基于协同训练的文本蕴含识别 *

任函[1]，万菁[2]，吴泓缈[1,2]，冯文贺[3]

（1. 武汉大学 外国语言文学学院，湖北省武汉市 430072；

2. 武汉大学 湖北省语言与智能信息处理研究基地，湖北省武汉市 430072；

3. 武汉大学 计算机学院，湖北省武汉市 430072）

摘要：针对文本蕴涵的训练数据不足的问题，本文提出了基于协同训练的文本蕴含识别方法。该方法利用少量已标注的蕴含数据和大量未标注数据进行协同训练。为此，本文利用改写视图和评估视图，从结构和非结构两个角度考察蕴含关系，并将语义树核分类器和基于统计特征的分类器应用于两个视图，同时利用协同训练的结果训练一个综合分类器，用于对新数据进行预测。实验表明，基于协同训练的蕴含识别方法能在少量训练数据的情况下获得较好的识别性能。

关键词：文本蕴含识别；协同训练；语义树核

中图分类号：TP391　　**文献标识码：**A

A Co-training Based Approach for Recognizing Textual Entailment

Han Ren[1], Jing Wan[2], Hongmiao Wu[1,2], Wenhe Feng[3]

(1. School of Foreign Languages and Literature, Wuhan University, Wuhan, Hubei 430072, China;

2. Hubei Research Base of Language and Intelligent Information Processing,

Wuhan University, Wuhan, Hubei 430072, China;

3. School of Computer, Wuhan University, Wuhan, Hubei 430072, China)

Abstract: This paper introduces a co-training based approach for recognizing textual entailment. In this approach, a small labeled entailment dataset as well as a large unlabeled one are employed for co-training, which aims at solving the lack of entailment data . Two different views, rewriting view and assessing view, are proposed to measure structural and non-structural entailment relations, and two classifiers, namely semantic tree kernel based classifier and statistical features based classifier, are applied to train under the two views separately. For predication, a global classifier is built, trained by the results of co-training. Experiments show that the co-training based approach achieves a good performance in the case of a small training dataset.

Key words: Recognizing Textual Entailment；Co-training；Semantic Tree Kernel

1 引言

文本蕴含可以看成是一个连贯的文本 T 和一个被看作是假设 H 之间的一种关系，如果 H 的意义可以从 T 的意义中推断出来，那么就说 T 蕴含 H，即 H 是 T 的推断，记作 T→H。文本蕴含识别提供了一种理解自然语言中的多样化表达的有效手段，可以广泛应用于自动问答、信息抽取、自动文摘等众多自然语言处理应用中。

文本蕴含识别的一种主要策略是采用有监督的机器学习方法。该方法将文本蕴含问题看作一个两类（蕴含和不蕴含）或三类（蕴含、矛盾和未知）的分类问题，根据已标注的训练实例和蕴含特征进行学习。在文本蕴含识别的主要评测竞赛 RTE 中，大多数系统均采用该方法建立自己的文本蕴含识别系统[1,2,3]。

基于有监督学习的文本蕴含识别的一个关键问题是分类学习的性能。由于蕴含和非蕴含

* **收稿日期：**2014-06-20　**定稿日期：**2014-07-28

基金项目：中国博士后科学基金资助项目（2014M552073, 2013M540594），中央高校基本科研业务费专项资金（2012GSP017）

作者简介：任函（1980—），男，博士，主要研究领域为自然语言处理；万菁（1981—），女，硕士，主要研究领域为应用语言学；吴泓缈（1954—），男，博士，教授，主要研究领域为应用语言学；冯文贺（1976—），男，博士，讲师，主要研究领域为自然语言处理。

两个类都比较庞杂，实例间的相似性难以保证，据此建立的分类器的性能难以进一步提高。造成这一问题的根本原因是训练数据不足以及学习结果不充分。为此，可以采取两种方法：第一种方法是增加训练数据，以保证有足够的相似实例及可区分实例。然而，标注蕴含数据会增加大量的人工成本。第二种方法是选择更合适的蕴含特征，但是在训练数据集的无法提供足够实例的情况下，即使选择了有效的分类特征，学习效果也无法得到有效改进。

