基于粗糙集方法的共指消解*

贾修一，张亚兵，陈家骏，商 琳
南京大学 软件新技术国家重点实验室，江苏 南京 210093
南京大学 计算机科学与技术系，江苏 南京 210093
E-mail: {jiaxy, zhangyb}@nlp.nju.edu.cn; {chenjj, shanglin}@nju.edu.cn

摘 要：选择合适的特征是共指消解任务中一个重要的组成部分。特征不是越多越好，反映本质的特征很重要；对于不同种类的语料，一个公共的特征集往往难以适应，为了提高特征对语料的针对性，对不同的语料应选择不同的特征。本文基于上述观点，采用粗糙集理论中的属性约简方法来解决共指消解的特征选择问题，它一方面能解决特征冗余问题，另一方面可以实现针对不同语料选择具有适应性的特征。论文在特征选择之后，利用基于粗糙集的LEM2 规则提取算法学习规则构建分类器进行共指消解。在 ACE-2003 语料库上的实验说明了粗糙集方法对共指消解任务的有效性。

关键词：共指消解；粗糙集理论；特征选择；规则提取

Rough Set Approach to Coreference Resolution

Jia Xiuyi, Zhang Yabing, Chen Jiajun, Shang Lin
State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093
Department of Computer Science and Technology, Nanjing University, Nanjing 210093
E-mail: {jiaxy, zhangyb}@nlp.nju.edu.cn; {chenjj, shanglin}@nju.edu.cn

Abstract: Selecting a proper set of features is more important than adopting more and more features in coreference resolution. For different corpus, it is not a good idea to apply a same feature set. The feature selection procedure should be considered when learning from each given corpora. This paper applies rough set approach to coreference resolution, in which the attribute reduction in rough set theory is used to remove redundant features for each corpora first, and then classification rules are induced from these corpus to provide a mention-pair based solution for the coreference resolution problem. Experiments on ACE-2003 corpus show the effectiveness of the rough set approach to mention-pair based solutions.

Keywords: coreference resolution; rough set; feature selection; rule induction

1 引言

共指消解是指将多个命名实体指向现实世界中的同一实体，其目标是识别出文档中所有存在的共指关系[1]。共指消解在自然语言处理任务中有着广泛的应用，因此作为一个研究的热点，很多学者从不同方面对其进行了研究：构建新的消解模型[2]，寻找新的特征[3]和应用新的机器学习算法[4]等。

Soon et al.提出了 mention-pair 模型[5]，将共指消解问题转换成为一个分类任务；Yang et al.考虑 mention 和 entity 两个层面提出了 entity-mention 模型[2]，该模型和 mention-pair 相比，本质上是增加了在 entity 层面的特征。在选取特征方面，Soon et al.在他们的系统中首先应用了 12 个特征，Ng 和 Cardie 他们将特征扩充到了 53 个[3]，而 Bengtson 和 Roth 在他们的工作中发现只需要选取一些比较好的特征就能够取得更好的性能[1]。

特征不是越多就越好。特征太多还可能带来数据稀疏问题和过拟合问题等。在文献[1]中，他们在人工挑选的一些较好的特征上用基本的 mention-pair 模型进行学习，得到了比 Culotta 等应用一个复杂的模型更好的结果，在 B-cubed 的 F 值上提高了 1.5 个百分点以上。在共指消解任务中，目前有很多学者在寻找合适的特征来表示问题上做了许多工作，这些特征都独立于所要学习的语

* 本文承国家 973 课题资助项目（2010CB327903）；江苏省自然科学基金项目（BK2009233）资助。

料，也就是说特征的选取是由专家给定的，与语料无关，这样在一定程度上虽能够保证选取特征的泛化性，但是由于是手工选取，从而具有一定的随意性。另外，对于不同种类的语料，一个公共的特征集往往难以适应，为了提高特征对语料的针对性，对不同的语料应选择不同的特征。

在机器学习领域，有许多自动的特征选择方法[6]，而粗糙集理论[7]中的属性约简就是其中一种特征选择的方法。和其他方法相比，属性约简被形式化定义为寻找保持分类能力不变的最小属性（特征）集合，该集合不包含任何冗余特征。属性约简在保持原有特征语义不变的前提下，去除了冗余特征，使得选择的特征更具有针对性。在选择合适的特征以后，一个好的机器学习算法对消解性能的影响也非常重要。很多算法以规则的形式来表达学习到的知识，而规则提取又是粗糙集理论的另一个重要应用。基于此，本文应用粗糙集方法在共指消解任务上，采用基本的 mention-pair 模型，先通过属性约简，去除冗余的特征，再利用学习到的规则进行分类，判断两个命名实体是否共指。

