基于最大熵模型的介词纠错系统①

李 悦[1], 吴 敏[1], 吴桂兴[2], 郭 燕[2]

[1](中国科学技术大学 现代教育技术中心, 合肥 2300260)
[2](中国科学技术大学 苏州研究院, 苏州 235123)

摘 要: 英语介词纠错系统, 针对英语学习者英语语言中常见的介词错误进行计算机自动纠正. 首先, 对标注过得语料库中介词错误进行了分类统计, 总结出 21 种常见介词, 在英语 wiki 语料库中利用计算机自动错误插值算法获得训练集合. 然后在训练集合基础之上, 通过使用基于最大熵模型的分类器, 选择了包括上下文、介词补足语等特征, 在训练集上进行模型的训练, 最后使用模型对于输入句子进行预测并纠正存在的使用错误. 在 NUCLE 语料的实验中, 给出了语料处理、模型特点、训练语料的大小、迭代次数对于测试集效果的影响, 并且比较了朴素贝叶斯模型的结果, 最后在测试数据达到 27.68 的 F 值, 相对于 CoNLL2013 的 shared task 中最好结果有小幅提升.

关键词: 介词错误; 计算机自动纠正; 最大熵模型

Preposition Error Correction System Based on Maximun Entropy Model

LI Yue[1], WU Min[1], WU Gui-Xing[2], GUO Yan[2]

[1](Center of Modern Educational Technology, University of Science and Technology of China, Hefei 230026, China)
[2](Suzhou Institute, University of Science and Technology of China, Suzhou 235123, China)

Abstract: English preposition error correction system is to help English language learners to correct automatically the common mistakes of English prepositions. First, to classsify and count up the preposition errors in the marked corpus files, sum up 21 kinds of common prepositions, with English wiki corpus and use of computer algorithms automatically error interpolation algorithm to get the training set, and then based on the training set, by using a classification based on the maximum entropy model chosen, including context, prepositions complement other features, training model on the training set, and then use the model to predict the input sentence and correct use of the presence of errors. In NUCLE corpus experiment, given corpus processing, model features, size, number of iterations to test the effect of the impact of training data set, and compare the results of the Naive Bayes model, and finally to the F value 27.68 in the test data with respect to the shared task CoNLL2013 best results have slightly improved.

Key words: preposition errors; computer automation correction; maximum entropy model

语法检查一直是自然语言处理领域研究的热门问题, 随着机器学习领域的发展, 利用机器学习算法来解决传统的语法检查问题已成为提高目前语法检查系统的主要方法. 在英语学习使用过程中, 介词无疑是一大难点. 虽然介词的数量不多, 但介词是英语语言中最为活跃的词汇. 在一定程度上讲, 英语就是介词的语言[1]. 因此, 解决介词纠错成为了英语纠错系统中重要的环节. 介词之所以难以掌握, 一方面由于多数介词都有几种意义. 另一方面, 介词可以和多种词类进行搭配. 利用自然语言处理技术和机器学习算法, Alla Rozovskaya, Kai-Wei Chang 等[2]使用基于平均感知器[3]和 Naïve Bayes 的组合来对介词错误进行纠错. 本文尝试发觉介词特征用于表述介词的使用情况, 利用最大熵模型学习并训练可用模型, 对介词错误进行

① 收稿时间:2015-05-05:收到修改稿时间:2015-06-08

计算机自动纠正.

本文组织结构如下, 第一节介绍介词错误纠正的相关研究, 第二节介绍最大熵模型及处理过程, 第三节介绍系统模块以及系统整个流程, 第四节给出在测试集语料上的实验结果并进行相关分析, 最后进行总结和展望.

1 问题定义及模型的建立

英语语句中的介词错误可以归纳为三种.

①介词多余,如“Despite of/NULL being tiring, it was rewarding”这里应省略掉介词 of.

②介词缺失, 如“I will be waiting NULL/for your call”丢失介词 for.

③介词错用, 如“I can see at/on the list a lot of interesting sports”在这里应该使用 on 代替 at.

三种错误在语料库中所占介词总体错误比例, 如表 1 所示.

