节点， $ch(n_2,j,k)$ 也一样。
- 与 **R-A** 类似， λ $(0 \leq \lambda \leq 1)$ 使得核不随子树大小发生剧烈变化。

R-C: 否则 $\Delta'(n_1,n_2) = 0$ 。

R-A 表示了相似节点匹配而 **R-B** 则表示相似子结构匹配。在 **R-B** 中，公式 (4-6) 通过递归算法能够计算多层子树相似匹配的情况，而公式 (4-1) 只对两层的子树有效。为了高效的计算，我们将公式 (4-4) 重写成公式 (4-5) 和公式 (4-6) 。因为公式 (4-4) 是点积，我们能够很容易地用 Haussler[68] 所给的方法证明在公式 (4-3) 中我们定义的核是合法的核。

可以看到，标准卷积树核和句法驱动树核包含的子结构是不同的。图 4-1 显示句法驱动核能获得额外的 17 个有适当加权的句法子结构。

4.3 句法驱动的卷积树核的有效计算

句法驱动卷积树核一个主要的问题是它的计算复杂性偏高，由于相似子结构和节点匹配机制，很明显如果采用蛮力的计算方法，句法驱动卷积树核计算是指数型的。特别是计算公式 (4-6) 需要指数时间，因为要通过删除一个或多个可选节点来生成当前子树的所有可能变种，尽管在树库中，有限的产

生式数量限制了简化产生式的数量。这里，我们用动态规划算法在多项式时间里计算句法驱动核。与部分树 (PT) 核[100] 相似，为了在两个具有可选节点的句法分析树间的找到共同子树，我们重新改造公式 (4-6)，并重写成：

$$\Delta'(n_1, n_2) = \lambda \times \sum_{p=lx}^{lm} \Delta_p(c_{n_1}, c_{n_2}) \tag{4-7}$$

其中：

- c_{n_1} 和 c_{n_2} 分别是 n_1 和 n_2 的子节点序列；
- $\Delta_p(\cdot,\cdot)$ 计算了根为 n_1 和 n_2 并且有 p 个子节点 (至少包括全部非可选节点) 的共同子树的个数；
- $lx = \max\{np(c_{n_1}), np(c_{n_2})\}$，其中 $np(\cdot)$ 是非可选节点的个数；
- $lm = \min\{l(c_{n_1}), l(c_{n_2})\}$，其中 $l(\cdot)$ 返回子节点的个数；

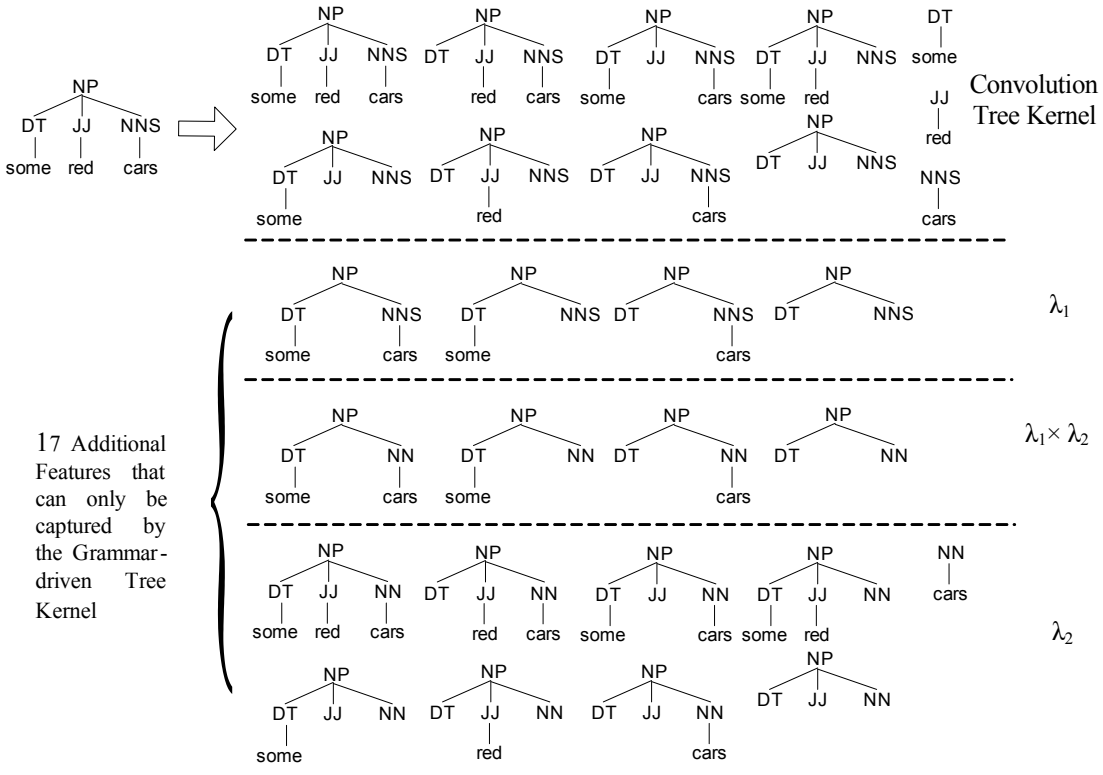


图 4-1 句法驱动卷积树核包含的全部子结构

Fig. 4-1 All of the Substructures Covered by the Grammar-driven Convolution Tree Kernel

• 同标准卷积树核相似，惩罚因子 λ ($0 \leq \lambda \leq 1$) 使得核不随子树大小发生剧烈变化。

现在问题变成如何有效地计算 $\Delta_p(c_{n_1}, c_{n_2})$ 。我们设计一个与 PT 核计算中使用的相似的动态规划算法。让我们首先看一下 PT 核中的动态规划算法。给出两个节点序列 $c_1 a = c_{n_1}$ 和 $c_2 b = c_{n_2}$ (a 和 b 分别是它们的最后一个子节点)，在 PT 核中动态规划算法递归计算 $\Delta_p(c_{n_1}, c_{n_2})$，如下：

$$\begin{aligned}\Delta_p(c_{n_1}, c_{n_2}) &= \Delta_p(c_1 a, c_2 b) \\ &= I(a,b) \times (1 + \Delta'(a,b)) \times \Delta_{p-1}(c_1, c_2) \\ &+ \Delta_p(c_1 a, c_2) + \Delta_p(c_1, c_2 b) - \Delta_p(c_1, c_2)\end{aligned} \tag{4-8}$$

两个停止标准为：

$$\Delta_p(c_1, c_2) = 0, \forall \text{ node sequence pair } c_1 \text{ and } c_2, \text{ if } \min(|c_1|, |c_2|) < p \tag{4-9}$$

$$\Delta_0(c_1, c_2) = 1, \forall \text{ any node sequence pair } c_1 \text{ and } c_2 \tag{4-10}$$

其中，$I(a,b)$ 是一个二元函数，当且仅当两节点匹配时为 1，其它情况为 0。明显地，公式 (4-8) 是一个典型的动态规划算法，公式 (4-9) 和 (4-10) 是两个动态规划算法的停止标准。此外，公式 (4-8) 表明：

• 如果最后的子节点节点 a 和 b 匹配，那么我们能够：1) 用 $\Delta'(a,b)$ 更一步的计算根为这两个相互匹配节点对的相同子树的个数；2) 用 $\Delta_{p-1}(c_1, c_2)$ 来求 c_1 和 c_2 间子节点个数为 $p-1$ 的共同子树的数目。

• 此外，$\Delta_p(\cdot,\cdot)$ 在删除节点 a 和 b 的短节点序列上被分别调用两次。因为 $\Delta_p(c_1, c_2)$ 将会在下一个递归调用中的 $\Delta_{p-1}(c_1 a, c_2)$ 和 $\Delta_{p-1}(c_1, c_2 b)$ 被调用两次，我们在公式 (4-8) 中删除一次调用来避免重复计算。实际上，这种两个计算问题在自然语言处理问题中经常出现，有时被称为 “spurious ambiguity”[101]。

• 计算 $\Delta'(n_1, n_2)$ 的时间复杂度为 $O(p|c_{n_1}| \cdot |c_{n_2}|)$，因为对于任意节点对

a 和 b，$\Delta'(a,b)$ 只调用一次。

与 PT 核相比，句法驱动树核有两点不同：

C1: 句法驱动核需要过滤可选节点而 PT 核对节点过滤没有限制；

C2: 句法驱动核处罚删除的可选节点 (包括内部和外部的可跳过节点) 而 PT 核将子序列的长度加权 (只考虑全部内部可跳过节点，而外部节点被忽略) 。

通过对两个因素的考虑，$\Delta_p(c_1,c_2)$ 在句法驱动核中计算如下：

$$\begin{aligned}\Delta_p(c_{n_1},c_{n_2}) &= \Delta_p(c_1a,c_2b)\\ &= I(a,b)\times(1+\Delta'(a,b))\times\Delta_{p-1}(c_1,c_2)\\ &+ \Delta_p(c_1a,c_2)\times opt(b)\times\lambda_1\\ &+ \Delta_p(c_1,c_2b)\times opt(a)\times\lambda_1\\ &- \Delta_p(c_1,c_2)\times opt(a)\times opt(b)\times\lambda_1^2\end{aligned} \tag{4-11}$$

两个停止标准为：

$$\Delta_p(c_1,c_2)=0,\forall \text{ node sequence pair } c_1 \text{ and } c_2,\ \text{if } \min(|c_1|,|c_2|)<p$$

$$\Delta_0(c_1,c_2)=1\times opt(c_1)\times opt(c_2)\times\lambda_1^{|c_1|+|c_2|},\forall \text{ node sequence pair } c_1 \text{ and } c_2$$

其中，$opt(w)$ 是一个二元函数，当非可选节点在节点序列 w 中出现时为 1，否则为 0 (C1)；λ_1 是反映跳过的可选节点的权重，λ_1 的值是跳过的可选节点的个数 (C2) 。与 PT 核相比，句法驱动程序加入了函数 $opt(\cdot)$ 和惩罚因子 λ_1 。这是为了确保句法驱动核只跳过可选节点和处罚那些被跳过的可选节点的子树。

很明确地，计算 $\Delta'(n_1,n_2)$ 的复杂度为 $O(p|c_{n_1}|\cdot|c_{n_2}|)$ 。这意味着句法驱动卷积树核的计算复杂度在最坏情况下为 $O(p\rho^2|N_1|\cdot|N_2|)$ 。$\rho=\max_i(|c_{n_i}|)$ 是两棵树的最大分枝。注意在自然语言处理问题中，树平均的 ρ 通常是很小的，通过避免计算两个具有不匹配标记的节点使全部的复杂度减小 (参考上

一节的 **R-C**) 。

4.3.1 与其它相关工作的比较

常规的卷积树核[70] 是我们句法驱动核的一个特殊情况。从核函数的观点上看，我们的核不仅考虑精确匹配，还考虑相似匹配。从特征提取观点上看，尽管它们都提取相同的子结构 (通过短语分析规则递归定义的) ，但是它们的子结构定义不同，因为句法驱动核试图提取符合语言学的子结构。图 4-2 比较了两个树上标准核和句法驱动核。它显示标准核只能够得到两个相同子树，而句法驱动核能获得另外 9 个共同的子结构。