本文采用粗糙集理论中的属性约简方法来解决共指消解的特征选择问题，它一方面能解决特征冗余问题，另一方面可以实现针对不同语料选择具有适应性的特征。论文在特征选择之后，利用基于粗糙集理论的规则提取算法学习规则作为分类器构建了共指消解系统。我们在 ACE-2003 语料上的实验说明了粗糙集方法在共指消解上的有效性，经过特征选择后的消解系统在 MUC 和 B-cubed 的 F 值上比不经过特征选择的系统能提高 2%以上。

文章首先介绍粗糙集方法的基本概念，其次通过不同的实验设置，对比了特征选择前后各个语料上的消解性能变化，表明了特征选择的必要性，最后通过和现有的一些方法的对比分析，说明了我们的方法的有效性。

2 粗糙集理论介绍

粗糙集理论作为一门处理不精确数据的数学方法，已被广泛应用到机器学习和数据挖掘等领域。在粗糙集理论中，知识的表示是通过决策表来表示的。一个决策表定义为一个四元组：$K=(U,At,\{V_a \mid a\in At\},\{I_a \mid a\in At\})$。在决策表中 U 是样例的论域，在共指消解任务中，每个样例就是一个命名实体对。$At=C\cup D$，C 是描述命名实体对的条件特征集合，如 *gender* 表示该样例是否具有相同的性别等。D 是决策特征，也就是类标号，1 表示共指，0 表示不共指。V_a 是特征 a 的可能的取值的集合。样例 x 在特征 a 上的取值用 $I_a(x)$ 来表示。

粗糙集理论中，属性约简被定义为寻找保持分类能力不变的最小属性集合，而分类能力不变是指约简前后整个决策表的正区域不发生变化。约简具有不唯一性，也就是说一个决策表的约简可能不止一个。在本文中我们采用最常用的属性约简算法：基于可辨识矩阵的属性约简[8]。

从决策表中提取规则是粗糙集理论的另一个重要的应用。LEM2 就是其中一种最常用的规则提取算法[9]。LEM2 算法的基本原理简单介绍如下：

规则的每个部分都是特征值对，如 *gender*=1 就表示是一个特征值对。对于给定样例 x，令 T 是一个条件特征值对的集合，$I_D(x)=w$ 是决策特征值对，则有 $T\rightarrow\{I_D(x)=w\}$ 当且仅当 $[T]\subseteq[I_D(x)=w]$ 和 $[T]\neq\varnothing$，其中 $[T]$ 表示满足条件特征值对 T 的样例集合。LEM2 的目的就是用最少的分类规则来覆盖所有的样例，而每条规则都由最少的特征值对来组成，更多的细节可以参阅文献[9]。和其他规则提取算法相比，LEM2 是一种局部覆盖算法[10]，学习到的规则集中每条规则都有较高的精确度，而通过学习更多的规则，也相应地提高了规则集的支持度。由于软件 RSES2[1]很好地集成了粗糙集理论中的属性约简算法和 LEM2 规则提取算法，本文将该系统应用到共指消解任务中去。

[1] http://logic.mimuw.edu.pl/~rses/.

3 粗糙集方法用于共指消解

本文采用粗糙集理论中的属性约简方法来解决共指消解的特征选择问题，它一方面能去除了冗余特征，另一方面可以针对不同语料选择具有适应性的特征。在特征选择之后，利用基于粗糙集理论的规则提取算法学习分类规则，基于此作为分类器构建了共指消解系统。

在本文中，我们采用的是基于 mention-pair 的共指消解模型，任务被刻画为二类分类问题。语料训练集以决策表形式存储，表中的元素的值代表每个命名实体对在相应特征上的取值。我们采用文献[5]中描述的 12 个特征，这很容易建立相应的可辨识矩阵和得到可辨识函数。

在去除冗余特征后，使用 LEM2 算法学习分类规则，该规则是一类典型的分类规则，一条基本的规则举例如下：(*appositive*=0)&(*alias*=0)&…&(*semanticClass*=1)& (*pair_definite*=2) ⇒ (*link*=0)。这种规则形式既利于用户理解，又易于在学习到的规则集上构建分类器。

