计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2017,26(9):145−150 [doi: 10.15888/j.cnki.csa.005951]
©中国科学院软件研究所版权所有.
E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于层次语言模型的英语动名词搭配纠错策略①

李灿润[1], 吴桂兴[2], 吴　敏[1]

[1](中国科学技术大学 现代教育技术中心, 合肥 230026)
[2](中国科学技术大学 苏州研究院, 苏州 235123)

摘　要: 搭配的正确使用是区分地道英语使用者和普通学习者的一个重要特征. 通过分析中国英语学习者语料库(CLEC), 可以发现动名词搭配错误是英语学习者易犯的错误. 本文提出一种可用于纠正英语学习者动名词搭配错误的层次语言模型. 该语言模型考虑到了句子内部词语之间的依赖关系, 将句子处理为不同的层次的子句, 同一个句子内部的单词高度相关, 不同子句内的单词相关性弱. 该语言模型对于句子成分的变化得到的结果更加稳定, 而且搭配信息得到浓缩, 得到的语言模型更加精确. 本文将模型用于生成分类器特征和结果排序. 这种层次语言模型应用到英语动名词搭配的检错纠错中, 对比传统语言模型, 会有更好的效果.

关键词: 动名词搭配错误; 层次语言模型; 自动纠错策略

引用格式: 李灿润,吴桂兴,吴敏.基于层次语言模型的英语动名词搭配纠错策略.计算机系统应用,2017,26(9):145–150. http://www.c-s-a.org.cn/1003-3254/5951.html

English Verb-Noun Collocation Error Correction Strategy Based on Hierarchical Language Model

LI Can-Run[1], WU Gui-Xing[2], WU Min[1]

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

Abstract: The correct use of collocation has been widely acknowledged as an essential characteristic to distinguish native English speakers from English learners. Through the analysis of CLEC, we can find that English learners often make mistakes on verb-noun collocations. In this paper, we propose a hierarchical language model that can be used to correct verb-noun collocation errors made by English learners. The language model takes the dependencies between words within a sentence into account. It parses sentences into different levels of clauses. The words within the same clause are highly correlated, and the relevance of words in different clauses is weak. The language model is more stable. Moreover, it is more accurate because collocation information is condensed. It can be used to re-rank candidates and generate classifier features. We apply this hierarchical language model to the correction of English verb-noun collocation errors. Compared with the traditional language model, the new model has better performance.

Key words: verb-noun collocation error; hierarchical language model; automatic correction strategy

据“中国学习者语料库”的统计, “在所有的言语失误中, 搭配错误在言语失误频率表中位居第六”[1], 可见搭配是英语学习的一个难点, 其中, 动名词搭配错误是搭配错误中频率最高的一类, 所以动名词搭配的检错纠错在英语语法检错纠错中有重要意义. 本文将引入一种层次语言模型, 可用于动名词搭配计算机自动纠正中构造分类器特征和最终结果排序. 这种语言模型能克服传统 n-gram 将句子当成线性模型所带来的缺

① 基金项目: 江苏省自然科学基金面上研究项目(BK20141209); 苏州市应用基础研究项目(SYG201543)
收稿时间: 2016-12-27; 采用时间: 2017-01-18

© 中国科学院软件研究所 http://www.c-s-a.org.cn

点. 本文将为常见动名词搭配构建一个搭配库, 并为搭配库中的动名词搭配训练对应的分类器, 并将层次语言模型应用于分类结果的排序上得到最终的结果.

本文组织结构如下, 第一节介绍语言模型用于英语语法错误纠正的相关研究, 第二节介绍层次语言模型的构造过程, 第三节介绍实验的各个模块以及纠错系统流程, 第四节给出在测试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 相关研究

在最近的研究中, 使用计算机辅助帮助英文写作得到了广泛关注, 如 CoNLL 2013 和 CoNLL 2014 的 shared task. 其中有一部分提交的论文采用了传统的语言模型或者对传统的语言模型进行了改进并取得了一定的效果. Longkai Zhang 等[2]采用了传统的 n-gram 语言模型进行纠错和检错, 但是传统 n-gram 语言模型比较简单, 所以对处理结果增加了最大熵分类器对冠词和介词错误分别进行二次处理. Grigori Sidorova 和 Francisco Velasquez 等[3]采用了一种基于规则的方法, 并考虑到了句法的树状关系, 引入了句法 n-gram 语言模型, 但是模型只考虑纵向依赖关系而忽略了横向的关系, 所以会丢失一些语义信息导致模型不够精确. Yashimoto 等[4]使用了一种树状语言模型用于主谓一致错误的纠正, 但是由于这种树状模型的结点需要增加除了句中单词外的附加语法信息, 所以会引起数据稀疏的问题, 影响模型实际使用效果.