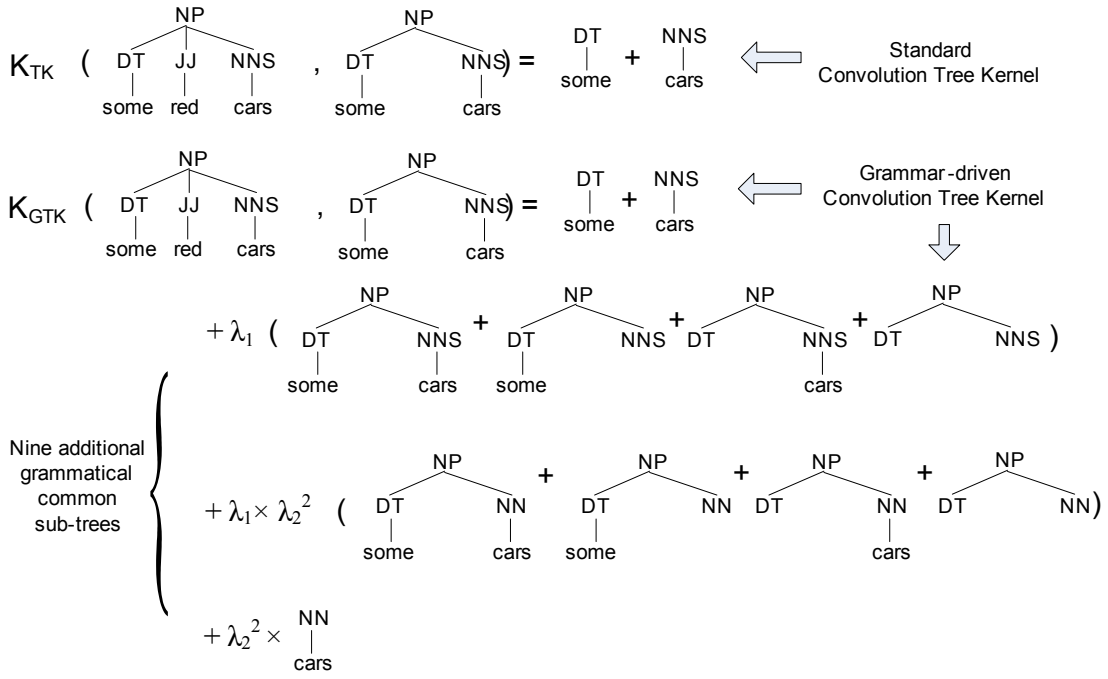


图 4-2 卷积树核与句法驱动核的比较

Fig. 4-2 Comparison of the Convolution Tree Kernel and the Grammar-driven Kernel

与允许子树间部分匹配的部分树 (PT) 核[100] 相比，标准卷积树核以及句法驱动的树核都产生较少的子结构。图 4-3 比较了 PT 核与标准卷积树核。在某种意义上，句法驱动树核是 PT 核的一种特殊情况。主要的不同是 PT 核不是句法驱动的，因此允许许多的不符合语言学知识的子结构匹配。这会影响语义角色标注的性能，由于缺少语言学的解释和约束，一些生成的非句法子

结构可能是噪声。另一个不同是 PT 核不允许相似节点匹配。因此说，句法驱动核比 PT 核使用更多的语言学知识。

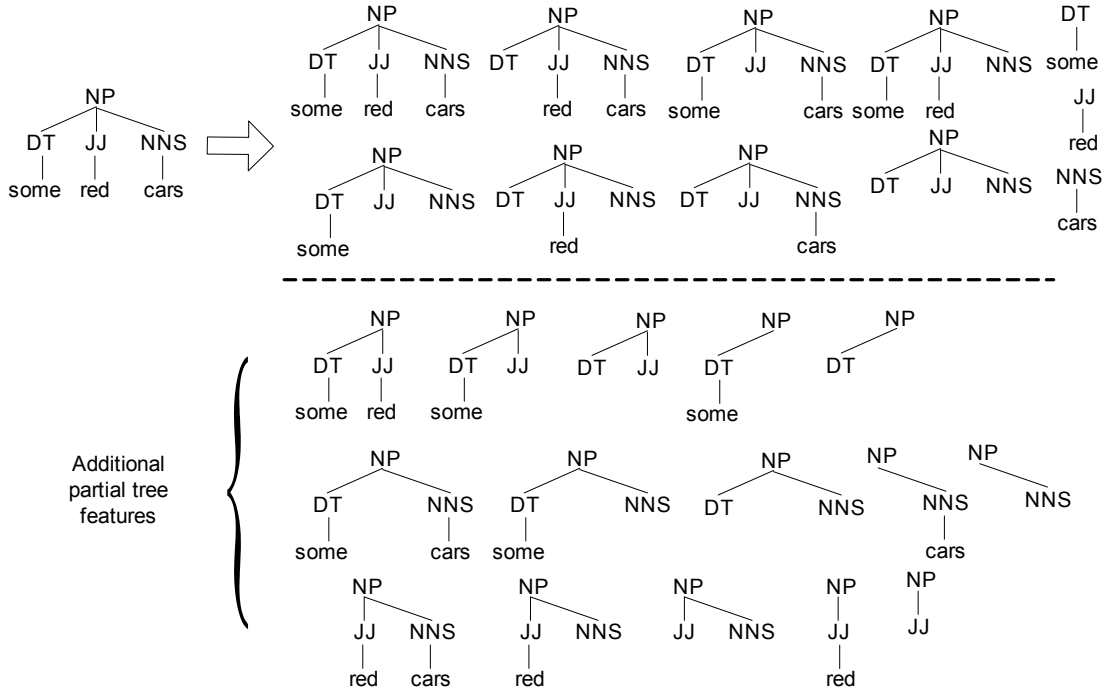


图 4-3 部分树 (PT) 核包括的全部子结构

Fig. 4-3 All of the Substructures Covered by the Partial Tree (PT) Kernel

4.4 实验及讨论

在这一节中，我们系统地对句法驱动卷积树核在语义角色标注中的性能进行了评测。

4.4.1 实验设置

本节实验中使用的语料库、语义角色标注的步骤等设置以及评测方法等均与前面章节所介绍的相同，主要改动在于我们将 SVM-Light-TK 工具 * 中的树核进行了修改，使之支持句法驱动树核。

* http://dit.unitn.it/∼moschitt/Tree-Kernel.htm

在此，我们使用上一章介绍的混合卷积树核 (K_{hybrid}) 作为基线系统，在其基础上增加句法驱动树核，并形成句法驱动的混合卷积树核 ($K_{G-hybrid}$)。

其中，为了进行相似结构匹配，我们在 CoNLL 2005 全部训练语料中提取了最少出现 5 次的大概 4,700 个产生式，包括最常见的 NP 、VP 和 ADJP 短语。最后，我们通过在每种短语中设置可选节点的方法，共获得 1,404 个有可选节点的产生式，每种短语中的可选节点如下：

- NP: JJ, ADJP, ADVP, CC
- VP: ADVP
- ADJP: ADVP, CC, RB

为了相似节点匹配，我们定义了如下三个等价的节点特征集合：

- JJ, JJR, JJS
- RB, RBR, RBS
- NN, NNS, NNP, NNPS, NAC, NX

这里，我们没有加入动词相关的等价节点特征集合 “VB, VBD, VBG, VBZ”, 因为动词语态信息在区别 Arg0 (Agent, operator) 和 Arg1 (Thing operated) 时往往是非常有用的。

4.4.2 实验结果

为了加快实验的参数调整速度，只有 4 个 WSJ sections (Section02-05) 用作训练集并在开发集上调整参数。当然，下面报告的所有性能都使用了全部训练数据和测试数据。

最终，我们设置树核的处罚因子 $\lambda = 0.4$ [72]，SVM 调整参数 $c = 2.4$，混合核参数 $\theta = 0.6$。另外，两个处罚因子 λ_1 (用于相似结构匹配) 和 λ_2 (用于相似节点匹配) 分别调为 0.6 和 0.3。

表 4-1 在测试数据上比较了不同方法对于语义角色分类的性能。它显示了：

1、句法驱动混合卷积树核性能明显 ($p = 0.05$ 下 χ^2 测试) 好于没有句法驱动的核，提高了 2.75% (87.96%-85.21%)。这说明附加语言学子结构对语义角色标注非常有用，并且由于在核设计中考虑语言学知识，句法驱动核在捕获有用的子结构上更加有效。

2、句法驱动相似节点和子结构匹配机制对语义角色分类非常有用，分别在性能上提高了 1.06% (86.27%-85.21%) 和 1.91% (87.12%-85.21%)。而且，他

们的贡献是累加的。

3、我们的相似子结构和节点匹配机制的句法驱动核性能分别只比目前通用的基于特征的线性核和基于特征的多项式核下降 0.55% (88.51%-87.96%) 和 1.96% (89.92%-87.96%) 。在文献上，语义角色标注中核方法的性能上比基于特征的方法差很多[72, 100, 102] 。这是核方法研究中第一次在语义角色标注上与基于特征的方法有可比性。目前基于特征的方法探究的很广泛，很难再提升性能了，核方法对于进一步提高性能有很大的潜力。我们的研究证明了这种潜力，并在正确方向上迈出一大步，因为在下一段中我们的复合核性能比之前的最好的基于特征的和基于核的方法都好。

4、用多项式核对扁平特征进行二元组合是非常有用的，导致在准确度上提升了 1.41% (89.92%-88.51%) 。

为了适当地用句法结构信息和不同的扁平特征，我们引入一个复合核来合并句法驱动混合树核和基于特征的多项式核 ($d = 2$) ：

$$K_{comp} = \gamma K_{G-hybrid} + (1 - \gamma) K_{poly}, (0 \leq \gamma \leq 1)$$

其中 γ 调整为 0.3 。表 4-1 显示复合核准确度是 91.02% ，它统计上 ($p = 0.05$ 下 χ^2 测试) 明显好于多项式核 ($\gamma = 0$ ，准确度为 89.92%) 和句法驱动混合卷积树核 ($\gamma = 1$ ，准确度为 87.96%) 。这也说明这两种核是相互补充的。

表 4-1 语义角色分类的性能比较

Table 4-1 Performance Comparison of Semantic Role Classification

分类方法	准确率(%)
基线系统(标准卷积树核): 非句法驱动的混合卷积树核	85.21
仅加入近似节点匹配	86.27
仅加入近似结构匹配	87.12
句法驱动的卷积树核(包括近似节点和结构匹配)	87.96
基线系统(基于特征): 线性核	88.51
基线系统(基于特征): 多项式核($d = 2$)	89.92
句法驱动核与二次多项式核结合	91.02

表 4-2 比较了在单一分析树上运行的不同方法。[57] 在 CoNLL 2005 SRL 所有参赛系统中总排名第 5 而在只用一个 Charniak 分析器返回的分析树是最好的系统，我们用这个系统作为基准系统。表 4-2 说明：

1、合并了非句法驱动混合卷积树核与基于特征的多项式核的复合核优于多项式核，还明显 ($p = 0.05$ 下 χ^2 测试) 优于在 CoNLL 2005 SRL 共享任务中只用一个分析树时的最好系统。

2、在所有系统中，我们的复合核合并了一个句法驱动混合卷积树核与基于特征的多项式核，它的性能是最好的。它明显 ($p = 0.05$ 下 χ^2 测试) 优于使用非句法驱动的复合核，其 $F_{\beta=1}$ 值高出了 0.72% (78.13%-77.41%) ，也明显 ($p = 0.05$ 下 χ^2 测试) 优于 CoNLL 2005 SRL 共享任务中的最好系统 (基于一个句法分析树) ，其 $F_{\beta=1}$ 值高出了 1.67% (78.13%-76.46%) 。这进一步验证了句法驱动卷积树核对 SRL 有效。

3、对节点个数的限制的放宽：在简化的产生式中容许至少有一个子节点，这导致性能下降 0.35% (78.13%-77.78%) 。通过实验验证我们的假设：两节点的限制能使简化的产生式很好的保留最初产生式相关的语义。在 CoNLL-2005 训练集上的统计显示 73.59% 的产生式有 3 到 6 个子节点节点而 18.59% 的产生式有 2 个子节点节点，只有 4.6% 的产生式有一个子节点。这指出将节点限制放宽到 1 个节点使得相似子结构机制有太多的弹性，因此影响了性能。