我们把命名实体对表示为特征值对的形式，为了进行和其他算法进行对比，采用的是 Soon et al. 采用的 12 个特征来进行学习。由于这 12 个特征中包含描述单个命名实体的特征，我们将其转化成命名实体对的形式。如：*pronoun* 是个描述单个命名实体的特征，命名实体 *i* 取值为 1 表示 *i* 是一个代词，0 表示不是。在我们的工作中将其替换为 *pair_pronoun*，则命名实体对(*i,j*)在特征 *pair_pronoun* 上的取值为：

0：*i* 和 *j* 都不是代词；1：*j* 是代词而 *i* 不是；2：*i* 是代词而 *j* 不是；3：*i* 和 *j* 都是代词。

相应的 *definite* 和 *demonstrative* 也被替换为 *pair_definite* 和 *pair_demonstrative*。

4 实验及讨论

4.1 实验设置

本文实验采用的语料来源于 ACE-2003，主要包括三个领域：news wire(NW)，newspaper(NP) 和 broadcast news(BN)。在文献[11]中使用相同的语料并实现了几个现有的系统，所以我们也采用相同的设置以进行对比。对于语料训练集和测试集的划分，命名实体识别等等都通过 Bart[12]来处理，Bart 所标记的命名实体并不完全和语料库本身标记的相同，因此我们取了两者的交集作为识别的命名实体进行学习，大约占语料库标记的 90%。Bart 同时也实现了 Soon 的方法，为了对比粗糙集方法的有效性，我们还同时实现了 Yang 的算法和 Zhang 的算法。

在 RSES2 系统中进行规则提取时需要设置一个覆盖率参数，该参数表示学习到的规则集所覆盖的样例占训练集的比率，我们采用默认值 1。

由于我们使用的是 RSES2 系统进行属性约简和规则提取，所以将标记的命名实体都转化成命名实体对的形式，构成一个决策表。在测试阶段，如果一个测试样本被不止一条规则所覆盖，则利用标准投票机制（一条规则有多少样例支持，则就有多少票）来解决冲突问题。评测方法主要采用两种常用的方法：MUC 和 B-cubed 方法。

为了检验特征选择的必要性，我们设计了两类实验，这两类实验都采用 LEM2 算法进行规则提取，第一类实验不考虑特征选择过程，也就是不采用属性约简，我们将其表示为 System 1；第二类实验考虑特征选择过程，在提取规则前先进行属性约简，我们将其表示为 System2。

4.2 不考虑特征选择的 System1

在 System1 里面我们并不考虑特征选择过程，只用 LEM2 算法进行规则提取，将学习到的规则集作为分类器，测试结果如表 1 所示。LEM2 学习到的规则中每个条件部分的位置前后按照覆盖的样例多少进行排序的，如规则(*appositive*=0)&(*alias*=0)&…& (*semanticClass*=1)&(*pair_definite*=2) ⇒ (*link*=0)，由特征值对(*appositive*=0)所覆盖的样例数要比由(*alias*=0)多。

表1　无属性约简的共指消解结果

	MUC			B-cubed		
	P	R	F	P	R	F
NW	0.759	0.605	0.673	0.724	0.651	0.685
NP	0.765	0.653	0.705	0.702	0.613	0.655
BN	0.587	0.390	0.468	0.561	0.521	0.541

4.3　经过属性约简的 System2

在 System2 里，我们引入了特征提取过程，应用基于可辨识矩阵的属性约简方法，再在约简过的基础上用 LEM2 算法进行规则提取。

通过对语料库中三个语料进行属性约简，我们得到结果为：语料 NW 和 NP 中不存在冗余特征，BN 中存在两个冗余特征：*alias* 和 *appositive*，所以在进行规则提取前将这两个特征从 BN 中删除，学习到的结果如表 2 所示。

表2　基于可辨识矩阵属性约简的共指消解结果

	MUC			B-cubed		
	P	R	F	P	R	F
NW	0.759	0.605	0.673	0.724	0.651	0.685
NP	0.765	0.653	0.705	0.702	0.613	0.655
BN	0.681	0.548	0.607	0.636	0.591	0.613

对比表 1 和表 2 我们可以看出，删除冗余特征后，在 BN 上 MUC 和 B-cubed 的值分别高了 14%和 7%。对于 NW 和 NP 而言，不存在冗余特征，为了测试下删除不是冗余特征后的性能，我们也将 *alias* 和 *appositive* 删除，得到的结果分别是：NW 在 MUC 值上降低了 3%，在 B-cubed 值上降低了 2%；NP 在 MUC 值上降低了 5%，在 B-cubed 值上降低了 3%。通过对比删除冗余特征结果可知，相同的特征对不同的语料来说作用是不一样的，特征 *alias* 和 *appositive* 对于语料 NW 和 NP 是有用的，而对语料 BN 则是冗余的。由此可见在进行对特定语料学习时引入特征选择过程的必要性。我们通过对语料的进一步分析，又做了些对比实验。