训练数据不足是机器学习的一个普遍问题。为此，可以采用半监督学习方法应对训练数据不足的问题。本文提出了一种基于协同训练（Co-training）的半监督学习方法进行文本蕴含识别。具体而言，协同训练过程分别利用文本蕴含的改写与评估两个视图考察蕴含关系。改写视图主要考察蕴含关系的结构化特征，而评估视图主要考察蕴含关系的统计特征；然后，利用协同训练方法在两个视图上进行半监督学习。实验表明，该方法在一定程度上提升了文本蕴含识别的效果。

2 文本蕴含识别的协同训练策略

协同训练方法需要定义两个不同的用于观察数据集的视图。对于每一个文本−假设对，我们可以从两个相对独立的视图上去观察，一个是改写视图，另一个是评估视图。

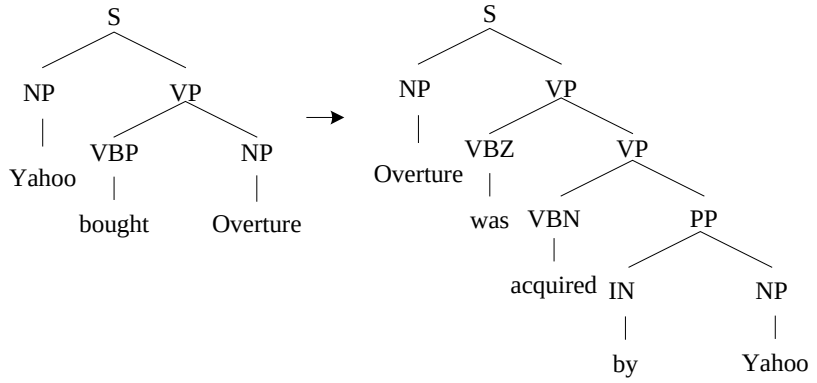

图 1 词汇和句法层面改写

2.1 改写视图

改写视图从蕴含语料构建者的角度来看待语料中的文本-假设对，认为蕴含的本质是对文本片断的改写，包括词汇、句法或者语义层面的改写。改写视图是 RTE 中的一种典型视图，很多参与 RTE 评测的蕴含识别系统都利用这种视图来构造分类特征[4,5,6]。另一方面，改写过程可由结构变化来表示。例如，图 1 中两句的蕴含关系体现在两个方面，一是词汇的改写，即 buy 和 acquire 具有蕴含关系，其改写体现在叶节点变化上；二是句式的变化，即由一般句式变为被动句式，其改写体现在非叶节点变化上。具有深度语义蕴含关系的文本也可以基于改写视图进行观察，只需用更复杂的结构变化表示即可。

2.2 评估视图

评估视图从评价蕴含关系的角度观察文本-假设对，即标注者如何去评价一个文本-假设对是否存在蕴含关系。其理由是，蕴含关系并不总是能够由句法或语义的改写来判断。例如，在下例中，T1 为列表形式，因此无法采用句法或语义分析得出蕴含关系。

T1：Rosanjin Kitaoji (*Kana readings for Rosanjin Kitaoji*).
Born in Kyoto in 1883.
Ceramist.

H1：Rosanjin Kitaoji, who was born in Kyoto in 1883, is a ceramist.