在英语搭配纠错检错方面, 杜一民等[5]尝试了用传统的 n-gram 对分类器的结果进行排序产生最终的纠错结果. 但是由于搭配关系的词汇之间的位置关系比较多变, 所以传统的 n-gram 模型存在一定的局限性. 本文考虑将句法的树状层次关系保留到语言模型中, 并同时保留不同层次纵向依赖关系也就是句法结构信息, 以及同一句法层次的横向的关系也就是语义信息, 从而建立一种层次语言模型. 采用分类器对备选结果进行筛选缩小备选集合, 最后利用该语言模型进行排序, 挑选出最合适的结果.

2 层次语言模型

2.1 传统的 n-gram 语言模型的不足之处

传统的 n-gram 语言模型将句子考虑为一个顺序的串, 其中 n 为取词窗口的大小. 以比较常用的 tri-gram[6] 为例, 此时的 n 等于 3, 模型的定义如下:

模型由语言中所有单词的集合 V 和参数 $q(w|u,v)$ 组成, $w \in V \cup \{STOP\}$, $u,v \in V \cup \{*\}$. STOP 定义为句子的结尾, *定义为句子的开头. $q(w|u,v)$为句子中的前两个单词为(u, v)的情况下, 下一个词是 w 的概率.

定义一个句子为$x_1, x_2, ...x_n$, 其长度为 n, 其中x_n恒为 STOP, 假设$x_0 = x_1 = *$, 根据二阶马尔科夫模型可得句子$x_1, x_2, ...x_n$的概率为:

$$p(x_1...x_n) = \prod_{i=1}^{n} q(x_i|x_{i-2}, x_{i-1})$$

对于概率$q(w|u,v)$的计算, 使用最大似然估计计算:

$$q(w|u,v) = \frac{c(u,v,w)}{c(u,v)}$$

其中, $c(u,v,w)$为用于训练的语料中的(u,v,w)三元组的出现次数, 而$c(u,v)$为用于训练的语料中的(u,v)二元组出现的次数. 由于在训练中可能会出现$c(u,v,w) = 0$的情况, 这将导致$q(w,|u,v) = 0$, 也就是数据稀疏问题. 对于这种问题, 需要使用参数平滑处理.

用于英语语法纠错时, 传统的 n-gram 语言模型将句子处理为线性模型. n 的大小决定了相关的单词间的最大单词跨度. 所以当 n 确定时, 对于句中单词间隔大于 n 的单词间的依赖关系将丢失; 而当 n 过大时将使得可用信息密度下降, 这将降低语言模型的准确性.

例句: I will give you an example of why I have come to that conclusion.

以句中的 give example 为目标搭配提取出的 3 元组如下:

give you an

you an example

an example of

可以看到没有办法取得一个能包含 give example 这两个目标词语的三元组, 也就是说语言模型没办法将这对搭配的信息保存下来.

但是如果将取词窗口扩大到 4, 则可以取得以下包含两个目标词汇的 4 元组:

give you an example

但是这个四元组同时包含了更多词汇, 故引入了更多噪声, 从而会使语言模型的准确性下降. 假设 give example 这对搭配在语料中出现几十次, 然而这个特定的四元组可能只出现一两次. 所以虽然长度增长能保

© 中国科学院软件研究所 http://www.c-s-a.org.cn

证取词窗口能包含对应搭配, 但是也会引起数据稀疏的问题.

2.2 层次语言模型

为了能让英语搭配词组的信息能够更有效的被描述出来, 我们考虑引入句法树描述句子中词语的依存关系, 再融合传统 n-gram 模型的优点对句法树进行描述. 也就是通过层次语言模型获得 *n* 元组, 而不是以文本中词语出现的先后顺序直接得到 *n* 元组.