4、相似节点匹配和相似子结构的机制都提升了性能(77.69% 和 77.51%

表 4-2 语义角色标注的性能比较

Table 4-2 Performance Comparison of Semantic Role Labeling

Methods	$F_{\beta=1}$ (%)
复合核：句法驱动卷积树核+ 基于特征的多项式核($d = 2$)	**78.13**
复合核 : 句法驱动卷积树核(放宽了在简化产生式中至少有两个子节点的限制，可以只有一个子节点) + 基于特征的多项式核	77.78
复合核 : 仅使用近似结构匹配	77.69
复合核 : 仅使用近似节点匹配	77.51
复合核 : 不使用句法驱动核	77.41
二次多项式核：	77.00

77.41%)，但不明显 ($p = 0.05$ 下 χ^2 测试) 。这可能是由于基于特征的多项式核使用大量的扁平特征从而降低了两个相似匹配机制的贡献。

图 4-4 举例说明在语义角色标注中句法驱动卷积树核有着非句法驱动所没有的优势。句法驱动核能够正确的标注 A1 而非句法驱动则不能。这很可能是由于：

1、产生式 “NP→DT ADJP JJR NN” 在训练数据中有非常低的频度 (只有 5 次) 。这可能使的非句法驱动核的失败。

2、原始产生式 “NP→DT ADJP JJR NN” 在相似子结构与相似节点匹配机制中与高频产生式“NP→DT JJ NN”(28,000 多次) 非常相似 (JJR 的变种 JJ 和除去当成可选节点的 ADJP，简化产生式为“NP→DT [ADJP] JJR NN”) 。这使得句法驱动能很好的工作。

最后，表 4-3 比较了两个复合核在训练中 (2.66G*8 CPU 和内存为 8G) 的时间。这显示为：

1、尽管句法驱动核在最坏的情况下的计算的时间复杂度为 $O(p\rho^2|N_1| \cdot |N_2|)$，但只增加很少的计算时间。这证实我们的动态规划算法的有效性。

2、在大的数据集中训练 SVM 分类器是非常消耗时的。

表 4-3 训练计算开销的比较

Table 4-3 Comparison of computational burden in training

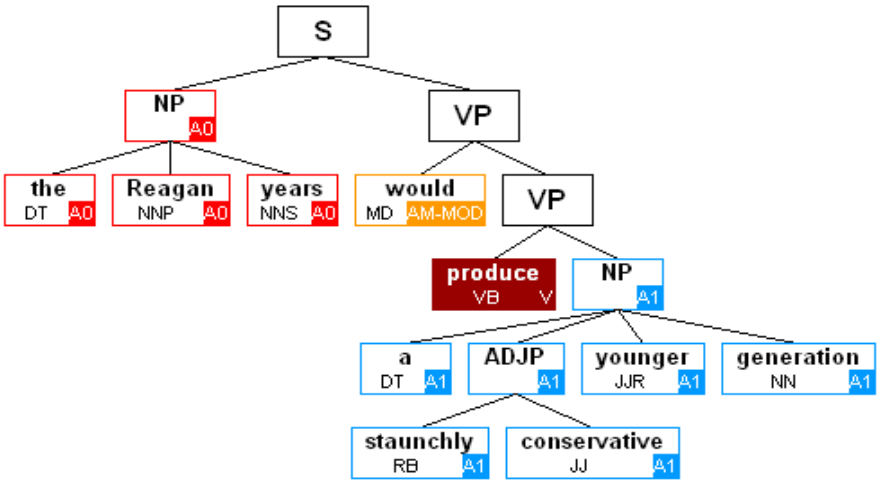
分类方法	训练时间	
	4 Sections	20 Sections
句法驱动的卷积树核	∼8 hours	∼7 days
非句法驱动的卷积树核	∼5 hours	∼6 days

4.5 本章小结

核方法在语音和语言处理中有广泛的研究，在许多自然语言处理问题中，研究如何利用结构化信息具有很大的潜力。在本章中，我们设计了一个句法驱动卷积树核来在核设计上为语义角色标注研究探究更多的语言学知识。实验结果证实句法驱动树核允许复合语言学知识的相似子结构和节点匹配，其性能在获取句法结构上超过了标准卷积核。

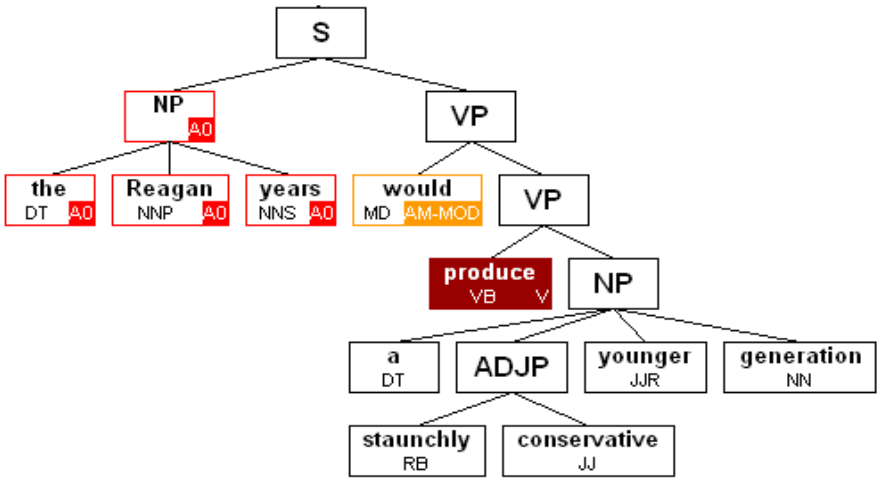
尽管在许多自然语言处理应用中树核达到了预期的效果，但它们在生成子结构时缺少对语言学的考虑。据我们所知，本研究在核设计中首次尝试结合语言学知识。

本章一个更大的研究目标是：句法树相似度的计算应该由语言学来获知。



a) 句法驱动混合卷积树核的标注结果

a) An SRL Result using the Grammar-driven Hybrid Convolution Tree Kernel



b) 非句法驱动混合卷积树核的标注结果

b) An SRL Result using the Non-grammar-driven Hybrid Convolution Tree Kernel

图 4-4 句法驱动与非句法驱动混合卷积树核的标注结果比较

Fig. 4-4 Comparison between the Results of using the Grammar-driven and Non-grammar-driven Hybrid Convolution Tree Kernel

我们使用核方法，对利用语言学知识来计算句法树相似度进行了初步研究。在设计句法驱动卷积树核中，仍然有许多方面值得在未来进行深入探索。

第 5 章　基于核方法的中文语义角色标注

5.1 引言

在前面的章节中，我们依次介绍了三种基于核的语义角色标注方法，分别是：多项式核，混合卷积树核，以及句法驱动的卷积树核。它们的有效性在英文语义角色标注问题上得到了验证。然而对于中文语义角色标注问题，基于核的方法是否可行还没有学者探讨过。因此，本章将集中将以上三种核方法在中文语义角色标注数据上进行实验。为此，我们首先介绍中文的语义角色标注语料库。接着构造了一个基于特征向量的中文语义角色标注基线系统，主要引入了若干中文相关的特征。最后，对多种方法进行了对比。实验同样证明了基于核方法的有效性。

5.2 中文语义角色标注语料库资源

与英文一样，中文同样具有谓词-论元结构[103]。同时进行中文语义角色标注，和其他基于有指导机器学习的自然语言处理任务一样，也需要语料资源的支持。目前，中文语义角色标注的研究主要使用三种资源：Chinese Proposition Bank (CPB)[104]，Chinese Nombank[105]，Chinese FrameNet[31]。

Chinese PropBank (CPB) 是宾夕法尼亚大学基于 Penn Chinese Treebank (PCT) 标注的汉语语义角色标注资源，在 Penn Chinese Treebank 句法分析树的对应句法成分中加入了语义角色信息。Penn Chinese Treebank 的标注数据主要来自新华新闻专线、Sinorama 新闻杂志和香港新闻。图 5-1 是 CPB 中一个句子的标注实例。CPB 的语料是以动词性质的谓语动词为核心，标注其语义角色的。其中，谓语动词有四类：状态动词 (VA)，系动词 (VC)，{有，没有，无} 作动词 (VE) 以及其他动词 (VV)。其他动词包括情态动词、可能性动词、行为动词等。不同类型的动词，其框架结构 (FrameSet) 不同，所带的语义角色个数和类型也不同。CPB 包含 20 多个语义角色，相同的语义角色对于不同目标动词有不同的语义含义。其中核心的语义角色为 Arg0-5 六种，Arg0 通常表示动作的施事，Arg1 通常表示动作的影响等等。其余的语义角色为附加语义角色，用前缀 ArgM 表示，后面跟一些附加标记 (Secondary Tags) 来表示这些

参数的语义类别，如 ArgM-LOC 表示地点，ArgM-TMP 表示时间等。具体标记以及含义如表 5-1 所示。

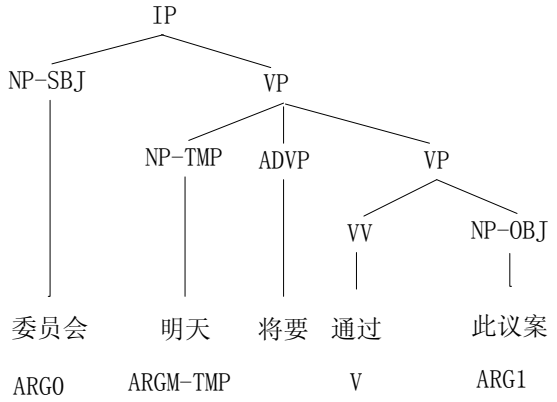


图 5-1 Chinese PropBank 中的一个标注实例

Fig. 5-1 An Instance Illustrating in the Chinese PropBank

表 5-1 Chinese PropBank 附加标记列表

Table 5-1 List of Secondary Tags in the Chinese PropBank

附加标记	标记的具体含义
ADV	Adverbials (附加的，默认标记)
BNE	Beneficiary (受益人)
CND	Condition (条件)
EXT	Extent (扩展)
FRQ	Frequency (频率)
LOC	Locative (地点)
MNR	Manner (方式)
PRP	Purpose or Reason (目的或原因)
TMP	Temporal (时间)

Chinese Nombank 把传统 English Proposition Bank 和 English Nombank 的标注框架，扩展到对中文名词性谓词的标注。Chinese Nombank 在 PCT 数据上加入了语义角色层的标注信息，与 CPB 一样，也标注了两类语义角色：核心语义角色和附加语义角色。Chinese NomBank 还标注了名词性谓词的框架，不过规模只是 CPB 中对应动词性谓词标注框架集的一少部分。Chinese

NomBank 中的角色位置有两类情况。第一类，角色在以名词性谓词为核心词 (Head word) 的名词短语中。第二类，当以名词性谓词为核心词的名词短语作支持动词 (Support Verb) 的主语时，允许语义角色在名词短语外。图 5-2 是 Chinese NomBank 的一个标注实例。

