

基于相反分类器的数据流分类方法

王 勇 李战怀 张 阳 蒋 芸

(西北工业大学计算机科学与软件系 西安 710072)

摘 要 目前挖掘概念流动的数据流已经成为研究的热点。概念流动的数据流分类在预防信用卡欺诈,网络入侵发现等应用中具有重要的应用。本文定义了一种相反分类器来从错误中学习,提出了训练一个集合分类器来对具有概念流动的数据流进行分类的算法 IWB。通过在合成数据集和 benchmark 上的实验,与 Weighted Bagging 算法^[13]比较,表明我们的算法具有更高的准确度,更快地收敛到新的目标概念的性能。

关键词 集合分类器,相反分类器,概念流动

A Method for Classifying Data Stream Based on Reverse Classifier

WANG Yong LI Zhan-Huai ZHANG Yang JIANG Yun

(Department of Computer Science & Software, Northwestern Polytechnical University, Xi'an 710072)

Abstract Recently, mining data streams with concept drifts for actionable insights has become an important and challenging task for a wide range of applications including credit card fraud protection, network intrusion detection, etc. In this paper, we define the reverse classifier, which can help an algorithm to learn from error, and propose a IBW (improved weighted bagging) algorithm for classifying concept drift data streams using weighted ensemble classifiers. We evaluate IBW algorithm and Weighted Bagging algorithm on the STAGGER Concepts and synthetic data. The experiment results show that the proposed method have substantial advantage over Weighted Bagging approach in prediction accuracy, and it can converge to target concepts with high accuracy and speed.

Keywords Ensemble classifier, Reverse classifier, Concept drifting

能在概念流动的条件下,进行准确分类的学习算法在很多领域内有重要的应用。这些应用,包括在网络事务日志中区分正常访问和黑客攻击;在信用卡数据流中区分出欺诈申请。由于在这些数据流中进行分类面临巨大的数据量和概念流动的挑战,所以使得传统的数据挖掘方法不再适用。这些应用要求分类算法能够处理大量的数据,以及在遇到概念流动的时候,能很快收敛到新的目标概念。

目前,研究者已经提出了一些方法对具有概念流动的数据流进行分类。FLORA 算法^[1]是通过维持和滑动窗口中的实例一致的规则系统来进行分类。Domingos, Hulten 等人提出了 VFDT 算法^[2~4],该算法是基于 Hoeffding 树的在线决策树学习系统。文[5]中提出了用模糊信息网络来建立在线更新的分类器。正如文[6]中指出的,集合分类器和单个分类器方法相比,更准确,更灵活,有时是更有效率的。文[7,8]提出了集合分类器 Weighted Majority 算法,该算法对集合中的每个分类器按精度赋权,但是该方法不能动态地增加和减少集合中的分类器。文[9]改进了这个方法,但加入了一些在很多情况下不符合实际的假定。比如文[9]中假定实例的每个属性服从某种高斯分布。文[10]中提出了基于交叉验证的决策树集合方法,指出了在新的概念和旧的概念分布相同,以及新的数据不足的情况下,旧的数据才会帮助产生更准确的模型。文[11]使用推进的方法建立集合分类器。文[12,13]是通过装袋的方法来处理数据流中的概念流动的集合分类器方法。分别被称为 Bagging 和 Weighted Bagging 方法。上面提到的算法面临一个重要的问题,即当概念出现巨大的变化时,分类的精度将出现突然下降的现象,而在理想的情况下,预测错误率应该和概念流动不相关^[10]。

本文中提出了基于集合分类器的 IWB 算法,定义了相反分类器。通过对相反分类器的使用,在遇到突然的概念流动时,IWB 算法可以从错误中学习,具有更快收敛到新的目标概念的能力,而当概念稳定时没有大的波动。本文最后通过在标准数据集和大数据集上与 Weighted Bagging 算法做比较验证了我们的结论。