我们认为句子的主干中的核心词语之间有直接的关系, 它们应该放在一起, 形成一个浓缩的子句; 而各个核心词汇的修饰成分也应该以这个核心词汇为中心放在一起, 形成一个浓缩的子句; 如果还有冗余的成分则放在下一个层次, 递推形成层次结构. 根据语法关系放在一起的词汇组成新的子句, 子句内部的词语相互约束; 下层的子句是上层的补充和修饰部分.

以“give you an example”和“give you the example”这两个短语为例, 建立句法树如图 1.

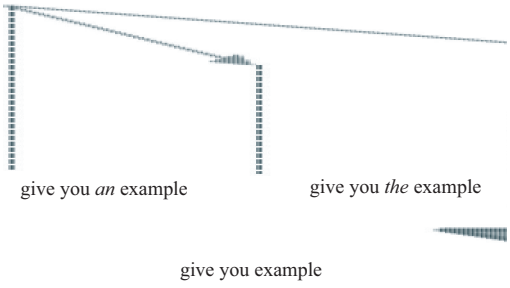

图 1 依存关系图

从图 1 可以看到这两个短语中各个词语的依存关系, 如果只保存根结点词语和根结点的子结点词语, 并且按照原来的位置关系重新抽取出新的短语, 则这两个原先不同短语都可以得到一个新的短语“give you example”, 也就是说, 主要成分被保留下来, 次要成份被忽略了.

句子表面意义上的相邻, 并不是真正语法意义上的相邻关系, 正是这种表面意义上的相邻导致了传统 n-gram 的低效. 我们如果利用依存关系就能有效获取这种语法上的相邻关系, 并把这种更深层次的相邻保存在新的子句中, 就能有效去除冗余.

接下来我们来看一个完整的例子, 由句子“I will give you an example of why I have come to that conclusion.”可以建立图 2 所示的树 .

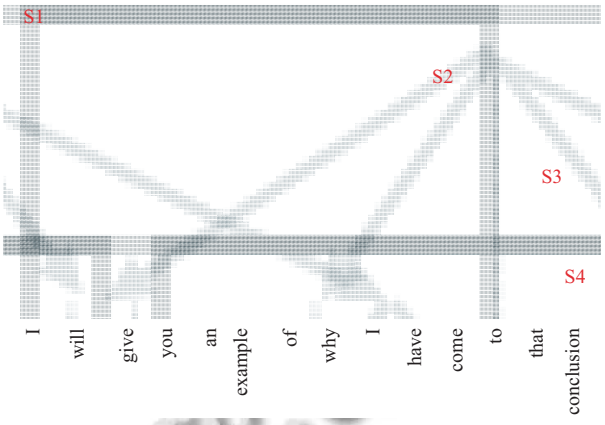

图 2 句法树例子图

然而不同于 Grigori[7]纵向由根到每个叶子结点的路径生成 *n* 元组的方法, 我们首先由这棵树横向构建子序列, 然后再生成 *n* 元组.

从句子的根结点出发, 根结点和所有的根结点的直接子结点按照原来的顺序构成一个子句, 第一层子句也是整个句子的主干, 也就是“I will give you example”, 这个子句去除了所有的子结点修饰成分, 句子内容被浓缩, 语义信息也可以保留下来; 遍历所有的根结点的直接子结点, 如果子结点还有孩子就递归地按照相同的方法建立子句, 这些低层次的子句是上层子句的修饰部分. 注意到介词不是实词, 所以处理的时候将介词和后边的依赖词当成一个整体放入介词所在的子句, 否则会产生大量两个词语的子句. 最终可以得到 4 个层次, 总共 4 个子句.

第一层子句:

S1: I will give you example.

第二层子句:

S2: an example of come.

第三层子句:

S4: why I have come to conclusion.

第四层子句:

S5: that conclusion.

考虑到传统 n-gram 易于训练, 以及有比较成熟的平滑处理方法, 本文参考传统 n-gram 的对子句进行 *n* 元组提取就可以得到最终的层次语言模型 *n* 元组.

定义层次语言模型中句子的概率为:

$$p(S)=\prod_{i=1}^{n} p(S_i)*Weight(S_i)$$

© 中国科学院软件研究所 http://www.c-s-a.org.cn

其中 S 为原句子, 下标 i 表示对应 S 分解出来的第 i 个子句, p 函数的定义采用和传统语言模型相同的方法; 其中, *Weight* 是第 i 个子句的权重, 权重跟句子所在层次相关.

