基于证据理论的单词语义相似度度量

王俊华 [1,2,3] 左祥麟 [1,2] 左万利 [1,2]

摘 要 单词语义相似度度量一直是自然语言处理领域的经典和热点问题, 其成果可对词义消歧、机器翻译、本体映射、计算语言学等应用具有重要影响. 本文通过结合证据理论和知识库, 提出一个新颖的度量单词语义相似度度量途径. 首先, 借助通用本体 WordNet 获取证据; 其次, 利用散点图分析证据的合理性; 然后, 使用统计和分段线性插值生成基本信任分配函数; 最后, 结合证据冲突处理、重要度分配和 D-S 合成规则实现信息融合获得全局基本信任分配函数, 并在此基础上量化单词语义相似度. 在数据集 R&G(65) 上, 对比本文算法评判结果与人类评判结果的相关度, 采用 5 折交叉验证对算法进行分析, 相关度达到 0.912, 比当前最优方法 P&S 高出 0.4 个百分点, 比经典算法 reLHS、distJC、simLC、simL 和 simR 高出 7 %～13 %; 在数据集 M&C(30) 和 WordSim353 上也取得了比较好的实验结果, 相关度分别为 0.915 和 0.941; 且算法的运行效率和经典算法相当. 实验结果显示使用证据理论解决单词语义相似度问题是合理有效的.

关键词 词计算, 统计学习, 证据理论, 不确定性度量

引用格式 王俊华, 左祥麟, 左万利. 基于证据理论的单词语义相似度度量. 自动化学报, 2015, **41**(6): 1173−1186

DOI 10.16383/j.aas.2015.c131141

Word Semantic Similarity Measurement Based on Evidence Theory

WANG Jun-Hua[1,2,3] ZUO Xiang-Lin[1,2] ZUO Wan-Li[1,2]

Abstract Measuring semantic similarity between words is a classical and hot problem in nature language processing, the achievement of which has great impact on many applications such as word sense disambiguation, machine translation, ontology mapping, computational linguistics, etc. This paper proposes a novel approach to measure words semantic similarity by combining evidence theory with knowledge base. Firstly, we extract evidences based on WordNet; secondly, we analyze the reasonableness of the extracted evidence using scatter plot; thirdly, we generate basic probability assignment by statistics and piecewise linear interpolation technique; fourthly, we obtain global basic probability assignment by integrating evidence conflict resolution, importance distribution, and D-S combination rules; finally, we quantify word semantic similarity. On data set R&G(65), we conducted experiment through 5-fold cross validation, and the correlation of our experimental results with human judgment was 0.912, with 0.4 % improvements over existing best practice P&S, 7 %～13 % improvements over classical methods (reLHS、distJC、simLC、simL, simR); the experimental results based on M&C(30) and WordSim353 were also good with correlations being 0.915 and 0.941. The operational efficiency of our method is as good as classical methods′, showing that using evidence theory to measure word semantic similarity is reasonable and effective.

Key words Computing with word, statistical learning, evidence theory, uncertainty modeling

Citation Wang Jun-Hua, Zuo Xiang-Lin, Zuo Wan-Li. Word semantic similarity measurement based on evidence theory. *Acta Automatica Sinica*, 2015, **41**(6): 1173−1186

单词语义相似度度量一直是自然语言处理领域的热点问题, 其成果可推动机器翻译[1]、词义消歧[2−4]、信息抽取[5] 等应用的发展. 国内外对单词语义相似度度量的相关研究基本分为基于知识的方法[6−24] 和基于语料库的方法[7, 25−30].