1 相反分类器和分类器精度赋权公式

当概念流动或者分布发生变化时,使用原来训练的分类器来分类新的实例必然会导致分类精度的突然下降。如果只依赖数据流中新到来的实例训练新的分类器去更新集合分类器,则会导致收敛到新的目标概念的速度下降。本文的基本思想,是通过建立相反分类器使分类算法可以从错误中学习。当一个分类器的期望分类精度较低时,不是简单放弃该分类器而应建立并验证相反分类器。当相反分类器和现有概念一致时,通过更新集合分类器,就可能提高集合分类器的分类精度。

定义 1(相反分类器) 一个分类器用 C 来表示,对于一个实例 \vec{x} , $C(\vec{x}) = c_i, c_i \in c$, 则分类器 C 的相反分类器记为 \bar{C} , 其中 $\bar{C}(\vec{x}) = c'_i, c'_i \neq c_i, c'_i \in c$ 。特别当 $c = \{0, 1\}$ 时,分类器 C 的相反分类器是 $\bar{C}(\vec{x}) = \begin{cases} 0, & \text{当 } C(\vec{x}) = 1 \text{ 时} \\ 1, & \text{当 } C(\vec{x}) = 0 \text{ 时} \end{cases}$ 。

图 1 中的直线表示目标概念 1,在实际应用中一般是未知的。折线表示根据实例训练得到的分类器 C_1 ,是对目标概念 1 的一个近似。图 2 是分类器 C_1 的相反分类器,它将折线上面的实例都标记为负例,而将折线下面的实例都标记为正例。图 3 是目标概念 2 以及对应的分类器 C_2 。数据流中经

常面临概念变化的情况,当概念由 1 变化到 2 时,用分类器 C_1 去分类,实际上由概念 2 决定的实例,这就会出现很大的错误率。此时 weighted Bagging 方法就会舍弃分类器 C_1 ,而逐步训练新的分类器以适应概念 2。从而会导致分类器收敛速度下降和分类器 C_1 中信息的浪费。而分类器 C_1 的相反分类器 \bar{C}_1 却和概念 2 比较接近,可以先用 \bar{C}_1 去分类概念 2 标记的实例,等得到与概念 2 更近似的高精度分类器时再替换分类器 \bar{C}_1 。判断一个分类器的相反分类器是否与最近的概念相近,必须计算它的期望预测错误率,只有该相反分类器的分类能力较高时,才将它加入集合中更新集合分类器。

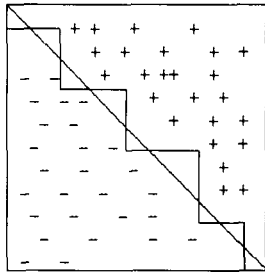


图 1 目标概念 1 及对应的分类器 C_1

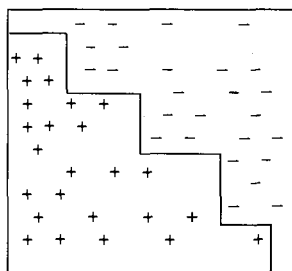


图 2 分类器 C_1 的相反分类器 \bar{C}_1

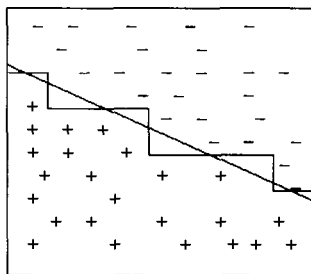


图 3 目标概念 2 及对应的分类器 C_2

当数据流逐步的流入时,到来的数据被分成数据段 $S_1, S_2, \dots, S_n, \dots$, 其中 S_n 是最近时间步到来的数据段。记每个段的大小为 $ChunkSize$, 在每个数据段上训练一个分类器 C_i , 组成一个分类器集合。为了对每个分类器进行准确度赋权,可在一个检测数据集合上计算它的期望预测错误率。若假设最近的训练数据段和检测集合的类分布非常近似,则分类器的准确率赋权可以通过在 S_n 上的分类错误率近似得到。具体来说,一个分类器 C 在 S_n 上的平均平方错误率可定义为

$$MSE(C, S_n) = \frac{1}{|S_n|} \sum_{(\vec{x}, c_i) \in S_n} (1 - f_C(\vec{x}))^2 \quad (1)$$