人工对该例进行评价时，主要根据 H 中描述的人物的各属性（姓名、时间、职业等）是否与 T 中的相同，得出是否具有蕴含关系的结论。因此，评估视图主要考察的是词汇的重叠程度，此时利用统计特征往往是有效的，例如词袋（bag-of-words）特征。

一些文本蕴含的研究也证明这种视图是有效的，如 Zanzotto 等[7]联合这种视图，从 Wikipedia revision corpus 中自动学习具有蕴含关系的文本和假设。Malakasiotis 等[8]在 RTE-3 评测中，利用字串相似性特征和支持向量机判断文本蕴含。因此，这种视图也属于强学习器，可以独立识别蕴含关系。

3 分类过程

在协同训练方法中，需要对每个视图单独训练分类器。这里我们采用 SVM 作为分类器，依据是 Gaona 等[9]对蕴含分类模型的研究，他利用 RTE-3 的数据训练了 SVM、朴素贝叶斯等八种机器学习模型，并采用 10 次交叉迭代验证得到蕴涵识别的准确度。实验结果表明，SVM 的准确度高于其他几种算法。

分类器设计包括两个部分，即核函数和分类特征的设计。核函数的设计可以围绕视图的特点来进行。对于改写视图，其主要涉及的是结构化特征，我们采用树核作为分类器的核函数。对于评估视图，其主要涉及的是统计特征，我们可以采用多项式核作为分类器的核函数。分类特征的选择同样围绕视图的特点进行，现分别就核函数及分类特征进行说明。

3.1 核函数

评估视图中的统计特征如词汇重叠等，主要用于表现非结构化信息，因此可采用如线性核、多项式核等核函数。本文采用多项式核作为分类的核函数。事实上，在文本蕴含识别系统中，也经常采用多项式核作为分类器的核函数[10,11]。而改写视图需要描述数据的结构信息，若仍采用多项式核，则可能导致在句法结构上非常相似的两个句子会被表示成完全不同的特征，由此会造成数据稀疏问题。为此，本文采用树核作为改写视图中的分类器核函数。标准树核函数的形式化描述为：

$$k(T_1, T_2) = \sum_{n_1 \in N_1} \sum_{n_2 \in N_2} \Delta(n_1, n_2) \tag{1}$$

其中，N_1 和 N_2 分别是树 T_1 和 T_2 中节点的集合，$\Delta(n_1, n_2)$表示以 n_1 和 n_2 为根节点的相同子树的数量。

然而，标准树核函数无法刻画语义信息，如图 1 中，buy 和 acquire 存在上下位关系，反应在树核函数中则是两棵不同的子树。为此，Mehdad 等[12]提出了句法语义树核来考察那些叶结点上相似（如同义关系、上下位关系）的术语，试图解决因部分叶子结点的不同而导致的整个子树不匹配的问题。受此启发，我们也将语义信息融入树核中，以提高子树匹配的效果。与 Mehdad 等的方法不同的是，我们不仅考虑了词义相似性，也考虑了语义相似而结构不同的蕴含关系。具体而言，Mehdad 等的方法仅考虑词或术语的相似性，而我们的方法还考虑了具有蕴含关系的短语。例如，对于以下例子：

T2：Yahoo's acquisition of Overture

H2：Yahoo acquired Overture

Mehdad 等的方法无法得到 T2 和 H2 存在语义相同的子树，而事实上 X's acquisition of Y 蕴含了 X acquire Y，通过对短语所在子树进行匹配，我们的树核函数可以得到 T2 与 H2 存在语义相同子树的结论。

本文采用的语义树核函数的定义如下：

$$\Delta(n_1, n_2) = \lambda K_P(c_{n_1}, c_{n_2}) K_W(ch_{n_1}, ch_{n_2}) \qquad (n_1 \neq n_2) \tag{2}$$

其中，K_P 和 K_W 为短语语义树核和词汇语义树核，c_{n_1}和c_{n_2}分别为 n_1 和 n_2 的非叶节点，ch_{n_1}和ch_{n_2}分别为 n_1 和 n_2 的叶节点，λ为衰退因子。K_P 考察短语的蕴含关系，若 n_1 和 n_2 的结构具有蕴含关系，则 K_P 为 n1 和 n2 的全部子树（除叶结点外）个数的乘积，否则，K_P 为 0。K_W 考察词义蕴含关系，若叶节点上的词汇的词义完全相同或具有蕴含关系，K_W 则为 1，否则为 0。对于词义蕴含关系，可以采用如 WordNet 等词义资源，考察两个词是否具有上下位或蕴含等关系；对于短语蕴含关系，可以利用蕴含规则库和外部知识，如 DIRT 等，考察两个文本片断是否存在蕴含规则。