基于知识的单词语义相似度度量算法的基础是语义词典, 如 WordNet、MindNet、FrameNet. 早期 Rada 等[8] 借助 WordNet, 首次提出通过计算概念间的语义距离获得概念所指代单词的相似度. 在 Rada 算法的基础上, Resnik[9] 综合考虑了概念共

收稿日期 2013-12-13 录用日期 2014-10-27
Manuscript received December 13, 2013; accepted October 27, 2014
国家自然科学基金 (60903098, 60973040, 61300148, 61472049), 吉林省重点科技攻关项目 (20130206051GX), 吉林省科技计划青年基金项目 (20130522112JH) 资助
Supported by National Natural Science Foundation of China (60903098, 60973040, 61300148, 61472049), Key Scientific and Technological Break-through Program of Jilin Province (20130206051GX), and Science and Technology Planning Youth Fund Project of Jilin Province (20130522112JH)
本文责任编委 赵铁军
Recommended by Associate Editor ZHAO Tie-Jun
1. 吉林大学计算机科学与技术学院 长春 130012 2. 符号计算与知识工程教育部重点实验室 (吉林大学) 长春 130012 3. 长春工业大学计算机科学与工程学院 长春 130012
1. College of Computer Science and Technology, Jilin University, Changchun 130012 2. Key Laboratory of Symbolic Computation and Knowledge Engineering (Jilin University), Ministry of Education, Changchun 130012 3. School of Computer Science and Engineering, Changchun University of Technology, Changchun 130012

有祖先结点所拥有的信息量, 提出了 simR 方法. Wu 等[10] 则基于 WordNet 概念上下位关系, 利用概念及其最近邻共有祖先的深度度量其语义相似度. Agirre 等[11] 考虑得更为全面, 除了概念结点间的路径长度、概念深度, 还引入了局部密度等. Jiang 等[12] 则通过度量概念及其最近邻共有祖先的信息量计算概念相似度, 提出了 distJC 方法. Lin[13] 也利用了概念信息量, 提出了 simL 方法. Leacock 等[2] 在概念间最短距离的基础上引入了概念体系最大深度用以计算概念语义相似度, 提出了 simLC 方法. Hirst 等[14] 认为如果概念间路径长度较短且方向改变不频繁则其语义相似, 提出了 relHS 方法. Li 等[15] 结合路径长度、概念深度和信息量等多种资源度量概念语义相似度. Yang 等[16] 基于三种关系类型 (hyper/hyponym, hol/meronym, syn/antonym) 利用概念间距离结合 7 个参数度量概念语义相似度, 提出了 simmaxB 方法. Budanitsky 等[17] 对比分析了早期 5 种典型的基于 WordNet 的单词语义相似度度量方法, 分别由 Jiang 和 Conrath、Hirst 和 St-Onge、Leacock 和 Chodorow、Lin、Resnik 提出. 近年来部分工作者对前人所提方法进行了不同程度的改进. Alvarez 等[18] 利用单词在 WordNet 中的所有释义、释义间关系和释义描述构建单词语义关联权图, 在此基础上定义单词间距离, 实现单词语义相似度度量, 提出了 SSA (A graph modeling of semantic similarity between words) 方法. Qin 等[19] 利用 WordNet 结合有向无环图理论, 使用语义距离和特征信息量度量单词语义相似度. Pirró 等[20] 结合基于特征的相似度度量方法和信息论理论, 提出了 P&S 方法. Cai 等[21] 利用 WordNet 的有向无环图, 结合改进的基于距离的和基于信息的度量方法, 实现单词语义相似度度量. Sánchez 等[22−23] 分别对基于信息的和基于特征的度量方法进行了改进. Liu 等[24] 结合 WordNet 给出了概念向量模型, 并利用向量余弦相似度量化概念的语义相似度.

基于语料库的单词语义相似度度量算法的思想是借助存储了词语上下文信息的大规模语料库 (ACL/DCI 千万级语料库, COBUILD、Longman 百万级语料库), 利用统计技术, 通过量化上下文特征间的相似程度获得单词语义相似度. Dagan 等[25] 采用概率模型来计算单词间的距离. Brown 等[26] 利用平均互信息度量单词语义相似度. Lee[27] 则采用了相关熵模型. Liu 等[28] 通过建立模式向量空间模型度量单词语义相似度. Xu 等[29] 则通过构建词义向量计算单词语义相似度. Radinsky 等[30] 则通过建立概念时序动态发现词间的关联.