4.4　进一步移除特征 *sentDist* 的实验

重新检验在实验过程中使用的特征，除了特征 *sentDist* 外，其他特征都是离散型的。对于 *sentDist*，对应的语义是“距离”，虽然取值范围是整数，但是将其看作连续值更加合适。在文[18]和我们上面的实验中都将其看作离散型特征进行处理，因其取值范围较大，基于该特征将会产生许多不同的特征值对，因此在规则集里，该特征将因此被认为是个很重要的区分性特征。在产生的规则集中会有许多精确的距离值，这由此就可能带来一个过拟合问题。

举例而言，有两个命名实体对在特征 *sentDist* 上分别取值是 159 和 160，两个实体对的距离都非常远，直观上可以说特征 *sentDist* 在这两个取值的基础上对于决策特征 *link* 应该具有相同的影响。但是在学习的过程中，我们却将其看作两个不同的取值，如果这两个命名实体对具有不同的 *link* 值，则 *sentDist* 的取值 159 和 160 就具有重要的区分性，这样的规则显然没有任何的泛化性，为了解决这种问题，可以使用能够处理连续值的学习方法，也可以将该特征进行离散化处理，将取值离散化为区间值的形式。因为有多种离散化处理方式，而且每种离散化的结果可能会不同，因此本文采用最简单的方法：直接去除 *sentDist* 特征。

对于语料 NW 和 NP，移除特征 *sentDist*，对于语料 BN，移除特征 *sentDist* 和冗余特征 *alias* 和 *appositive*，学习到的结果如表 3 所示。

表3　移除特征 *sentDist* 的共指消解结果

	MUC			B-cubed		
	P	R	F	P	R	F
NW	0.825	0.610	0.701	0.794	0.661	0.722
NP	0.822	0.651	0.726	0.744	0.630	0.682
BN	0.801	0.539	0.646	0.760	0.590	0.664

实验 3 的结果和实验 2 提供的结果相比明显提高很多。由此可以看出将 *sentDist* 作为一个离散型的特征处理时对于算法 LEM2 在这三个语料上的性能而言是不合适的。直接将其删除，反而取得更好的效果。如果考虑语料背景，进行合适的离散化处理，或许能够取得更好的性能。

4.5　和其他方法对比

为了说明 LEM2 算法的有效性，我们将实验 3 的结果和其他已有的规则类算法进行对比，对比算法有采用决策树(C5)算法做分类器的 Soon 方法，采用 ILP 算法的 Yang 方法和采用 ILM+MLN 的 Zhang 方法。结果如表 4，5，6 所示。

表4　在语料 NW 上的方法比较

	MUC			B-cubed		
	P	R	F	P	R	F
Soon(Bart)	0.808	0.520	0.633	0.793	0.600	0.683
Yang(ILP)	0.733	0.598	0.658	0.647	0.666	0.656
Zhang(ILP+MLN)	0.745	0.600	0.700	0.699	0.676	0.687
Our method	0.825	0.610	**0.701**	0.794	0.661	**0.722**

表5　在语料 NP 上的方法比较

	MUC			B-cubed		
	P	R	F	P	R	F
Soon(Bart)	0.791	0.570	0.663	0.755	0.566	0.647
Yang(ILP)	0.762	0.639	0.695	0.575	0.634	0.603
Zhang(ILP+MLN)	0.741	0.686	0.712	0.665	0.637	0.651
Our method	0.822	0.651	**0.726**	0.744	0.630	**0.682**

表6　在语料 BN 上的方法比较

	MUC			B-cubed		
	P	R	F	P	R	F
Soon(Bart)	0.849	0.459	0.596	0.862	0.557	0.677
Yang(ILP)	0.717	0.393	0.508	0.592	0.553	0.572
Zhang(ILP+MLN)	0.790	0.655	**0.716**	0.714	0.654	**0.682**
Our method	0.805	0.539	0.646	0.760	0.591	0.664