公式(2)修改了标准树核函数中$n_1 \neq n_2$情况下的计算方法。对于其它情况，仍可按照标准树核函数定义进行计算。

3. 2 训练过程

利用 2.1 节给出的两种视图，我们可以分别构造两个特征集，用于训练两个分类器。正例和反例则分别是具有和不具有蕴含关系的文本-假设对，这些数据可以从 RTE 评测数据集中挑选。算法首先从未标注数据集中选择一个子集，然后分别利用两个分类器进行分类，得到一种类别划分；然后，从分类结果中选择 k 个最优的标注结果（包括正例和反例），加入到初始标注集中，并且从未标注集中移除。该过程反复进行，直到达到停止条件。这里的停止条件可以设定为迭代次数。

除了两个视图下的分类器，我们还需要训练一个综合分类器，以整合全部未标记数据的训练结果，并对新数据进行预测。综合分类器将两种核函数进行混合，公式如下：

$$K_{hybrid} = \alpha K_{poly} + (1 - \alpha) K_{tree} \tag{3}$$

其中，K_{poly}为多项式核，K_{tree}为树核，α为混合系数。于核函数在线性运算下是封闭的，因此混合核函数也满足 Mercer 条件。

3. 3 分类特征

基于树核的分类器只需选取适当的句法子树，因此无需细致的特征构建和选择工作。而基于多项式核的分类器主要通过扁平特征进行学习（如字串重叠特征和词义相似度特征），为了增强分类器的学习能力，我们引入了句法和语义相似性特征。

句法相似度的算法如下：首先，统计文本 T 和 H 中的依存关系对，用 S_T 和 S_H 分别表示 T 和 H 的依存子树集合；然后，对每一 $p_h \in S_H$，计算 p_h 与每一 $p_t \in S_T$ 的相似度，并将最大的相似度值作为 p_h 的相似度；最后，计算 T 与 H 的整体相似度。公式表示如下：

$$Sim_{syndep}(T, H) = \frac{\sum_{p_h \in S_H} max_{p_t \in S_T}\{sim_p(p_t, p_h)\}}{|S_H|} \tag{4}$$

其中，$sim_p(p_t, p_h)$为依存子树 p_t 和 p_h 的相似度。一种计算方法是，相似度为依存关系对是否完全匹配的布尔值。然而当叶结点为同义词时，这种匹配将失败。为此，我们利用 WordNet 计算 p_t 和 p_h 中的词汇相似度，公式为：

$$sim_p(p_t, p_h) = \frac{max\{sim_w(t_1, t_1') + sim_w(t_2, t_2'),\ sim_w(t_1, t_2') + sim_w(t_2, t_1')\}}{2} \tag{5}$$

其中，$t_1, t_2 \in p_t$，$t_1', t_2' \in p_h$。WordNet 词义相似度采用 Wu-Palmer 相似度[13]进行计算。

语义相似度考察谓词论元结构的相似性，计算方法与句法相似度类似，只不过将句法依存子树变成语义依存子树。

4 实验结果及分析

实验语料来自 RTE-5，其中 RTE-5 中训练集和测试集分别为 600 个文本-假设对，训练集和测试集中蕴含和不蕴含的样本个数分别为 300。实验的评测指标采用正确率（正确的评价结果占总测试样本个数的比值）、准确率、召回率和 F 值作为评测指标。各参数根据最优实验结果进行设置，其中衰退因子λ取 0.4，混合系数α取 0.6。

词义蕴含关系利用 WordNet 进行计算，具体方法是：若两个词在 WordNet 中存在上下位关系或蕴含关系，则认为两词具有蕴含关系，否则不具有蕴含关系。为获取用于判断结构蕴含关系所需的知识，本文利用 DIRT 复述库，用于搜索两个子树是否存在蕴含转换规则。