基于语料库的方法受制于所采用的语料库, 难以避免数据稀疏问题. 基于知识的方法简单有效, 无需用语料库进行训练, 较直观且易于理解; 但该类方法多受人的主观意识影响. 考虑到反映单词语义相似度的单词对特征是多样化的, 单个特征不足以量化单词语义相似度, 需要融合多个特征进行量化; 且相似问题具有不确定性, 这由 “相似” 的不确定性决定, 我们将 “相似” 的不确定性转化为对 “相同” 的不确定性推理计算, 即建模为对 “是否相同” 这个计算机 “不知道” 问题的不确定性推理. 而证据理论[31] 在处理特征差异不足以给出决策时有着较大优势, 亦可灵活地度量决策的不确定性, 同时证据合成规则也为融合多个决策特征提供了理论依据. 我们结合以上两类方法的思想首次提出了基于证据理论的单词语义相似度度量方法, 基本框架如图 1 所示.

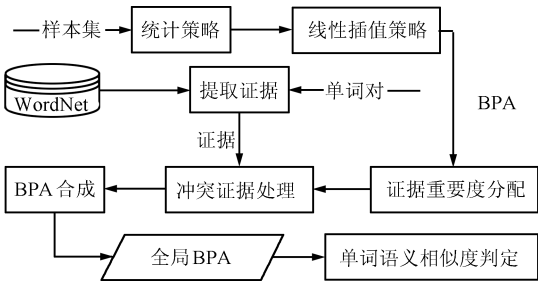


图 1 单词语义相似度建模

Fig. 1 Word Semantic similarity modeling

证据理论通过定义识别框架建立命题和集合之间的一一对应关系, 通过引入基本信任分配函数区分不确定和不知道的差异, 并用 D-S 合成规则代替贝叶斯公式来更新信任函数获得全局基本信任函数. 本文第 1 节介绍用于判别单词语义相似性的证据, 即单词对特征; 第 2 节介绍识别框架、基本信任分配函数、证据合成过程和单词语义相似度度量模型.

1 单词对特征选择和量化

通过上文总结的现有工作可以看出, 当前基于知识的方法多利用概念距离和概念深度. 这是由于概念距离和概念深度易于获得且对概念相似性的区分度较高. 鉴于此, 本文在概念距离和概念深度的基础上定义单词对距离与单词对深度, 并通过 R&G[32] 数据利用散点图分析其对单词语义相似性的区分度. 概念距离和概念深度都是针对概念上下位关系图而言, 且在概念路径及概念路径长度的基础上定义的.

定义 1. 概念上下位关系图 (HG). 在 WordNet 中, 概念间上下位关系数占所有关系总量的近 80 %, 概念通过上下位关系形成图 $HG = (V, E, r)$ (如图 2). 其中, V 为概念节点集, E 为 V 上的二元关系集, r 是根结点.

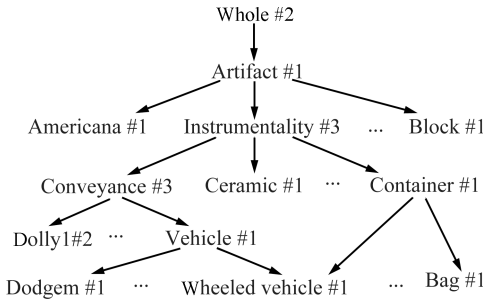


图 2　概念上下位关系图 (局部)

Fig. 2　Concept hyponymy graph (Part)

定义 2 (概念路径). 已知概念节点序列 $P=(v_1, v_2, \cdots, v_n)$, P 为概念节点 v_1 到 v_n 的概念路径, 当且仅当 $E(v_i, v_{i+1})$ $(0<i<n)$ 成立.

定义 3 (概念路径长度). 已知概念节点 v_0 到 v_n 的概念路径 $P=(v_0, v_1, v_2, \cdots, v_n)$, 则 P 的长度为 n, 记为 $L_P=n$.

定义 4 (概念节点深度). 已知概念节点 v 和 r 到 v 的最短概念路径 $P=(r, v_1, v_2, \cdots, v)$, 则将 v 的深度定义为 P 的长度, 记为 $D_v=L_P$.

定义 5 (概念节点距离). 已知概念节点 v_1、v_2 及 v_1 与 v_2 的最近邻公共祖先概念节点 v, 将 v 到 v_1 和 v 到 v_2 的最短路径 P_1 与 P_2 的长度之和定义为 v_1 与 v_2 之间的距离 $L(v_1, v_2)$. 规定概念节点到其自身的距离为零.