采用不同的权重是考虑到不同的子句对整个句子的影响有所不同, 比如整个句子的主干对整个句子的正确程度的贡献应该比主干的修饰成分贡献还要高, 一般来说, 层次越高权重越大. 由于本文只处理动名词搭配, 经过处理之后搭配中的动词名词都在同一个子句中, 为了简化处理, 本文使用的权重函数设为常数 1.

表 1 传统语言模型及层次语言模型二元组例子

模型	示例
传统语言模型2元组	I-will, will-give, give-you, you-an, an-example, example-of, of-why, why-I, I-have, have-come, come-to, to-that, that-conclusion
层次语言模型2元组	I-will, will-give, give-you, you-example, an-example, example-of, of-come, why-I, I-have, have-come, come-to, to-conclusion, that-conclusion

从给出的 give you example 这个例子可以知道, 层次语言模型相比传统的 n-gram 在元组的表示上更加稳定. 比如 example 前插入一个形容词得到新的句子, 由于变化被隔离开, 层次语言模型只会增一个新的元组. 而使用传统的 n-gram 处理时, 插入位置附近的元组会发生改变.

可以发现, 当 n 为 2 的时候, 传统的 n-gram 和我们的层次语言模型得到的二元组相等, 而当 n 大于 2 的时候我们的层次语言模型得到的 n 元组会少于或等于传统 n-gram. 因为传统的 n-gram 只考虑对原句子相邻的词语进行组合, 而层次语言模型只对原句子中关系比较 “紧密”的词语进行组合. 比如上述的例句, 当 n 为 3 时, 传统 n-gram 得到的元组是 12, 而层次语言模型得到的元组数是 9; 当 n 为 4 时, 传统 n-gram 得到的元组是 11, 而层次语言模型得到的元组数是 6. 比较少的元组能够使模型得到简化, 易于使用, 并且在使用中能节省更多的资源. 同时, 模型也可应用于抽取分类器所使用的特征.

3 纠错流程设计

首先, 构建一个动名词搭配库; 然后根据待改句子的动名词搭配从动名词搭配库中粗选出语义较为相似的一部分搭配构成相似搭配集; 接着利用语言模型抽取待改句子上下文信息作为分类器特征, 用分类器对相似搭配集进行一轮比较细致的挑选得到候选结果集. 最后, 用语言模型对候选结果集的重排序得到最终的结果. 本文在纠错框架上参考了文献[5]的流程, 并对分类筛选和语言模型排序部分进行了改进.

提取搭配和建立依赖树的过程, 本文使用了 Stanford Parser. 文献[8]对 Stanford Parser 的可靠性进行了测试, 实验中挑选了 1000 条句子, 随机对句子中的动词或者名词修改成错误的形式, 并用 Stanford Parser 对修改后的句子进行处理, 并请专业人员进行分析, 结果显示 99% 的依赖树是可靠的. 同时, 考虑到动词的时态和名词单复数变化, 本文使用了 Stanford-CoreNlp 提供的方法还原词干.

3.1 构建搭配库以及训练和测试数据准备

构建搭配库, 首先从 CLEC 中挑选出错误标签为 cc3, 也就是有动名词搭配错误的句子, 得到 77 个错误搭配, 并由专业人员进行改正, 得到对应的 77 个改正搭配. 同时, 针对每组改正搭配从牛津搭配词典中选择 3 到 5 个相似的搭配构成改正搭配的混淆集. 这些改正搭配和相似搭配构成一个小型搭配库, 一共 230 个搭配. 针对搭配库中的每一个搭配, 再从网络语料库中挑选 50 句包含该搭配的句子, 作为每个搭配的训练集.

对改正搭配收集若干条句子, 其中的一半句子不做修改作为正测试例, 另一半句子注入该错误搭配作为测试的负测试例, 构成测试数据集.

本文使用 BNC 语料库来训练用于排序的层次语言模型和用于对比传统语言模型.

3.2 生成相似搭配集

通过对中国学习者语料库中的动名词搭配错误的分析可知[9], 动名词搭配使用错误主要是由于动词或者名词词意使用错误或者直译错误, 所以改正的搭配与原搭配具有比较相近的语义.