其中: S_n 中的每个实例具有形式 (\vec{x}, c_i) , \vec{x} 是一个向量表示一个实例, c_i 是该实例的类标号; $f_C(\vec{x})$ 是分类器 C 计算得出的

实例 \vec{x} 属于类 c_i 的概率。 $c_i \in c(S_n)$, $c(S_n)$ 是 S_n 中实例的类标记的集合。分类器 C 的权应该和平均平方错误率成反比。

另一方面,当用一个随机分类器来分类数据段 S_n 时,它的平均平方错误率是

$$MSE_{random}(S_n) = \frac{1}{|c(S_n)|} \sum_{c_i \in c(S_n)} p(c_i)(1 - p(c_i))^2 \quad (2)$$

其中: $p(c_i)$ 是 S_n 中类标记为 c_i 的实例的比例, $c_i \in c(S_n)$, $c(S_n)$ 是 S_n 中实例的类标记的集合。

例如,当 $S_n = \{(\vec{x}_1, 0), (\vec{x}_2, 1)\}$ 且类分布相同时,则 $MSE_{random}(S_n)$ 为 0.25。若使用随机分类器的错误率作为一个域值,则对集合中一个分类器赋权通过式(3)来计算得到。

$$w(C, S_n) = MSE_{random}(S_n) - MSE(C, S_n) \quad (3)$$

当一个分类器的权为负时,表示该分类器的分类精度比一个随机分类器的精度还要差,所以 weighted Bagging 会放弃权为负的所有分类器,这也反映了该分类器的相反分类器与当前的概念比较接近,如果这个相反分类器的精度较高,就可以帮助改进分类的精度。

2 Improved Weighted Bagging 算法

装袋和推进是两种经常使用的产生集合分类器的方法。而推进方法要求对数据点进行重新赋权和对噪音敏感,所以不适用于数据流分类^[12]。本文中使用的装袋方法来产生集合分类器。数据流分类算法要求处理无限多的数据流中的点,也要生成无限多的分类器,但是不可能也不必要使用所有产生的分类器去进行分类。而只是使用在目前的训练集合上具有最高的分类精度的 K 个分类器来组成分类器的集合。

算法 1 IWB(improved weighted bagging)算法

输入: S // 数据流中新到来的 $ChunkSize$ 个点构成一个数据集合
ensemble // K 个预先训练的分类器组成的集合
 K // ensemble 中的分类器的数目的最大值
输出: ensemble // 分类器的集合,其中每个元素是一个分类器及它的权
 $C = Classifier(S_n)$; // 在第 n 个时间步到来的数据集合 S_n 上训练分类器 C

$MSE(S_n, C)$; // 通过交叉验证计算 C 在 S_n 上的平均平方错误率

$w(C, S_n) = MSE_{random}(S_n) - MSE(S_n, C)$

```

Foreach  $C_i \in ensemble$  {
    ensemble = Remove( $C_i$ , ensemble);
     $w(C_i, S_n) = MSE_{random}(S_n) - MSE(C_i, S_n)$ ;
    if ( $w(C_i, S_n) > 0$ ) {
        ensemble ← ( $C_i$ ,  $w(C_i, S_n)$ );
    }
    else {
         $\bar{C}_i = ReverseClassifier(C_i)$ ;
         $w(\bar{C}_i, S_n) = MSE_{random}(S_n) - MSE(\bar{C}_i, S_n)$ ;
        if ( $w(\bar{C}_i, S_n) > 0$ ) {
            ensemble ← ( $\bar{C}_i$ ,  $w(\bar{C}_i, S_n)$ );
        }
        else {
            ensemble ← Max weight(( $C_i$ ,  $w(C_i, S_n)$ ), ( $\bar{C}_i$ ,  $w(\bar{C}_i, S_n)$ ));
        }
    }
}

```

$wnsemble \leftarrow (C, w(C, S_n))$;

ensemble = TopK(ensemble);

return ensemble;

算法 1 是挖掘概念流动的数据流的集合分类器算法。在每一个时间步,一个新的数据段到达时,首先由这个数据段训练一个分类器。其次在这个数据段上基于交叉验证的方法来评价这个分类器。然后基于这个数据段,根据式(3)更新集合中其它分类器的权,当这个权小于 0 时,先计算它的相反分类器,再在这个数据段上进行评价,只有当相反分类器的权大于 0 时,它才被加入集合中。最后返回这个集合中权最高的 k 个分类器。