定义 6 (单词对距离 (LW)). 在上下位关系图中每个单词对应一个或多个概念, 每一单词对也就对应一个或多个概念对, 我们将其对应的所有概念对中距离最短的概念节点距离定义为该单词对的距离.

定义 7 (单词对深度 (DW). 单词对的深度为其对应的所有概念对中距离最短的概念对间最短路径上公共祖先节点的概念节点深度. 概念对间最短路径上的公共祖先节点有且仅有一个.

基于定义 6 和 7, 易理解单词对距离和单词对深度可通过调用 WordNet 的查询接口获得. 给定单词 w_1 和 w_2, 可通过以下步骤获得其特征 LW 和 DW. 1) 分别获得与 w_1 和 w_2 所有语义相对应的概念序列 $C_1=(c_{11}, c_{12}, \cdots, c_{1n})$ 和 $C_2=(c_{21}, c_{22}, \cdots, c_{2m})$, 其中 n 是 w_1 的语义数, m 是 w_2 的语义数. 2) 获得 C_1 和 C_2 中各概念 c_{ij} 到 WordNet 根节点的概念路径序列 $P=(p_1, p_2, \cdots, p_k)$, 其中 k 是概念 c_emphij 到 WordNet 根节点的概念路径数. 3) 统计概念 c_{1h} 到 c_{2t} 的最短路径并记录路径长度 L_{ht} 和路径上的公共祖先节点的深度 D_{ht}. 4) 最后经比较将路径长度最小值赋值给单词对距离 LW, 同时将对应的公共祖先节点的深度赋值给单词对深度 DW:

$$LW=\min_{h\le n, t\le m} L_{ht} \tag{1}$$

$$DW=D_{ht}\big|_{\min_{h\le n, t\le m} L_{ht}} \tag{2}$$

我们针对 R&G 数据集的 65 对单词基于 WordNet1.6 分别求取了每对单词的 LW 和 DW 值 (见表 1). 选用 WordNet1.6 是由于 WordNet 的高版本删除了单词 woodland, 从而无法获得单词对 shore-woodland、cemetery-woodland、bird-woodland、hill-woodland 和 forest-woodland 的特征. Word 列数据记录了 R&G 数据集的 65 对单词, S 列数据记录了相应单词对的人工标注相似度值, LW 列数据记录了相应单词对的距离特征, DW 列数据记录了相应单词对的深度特征.

表 1　R&G 数据集

Table 1　R&G data set

Word	S	LW	DW
cord-smile	0.02	12	0
noon-string	0.04	30	0
rooster-voyage	0.04	30	0
fruit-furnace	0.05	6	2
autograph-shore	0.06	30	0
automobile-wizard	0.11	11	0
mound-stove	0.14	6	2
grin-implement	0.18	30	0
asylum-fruit	0.19	6	2
asylum-monk	0.39	10	0
graveyard-madhouse	0.42	12	1
glass-magician	0.44	8	0
boy-rooster	0.44	11	1
cushion-jewel	0.45	6	2
monk-slave	0.57	4	2
asylum-cemetery	0.79	9	1
coast-forest	0.85	6	1
grin-lad	0.88	30	0
shore-woodland	0.9	5	1
monk-oracle	0.91	7	2
boy-sage	0.96	5	2
automobile-cushion	0.97	7	3
mound-shore	0.97	4	3
lad-wizard	0.99	4	2
forest-graveyard	1	7	1
food-rooster	1.09	12	0
cemetery-woodland	1.18	7	1
shore-voyage	1.22	30	0
bird-woodland	1.24	7	1