计算词之间的语义相似度时使用了 Jiang-Conrath[10] 估计方法. 对两组动名词搭配之间的相似度分为两个动词之间的相似度加两个名词之间的相似度的和.

纠错时, 首先利用 Stanford Parser 从待改句子中把动名词搭配抽取出来, 并与搭配库中的每个搭配对比计算出语义相似度, 得到语义相似度最高的前 15 个搭配构成相似搭配集合.

© 中国科学院软件研究所 http://www.c-s-a.org.cn

3.2 语言模型以及分类器

相似搭配集合的挑选过程并未考虑搭配出现的上下文信息, 所以需要利用分类器再对这个相似搭配集进行比较细致的筛选得到一个候选结果集, 最后再由语言模型进行最终排序. 本文分类器使用了与感知机算法具有相同算法结构的被动主动算法(PA-I)[11], PA 算法结合了感知机算法和 SVM 的优点, 学习速度快, 效果好.

本文分类器所使用的特征, 由层次语言模型中目标搭配词上下文的一元组和二元组所组成. 同时, 目标搭配周围的标点符号不会被计入 *n* 元组中. 下面是一个例子:

例句: I will give you an example of why I have come to that conclusion.

以句中的 give example 为目标搭配提取出的 n 元组特征如表 2 所示.

其中, Uni 代表一元组, Bi 代表二元组; V 代表目标搭配的动词, N 代表目标搭配的名词, L 代表目标词左边第一个词, LL 代表目标词左边第二词, 同理, R 和 RR 代表目标词右边第一个词和目标词右边第二个词; I 代表目标词在二元组的中间; Ch 表示中心词与其子结点形成的子句, 比如 ChUniVL 表示以动词为中心词的子句左侧一词.

表 2 提取出的 *n* 元组特征例子

模型	示例
传统语言模型	UniVL=will, UniVLL=I, BiVL=I will, UniVR=you, UniVRR=an, BiVR=you an, UniNL=an, UniNLL=you, BiNL=you an, UniNR=of, UniNRR=why, BiNR=of why, BiVI=will you, BiNI=an of
层次语言模型	ChUniVL=will, ChUniVLL=I, ChBiVL=I will, ChUniVR=you, ChUniVRR=example, ChBiVR=you example, ChUniNL=an, ChUniNR=of, ChBiVI=will you, ChBiNI=an of

本文使用 BNC 语料库来训练用于排序的语言模型. 根据上节层次语言模型的分析, 利用 Stanford Parser 解析语料, 建立层次语言模型. 本文排序采用的层次语言模型和用于比较对照的传统语言模型都采用 3 元语言模型. 3 元语言模型的具体实现参照伯克利大学的 n-gram 语言模型[12], 在模型训练中使用了 Kneser-Ney 平滑方法[13].

4 实验结果分析

本文通过收集包含改正搭配的句子, 再对每个搭配对应的收集到的一半的句子修改成错误的形式构成测试集. 具体的, 本文从网络语料中为每个改正搭配收集了 20 条句子, 其中 10 条句子不做修改作为正测试例, 另外 10 条句子将正确搭配改为错误搭配作为测试的负测试例, 一共 1540 条句子, 作为测试数据.

进行实验时, 本文对分类器特征分别选择了传统语言模型提取的搭配上下文信息和用层次语言模型提取的搭配上下文信息进行测试. 同时对排序部分分别采用传统语言模型和层次语言模型进行了测试.

对最终语言模型的排序结果进行评判使用的是平均倒数排名 MRR[14], MRR 是一种对排序结果进行评价的方法, 用于评估正确结果是否被包含到结果列表中, 以及正确结果在排序中有多靠前. 即: 最终排序后, 正确搭配如果排第一名则得到 1 分, 拍第二名得 0.5 分, 排第 *n* 则得 1/*n* 分, 如果正确结果没存在排序集中则得分为 0. 本文的 MRR 的最终结果为所有测试句的 MRR 的平均值.

由表 3 可以看到, 与分类器特征和排序都采用传统语言模型相比, 当分类器特征改为层次语言模型抽取的特征的时候 MRR 会有略微的提高, 而当排序使用层次语言模型时会有更大的提升. 这主要是因为排序部分的变化对 MMR 结果的影响比较直接. 当分类器特征由层次语言模型得到, 并且排序使用的语言模型也为层次语言模型的时候效果最好, 这也印证了层次语言模型相比传统语言模的优势.