第一个实验评测了基于协同训练的文本蕴含识别系统的性能。协同训练分类器 Co-training 采用第 3 章提出的方法进行协同训练。SVM1、SVM2 和 SVM3 分别采用混合核、多项式核和树核作为分类器的核函数。

标注数据随机从训练样本中选取 50%，即 300 个样本作为训练数据，并保证蕴含和非蕴含的样本数大致相同，其余样本则作为未标注数据。测试样本为全部测试集合。为减少随机数据选择的影响，每次样本选择过程都独立进行 10 次，然后进行学习，最后各评测值取 10 次各评测值的平均值。协同训练的迭代次数设为 30 次。实验结果如表 1 所示。

		Co-training	SVM1	SVM2	SVM3
正确率		**0.3763**	0.3385	0.3118	0.2325
蕴含	P	**0.3515**	0.313	0.3016	0.2207
	R	**0.4429**	0.4024	0.3522	0.2746
	F	**0.3919**	0.3521	0.3249	0.2447
非蕴含	P	**0.3858**	0.3496	0.3203	0.2495
	R	**0.2873**	0.2675	0.2342	0.1722
	F	**0.3293**	0.3031	0.2706	0.2038

表 1 协同训练分类器、混合核分类器和单核分类器的分类结果

实验结果显示，基于协同训练的分类器的正确率比混合核分类器 SVM1 高出 3.78%，比单核分类器 SVM2 和 SVM3 分别高出 6.45%和 11.41%。就准确率、召回率和 F 值而言，基于协同训练的分类器的性能比其它分类器也有不同程度的提高。这表明，基于协同训练的分类器能在蕴含数据不足的情况下获得更好的识别性能。

其次，混合核分类器 SVM1 的正确率比单核分类器 SVM2 和 SVM3 分别高出 2.67%和 10.6%，蕴含和非蕴含两类的准确率和召回率也高于单核分类器。另一方面，采用统计特征的分类器 SVM2 的正确率比基于树核的分类器 SVM3 高出 7.93%，两类的准确率和召回率也比 SVM3 大辐提高。这表明：1）蕴含关系的类别丰富，一些蕴含关系可以由统计特征表现出来，另一些则体现在结构信息中，即蕴含关系需要从统计信息和结构信息进行综合评价，因此综合考察了蕴含数据的结构和非结构特征的混合核分类器能够获得更好的性能。2）基于统计特征的分类器 SVM2，其性能相比基于树核的分类器 SVM3 有较大提高，其原因在于，文本与假设里往往存在大量的字串和词汇重叠，对于那些具有蕴含关系而句式结构不一致的文本对，即使难以找出蕴含结构，但其中大多数字串仍然是相同的，因此基于非结构特征的分类器仍然可以得到正确结果；但对于基于树核的分类器而言，一旦难以找出蕴含结构，分类器就可能给出错误的判断。因此，采用统计特征对实验数据进行分类识别更有效。

从实验数据中还可以看出，各分类器的蕴含类的准确率都低于召回率，而非蕴含类的准确率都高于召回率。显然，各分类器在将蕴含样本判断为蕴含关系的同时，也将不少非蕴含样本判断为蕴含关系。事实上，蕴含关系不仅体现在字串重叠、词义相似和短语结构上的蕴

含，还包括数量、地理信息、背景知识等等。因此，不论是统计特征还是结构信息，都难以应对比较复杂的蕴含关系。为此，需要对数据进行更多的预处理，以使待分类的数据更易于学习。