表 1 介词三种错误比例

类型	介词多余	介词缺失	介词错用
错误比例(%)	18.1	24.0	57.9

在介词缺失的问题中, 介词后面往往接名词或者名词短语, 系统将对语料库中每个名词或者名词短语进行考察, 判断是否有介词缺失的情况. 另外, 由于英语语法中介词使用方法较为多样, 介词可以作为动词不定式标记, 例如“He wants NULL/to go there”, 本系统将这类介词缺失定义为动词错误, 所以本系统的纠错不包括这类错误. 另外, 英语介词还可以作为表语使用, 如“The matter is of great importance”, 这里 of 的使用也归纳到动词错误中, 不考虑其缺失的情况, 但会考察其错用的情况.

通过对语料库中, 介词出现次数统计获取了 21 种常用的介词, 如表 2 所示.

表 2 NUCLE 语料库介词数量统计

	of	to	in	for	at	on
数量	13920	13903	7046	4083	3621	2979

表 2 列举了最常用的 7 种介词, 此外还有'by', 'from', 'into', 'as', 'about', 'towards', 'through', 'between', 'under', 'during', 'over', 'upon', 'among', 'within', 此外还有不少数量的介词, 但是由于出现频率极低, 没有归纳在系统之中.

将介词多余, 介词缺失与介词错用归等效考虑为一个分类问题进行处理. 定义 21 种常用介词与 NULL 为一个集合, 不同的错误对应集合里单词的映射. 那么, 介词的纠错就可以通过提取介词的上下文等特征, 根据特征利用分类器进行分类来判断当前的映射类型, 从而实现纠错.

1.1 最大熵模型

最大熵分类器[4]模型是朴素贝叶斯分类器模型的泛化模型. 像朴素贝叶斯模型一样, 最大熵分类器通过将适合于输入值和标签的参数乘在一起, 为给定的输入值计算每个标签的可能性. 最大熵分类器模型为每个标签定义一个参数, 指定其先验概率, 为每个(特征, 标签)对定义一个参数, 为标签的似然性指定其独立特征的贡献.

特别的, 它查出能使训练语料的整体似然性最大的参数组. 其定义如下:

$$P(features)=\sum_{x\in corpus}P(label(x)\mid features(x))$$

其中 $P(label\mid features)$, 特征为 features 的输入, 并且类标签为 label 的概率, 被定义为:

$$P(label\mid features)=P(label,features)/\sum_{label}P(label,features)$$

在介词问题中, 将一系列可能影响冠词使用判断的因素, 表示为一个特征向量 X, 那么就要找到使得 p(冠词|X)达到最大的那个冠词, 在最大熵模型中要求 p(冠词|X)满足一定约束条件的情况下, 使得上式定义的概率熵取得最大值.

由于模型在参数优化上计算复杂度较高, 没有办法直接计算能最大限度地提高训练可能性的模型参数. 因此, 最大熵分类器采用迭代优化技术选择模型参数. 该技术利用随机值初始化模型的参数, 然后反复优化这些参数, 使他们更接近最优解. 这些迭代优化技术保证每次参加优化都会使它们更接近最佳值, 但不一定能够提供方法来确定是否已经达到最佳值. 在参数优化过程中, 通常花费的时间都较长. 当训练集的大小、特征的数目及标签的数目都对模型训练的时间有着显著的影响.

目前关于迭代优化的技术有很多, 常见的有广义迭代缩放(Generalized Iterative Scaling, GIS)或者改进的迭代缩放(Improved Iterative Scaling, IIS), 这两种方法相对于共轭梯度(Conjugate Gradient, CG)和 BFGS 优化要慢很多. 由于最大熵模型的迭代优化问题不是本文研究的重点, 故不做详细论述. 在这里指出的是,

本文使用的是 IIS 与 BFGS 两种迭代方法.

1.2 错误插值算法

在实际模型训练过程中, 标注语料库错误数量往往少于正确使用的数量, 不少语料库未能被人工标注缺少错误实例, 直接进行训练存在正负样本均衡的问题, 负样本远远小于正样本, 以至于模型很难获得很好的效果.