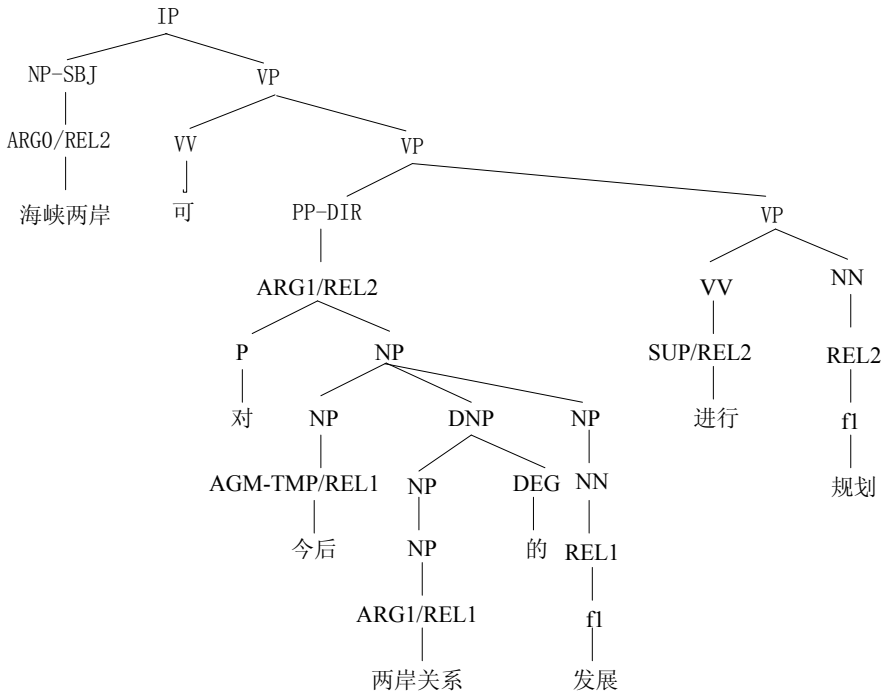


图 5-2 Chinese NomBank 中的一个标注实例

Fig. 5-2 An Instance Illustrating in the Chinese NomBank

山西大学构建的 Chinese FrameNet 是基于框架语义的，是一种 FrameNet 风格的中文词典。它描述了词汇单元以及参与者框架元素之间的关系，也包含了框架元素的详细句法信息。Chinese FrameNet 的架构和 English FrameNet 相似，并且有许多来自 English FrameNet 的翻译，但是作了一些相应的修改和创新，增加了相应语义角色的汉语名称。目前 Chinese FrameNet 已经有 130 多个汉语框架，并且还在不断增加，可惜的是 Chinese FrameNet 还没有在网络上共享。

在本文中，我们以 CPB 作为实验数据，主要考虑到其具有较大的数据规模以及相应的句法分析结果等优点。

5.3 标注步骤

为了提高系统召回率，避免剪枝阶段所造成的语义角色丢失，我们在中文语义角色标注系统中没有使用剪枝策略，因为如果某些句法成分被剪掉，则无论如何也不会将其标注为语义角色了。同时由于相同的原因，为了避免识别步骤对系统召回率造成的影响，我们将识别和分类两阶段合二为一，直接对与谓词相关的句法成分进行分类，属于语义角色的成分被分到对应类别，不属于任何角色的成分被赋予空类别。也就是说，我们对英文语义角色标注系统的四个步骤进行了一定的合并，变为了两个步骤，分别为识别和后处理。这也是基于 Chinese PropBank 数据量远没有英文的大的事实，才使得我们对步骤的合并成为了可能，否则分类器会有太多的输入，使得计算时间大幅增加。图 5-2 对两个数据的规模进行了对比，可以看出，PropBank 的数据量大概为 Chinese PropBank 的 4 倍左右。

表 5-2 PropBank 与 Chinese PropBank 数据规模的对比
Table 5-2 The Corpus Size Comparison between PropBank and Chinese PropBank

	PropBank	Chinese PropBank
Sentences	43,594	10,367
Tokens	1,039,565	269,129
Propositions	99,265	36,849
Arguments	262,281	104,007

5.4 中文语义角色标注特征集

特征一直是决定统计自然语言处理系统性能的重要因素。下面我们介绍一些基本特征并简要分析其有效性。这些特征部分来自英文语义角色标注系统，部分来自其它中文系统。接着着重介绍我们新加入的针对中文的特征。

- 句法成分相关特征

1. 短语类型

2. 中心词及其词性：在中心词提取中，我们使用 Sun 等人[85] 使用的核心词提取规则 (Head rules for Chinese)

3. 句法成分第一个词及其词性

4. 句法成分最后一个词及其词性

5. 句法分析树中左、右兄弟句法成分的短语类型

- 谓词相关特征

1. 谓词

2. 子类框架：谓词父节点及其子节点。如图 5-1 中，“通过”的子类框架是 VP→VV-NP-OBJ

3. 谓词的类别信息：目前的中文语义角色标注任务中还没有统一规范的动词分类，我们使用 Xue 等人[74] 的方法，从如下三个方面来对动词分类：

(a) 动词的框架数

(b) 动词每个框架的角色数

(c) 动词每个框架的句法交替 (Syntactic alternations)

通过统计，测试语料中总会有动词在训练语料中没有出现过，从训练数据中学习的模型就不能很好的对这些动词进行预测。CPB 中许多动词有相似的语义结构，比如动词“显现”和“显示”都带两个核心语义角色，主语指描述的实体，宾语指所描述实体的特性。这样，谓词类别信息就可以在动词稀疏的情况下正确预测角色类别。

- 谓词-句法成分关系特征

1. 路径

2. 部分路径

3. 路径长度

4. 位置

5. 距离

6. 句法成分的句法框架：句法框架特征包含谓词和围绕谓词的名词短语。句法框架特征中，谓词和这些名词短语作为核心，当前句法成分和它们相关联。如图 5-1 中：句法成分 NP-OBJ 的句法框架是 np_np_v_NP-OBJ

针对中文的特点，我们加入下列新的特征：

1. 句法成分的句法功能：CPB 手工标注的句法分析中，短语类型后缀有功能标记，比如 -IO 表示间接宾语，-OBJ 表示直接宾语，-SBJ 表示主语等。这些功能标记作为特征能够有效暗示语义角色的类型

2. 句法成分前一个词和后一个词

3. 从句层数：在 Xue 等人 [38] 有关 Penn Chinese Treebank 的句法标注文章中，对汉语句子提出了几种类型：带补语的子句 (CP) 、简单子句 (IP) 、不带

疑问次的疑问句 (IP-Q) 等。我们把句法成分到谓词的路径上经历的子句 IP 、CP 、IP-Q 等的个数作为特征

4. 句法成分到谓词的路径上出现的名词短语个数

5. 句法成分和谓词的相对位置：我们从三方面来考察他们的相对位置：它们是否兄弟节点关系，是否属于相同动词短语 (VP) 的儿子节点，是否属于相同子句 IP 或 CP 短语的儿子节点

6. 句法成分和谓词的共同最近父节点

7. 谓词的搭配模式：CPB 语料数据中，Arg2 大多情况在含有下面 5 种结构的句子中出现：介词-动词结构、使-动词结构、把-动词结构、被-动词结构、动词-数量词结构五种搭配结构。这种搭配模式能够提高对 Arg2 的预测效果，比如对于动词“修到”，Arg2 表示修建的地点，那么在语句“把公路修到山顶上”中“把-动词结构”就暗示句法成分“公路”属于角色 Arg2 。

5.5 基于核方法的中文语义角色标注

本节介绍如何将上面三章的基于核的学习方法在中文语义角色标注上进行应用。

首先是基于多项式的核，由于它是建立在特征向量之上的，因此需要首先根据中文的特点，构造适合中文的特征。在此我们使用上一节介绍的特征，但是由于各个特征之间往往是互相影响的，因此新加入的特征未必有效，需要首先进行特征选择工作，但这往往是一项繁琐且艰苦的工作。我们在 5.6.2 节给出了中文语义角色标注的特征选择结果，但是各种特征的选择和组合情况和特征的个数成指数关系，所以不可能获得最优解，因此在此使用的是一种贪心的方法，即对每个特征分别独立的考查。

然而，在使用混合卷积树核进行语义角色标注时，我们就不需要进行细致的特征构造和选择工作，只需要提取适当的部分句法分析子树即可。由于中文的句法分析树结构与英文的非常相似，所以我们几乎不需要将英文的程序进行任何改动，就可以将该方法移植到中文上。由此可见，基于卷积树核的方法，具有很好的语言可移植性，对于快速的开发语义角色标注系统，有很好的帮助。

最后是句法驱动的卷积树核，由于中文句法分析树的句法成分标记，包括短语类型和词性类型等，与英文的完全相同，因此我们也不需要对程序进行任何改动，只需要在中文训练语料库上自动提取可选节点，即可构造中文

的可选节点产生式集合，从而应用句法驱动的卷积树进行语义角色标注。

5.6 实验及讨论

本节首先介绍中文语义角色标注实验语料库的构造过程，然后通过实验结果，选择了合适的中文特征集构成基于多项式核的基线系统。接着使用混合卷积树核以及句法驱动的卷积树核，并融合了基线系统进行了对比实验。

5.6.1 实验设置

为了与 Xue 等人[74] 的工作进行对比，我们按照他们的做法，将 760 个文档的 CPB 语料中前 99 个 (从 chtb 001.fid 到chtb 099.fid) 作为测试数据，剩余 661 个(从 chtb 100.fid 到chtb 760.fid) 作为训练数据。

与英语相同，实用的语义角色标注系统必须构建在自动的句法分析之上，在此我们选用开源的 DBParser* 作为我们的自动短语结构句法分析器，DBParser (Daniel Bikel's Parser) 是 Daniel Bikel 设计实现的一个多语短语结构句法分析器[106] 。它提供多种已实现的统计分析模型，比如对 Michael Collins 的模拟，它也能很方便的扩展到多个领域和多种语言。目前的 DBParser 句法分析器通过 Java 实现，提供了英语、汉语和阿拉伯语的设置文件很相关资源，并且分析性能较高。

我们将基于手工标注短语结构句法的语义角色标注语料库称为 CPB (Chinese PropBank) 语料。而将基于 DBParser 的自动句法分析的语料资源称为 DB-CPB (DBParser based Chinese PropBank) 。为了进行实验，还需要进行以下三步工作：

1、构建自动句法分析资源，我们使用除了测试数据外的所有 Chinese Penn Treebank 作为句法分析的训练数据，训练一个汉语句法分析器。然后再对测试数据部分的 99 个文档进行自动的句法分析。需要注意的是，这里我们使用的分词和词性标注与 Xue[74] 的工作有所不同，我们使用的是由哈工大信息检索研究室开发的分词和词性标注系统 †，而 Xue 使用的是一体化分析方法，即将分词、词性标注以及句法分析工作融入到一个统一的模型之中。

2、在训练句法分析模型时，还需要删除 CPB 句法标注中的空节点 (-NONE-)，这是由于空节点是为了层次结构的完整而手工加入的，而自动分析

*http://www.cis.upenn.edu/∼dbikel/software.html#stat-parser

†http://ir.hit.edu.cn/demo/ltp

时无法得到。如图 5-3 所示的句法分析标注结果中：(-NONE- *-1) 和 (-NONE- *-2) 表示指代关系的索引，指代节点 (NR 张三) 。

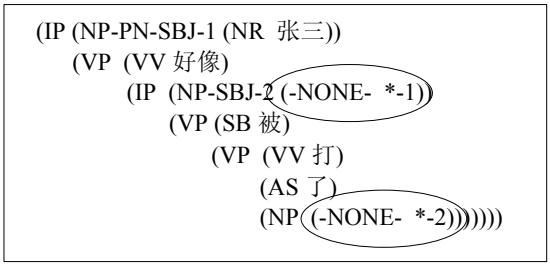


图 5-3 CPB 中的空标记

Fig. 5-3 Null Elements in the CPB

对空节点的处理规则为：

(1) 如果空节点有非空兄弟节点，则直接删除空节点

(2) 如果空节点没有非空兄弟节点，则删除空节点并删除其父节点。

由于句法分析器的限制，经过自动句法分析的测试数据中也不再含有空节点信息。

3、构建语义角色标注的资源。把训练数据中所有句子中的空节点删除，并将这些空节点对应的角色标记经过处理后的语料做训练语料。处理过程遵守如下三个规则：

(1) 如果空节点是一个完整的角色类型或者不是角色类型，删除对应角色标记。

(2) 如果空节点是角色的开始，则角色开始标记移到后一个词。

(3) 如果空节点是角色的结尾，则角色结尾标记移到前一个词。

根据空标记处理规则，处理测试数据中角色和分词结果的对齐，并加入自动句法分析得到句法结果得到测试数据。同时，我们保持 DB-CPB 中谓词和 CPB 中谓词一致。图 5-4 是其中一个句子的标注结果。

另外，为了验证我们新加入特征的性能，并找到最优的特征集，我们在此首先使用了最大熵分类器，在 CPB 语料库上进行实验，找到最优的特征集合，其主要目的是利用最大熵分类器训练速度较快的特点，以加快特征集的选择速度。在选择出最优的特征集后，我们使用前面介绍的基于核的分类器进行对比实验。