在预测一个新的实例的类标号时,使用集合中所有权大于 0 的分类器进行赋权投票来决定最终的类标号。

3 试验结果与讨论

为了验证我们的算法,首先在 STAGGER Concept 上和 Weighted Bagging 方法进行比较。STAGGER Concept 是对具有概念流动的数据流算法的分类能力的标准检测方法^[9]。在 STAGGER Concept 中,每个实例有三个属性值, $color \in \{gree, blue, red\}$; $shape \in \{triangle, circle, retangle\}$; 和 $size \in \{small, medium, large\}$ 。训练实例陆续在 120 个时间点上到达,在每个时间点,学习器收到一个实例。在前 40 个时间点上,目标概念是 $color = red \wedge size = small$,接下来的 40 个时间点,目标概念是 $color = green \vee shape = cicle$,在最后的 40 个时间点上目标概念是 $size = medium \vee size = large$ 。为了评价一个学习器,在每个时间点上随机的产生 100 个检测实例,这 100 个实例是以当前时间的目标概念标记的。在学习一个新的训练实例后,用集合中的分类器在相应的 100 个检测实例上计算正确预测的百分比。在整个的实验中,重复这个过程 50 次,取其平均值。集合分类器中的每个分类器都是采用决策树分类器 C4.5。

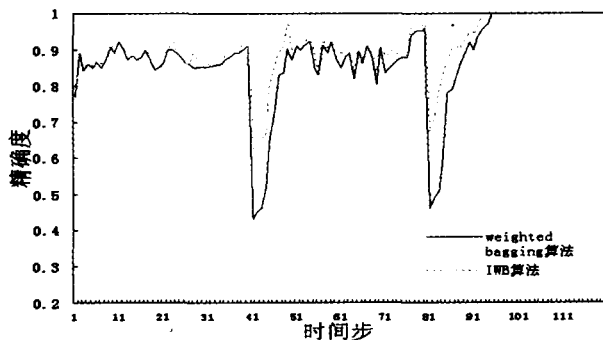


图4 Weighted Bagging 算法和 IWB 算在 stagger 概念上的分类精确度比较

图4是在标准数据集上,将 Weighted Bagging 算法和 IWB 算法所预测的精度进行比较。其中横坐标是时间步,纵坐标是在检测集合上进行正确分类的百分比。从中可以看出,在时间步 40 和 80 的时候,概念发生了变化,Weighted Bagging 方法的预测精度突然降低。和利用了相反分类器的 IWB 算法相比,IWB 算法预测精度更高,收敛到新的目标概念的速度更快。IWB 算法不是通过在线更新旧有的分类器的方法来改进一个分类器,而是通过利用相反分类器的概念来改变旧有的分类器以建立一个新的分类器的方法。在线更新旧有的分类器首先要求这个分类器逐步忘记以前的训练实例。当概念突然变化时,这种方法收敛速度较慢。而 IWB 算法中建立相反分类器非常简单,只需验证这个相反分类器的分类准确度就可以直接应用。同时在新的分类器被训练得到以后,如果它的分类精度较高,还可以替换旧有的分类器。

为了检验 IWB 算法在大数据集合上^[13]的分类能力,我们还在大数据集合上和 Weighted Bagging 方法进行了比较。大数据集合的生成方法与文^[13]中的方法一致。在 d 维空间上,一个超平面被定义为 $\sum_{i=1}^d a_i x_i = a_0$ 。对于满足条件 $\sum_{i=1}^d a_i x_i \geq a_0$ 的点 (x_1, x_2, \dots, x_d) ,我们将其标记为正例,否则标记为负例。由于在真实的数据集里,样本通常不是均匀分布于多维空间 $[0, 1]^d$ 上的,一些先前未知的分布规则有可能应用于数据流中。为此在实验中,引入另一个 d 维超平面 $\sum_{i=1}^d b_i x_i = b_0$,仅当样本满足 $\sum_{i=1}^d b_i x_i \geq b_0$ 时,才能出现在数据流中,我们用这个超平面来模拟分布规则的变化。由于这两个