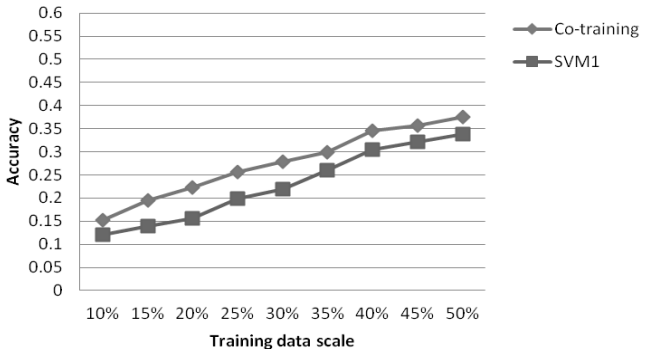

图 2 不同训练数据规模下的 Co-training 和 SVM1 的正确率

第二个实验考察不同训练数据规模下基于协同分类的分类器和混合核分类器的分类性能。本实验设置与第一个实验的设置基本相同，区别仅在于所选取的训练样本占样本总数的比例从 10%到 50%，以 5%的比例递增，剩下的则作为未标记样本。实验结果如图 2 所示。

实验结果显示，在训练集不足的情况下，基于协同训练的分类器 Co-training 的性能明显优于基于有监督的分类器 SVM1。事实上，由于蕴含现象庞杂，而人工标注的训练数据非常有限，因此往往难以满足训练要求。而基于协同训练的方法能够利用现有未标注数据进行自学习，在一定程度上克服数据不足的问题，因此更适合于文本蕴涵识别。

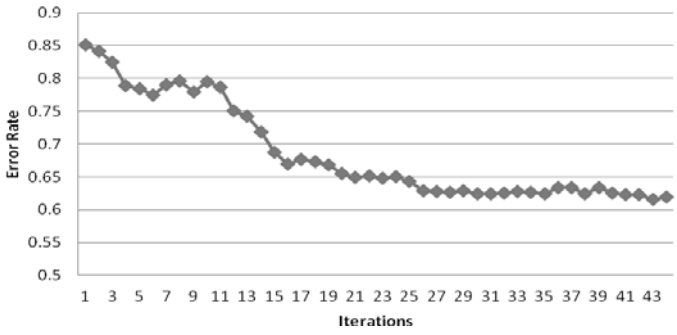

图 3 不同迭代次数的协同训练算法性能

第三个实验评估协同训练算法的迭代次数对性能的影响。在实验中，标注数据的比例设为 50%。与第一个实验相同，标注数据和未标记数据随机从训练样本中选取，并保证正例和负例的样本数大致相同，每次样本选择过程都独立进行 10 次，整体的正确率取 10 次正确率的平均值。实验结果如图 3 所示。

实验结果显示，迭代次数在 25 至 35 次之间时，系统的性能相对稳定；当迭代次数超过 35 次时，系统的错误率开始不稳定，这种情况是由于噪音的累积而造成的。虽然我们看到，迭代次数达到 40 次时，错误率有所下降，但总体上看，随着噪音不断累积，其负作用会越来越大。因此，我们需要选择一个相对稳定的迭代次数。在本实验中，30 次迭代可以将错误率维持在较低的水平。

5 结论

针对文本蕴涵的训练数据不足的问题，本文提出了基于协同训练的文本蕴含识别方法，利用少量已标注的蕴含数据和大量未标注数据进行半监督学习。为满足协同训练算法的学习条件，本文分别采用改写视图和评估视图来考察结构信息和统计信息。针对改写视图的分类器，本文提出一种语义核函数，同时考察词汇和短语的语义蕴含关系。实验表明，相比有监督的学习方法，基于半监督的协同训练方法能让蕴含识别系统在数据不足的情况下获得更好的识别性能。同时，蕴含关系需要从统计上的相似度和结构上的蕴含关系进行综合评价，因此综合考察了结构和非结构特征的混合核分类器能够获得更好的性能。

另一方面，蕴含关系非常庞杂，而统计上的相似度和结构上的蕴含关系仅是蕴含关系的两种表现形式。因此，不论是统计特征还是结构信息，都难以应对比较复杂的蕴含关系。为此，需要对数据进行更多的预处理，以使待分类的数据更易于学习。

参考文献

[1] Shachar Mirkin, Roy Bar-Haim, Jonathan Berant, Ido Dagan, Eyal Shnarch, Asher Stern and Idan Szpektor. Bar-Ilan University's Submissions to RTE-5[C]//In Proceedings of The Text Analysis Conference 2009. Gaithersburg, Maryland, USA, 2009.