5.6.2 实验结果

首先，研究新加入的特征对系统的影响，表 5-3 给出了在使用基本特征的基础系统上分别加入扩展特征及组合特征对系统性能的影响，其中黑体表示加入扩展特征后，系统系能提高，并在显著提高的后面加入“*”。

从表 5-3 可以看出，加入句法成分后一个词、谓词和短语类型的组合、谓词类别信息和路径的组合都显著提高了系统的性能 $F_{\beta=1}$ 值。其它特征或特征组合加入后，除了少数特征和特征组合的加入使得性能降低，多数都使系统性能提高。

句法成分后一个词能够显著提高系统的性能，一方面是由于汉语语法的一个重要特点就是十分重视词序，词序不同表达的意思就不同；另一方面，句法成分后一个词作为上下文特征，能够反映当前句法成分的特定语境意义。短语类型是一个非常有效的特征，这是由于不同句法类型的短语总是趋向于充当不同的语义角色，而当给定句子中谓词时，这种特性就更加明显，所以谓词和短语类型的组合显著提高了系统的性能。比如例句“今年比去年同期增长九十点七亿美元”，对于动词“增长”，其后的数词短语往往趋向于做 Arg2。路径特征在谓词已知时非常有效，但是两者组合会导致数据稀疏，所以我们采用谓词类别信息和路径组合，显著提高了系统的性能。

接着我们在基础系统上加入了使性能提高的扩展特征和组合特征，构成了新系统。表 5-4 列出了基础系统和新系统的性能。从中可以看出：尽管每个特征单独加入后，系统性能的增加幅度不是很大，但这些特征全部加入后，系统的性能就有了明显的改进，增加了 1.55 个百分点。

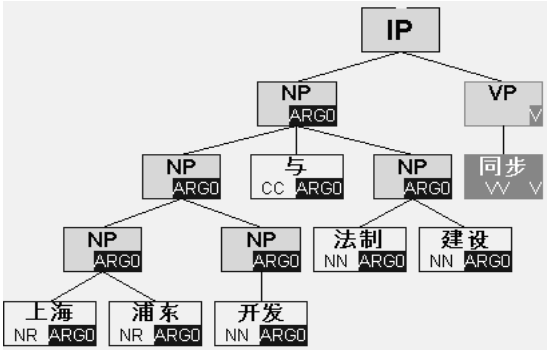


图 5-4 DB-CPB 中的一个标注实例

Fig. 5-4 An Instance Illustrating in the DB-CPB

表 5-3 新特征对系统性能的影响

Table 5-3 The Effect of New Features on the System Performance

特征	Precision (%)	Recall (%)	$F_{\beta=1}$ (%)
基础系统	90.94	88.62	89.76
逐个加入扩展特征			
+ 句法成分的功能	90.99	88.77	**89.87**
+ 句法成分前一个词	91.15	88.70	**89.91**
+ 句法成分后一个词	91.58	89.14	**90.34***
+ 句法成分前一个词的词性	90.97	88.71	**89.82**
+ 句法成分后一个词的词性	91.05	88.86	**89.94**
+ 从句层数	91.01	88.91	**89.95**
+ 谓词到句法成分的路径上名词短语个数	90.93	88.69	**89.80**
+ 句法成分和谓词的相对位置	90.97	88.81	**89.88**
+ 句法成分和谓词的共同最近父节点	90.87	88.68	89.76
+ 谓词的搭配模式	91.08	88.81	**89.93**
逐个加入组合特征			
+ 谓词：短语类型	91.64	89.19	**90.40***
+ 谓词：中心词	91.48	88.75	**90.09**
+ 谓词：中心词：中心词词性	91.46	88.71	**90.06**
+ 谓词：位置	91.27	88.79	**90.01**
+ 谓词：路径	91.44	88.98	**90.19**
+ 中心词：位置	90.98	88.44	89.69
+ 谓词类别信息：路径	91.58	89.06	**90.30***
+ 谓词类别信息：句法成分的句法框架	91.17	88.83	**89.98**
+ 短语类型：左兄弟成分的类型	90.84	88.63	89.72
+ 短语类型：右兄弟成分的类型	90.90	88.78	**89.83**
+ 谓词的搭配模式：位置	91.01	88.45	89.71
+ 句法成分的句法框架：短语类型	90.81	88.62	89.70
+ 谓词：句法成分的句法框架	91.22	88.87	**90.03**
+ 中心词: 中心词词性：路径	91.15	88.59	**89.85**
+ 短语类型：路径	91.09	88.75	**89.90**

从我们实验的结果可以看出，在汉语手工标注句法语料上性能能够达到91.31%，主要在于：

首先，动词类别信息能够有效提高系统性能。因为汉语中动词一词多义的现象比较少，在所有语料数据集中，共有 4,858 个动词。其中，只有 62 个动词有 3 个或 3 个以上的框架，如表 5-5 所示。这样对于大量只有 1 个框架的动词，其语义角色就有相对固定的句法形式，在手工标注的准确句法分析情况下，角色预测就比较容易。同时，汉语中形容词作谓语的情况很多，这样谓词的角色相对单一，句法实现也简单。在 CPB 语料中，谓词的词性有 VA，VV，VC，VE 四种，其中形容词性谓词 VA 占有很大比例，这样对应谓词的语义角色就很容易预测。

对表 5-5 进一步进行比较，可以看出框架数多的动词只是占了很少一部分比例。这部分动词对应的角色也很少，这样大部分语义角色是属于简单框架的动词。因而系统性能较高，这也是一个重要影响因素。

表 5-4 加入扩展特征和组合特征前后的系统性能比较

Table 5-4 The Performance Comparison after Adding the New Features

实验	Precision (%)	Recall (%)	$F_{\beta=1}$ (%)
基础系统	90.94	88.62	89.76
新系统	92.68	89.97	91.31

表 5-5 CPB 中动词框架个数统计

Table 5-5 The Number of Verb Frames in CPB

框架个数	动词个数
1	4,511
2	285
3	41
4	13
5	5
6	2
7	1
≥8	0

其次，Penn Chinese Treebank 使用了更加层次化的结构方式，在完全句法分析树中，使用了许多空标记 (-NONE-) 来表示深层的含义。

并且，在中文语义角色标注的训练语料数据中，有更多附加角色 ArgMs，相对少的而又难分辨的核心角色 Arg3，Arg4，如表 5-6 所示。这样使得角色的识别和分类更加容易。

表 5-6 训练集中主要角色的分布情况
Table 5-6 List of Main Roles in Training Data

角色类型	数量	所占比例 (%)
Arg0	23,239	34.36
Arg1	21,018	31.08
Arg2	2,428	3.59
Arg3	188	0.28
Arg4	26	0.04
ArgM-ADV	9,003	13.31
ArgM-MNR	1,264	1.87
ArgM-LOC	1,717	2.54
ArgM-TMP	4,334	6.41

通过对分类错误的语义角色进行分析，这些错误主要有如下几方面引起：首先，一般动词的主语 (Subject) 被标为 Arg0，宾语 (Object) 被标为 Arg1。但也有一些动词例外，比如“出现”。例如：这支新队伍以新面孔 **出现** 在世人面前。其中“这支新队伍”做主语却被标注为 Arg1。其次，占比例较高的角色 Arg2 的召回率较低。在汉语里中，Arg2，Arg3，Arg4 这几类角色非常灵活，对于不同的动词表示不同的含义，这种灵活性增加了分析的难度。下面是角色 Arg2 的例子：

(1) 他们都 **给** *我* 肯定的答复。
(2) 外商独资企业 **增加** 了 *百分之四点一二*，达八千四百八十四个。
(3) 中国对外贸易合作部 **派驻** *澳门* 的直属企业。

在上面的三个句子中，例句 (1) 中 Arg2 表示“给”的接受者；例句 (2) 中 Arg2 表示“增加”的数额；例句 (3) 中 Arg2 表示“派驻”的地点。

表 5-7 给出了基于 CPB 和 DB-CPB 的不同算法的对比实验，其中 PAF 和

MPAF 分别代表 PAF 核以及改进的 PAF 核系统；Hybrid 代表混合卷积树核系统；GTK (Grammar-driven Tree Kernel) 代表句法驱动的卷积树核系统；ME (Maximum Entropy) 代表最大熵系统；Poly 代表二次多项式核系统；Xue 代表 Xue 等人[74] 的系统；Comp 代表多项式核与句法驱动核的复合核系统。

从中我们可以得出以下结论：

1、DB-CPB 语料的系统和 CPB 语料系统相比，性能差近 30% 。主要影响因素有：

(1) CPB 句法分析中，句法节点类型有丰富的功能后缀，比如 NP-SBJ 、NP-OBJ 、PP-MNR 、LCP-TMP 等，这些功能后缀给角色标注提供了丰富的信息，非常有利于角色标注。而在 DBParser 自动句法里，这些后缀信息是无法得到的。

(2) CPB 句法为了句子层次结构完整而加入的空标记，这些标记也有助于进行语义角色标注，而自动句法的结果里这种层次上的完整化信息也是无法得到的。

(3) DBParser 句法分析的准确性还不够理想，会引入许多错误信息。

因此，语义角色标注是对自动句法分析的一个考验和挑战，同时高质量的自动句法分析结果也是提高角色标注性能的一个努力方向。

表 5-7 中文语义角色标注系统性能比较

Table 5-7 The Chinese PropBank Performance ($F_{\beta=1}$) Comparison

	PAF	MPAF	Hybrid	GTK	ME	Poly	Xue[74]	Comp
CPB (Gold)	84.43	84.77	85.85	86.13	91.01	91.13	91.3	**91.67**
DB-CPB (Auto)	58.83	59.21	60.12	60.75	64.56	64.79	61.3	**65.42**

2、在中文数据集上进一步验证了我们的混合卷积树核 (Hybrid) 性能要优于 PAF 核以及其改进版本 MPAF 。同时，如果在核设计的时候加入语言学知识，所构造的句法驱动的混合卷积树核 (GTK) 在单纯使用树核的方法中，会获得最好的性能。

3、在基于人工标注的句法分析上，基于多项式核 (Poly) 的方法虽然性能低于 Xue 等人[74] 的方法，但是还是要优于最大熵 (ME) 的方法。但是在基于自动句法分析时，无论是哪种基于特征的方法，性能都要比 Xue 等人的方法

高很多，这主要是由于我们采用的句法分析器，性能要高于 Xue 等人使用的一体化的句法分析器。

4、最终多项式核与基于句法驱动的卷积树核合并构成的复合核 (Comp) 在人工标注句法分析上，性能要优于 Xue 等人[74] 的方法，并在中文语义角色标注问题上获得了最好的性能。

不同方法在中文和英文数据上的性能趋势如图 5-5 所示。从中可以看出，CPB 语料库上的性能最高，这主要是因为 CPB 基于的是手工标注的句法分析结果，因此，语义角色标注的结果几乎不受错误句法分析的影响，所以准确率比较高。另外，同样是基于自动的句法分析结果，PropBank 的性能要高于 DB-CPB，这也主要是由于英文句法分析器的性能要高于中文的原因。通过各条曲线的变化趋势来看，不同方法在两种语言数据上的趋势是一致的，也就是说我们的方法在不同语言上具有较好的稳定性，从而也证明了我们的方法能够移植到不同的语言上。