超平面是互相独立的,通过改变概念平面系数 a_i ,分布平面系数 b_i ,或者同时改变 a_i 和 b_i ,可以模拟不同的数据流。我们采取同时改变概念平面系数和分布平面系数,这样可以最大限度地模拟真实的数据流。权 $a_i, b_i (1 \leq i \leq d)$ 的初始值在范围 $[0, 1]$ 内任意选取。我们选取 a_0 和 b_0 ,使两个超平面将多维空间划分为几乎相等的两个子空间,即 $a_0 = \frac{1}{2} \sum_{i=1}^d a_i, b_0 = \frac{1}{2} \sum_{i=1}^d b_i$,再通过一系列的参数来模拟概念和分布流动的数据流。随机引入的噪声会改变样本的类标号,噪声率选为 5%,参数 D 指权值改变的维数的总和;参数 $s_i \in R$ 表示每个样本的权值 a_i 和 b_i 的变化量。例如,在每个样本产生后,以 s_i/N 调节 a_i 和 b_i 的值。另外,在 N 个样本产生以后, a_i 和 b_i 可能以 10% 的概率朝着相反的方向变化,也就是说 s_i 被 $-s_i$ 替换的概率是 10%。同时,每当改变 a_i 或者 b_i 的权值后, a_0 和 b_0 也随之改变。在实验时每隔 40 个时间步按照上面的方法模拟概念的变化。为了评价一个学习器,在每个时间步上随机产生 1000 个检测实例,这 1000 个实例是以当前时间的目标概念标记的。在每个时间步运行算法 IBW 后,使用集合中的分类器来分类检测实例集合并计算正确预测的百分比。当每个时间点到来的实例数是 1000, 2000, 5000, 10000 时,得到了相似的结果,下面给出当数据流中的数据段的大小为 2000 时的结果。

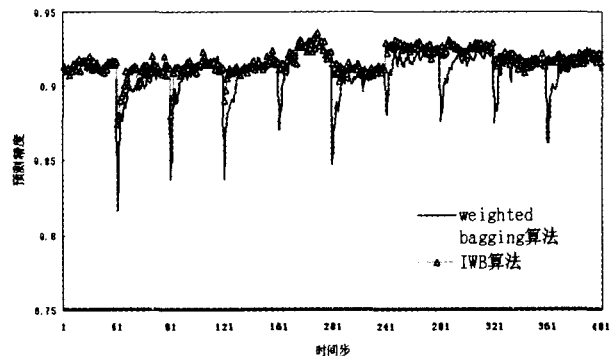


图5 Weighted Bagging 算法和 IWB 算法在大的合成数据集上的预测精度比较

图5是 Weighted Bagging 算法和 IWB 算法在大的数据流上的预测精度比较结果。其中横坐标是时间步,纵坐标是在检测集合上进行正确分类的百分比。从图中可以看出,每当概念发生变化,Weighted Bagging 算法重复出现突然的预测精度降低。而在概念发生流动的时候,IWB 算法具有更快的收敛到目标函数的能力。不仅在发生概念流动的时候,预测的精度下降比较低,而且收敛到新的目标概念的速度比较快。更重要的是随着时间的推移,IWB 的学习能力越来越强。在时间步 240 后,预测的精度不再随着概念的变化而出现突然降低,一直保持较平稳的预测精度。正如文^[10]中 FanWei 所指出的理想情况下,预测错误率应该和概念流动的数目不相关。

结论 本文定义了相反分类器,设计并实现了基于相反分类器的集合分类器算法 IBW。在标准数据集和合成数据集上进行了检验。实验结果表明,IWB 算法比 Weighted Bagging 方法在遇到概念流动的时候具有更快的收敛到目标函数的能力并具有更高精度。进一步的工作是使用相反分类器的概念来改进其它的集合分类器方法。