[2] Han Ren, Donghong Ji and Jing Wan. WHU at TAC 2009: A Tri-categorization Approach to Textual Entailment Recognition[C]//In Proceedings of Text Analysis Conference 2009. Gaithersburg, Maryland, USA, 2009.

[3] Mark Sammons, V.G.Vinod Vydiswaran, Tim vieira, Nikhil Johri, Ming-Wei Chang, Dan Goldwasser, vivek Srikumar, Gourab Kundu, Yuancheng Tu, Kevin Small, Joshua Rule, Quang Do and Dan Roth. Relation Alignment for Textual Entailment Recognition[C]//In Proceedings of the Text Analysis Conference 2009. Gaithersburg, Maryland, USA, 2009.

[4] Alicia Ageno, David Farwell, Daniel Ferres, Fermin Cruz, Horacio Rodriguez and Jordi Turmo. TALP at TAC 20008: A Semantic Approach to Recognizing Textual Entailment[C]//In Proceedings of the 4th PASCAL Challenges Workshop on Recognizing Textual Entailment. Gaithersburg, Maryland, USA, 2008.

[5] Eugene Agichtein, Walt Askew and Yandong Liu. Combining Lexical, Syntactic, and Semantic Evidence for Textual Entailment Classification[C]//In Proceedings of the 4th PASCAL Challenges Workshop on Recognizing Textual Entailment. Gaithersburg, Maryland, USA, 2008.

[6] Fabio Massimo Zanzotto. PeMoZa submission to TAC 2008[C]//In Proceedings of the 4th PASCAL Challenges Workshop on Recognizing Textual Entailment. Gaithersburg, Maryland, USA, 2008.

[7] Fabio Massimo Zanzotto and Marco Pennacchiotti. Expanding Textual Entailment Corpora from Wikipedia using Co-training[C]//In Proceedings of the COLING-Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources. Beijing, China, 2010.

[8] Prodromos Malakasiotis and Ion androutsopoulos. Learning Textual Entailment using SVMs and String Similarity Measures[C]//In Proceedings of the The ACL-PASCAL Workshop on Textual Entailment and Paraphrasing. Prague, Czech, 2007.

[9] Miguel Angel Ríos Gaona, Alexander Gelbukh and Sivaji Bandyopadhyay. Recognizing Textual Entailment Using a Machine Learning Approach[C]//In Proceedings of the 9th Mexican International Conference on Artificial Intelligence Conference on Advances in Soft Computing: Part II, Pachuca, Mexico, 2010.

[10] Roy Bar-Haim, Jonathan Berant, Ido Dagan, Iddo Greental, Shachar Mirkin, Eval Shnarch and Idan Szpektor. Efficient Semantic Deduction and Approximate Matching over Compact Parse Forests[C]//In Proceedings of the 4th PASCAL Challenges Workshop on Recognizing Textual Entailment. Gaithersburg, Maryland, USA, 2008.

[11] Alvaro Rodrigo, Anselmo Penas and Felisa Verdejo. Towards an Entity-based Recognition of Textual Entailment[C]//In Proceedings of the 4th PASCAL Challenges Workshop on Recognizing Textual Entailment. Gaithersburg, Maryland, USA, 2008.

[12] Yashar Mehdad, Alessandro Moschitti and Fabio Massiomo Zanzotto. SemKer: Syntactic/Semantic Kernels for Recognizing Textual Entailment[C]//In Proceedings of the Text Analysis Conference 2009. Gaithersburg, Maryland, USA, 2009.

[13] Zhibiao Wu and Martha Palmer. Verb Semantics and Lexical Selection[C]//In Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics. Las Cruces, New Mexico, 1994.