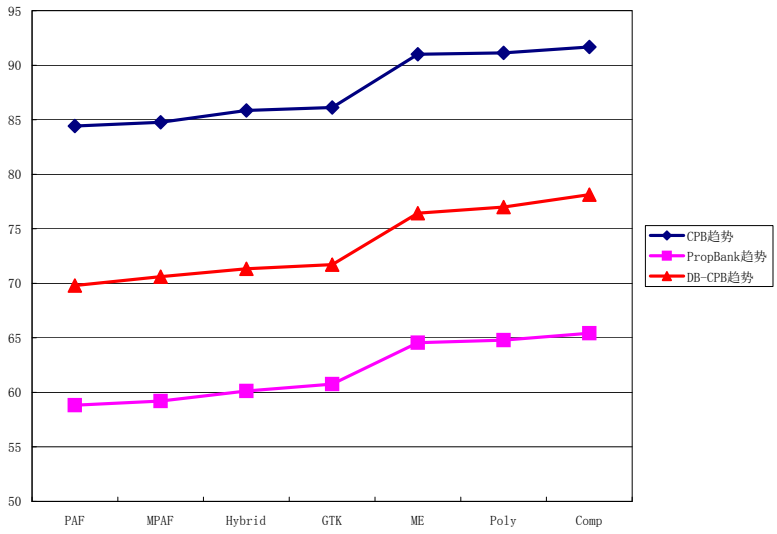


图 5-5 不同方法在中文和英文数据上的性能趋势

Fig. 5-5 The Trences of Different Methods on Chinese and English Corpus

5.7 本章小结

本章着重介绍了中文语义角色标注问题，我们首先针对中文的特点，构造了更适合的特征，并首次将基于核的方法在中文语义角色标注上进行了应

用。在 Chinese PropBank 上的实验结果证明，我们的提出的三种核方法，在中文上获得了与英文相同的趋势，也就是证明了每种方法的有效性。同时，无论基于人工标注还是自动的句法分析结果，我们最终的复合核都取得了最好的性能，从而也说明，在目前基于特征的方法性能难有提高的情况下，使用基于核的方法能够利用更多的特征，是进一步提高系统性能的一个很好的方向。

结 论

语义角色标注是自然语言处理领域的一个新兴的热门课题，它是浅层语义分析的一种实现方式，即不对整个句子进行详细的语义分析，而只是标注自然语言短语为给定谓词的语义角色，这个短语作为此谓词框架的一部分被赋予一定的语义含义。因此具有问题定义清晰，便于人工标注和评测等优点。从理论上说，语义角色标注是让机器理解自然语言的关键，从应用上讲，语义角色标注是机器翻译、信息抽取和自动问答的基石。

近年来，随着自然语言处理各种底层技术，如词性标注，句法分析等的日臻成熟，特别是机器学习技术的发展，使得进行较为精准的语义角色标注成为了可能。同时随着研究的逐步深入，人们越来越感觉到寻找合适的特征对系统性能的巨大帮助，然而，随着众多特征的加入，单纯通过特征的选取已经很难使系统性能有太大的提高了。为此必须寻找新的方法才有可能满足人们对系统性能无尽的需求。

基于核的方法是一种可行的解决方案，它的主要思想是将低维特征空间映射到高维特征空间，从而将在低维空间不容易区分的问题在高维空间加以解决。此映射的方法既可以是对特征进行组合，也可以是对特征进行分解。

本文使用并提出多种基于核的方法，逐步深入的应用于语义角色标注这一问题。最后针对中文的特点，提取合适的特征，并进一步验证了各种核方法的有效性。具体地讲：本论文的贡献主要表现在以下几个方面:

1、使用基于二次多项式核的语义角色标注方法。多项式核能够自动的对特征进行组合，使得该方法取得了较好的语义角色标注结果。实验结果表明，基于二次多项式核的语义角色标注系统是目前已知的最好的基于单句法分析器的语义角色标注系统之一；

2、提出了基于混合卷积树核的语义角色标注方法。卷积树核能够自动将较大的结构特征进行分解，并能够在多项式时间内进行核函数的计算，最终获得较好的性能。混合卷积树核融合多种树核，对语义角色标注中的不同种类的特征分别进行建模，最终获得优于标准卷积树核的性能。然后将混合卷积树核与多项式核进行融合，得到的复合核取得了比单独使用两种核都好的结果；

3、提出了基于句法驱动卷积树核的语义角色标注方法。针对标准卷积树

核要求两棵子树之间必须是精确匹配的，而不考虑结构近似，语义角色相同的情况。因此我们提出了新的句法驱动卷积树核。在核函数的设计过程中，融入了语言学知识，容许结构和节点的近似匹配，最终取得了较标准卷积树核更好的性能。最后同样与多项式核进行融合，也取得了更好的性能；

4、针对中文的特点，提出了更适用的新特征，并首次将核方法应用于中文语义角色标注，最终获得了目前最好的中文语义角色标注系统，从而也证明了我们方法对不同语言的有效性。

虽然本文的研究内容较好地提高了语义角色标注系统的性能，但是通过对语义角色标注问题较长时间的深入研究，我们认为对以下几个问题，还需要做进一步的研究：

1、领域自适应。为了更好的应用语义角色标注，必须解决领域自适应问题，也就是说解决测试语料和训练语料属于不同的领域，其性能下降较多的问题。这种现象在历次 CoNLL 评测中的表现尤为明显，Brown 语料上的评测结果均较之 WSJ 语料结果低 10% 左右。但是，我们不可能针对不同的领域，都标注大量的训练语料，所以领域的自适应成为一种必需。

2、句法和语义分析互动。目前人们普遍认为语义角色标注在很大程度上依赖句法分析的结果，为了尽量减小由于句法分析错误对语义角色标注造成的不利影响，人们曾经试图融合多个句法分析的结果，并取得了积极的进展。但是这种提高是单方向的，也就是说仅仅提高了语义角色标注的性能，可否利用语义角色标注帮助句法分析目前还鲜有人研究。通过在此方向上的探索可以使得句法和语义分析充分互动，互为补充。

3、语义依存分析。目前，语义角色标注结果的表示形式还存在一些不足，包括：一、语义角色的名称和种类不统一，命名过于随意。二、语义角色作为统一的整体，没有考虑其内部的结构。因此，我们认为语义依存分析是一种更合适的语义分析表示方式。它统一了依存句法分析和语义角色标注，使用数量不多的语义关系表示任意两个词之间的抽象语义关系。

参考文献

1 R. F. Simmons. Answering English Questions by Computer: A Survey. Commun. ACM. 1965, 8(1):53–70

2 D. L. Waltz. An English Language Question Answering System for a Large Relational Database. Commun. ACM. 1978, 21(7):526–539

3 R. C. Schank, R. P. Abelson. Scripts, Plans, Goals and Understanding: An Inquiry Into Human Knowledge Structures. Hillsdale, NJ: L. Erlbaum, 1977

4 M. G. Dyer. In-depth Understanding: A Computer Model of Integrated Processing for Narrative Comprehension. MIT Press, 1983

5 F. Jelinek. Statistical Methods for Speech Recognition. MIT Press, 1997

6 C. D. Manning, H. Schütze. Foundations of Statistical Natural Language Processing. Cambridge, Massachusetts: The MIT Press, 1999.

7 K. Mahesh. Natural Language Processing for the World Wide Web : Papers from the 1997 Spring Symposium. CA: AAAI Press, 1997

8 R. A. Baeza-Yates, B. A. Ribeiro-Neto. Modern Information Retrieval. ACM Press / Addison-Wesley, 1999.

9 K. Sparck Jones, P. Willett. Readings in Information Retrieval. Morgan Kaufmann Publishers Inc., 1997

10 H. T. Ng, J. Zelle. Corpus-based Approaches to Semantic Interpretation in Natural Language Processing. AI Magazine. 1997, 18(4):45–64

11 N. Ide, J. Veronis. Introduction to the Special Issue on Word Sense Disambiguation: The State of the Art. Computational Linguistics. 1998, 24(1):2–40

12 C. Cardie. Empirical Methods in Information Extraction. AI Magazine. 1997, 18(4):65–80.

13 D. Freitag. Toward General-purpose Learning for Information Extraction. C. Boitet, P. Whitelock, (Editors) Proceedings of the Thirty-Sixth Annual Meeting of the Association for Computational Linguistics and Seventeenth International Conference on Computational Linguistics. San Francisco, California, 1998:404–408.

14 D. M. Bikel, R. L. Schwartz, R. M. Weischedel. An Algorithm That Learns What's in a Name. Machine Learning. 1999, 34(1-3):211–231.

15 M. E. Califf, R. J. Mooney. Relational Learning of Pattern-match Rules for Information Extraction. Working Notes of AAAI Spring Symposium on Applying Machine Learning to Discourse Processing. Menlo Park, CA, 1998:6–11.

16 D. Gildea, D. Jurafsky. Automatic Labeling of Semantic Roles. Computational Linguistics. 2002, 28(3):245–288

17 M. Surdeanu, S. Harabagiu, J. Williams, et al. Using Predicate-argument Structures for Information Extraction. Proceedings of ACL 2003. 2003

18 于江德, 樊孝忠, 庞文博. 事件信息抽取中语义角色标注研究. 计算机科学. 2008, 35(3):155–157

19 S. Narayanan, S. Harbabagiu. Question Answering Based on Semantic Structures. Proceedings of Coling 2004. 2004

20 D. Shen, M. Lapata. Using Semantic Roles to Improve Question Answering. Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL). 2007:12–21.

21 J. Hajic, M. Cmejrek, B. Dorr, et al. Natural Language Generation in the Context of Machine Translation. Tech. rep., Center for Language and Speech Processing, Johns Hopkins University, Baltimore, 2002

22 M. W. Bilotti, P. Ogilvie, J. Callan, et al. Structured Retrieval for Question Answering. SIRIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. New York, NY, USA, 2007:351–358

23 Y. W. Y. L. A. S. Gabor Melli, Zhongmin Shi, F. Popowich. Description of Squash, the Sfu Question Answering Summary Handler for the Duc-2006 Summarization Task. Proceedings of the Document Understanding Conference 2006 (DUC-2006). 2006

24 C. F. Baker, C. J. Fillmore, J. B. Lowe. The Berkeley FrameNet Project. Proceedings of the ACL-Coling-1998. 1998:86–90.

25 M. Palmer, D. Gildea, P. Kingsbury. The Proposition Bank: An Annotated Corpus of Semantic Roles. Comput. Linguist. 2005, 31(1):71–106

26 A. Meyers, R. Reeves, C. Macleod, et al. The Nombank Project: An Interim Report. A. Meyers, (Editor) HLT-NAACL 2004 Workshop: Frontiers in Corpus Annotation. Boston, Massachusetts, USA, 2004:24–31

27 K. Erk, A. Kowalski, S. Pado, et al. Towards a Resource for Lexical Semantics: A Large German Corpus with Extensive Semantic Annotation. Proceedings of at ACL 2003. Sapporo, 2003.

28 E. Hajicova. Prague Dependency Treebank: From Analytic to Tectogrammatical Annotation. Proceedings of the First Workshop on Text, Speech, Dialogue. 1998:45–50

29 N. Xue, M. Palmer. Annotating the Propositions in the Penn Chinese Treebank. Q. Ma, F. Xia, (Editors) Proceedings of the Second SIGHAN Workshop on Chinese Language Processing. 2003:47–54.

30 N. Xue, F. Xia, F. dong Chiou, et al. The Penn Chinese Treebank: Phrase Structure Annotation of a Large Corpus. Nat. Lang. Eng. 2005, 11(2):207–238

31 L. You, K. Liu. Building Chinese Framenet Database. Conference on Natural Language Processing and Knowledge Engineering (IEEE NLP-KE). 2005:301–306

32 N. Xue, M. Palmer. Calibrating Features for Semantic Role Labeling. Proceedings of EMNLP 2004. 2004

33 V. Punyakanok, D. Roth, W.-t. Yih, et al. Semantic Role Labeling Via Integer Linear Programming Inference. Proceedings of Coling-2004. 2004:1346–1352.

34 T. Liu, W. Che, S. Li, et al. Semantic Role Labeling System Using Maximum Entropy Classifier. Proceedings of CoNLL-2005. 2005:189–192.

35 X. Carreras, L. Màrquez. Introduction to the CoNLL-2005 Shared Task: Semantic Role Labeling. Proceedings of CoNLL-2005. 2005:152–164.