从表中我们可以看到：

- 和Soon 的方法比，我们的方法在三个语料上MUC取值提高了5%，在语料NW和NP上B-cubed取值提高了 4%，在语料 BN 上我们的方法低了 1%。
- 和 Yang 的方法比，我们的方法在三个语料上 MUC 取值提高了 3%以上，B-cubed 取值提高了 7%以上。
- 和 Zhang 的方法比，我们的方法在语料 NW 和 NP 上取值要高，在 BN 上要略低。除此之外，在训练时间上，本文的方法要远远少于 Zhang 的方法。在参数设置上，本文的方法也要远远比 Zhang 方法简单。限于篇幅，关于训练时间和参数设置的对比本文不予详述，只将结果陈述于此。

4.6 讨论

从 4.2 和 4.3 部分实验我们可以看出，即使只有 11 个条件特征和 1 个决策特征，特征选择过程对于不同的语料而言仍然是必须的，当然这也会与最终的分类器学习算法有关，就本文使用的 LEM2 规则提取算法而言，特征选择的结果除了精简了规则外，还提高了学习性能。

在规则提取类算法中，决策树方法是基于每个特征的“重要度”来构建树的，Yang 和 Zhang 文献中采用的 ILP 方法提取的是一阶规则，这两种规则提取方法都是基于全局考虑的覆盖算法，这在某种程度上意味着抽取的规则具有较高的支持度和较低的精确度。基于粗糙集理论的 LEM2 算法是基于特征值对的，其搜索空间是所有特征值对的集合，是一种局部覆盖算法，提取的每条规则具有较高的精确度和较低的支持度。为了提高规则集的支持度，LEM2 算法学习出较多的规则，从而提高了整个规则集的支持度。表 4，5，6 中的结果也显示了本文方法可以得到较高的准确率和很好的召回率，使得 F 值比较高。

5 结论

本文在共指消解任务中考虑特征选择过程，利用粗糙集理论中的属性约简方法去除冗余特征，并使得选取的特征对不同语料具有针对性。在特征选择的基础上，采用基于粗糙集理论的 LEM2 规则提取算法进行学习，利用学习到的规则集作为分类器进行共指消解。和现有的利用规则作为分类器的共指消解系统相比，粗糙集方法在共指消解任务中能够取得较好的性能。

在本文中使用粗糙集方法进行处理时，并没有考虑领域知识，结合语料构建合适的特征并分析每个特征在共指消解任务中的重要性和相关性是我们下一步的工作。

参 考 文 献

[1] E. Bengtson and D. Roth. Understanding the value of features for coreference resolution[A]. In: Proc. of EMNLP[C]. 2008. 294-303.

[2] X. Yang, J. Su, J. Lang, C.L. Tan, T. Liu and S. Li. An entity-mention model for coreference resolution with inductive logic programming[A]. In: Proc. of ACL-HLT[C]. 2008. 843-851.

[3] V. Ng and C. Cardie. Improving machine learning approaches to coreference resolution[A]. In: Proc. of ACL[C]. 2002. 104-111.

[4] A. Culotta, M. Wick, R. Hall and A. McCallum. First-order probabilistic models for coreference resolution[A]. In: Proc. of NAACL[C]. 2007. 81-88.

[5] W.M. Soon, H.T. Ng and D.C.Y. Lim. A machine learning approach to coreference resolution of noun phrases[J]. Computational Linguistics, 2001, 27(4): 521-544.

[6] I. Guyon and A. Elisseeff. An introduction to variable and feature selection[J]. Journal of Machine Learning Research. 2003, 3: 1157-1182.

[7] Z. Pawlak. Rough sets[J]. International Journal of Computer and Information Sciences, 1982, 11: 341-356.

[8] A. Skowron and C. Rauszer. The discernibility matrices and functions in information systems[J]. Intelligent decision support: Handbook of applications and advances of rough set theory, Kluwer Academic Publishers, Dordrecht, 11: 331-362.

[9] J.W. Grzymala-Busse. A new version of the rule induction system LERS[J]. Fundamenta Informaticae. 1997, 31(1): 27-39.

[10] C.C. Chan and J.W. Grzymala-Busse. On the two local inductive algorithms: Prism and LEM2[J]. Foundations of Computing and Decision Sciences, 1994, 19: 185-203.Y.

[11] Zhang, J. Zhou, S. Huang and J. Chen. Combing ILP and MLN for coreference resolution[A]. In: Proc. of IALP[C]. 2009. 59-64.

[12] V. Yannick S. Ponzetto, M. Poesio, V. Eidelman, A. Jern, J. Smith, X. Yang and A. Moschitti. Bart: a modular toolkit for coreference resolution[A]. In: Proc. of ACL-HLT: Demo Session[C]. 2008. 9-12.