造成这种训练效果不佳的主要原因在于较低的召回率. 通常情况下有两种较为极端的方式来解决训练过程中低召回率的情况: 一种是在训练过程中, 原文介词不作为特征引入训练; 另一种方式是, 保留原文介词特征, 并使得分类器不过分依赖于该特征. 第一种方法往往会造成准确率大幅下降. 使用最大熵模型训练的结果, 往往有不错的召回率, 系统需要尽可能不降低准确率, 同时提高召唤率. 因此, 系统选择保留原文介词特征, 并引入错误插值算法提高召回率, 从而提高整体纠错效果.

错误插值算法针对语料库进行处理, 根据已标注的语料库获取错误类型, 以及错误所占比例插入错误用例. 这种做法的目的, 在于减少模型在训练过程中对原始 介词特征的依赖, 从而更加依赖于介词的上下文环境. 为了实现这个目标, 在错误插值算法中, 错误的出现模仿真实获取的错误. 基于 NUCLE 语料库, 表格 3 给出介词错误的统计, 这里只给出了四行, 实际表格为 22 行列的矩阵. 针对每一个介词, 获取原文介词使用正确的概率.

表 3　介词错误矩阵

	介词标签						
	of	in	on	to	for	with	by
of	0.967	0.091	0.007	0.002	0.023	0.005	0.003
in	0.004	0.947	0.022	0.002	0.006	0.006	0.008
on	0.002	0.009	0.903	0.002	0.007	0.007	0.002
to	0.002	0.004	0.129	0.979	0.153	0.008	0.003

根据表格 3 所获得的矩阵, 使用错误插值算法对语料库进行错误插入. 具体算法描述如下:

算法 1　错误插值算法.

输入: 标注过的训练样本 E, 介词错误矩阵 CM, 系数 C

输出: 错误扩充后训练样本 E

```
for Example e in E do
  初始化 lab←e.label,e.source←e.label
  随机化 targets in CM[lab]
  初始化 flag←False
  for target t in targets do
    if flag equals True then
      Break
    end if
    if t equals lab then
      Prob(t) = CM[lab][t]·C
    else
    end if
    x←Random[0,1]
    Prob(t) = (1.0−CM[lab][lab]·C)/(1.0−CM[lab][lab]) · CM[lab][t]
    if x<Prob(t) then
      e.source←t
      flag←True
    end if
  end for
end for
return E
```

使用系数 C<1.0, 修改系数 C 调整差错比率. 扩充后, 错误数量如表 4 所示.

表 4　NUCLE 语料库错误数量

1.0	0.9	0.8	0.7	0.6	0.5
3332	3890	13716	19445	25334	31637

1.3 实验所用语料

实验采用 NUCLE-release2.2[5]语料, 语料包括共 28 种错误类型, 其中 prep 表示介词错误, 由人工收集并标注. 由于本文研究限于介词用法, 我们在使用该语料前, 将除介词以外错误预先纠正.

语料库包括了训练集和测试集, 在预处理过程中, 纠正非介词错误, 将语句转换为小写, 并去掉了过长语句. 语料库训练集与测试集中, 语句数量、分词数量以及介词错误数量统计如下表 5 所示.

表 5　NUCLE 语料库统计

	训练集	测试集
语句数量	57151	1381
分词数量	1161567	29207
Prep 错误数	133665	3332

2　系统设计与实现

2.1 系统模块

语料预处理: 对每一个句子, 将其转为小写, 进行

拼写检查以及合法检查, 过滤掉 tokens 大于 100 的过长句子. 对输入句子使用 Stanford parser[6]进行词性标注与语块划分. 例如: My dog also likes eating sausage.标注为:

```
(ROOT
  (S
    (NP (PRP$ My) (NN dog))
    (ADVP (RB also))
    (VP (VBZ likes)
      (S
        (VP (VBG eating)
          (NP (NN sausage)))))
    (. .)))
```

由于英语中, 介词往往出现在名词或者名词短语之后, 可以通过寻找名词(NN)或者名词短语(NP), 找到可能缺失介词的位置.