36 V. Punyakanok, D. Roth, W. tau Yih. The Necessity of Syntactic Parsing for Semantic Role Labeling. Proceedings of IJCAI-2005. 2005:1117–1123.

37 X. Carreras, L. Màrquez. Introduction to the Conll-2004 Shared Task: Semantic Role Labeling. H. T. Ng, E. Riloff, (Editors) HLT-NAACL 2004 Workshop: Eighth Conference on Computational Natural Language Learning (CoNLL-2004). Boston, Massachusetts, USA, 2004:89–97

38 K. Hacioglu, S. Pradhan, W. Ward, et al. Semantic Role Labeling by Tagging Syntactic Chunks. H. T. Ng, E. Riloff, (Editors) HLT-NAACL 2004 Workshop: Eighth Conference on Computational Natural Language Learning (CoNLL-2004). Boston, Massachusetts, USA, 2004

39 O. Y. Kwong, B. K. Tsou. Semantic Role Tagging for Chinese at the Lexical Level. Proceedings of IJCNLP 2005. 2005

40 K. Hacioglu. Semantic Role Labeling Using Dependency Trees. Proceedings of Coling 2004. 2004

41 D. Klein. The Unsupervised Learning of Natural Language Structure. Ph.D. thesis, Stanford University. 2005

42 R. S. Swier, S. Stevenson. Unsupervised Semantic Role Labelling. D. Lin, D. Wu, (Editors) Proceedings of EMNLP 2004. Barcelona, Spain, 2004

43 陈耀东, 王挺, 陈火旺. 半监督学习和主动学习相结合的浅层语义分析. 中文信息学报. 2008, 22(2):70–75

44 D. Gildea. Probabilistic Models of Verb-argument Structure. Proceedings of the 19th international conference on Computational linguistics. 2002:1–7

45 C. A. Thompson, R. Levy, C. D. Manning. A Generative Model for Semantic Role Labeling. Proceedings of ECML-2003. 2003

46 R. O. Duda, P. E. Hart, D. G. Stork. Pattern Classification (2nd Edition). Wiley-Interscience, 2000

47 V. N. Vapnik. The Nature of Statistical Learning Theory. Springer-Verlag, Berlin, 1995

48 T. Joachims. Text Categorization with Support Vector Machines: Learning with many Relevant Features. C. Nédellec, C. Rouveirol, (Editors) Proceedings of ECML-98, 10th European Conference on Machine Learning. Chemnitz, DE, 1998:137–142.

49 S. Pradhan, K. Hacioglu, V. Krugler, et al. Support Vector Learning for Semantic Argument Classification. Machine Learning Journal. 2005

50 F. Rosenblatt. Principle of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Washington, D.C.: Spartan Books, 1962

51 Y. Freund, R. E. Schapire. Large Margin Classification Using the Perceptron Algorithm. Computational Learning Theory. 1998:209–217.

52 X. Carreras, L. Màrquez, G. Chrupała. Hierarchical Recognition of Propositional Arguments with Perceptrons. H. T. Ng, E. Riloff, (Editors) HLT-NAACL 2004 Workshop: Eighth Conference on Computational Natural Language Learning (CoNLL-2004). Boston, Massachusetts, USA, 2004

53 D. Roth. Learning to Resolve Natural Language Ambiguities: A Unified Approach. AAAI '98/IAAI '98: Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence. Menlo Park, CA, USA, 1998:806–813

54 P. Koomen, V. Punyakanok, D. Roth, et al. Generalized Inference with Multiple Semantic Role Labeling Systems. Proceedings of CoNLL-2005. Ann Arbor, Michigan, 2005

55 R. E. Schapire, Y. Singer. Improved Boosting Algorithms Using Confidence-rated Predictions. Mach. Learn. 1999, 37(3):297–336

56 L. Màrquez, P. Comas, J. Giménez, et al. Semantic Role Labeling as Sequential Tagging. Proceedings of CoNLL-2005. Ann Arbor, Michigan, 2005

57 M. Surdeanu, J. Turmo. Semantic Role Labeling Using Complete Syntactic Analysis. Proceedings of CoNLL-2005. Ann Arbor, Michigan, 2005

58 A. L. Berger, S. A. Della Pietra, V. J. Della Pietra. A Maximum Entropy Approach to Natural Language Processing. Computational Linguistics. 1996, 22(1):39–71.

59 M. Fleischman, N. Kwon, E. Hovy. Maximum Entropy Models for FrameNet Classification. M. Collins, M. Steedman, (Editors) Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing. 2003:49–56.

60 N. Kwon, M. Fleischman, E. Hovy. Senseval Automatic Labeling of Semantic Roles Using Maximum Entropy Models. R. Mihalcea, P. Edmonds, (Editors) Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text. Barcelona, Spain, 2004

61 J. R. Quinlan. Induction of Decision Trees. Mach. Learn. 1986, 1(1):81–106

62 J. Chen, O. Rambow. Use of Deep Linguistic Features for the Recognition and Labeling of Semantic Arguments. Proceedings of EMNLP-2003. Sapporo, Japan, 2003

63 S. Ponzetto, M. Strube. Semantic Role Labeling Using Lexical Statistical Information. Proceedings of CoNLL-2005. Ann Arbor, Michigan, 2005

64 L. Breiman. Random Forests. Machine Learning. 2001, 45(1):5–32

65 R. D. Nielsen, S. Pradhan. Mixing Weak Learners in Semantic Parsing. Proceedings of EMNLP-2004. 2004

66 G. Ngai, D. Wu, M. Carpuat, et al. Semantic Role Labeling with Boosting, Svms, Maximum Entropy, Snow, and Decision Lists. R. Mihalcea, P. Edmonds, (Editors) Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text. Barcelona, Spain, 2004

67 T.-H. Tsai, C.-W. Wu, Y.-C. Lin, et al. Exploiting Full Parsing Information to Label Semantic Roles Using an Ensemble of ME and SVM Via Integer Linear Programming. Proceedings of CoNLL-2005. Ann Arbor, Michigan, 2005

68 D. Haussler. Convolution Kernels on Discrete Structures. Tech. Rep. UCSC-CRL-99-10, 1999.

69 C. Watkins. Dynamic Alignment Kernels. Tech. Rep. CSD-TR-98-11, 1999.

70 M. Collins, N. Duffy. Convolution Kernels for Natural Language. Proceedings of NIPS-2001. 2001.

71 H. Lodhi, C. Saunders, J. Shawe-Taylor, et al. Text Classification Using String Kernels. Journal of Machine Learning Research. 2002, 2:419–444

72 A. Moschitti. A Study on Convolution Kernels for Shallow Statistic Parsing. Proceedings of ACL-2004. 2004:335–342

73 M. Collins. Head-driven Statistical Models for Natural Language Parsing. Ph.D. thesis, Pennsylvania University. 1999

74 N. Xue, M. Palmer. Automatic Semantic Role Labeling for Chinese Verbs. Proceedings of IJCAI-2005. 2005

75 A. McCallum, D. Freitag, F. Pereira. Maximum Entropy Markov Models for Information Extraction and Segmentation. Proceedings of 17th International Conference on Machine Learning. 2000:591–598.

76 P. Blunsom. Maximum Entropy Markov Models for Semantic Role Labelling. Australasian Language Technology Workshop 2004. 2004

77 J. Lafferty, A. McCallum, F. Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. Proceedings of 18th International Conference on Machine Learning. 2001:282–289.

78 T. Cohn, P. Blunsom. Semantic Role Labelling with Tree Conditional Random Fields. Proceedings of CoNLL-2005. Ann Arbor, Michigan, 2005

79 董静, 孙乐, 吕元华, 等. 基于线性链条件随机场模型的语义角色标注. 中文信息处理前沿进展—中国中文信息学会二十五周年学术会议论文集. 2006

80 于江德, 樊孝忠, 庞文博, 等. 基于条件随机场的语义角色标注. 东南大学学报. 2007, 23(3):361–364

81 Z. P. Jiang, J. Li, H. T. Ng. Semantic Argument Classification Exploiting Argument Interdependence. Proceedings of IJCAI-2005. 2005

82 A. Haghighi, K. Toutanova, C. Manning. A Joint Model for Semantic Role Labeling. Proceedings of CoNLL-2005. Ann Arbor, Michigan, 2005

83 M. Surdeanu, L. Màrquez, X. Carreras, et al. Combination Strategies for Semantic Role Labeling. JAIR. 2007, 29:105–151

84 S. Pradhan, K. Hacioglu, W. Ward, et al. Semantic Role Chunking Combining Complementary Syntactic Views. Proceedings of CoNLL-2005. Ann Arbor, Michigan, 2005

85 H. Sun, D. Jurafsky. Shallow Semantic Parsing of Chinese. Proceedings of the HLT/NAACL 2004. 2004

86 M. Surdeanu, R. Johansson, A. Meyers, et al. The Conll 2008 Shared Task on Joint Parsing of Syntactic and Semantic Dependencies. CoNLL 2008: Proceedings of the Twelfth Conference on Computational Natural Language Learning. Manchester, England, 2008:159–177.

87 丁金涛, 周国栋, 王红玲, 等. 语义角色标注中有效的识别论元算法研究. 计算机工程与应用，. 2008, 44(18):153–156

88 V. N. Vapnik. Statistical Learning Theory. Wiley, 1998

89 R. Rifkin, A. Klautau. In Defense of One-vs-all Classification. J. Mach. Learn. Res. 2004, 5:101–141

90 M. Porter. An Algorithm for Suffix Stripping. Program. 1980, 14(3)

91 E. Charniak. A Maximum-entropy-inspired Parser. Proceedings of the first conference on North American chapter of the Association for Computational Linguistics. San Francisco, CA, USA, 2000:132–139

92 H. L. Chieu, H. T. Ng. Named Entity Recognition with a Maximum Entropy Approach. Proceedings of CoNLL-2003. 2003:160–163

93 J. Giménez, L. Màrquez. Fast and Accurate Part-of-speech Tagging: The Svm Approach Revisited. Proceedings of RANLP-2003. 2003

94 X. Carreras, L. Màrquez. Phrase Recognition by Filtering and Ranking with Perceptrons. RANLP-2003. 2003

95 T. Joachims. Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms. Norwell, MA, USA: Kluwer Academic Publishers, 2002

96 S. F. Chen, R. Rosenfeld. A Gaussian Prior for Smoothing Maximum Entropy Models. Tech. Rep. CMU-CS-99-108, 1999

97 T. Joachims, N. Cristianini, J. Shawe-Taylor. Composite Kernels for Hypertext Categorisation. Proceedings of ICML-2001. 2001:250–257

98 A. Moschitti, D. Pighin, R. Basili. Tree Kernel Engineering in Semantic Role Labeling Systems. Proceedings of the Workshop on Learning Structured Information for Natural Language Applications, Eleventh International Conference on European Association for Computational Linguistics. Trento, Italy, 2006:49–56

99 M. Zhang, J. Zhang, J. Su. Exploring Syntactic Features for Relation Extraction Using a Convolution Tree Kernel. Proceedings of the Human Language Technology Conference of the NAACL, Main Conference. New York City, USA, 2006

100 A. Moschitti. Efficient Convolution Kernels for Dependency and Constituent Syntactic Trees. Proceedings of the 17th European Conference on Machine Learning. Berlin, Germany, 2006

101 F. C. N. Pereira, M. D. Riley. Speech Recognition by Composition of Weighted Finite Automata. 1996.

102 W. Che, M. Zhang, T. Liu, et al. A Hybrid Convolution Tree Kernel for Semantic Role Labeling. Proceedings of the COLING/ACL 2006. Sydney, Australia, 2006

103 袁毓林. 论元角色的层级关系和语义特征. 世界汉语教学. 2002:10–22

104 N. Xue, M. Palmer. Annotating the Propositions in the Penn Chinese Treebank. Proceedings of the Second SIGHAN Workshop on Chinese Language Processing. Sapporo, Japan, 2003

105 N. Xue. Annotating the Predicate-argument Structure of Chinese Nominalizations. Proceedings of the LREC 2006. Genoa, Italy, 2006:1382–1387