表 3 不同分类器特征和排序语言模型 MRR 结果

排序	分类器特征	MRR(%)
传统语言模型	传统语言模型	76.1
传统语言模型	层次语言模型	78.7
层次语言模型	传统语言模型	82.6
层次语言模型	层次语言模型	83.9

5 总结

实验中本文建立了一个小型动名词搭配库用于搭配纠错, 并将层次语言模型应用于分类器特征选择和最终结果排序, 结果显示, 采用层次语言模型对比采用传统语言模型能取得更好的效果. 在未来的工作中, 可以尝试对语言模型以及提取分类器的特征部分做更深入的研究, 并且可以尝试将层次语言模型应用于其他

© 中国科学院软件研究所 http://www.c-s-a.org.cn

的搭配检错纠错中.

如上所述, 传统语言模型直接按照英文文本的表层含义处理文本, 忽略了文本的深层含义. 而层次语言模型考虑了英文句子中各个词之间的依赖关系, 同时参考了传统的语言模型的优点, 对子句进行建模. 这种层次语言模型的优点是当句子的变化时, 产生的结果更加稳定. 其次, 在 n 大于 2 的时候产生的 n 元组会少于传统的 n-gram, 因而得到的语言模型更加简化. 另外, 由于搭配信息得到浓缩, 模型训练中除去更多的噪声, 训练得到的语言模型将会更加精确.

参考文献

1 杨惠中, 桂诗春, 杨达复. 基于 CLEC 语料库的中国学习者英语分析. 上海: 上海外语教育出版社, 2005.

2 Zhang LK, Wang HF. A unified framework for grammar error correction. Proc. of the 18th Conference on Computational Natural Language Learning: Shared Task. Baltimore, Maryland, USA. 2014. 96–102.

3 Sidorov G, Gupta A, Tozer M, *et al*. Rule-based system for automatic grammar correction using syntactic N-grams for English language learning (L2). Proc. of the 17th Conference on Computational Natural Language Learning: Shared Task. Sofia, Bulgaria. 2013. 96–101.

4 Yoshimoto I, Kose T, Mitsuzawa K, *et al*. NAIST at 2013 CoNLL grammatical error correction shared task. Proc. of the 17th Conference on Computational Natural Language Learning: Shared Task. Sofia, Bulgaria. 2013. 26–33.

5 杜一民, 吴桂兴, 吴敏. 一种解决英语动名词搭配错误的模型. 计算机科学, 2016, 43(7): 230–233, 250. [doi: 10.11896/j.issn.1002-137X.2016.07.041]

6 Collins M. Language modeling: Course notes for NLP. Columbia: Columbia University, 2008.

7 Sidorov G, Velasquez F, Stamatatos E, *et al*. Syntactic dependency-based n-grams as classification features. Proc. of the 11th Mexican International Conference on Advances in Computational Intelligence. San Luis Potosí, Mexico. 2013. 1–11.

8 Wang LK, Wang HF. Go climb a dependency tree and correct the grammatical errors. Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing(EMNLP). Doha, Qatar. 2014. 266–277.

9 曹莉. 基于语料库的中国大学生英语四、六级考试作文中动名搭配错误分析[硕士学位论文]. 武汉: 华中科技大学, 2007.

10 Jiang JJ, Conrath DW. Semantic similarity based on corpus statistics and lexical taxonomy. Proc. of International Conference Research on Computational Linguistics (ROCLING X). Taiwan, China. 1997.

11 Crammer K, Dekel O, Keshet J, *et al*. Online passive-aggressive algorithms. The Journal of Machine Learning Research, 2006, 7(3): 551–585.

12 Pauls A, Klein D. Faster and smaller N-gram language models. Proc. of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Portland, Oregon, USA. 2011. 258–267.

13 Kneser R, Ney H. Improved backing-off for m-gram language modeling. Proc. of International Conference on Acoustics, Speech, and Signal Processing. Detroit, MI, USA. 1995. 181–184.

14 Craswell N. Mean reciprocal rank. Liu L, Özsu MT. Encyclopedia of Database Systems. Berlin, Heidelberg, Germany. Springer, 2009. 1703.

© 中国科学院软件研究所 http://www.c-s-a.org.cn