同时查找语句中存在的相关介词, 将这些包涵介词的语句以及可能存在介词缺失的语句, 进行标注用于下一步的处理.

特征提取模块: 特征上全部选择词汇作为特征, 主要包括 4-word window 的 N-gram 特征, 以及三个关于介词补语的特征. 详细特征如表 6 所示.

表 6 介词模型所用到的特征

特征类型	特征描述
Word n-gram	wB, w2B, w3B, wA, w2A, w3A, wBwA, w2BwB, wAw2A, w3Bw2BwB, w2BwBwA, wBwAw2A, wAw2Aw3A, w4Bw3Bw2BwB, w3Bw2BwBwA, w2BwBwAw2A, wBwAw2Aw3A, wAw2Aw3Aw4A
介词补语	prepcomp,wB&prepcomp,w2BwB&prepcomp

其中, wB 表示介词前一个词汇, w2B 表示介词前第二个词汇, wA 表示介词后一个词汇, prepcomp 表示介词补语(preposition complement).

模型训练模块 : 训练集提取特征, 模型训练模块负责通过特征训练分类器模型, 根据特征提取的结果以及原句介词(如果为缺失用 NULL 标示)作为模型训练的输入特征, 用标注的结果作为训练目标, 进行模型训练. 这里主要使用的是最大熵模型.

系统纠正模块: 在训练集中进行最大熵模型的训练. 对于测试集, 使用最大熵模型进行预测, 若预测结果与原结果一致, 则不进行纠正, 若预测结果与原结果不同, 则纠正模型预测结果. 根据模型预测的结果, 判断纠错.

整体系统实现了计算机自动化, 先利用训练集获得模型, 对输入的英语语句, 通过预处理、提取特征、模型预测、系统纠正, 来实现介词的纠错.

2.2 系统结构流程图

根据系统设计, 系统分为模型训练与模型预测两大部分, 模型训练针对于标注过的训练集, 英语语句的输入通过模型预测进行介词错误纠正. 模型英语介词错误纠正系统流程图如图 1 所示.

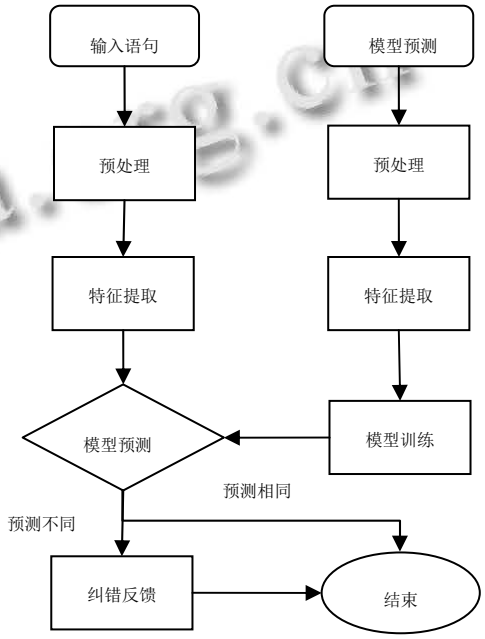

图 1 介词纠错系统的工作流程

3 实验结果及分析

实验评价标准, 通过语料库提供的脚本, 计算纠错结果的正确率、召回率以及 F 值.

对语料进行预处理, 去除了包括括号和超链接的句子根据系统设计. 表 7 给出处理前后的对比.

表 7 对语料进行预处理前后的测试集结果对比

	Precision	Recall	F
预处理前	68.13	13.47	21.82
预处理后	68.87	14.20	23.54

由表 7 所示, 对语料库处理去除多余句子并不影响实验结果, 相反减少了其他因素的干扰, 提高了实验结果.

为了测试最大熵模型训练中, 迭代次数对结果的影响, 对最大熵 的迭代次数进行了调整, 取值不同的迭代次数进行实验, 其结果如表 8 所示.

表 8 不同迭代次数下 F 值的变化

迭代次数	1	5	10	100
F	23.54	23.55	23.73	23.89

迭代次数的增加, 可以提高系统的 F 值. 由于每次迭代都要耗费大量的时间, 故后面的实验都统一迭代 1 次.