106 D. M. Bikel. On the Parameter Space of Generative Lexicalized Statistical Parsing Models. Ph.D. thesis. 2004

攻读博士学位期间所发表的论文

1 Wanxiang Che, Min Zhang, Ai Ti Aw, Chew Lim Tan, Ting Liu, and Sheng Li. Using a Hybrid Convolution Tree Kernel for Semantic Role Labeling. ACM Transactions on Asian Language Information Processing. 2008, 7(4):1-23. (EI Indexed)

2 Min Zhang, Wanxiang Che, Guodong Zhou, Ai Ti Aw, Chew Lim Tan, Ting Liu, and Sheng Li. Semantic Role Labeling Using a Grammar-Driven Convolution Tree Kernel. IEEE Transactions on Audio, Speech, and Language Processing. 2008, 16(7):1315-1329 (SCI Impact Factor: 0.8)

3 Wanxiang Che, Zhenghua Li, Yuxuan Hu, Yongqiang Li, Bing Qin, Ting Liu, and Sheng Li. A Cascaded Syntactic and Semantic Dependency Parsing System. In Proceedings of the Twelfth Conference on Computational Natural Language Learning (CoNLL-2008). 2008, pp238-242

4 Wanxiang Che, Min Zhang, Ai Ti Aw, Chew Lim Tan, Ting Liu, Sheng Li. Fast Computing Grammar-driven Convolution Tree Kernel for Semantic Role Labeling. In Proceedings of The Third International Joint Conference on Natural Language (IJCNLP-2008). 2008, pp781-786

5 Min Zhang, Wanxiang Che, Ai Ti Aw, Chew Lim Tan, Ting Liu, and Sheng Li. A Grammar-driven Convolution Tree Kernel for Semantic Role Classification. Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL-2007). 2007, pp200-207

6 Wanxiang Che, Min Zhang, Ting Liu, and Sheng Li. A Hybrid Convolution Tree Kernel for Semantic Role Labeling. In Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (Coling/ACL-2006). 2006, pp73-80

7 Ting Liu, Wanxiang Che, Sheng Li, Yuxuan Hu, and Huaijun Liu. Semantic role labeling system using maximum entropy classifier. In Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005). 2005, pp189-192

8 刘挺,车万翔,李生. 基于最大熵分类器的语义角色标注. 软件学报. 2007,

18(3), pp565-573 (EI Indexed)

9 刘怀军, 车万翔, 刘挺. 中文语义角色标注的特征工程. 中文信息学报. 2007, 21(2):79-85

10 车万翔, 刘挺, 李生. 浅层语义分析. 中国中文信息学会成立25周年学术年会. 2006, pp161-171

11 车万翔, 刘挺, 李生. 实体关系自动抽取. 中文信息学报. 2005, 19(2):1-6

12 车万翔, 刘挺, 秦兵, 李生. 基于改进编辑距离的中文相似句子检索. 高技术通讯. 2004,14(7):15-20 (EI Indexed)

哈尔滨工业大学博士学位论文原创性声明

本人郑重声明：此处所提交的博士学位论文《基于核方法的语义角色标注研究》，是本人在导师指导下，在哈尔滨工业大学攻读博士学位期间独立进行研究工作所取得的成果。据本人所知，论文中除已注明部分外不包含他人已发表或撰写过的研究成果。对本文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明。本声明的法律结果将完全由本人承担。

作者签字：　　日期:　　年　　月　　日

哈尔滨工业大学博士学位论文使用授权书

《基于核方法的语义角色标注研究》系本人在哈尔滨工业大学攻读博士学位期间在导师指导下完成的博士学位论文。本论文的研究成果归哈尔滨工业大学所有，本论文的研究内容不得以其它单位的名义发表。本人完全了解哈尔滨工业大学关于保存、使用学位论文的规定，同意学校保留并向有关部门送交论文的复印件和电子版本，允许论文被查阅和借阅，同意学校将论文加入《中国优秀博硕士学位论文全文数据库》和编入《中国知识资源总库》。本人授权哈尔滨工业大学，可以采用影印、缩印或其他复制手段保存论文，可以公布论文的全部或部分内容。

作者签名：　　日期:　　年　　月　　日

导师签名：　　日期:　　年　　月　　日

哈尔滨工业大学博士学位涉密论文管理

根据《哈尔滨工业大学关于国家秘密载体保密管理的规定》，毕业论文答辩必须由导师进行保密初审，外寄论文由科研处复审。涉密毕业论文，由学生按学校规定的统一程序在导师指导下填报密级和保密期限。

本学位论文属于　　保密□，在　　年解密后适用本授权书。

不保密□。

（请在以上相应方框内打“√”）

作者签名：　　日期:　　年　　月　　日

导师签名：　　日期:　　年　　月　　日

致　谢

六年的快乐、痛苦和兴奋的读博时光即将过去，值此论文完成之际，谨向所有关心、帮助过我的老师、同学、亲友表示衷心的感谢！

在我尊敬的导师李生教授的悉心指导和帮助下，我最终完成了博士学位论文。我的每一点工作与成绩都与他的辛勤汗水和培养分不开。李老师开阔的视野、敏捷的思维、渊博的学识无不是我学习的榜样！李老师的指导使我掌握了通用的科研方法，树立了宏伟的学术目标和人生目标，这些宝贵的财富使我终生受益！

更要感谢我的副导师刘挺教授，我与刘老师共事 8 年，亦师亦友。在我所有的工作中，都凝聚着他的心血和汗水。刘老师严谨的治学态度，科学的研究方法，渊博的学识、活跃的思维、创新的精神，永远是我学习的目标！刘老师还教会了我许多做人做事的道理，尤其是他能够不断地指出我在为人处世方面存在的诸多问题，让我在学会做研究的同时，也学会了做人！

感谢我在实习期间的导师，新加坡信息通讯研究所的张民研究员，微软亚洲研究院的李航研究员，周明研究员以及IBM中国研究中心的苏忠研究员。他们在我实习期间，对我在科研、学术和生活上给予了的一贯的指导和支持！

感谢本论文责任专家赵铁军教授对论文一丝不苟的审阅，还要感谢各位答辩委员会的专家以及匿名的评审专家对本论文提出的宝贵意见和改进建议。

感谢信息检索研究室的秦兵教授、张宇副教授多年来在学习和生活上的帮助和鼓励！特别感谢语言分析组的历届全体成员，他们不但给予本论文提供了强有力的支持，更是我研究上的好伙伴和坚强后盾。还要感谢研究室的历届毕业生、以及各位在校同学，他们共同营造了一个团结、温馨、积极、进取的氛围，使我在这里愉快地度过了七年美好的时光。谢谢你们，我亲爱的师兄弟姐妹们！

我深深的感谢我的父母，没有他们对我一贯的支持、理解、鞭策、鼓励和帮助，我不会有决心和信心拿到博士学位。他们积极乐观和热爱生活的态度对我影响至深。今后我将竭尽我的所能，回报这份我永远也还不清的亲情！

最后，我要向我单纯善良、温柔体贴的新婚妻子赵妍妍博士表示深深的谢意！你放弃了硕士毕业后可能获得的优厚待遇，毅然的伴我共同完成博士

学业而无怨无悔！是你始终给予我最直接的鼓励和精神上的安慰！我的每篇论文都是你牺牲了自己的时间来帮我修改的成果！

时光如水，日月如梭，六年的时光犹如人生旅途划过的一颗璀璨靓丽的流星。有太多的人和事值得回忆，也有太多的经验和教训值得我总结和汲取！

最后，向所有直接或间接帮助过我的老师、同学、同事、工作人员，以及本文的评阅老师表示感谢，谢谢您们！

个人简历

学习经历

1 2002 年 9 月–至今 哈尔滨工业大学计算机科学与技术学院 攻读工学博士学位

2 2002 年 7 月 哈尔滨工业大学计算机科学与技术学院 获工学学士学位

工作经历

1 2004 年 7 月–至今哈尔滨工业大学计算机科学与技术学院 教师

2 2007 年 1 月–2007 年 6 月新加坡信息通讯研究所 (I^2R) 访问学生

3 2004 年 3 月–2004 年 6 月IBM中国研究中心 访问学生

4 2002 年 4 月–2002 年 7 月微软亚洲研究院 访问学生

主要科研工作及成果

1 2009 年 1 月–2011 年 12 月 汉语依存句法分析若干关键技术研究，国家自然科学基金面上项目 （编号60803093）

2 2007 年 1 月–2009 年 12 月 汉语语义角色标注方法研究，国家自然科学基金面上项目 （编号60675034）

3 2006 年 1 月–2008 年 12 月 基于等价伪词的汉语全文无指导词义消歧技术研究，国家自然科学基金面上项目 （编号60575042）

4 2007 年 1 月–2008 年 12 月 基于XML的分层交互式中文处理开放平台，国家863探索类项目 （编号2006AA01Z145）

5 2008 年国际CoNLL句法分析核语义角色标注联合评测，共19加单位参赛，获第 2 名

6 2007 年国际SemEval，获词义消歧评测第 1 名

7 2005 年国际CoNLL语义角色标注评测，共19加单位参赛，获第 6 名

8 2003 年863计划中文与接口技术评测，获自动文摘第 1 名

学术论文

1 Wanxiang Che, Min Zhang, Ai Ti Aw, Chew Lim Tan, Ting Liu, and Sheng Li. Using a Hybrid Convolution Tree Kernel for Semantic Role Labeling. ACM Transactions on Asian Language Information Processing. 2008, 7(4):1-23. (EI Indexed)

2 Min Zhang, Wanxiang Che, Guodong Zhou, Ai Ti Aw, Chew Lim Tan, Ting Liu, and Sheng Li. Semantic Role Labeling Using a Grammar-Driven Convolution Tree Kernel. IEEE Transactions on Audio, Speech, and Language Processing. 2008, 16(7):1315-1329 (SCI Impact Factor: 0.8)

3 Wanxiang Che, Zhenghua Li, Yuxuan Hu, Yongqiang Li, Bing Qin, Ting Liu, and Sheng Li. A Cascaded Syntactic and Semantic Dependency Parsing System. In Proceedings of the Twelfth Conference on Computational Natural Language Learning (CoNLL-2008). 2008, pp238-242

4 Wanxiang Che, Min Zhang, Ai Ti Aw, Chew Lim Tan, Ting Liu, Sheng Li. Fast Computing Grammar-driven Convolution Tree Kernel for Semantic Role Labeling. In Proceedings of The Third International Joint Conference on Natural Language (IJCNLP-2008). 2008, pp781-786

5 Min Zhang, Wanxiang Che, Ai Ti Aw, Chew Lim Tan, Ting Liu, and Sheng Li. A Grammar-driven Convolution Tree Kernel for Semantic Role Classification. Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL-2007). 2007, pp200-207

6 Wanxiang Che, Min Zhang, Ting Liu, and Sheng Li. A Hybrid Convolution Tree Kernel for Semantic Role Labeling. In Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (Coling/ACL-2006). 2006, pp73-80

7 Ting Liu, Wanxiang Che, Sheng Li, Yuxuan Hu, and Huaijun Liu. Semantic role labeling system using maximum entropy classifier. In Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005). 2005, pp189-192

8 刘挺,车万翔,李生. 基于最大熵分类器的语义角色标注. 软件学报. 2007, 18(3), pp565-573 (EI Indexed)

9 刘怀军, 车万翔, 刘挺. 中文语义角色标注的特征工程. 中文信息学报. 2007, 21(2):79-85

10 车万翔, 刘挺, 李生. 浅层语义分析. 中国中文信息学会成立25周年学术年会. 2006, pp161-171

11 车万翔, 刘挺, 李生. 实体关系自动抽取. 中文信息学报. 2005, 19(2):1-6

12 车万翔, 刘挺, 秦兵, 李生. 基于改进编辑距离的中文相似句子检索. 高技术通讯. 2004,14(7):15-20 (EI Indexed)