参考文献

- 1 Widmer G, Kubat M. Learning in the presence of concept drift and

hidden contests. Machine learning, 1996, 23: 69~101
 2 Domingos P, Hulten G. Mining High-speed Data Streams. In: Proc. Of the sixth Intl. Conf. on Knowledge Discovery and Data Mining, 2000. 71~80
 3 Hulten G, Spencer L, Domingos P. Mining time-changing data Streams. In ACM SIGKDD, 2001
 4 Jin R, Agrawal G. Efficient Decision Tree Construction on Streaming Data. in proceedings of ACM SIGKDD, 2003
 5 Cohen L, Avrahami G, Last M. Incremental Info-Fuzzy Algorithm for Real Time Data Mining of Non-Stationary Data Streams. In Proc. of the Fourth IEEE International Conf. on Data Mining (ICDM2004), 2004. 39~43
 6 Kuncheva L I. Classifier Ensembles for Changing Environments. In: Proc. 5th Int. Workshop on Multiple Classifier Systems, 2004. 1~15
 7 Littlestone N, Warmuth M K. The weighted majority algorithm. Inform. Computation, 1994. 212~261

8 Littlestone N. Learning quickly when irrelevant attributes a bound: A new linear threshold algorithm. Machine learning. 1988. 285~318
 9 Kolrer J Z, Marcus A. Dynamic Weighted majority: A new Ensemble Method for Tracking Concept Drift. In: Proc. of the Third Int. IEEE Conf. on Data Mining, 2003. 123~130
 10 Fan Wei. Systematic data selection to Mine Concept-Drifting data Streams. In the proceeding of the Conf. KDD, 2004. 128~137
 11 Chu F, Zaniolo C. Fast and light boosting for adaptive mining of data streams. In: Proc. of the 5th Pacific-Asic Conference on Knowledge Discovery and Data Mining (PAKDD), 2004
 12 Street W, Kim Y. A streaming ensemble algorithm(sea) for large-scale classification. In Int'l Conf. on Knowledge Discovery and Data Mining (SIGKDD), 2001
 13 Wang H, Fan Wei, Yu P, Han J. Mining concept-drifting data streams using ensemble classifiers In int'l conf. on Knowledge Discovery and Data Mining (SIGKDD), 2003

(上接第 186 页)

证明:可以看出 K 是一个二重蕴涵式,根据基本语义定义,只须证

$M| =_w \Box(p \rightarrow q) \& M| =_w \Box p \rightarrow M| =_w \Box q$
 即可。

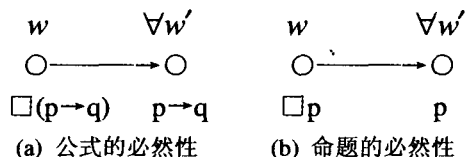


图 3 公式和命题的必然性

根据定义 3(4),由 $M| =_w \Box(p \rightarrow q)$ 可知,对于任意的 w', wRw' ,有 $M| =_w p \rightarrow q$ (见图 3a);再由 $M| =_w \Box p$ 可知,对上述的 w' ,都有 $M| =_w p$ (见图 3b)。因此有 $M| =_w q$ 。由于 w' 是 w 的任意可达世界,因此有 $M| =_w \Box q$,即 $M| =_w \Box(p \rightarrow q) \rightarrow (\Box p \rightarrow \Box q)$ 。又由于 M 和 w 都是任意选定的,因此 K 在所有的模型中有效。证毕。

命题 7 公式 $\Box \Box(p \rightarrow q) \rightarrow (\Box \Box p \rightarrow \Box \Box q)$ 在任意模型 M 中是有效的

证明:同理,只须证

$M| =_w \Box \Box(p \rightarrow q) \& M| =_w \Box \Box p \rightarrow M| =_w \Box \Box q$
 即可。