实验选取了出现次数最多的 21 种介词作为纠错目标, 为了测试不同数量下纠错效果的变化, 调整选取介词的数量, 按出现频率由高到底, 选取不同数量的介词进行测试, 结果如表 9 所示.

表 9 介词数量对纠错系统性能的影响

数量	10	15	20	21	25
F	23.31	23.37	23.40	23.41	22.41

通过介词数量对纠错系统性能的影响进行分析, 由于排名靠前的介词比重较大, 故介词种类数量对 F 值影响有限. 但是如果介词数量过少, 会导致相应的错误无法被检测, 过多则会造成错误样本的进一步稀疏, 通过实验验证选取 21 种介词时 F 值相对较高.

为了测试不同 C 取值情况下, 错误插值算法的引入对实验的影响, 对 C 取不同数值进行实验, 其结果如表 10 所示.

表 10 不同 C 取值下 F 值的变化

	0.98	0.95	0.9	0.8	0.7
F	25.27	26.74	23.47	18.54	10.87

由表 10 所示, C 在取值 0.95 时 F 值最高. 这一结果符合错误插入算法的特点: 插入过多的错误会导致正确率急剧下跌, 而插入过少会导致正确率提升有限.

对 C 取值 0.95, 对 Wiki 语料库进行错误填充并将其结果并入到训练集当中, 其结果如表 11 所示.

表 11 扩充训练集后 F 值的变化

	F
UNCLE	23.54
UNCLE+Wiki(10w 条)	24.57
UNCLE+Wiki(20w 条)	27.68

由表 11 所示, 训练集的扩大对结果有所提升.

表 12 朴素贝叶斯模型, 最大熵模型与 shared task 最好结果对比

	F
NB-priors	18.87
Maxent	27.68
Shared task	27.51

朴素贝叶斯模型是一种常用的分类模型, 我们用贝叶斯模型与最大熵模型做了对比, 并将结果与 CoNLL2013 shared task[7]最好的结果进行了对比, 其结果如表 12 所示.

4 结语

在本文中, 通过运用最大熵模型, 利用提取出的可表述介词特性的上下文特征训练模型, 并对英语语句中出现的介词错误进行纠正, 并深入探讨了模型参数、语料以及特征对于模型结果的影响, 并且引入了错误插值算法解决正负样本不均衡问题, 最终模型取得 27.68%的 F 值. 在实验过程中, 模型还是发生很多误判, 因此应该更深入的研究基本特征对于介词选择的影响, 以及一些基于语义上的特征,另外介词的一些使用还受到某些规则的约束, 可以考虑和规则系统的融合. 在后续的研究中, 还可用此类方法对于其他语法错误类型进行纠正.

参考文献

1 祝德勤.英语介词.北京:商务出版社,2004.
2 Rozovskaya A, Sammons M, Roth D. The UI system in the HOO 2012 shared task on error correction. Proc. of the Seventh Workshop on Building Educational Applications Using NLP. Association for Computational Linguistics. 2012. 272–280.
3 Freund Y, Schapire RE. Large margin classification using the perceptron algorithm. Machine Learning, 1999, 37(3): 277–296.
4 Wang P, Jia Z, Zhao H. Grammatical error detection and correction using a single maximum entropy model. CoNLL-2014. 2014. 74.
5 Berger AL, Della Pietra SA, Della PVJ. A maximum entropy approach to natural language processing. Computational Linguistics, 1996, 22(1): 39–71.
6 Dahlmeier D, Ng HT, Wu SM. Building a large annotated corpus of learner English: the NUS corpus of learner English. Proc. of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications. 2013. 22–31.
7 Klein D, Manning CD. Accurate unlexicalized parsing. Proc. of the 41st Annual Meeting on Association for Computational Linguistics-V1. Association for Computational Linguistics, 2003. 423–430.
8 Ng HT, Wu SM, Wu Y, et al. The conll-2013 shared task on grammatical error correction. Proc. of CoNLL. 2013.