根据定义 3(4'),由 $M| =_w \Box \Box(p \rightarrow q)$ 可知,对于任意的 w', wRw' 有 $M| =_w \Box(p \rightarrow q)$;再进一步,对于任意的 $w'', w'Rw''$,有 $M| =_w p \rightarrow q$ (参见图 4a)。由 $M| =_w \Box \Box p$ 可知,对上述的 w' ,都有 $M| =_w \Box p$;再进一步,对上述 $w'', w'Rw''$ 都有 $M| =_w p$ (参见图 4b),因此有 $M| =_w q$ 。由于 w'' 是 w' 的任意可达世界,因此有 $M| =_w \Box q$;又由于 w' 是 w 的任意可达世界,因此有 $M| =_w \Box \Box q$ 。这样我们就得到 $M| =_w \Box \Box(p \rightarrow q) \rightarrow (\Box \Box p \rightarrow \Box \Box q)$ 。又由于 M 和 w 都是任意选定的,因此 $\Box \Box(p \rightarrow q) \rightarrow (\Box \Box p \rightarrow \Box \Box q)$ 在所有的模型中有效。证毕。

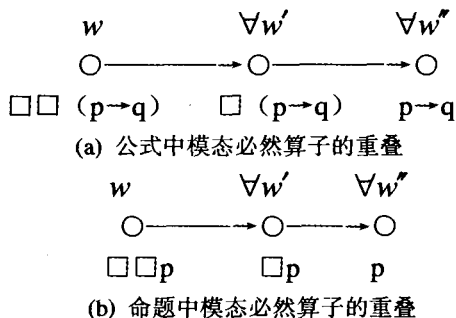


图 4 公式和命题中模态算子的重叠

结论 我们通过对可能世界语义学下特征公理的分析,得到了关于它们在框架上有效的一些对应条件,这些条件可以看作是特征公理有效的形式化解释。从另一方面看,也只有满足这些条件,相应的公理才是有效的,这样我们就可以确定在某一特定条件下将哪些公理作为正确的思维形式。通过这种分析,我们可以发现,一些直观上成立的公式(如 D、T)也是有条件成立的。由于在一般的 BDI 模型中,必然算子 \Box 代表 Agent 的信念,我们对文[1~3]中的换位推理的有效性进行了分析,并从模态逻辑和 Kripke 可能世界语义的角度给出换位推理规则有效性的语义证明。

虽然我们从语义上证明了换位推理规则在任意模型上的有效性,但是这是在默认了 Agent 对自己状况和行为知道的前提下对其它 Agent 状况和行为的“合理”推理。如果一个推理者不知道自己的状况和行为(即不满足认知反省公理),或者被推理者与推理者的认知状态差别很大,那么换位推理规则的有效性就不会得到满足。因此只有在满足弱 KD45 的前提下,换位推理的有效性才能得到保障。

参考文献

1 SHI Zhongzhi, TIAN Qijia, LI Yunfeng. RAO Logic for Multi-agent Framework. Journal of Computer Science and Technology, 1999, 14(4): 393~400
 2 王文杰,田启家,史忠植. 多主体系统中对其它主体的研究. 计算机研究与发展, 1998, 35(11): 971~974
 3 田启家,史忠植,王怀清. 多主体系统中的动作和知识推理. 见: 第四届中国人工智能联合学术会议论文集(96 人工智能进展). 清华大学出版社, 1996. 74~79
 4 杨鲲,陈建中,孙德刚,刘大有. 认知逻辑中逻辑全知问题及其解决方法. 吉林大学自然科学学报, 1999(3): 40~43
 5 周昌乐. 认知逻辑导论. 北京:清华大学出版社, 2001
 6 李金厚,蒋静坪. 从逻辑全知问题看当前基于逻辑的 Agent 研究的两个认识盲点. 北京科技大学学报, 2004, 26(2): 215~218
 7 周北海. 模态逻辑. 北京:中国社会科学出版社, 1997
 8 胡泽洪. 逻辑的哲学反思. 北京:中央编辑出版社, 2004. 127~135
 9 Blackburn P, de Rijke M, Venema Y. Modal Logic [M]. Cambridge University Press, 2001
 10 Kripke S. Semantic Analysis of Modal Logic II: Non-Normal Modal Propositional Calculi. In: Symposium on the Theory of Models, North-Holland, Amsterdam, 1965
 11 Konolige K. A Deduction Model of Belief. Pitman, London and Morgan Kaufmann, San Mateo, CA, 1986
 12 刘勇,蒲树祯,等. BDI 模型信念特性研究. 计算机研究与发展, 2005, 42(1): 54~59