续表 1 R&G 数据集

Table 1 (continued) R&G data set

Word	S	LW	DW
coast-hill	1.26	4	3
furnace-implement	1.37	5	2
crane-rooster	1.41	7	5
hill-woodland	1.48	5	1
journey-car	1.55	30	0
cemetery-mound	1.69	8	1
glass-jewel	1.78	7	2
magician-oracle	1.82	2	4
crane-implement	2.37	4	3
lad-brother	2.41	4	2
sage-wizard	2.46	5	2
oracle-sage	2.61	7	2
bird-cock	2.63	1	5
bird-crane	2.63	3	5
food-fruit	2.69	4	3
brother-monk	2.74	1	5
asylum-madhouse	3.04	1	7
furnace-stove	3.11	2	2
magician-wizard	3.21	0	4
hill-mound	3.29	0	7
cord-string	3.41	1	4
glass-tumbler	3.45	1	5
serf-slave	3.46	3	3
grin-smile	3.46	0	7
journey-voyage	3.58	1	5
autograph-signature	3.59	1	5
coast-shore	3.6	1	4
forest-woodland	3.65	0	3
tool-implement	3.66	1	4
cock-rooster	3.68	0	9
boy-lad	3.82	1	4
cushion-pillow	3.84	1	4
cemetery-graveyard	3.88	0	6
car-automobile	3.92	0	7
gem-jewel	3.94	0	6
midday-noon	3.94	0	7

我们利用表 1 提供的数据, 分别绘制了可反映 S 与 LW 和 S 与 DW 相关性的散点图 (见图 3). 散点图又称散点分布图, 是以一个变量为横坐标, 另一个变量为纵坐标, 利用散点的分布形态反映变量统计关系的一种图形. 它能直观表现出影响因素和预测对象间的总体关系趋势和变化形态, 为选用何种数学表达方式模拟变量间关系提供决策支持; 也是度量变量间关系强弱的最直观的图形. 图 3 中左图的横坐标为 LW, 纵坐标为 S, 从该图可以看出 LW 与 S 间的总体关系趋势呈现明显的相关性, 且单词对距离 LW 越大, 单词间的语义相似度 S 越小. 图 3 中右图的横坐标为 DW, 纵坐标为 S, 从该图可以看出 DW 与 S 间的总体关系趋势亦呈现明显的相关性, 而单词对深度 DW 越大, 单词间的语义相似度 S 也越大. 尽管 LW 和 DW 对 S 的影响趋势不同, 但均与 S 具有明显的相关性, 且 LW 和 DW 相互独立, 符合证据合成规则对证据独立性的要求. 鉴于如上分析, 我们最终选择 LW 和 DW 作为定量评价单词语义相似性的证据.

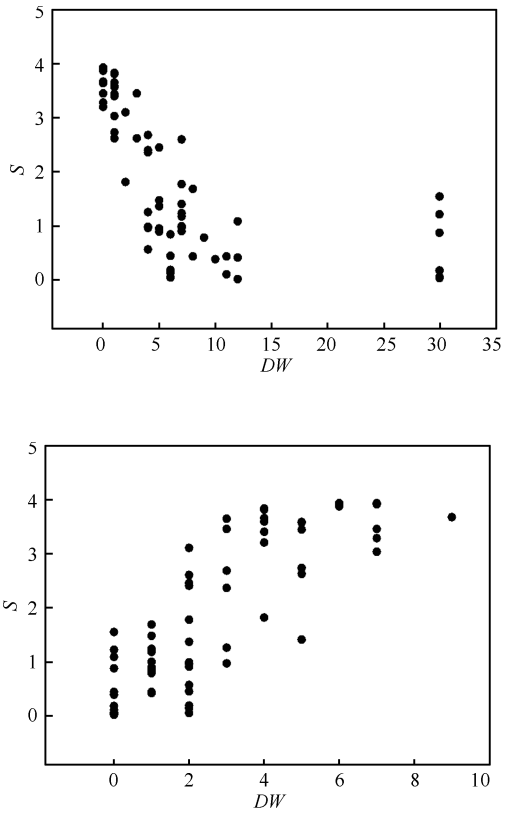


图 3 单词对特征分析散点图

Fig. 3 Scatter plot for word pair feature analyzing

考虑到人类的识别能力具有有界性, Yang 等[16] 研究得出人类基于概念距离区分概念语义相似度的上限是 12; 同时 WordNet 的最大概念深度为 16. 我们将样本集 R&G 中 LW 值大于 12 的设定为 12, 同时界定 LW 的值域为 $LV = \{i|0 \le i \le 12,\ i \in z\}$, DW 的值域为 $DV = \{i|0 \le i \le 16,\ i \in z\}$. 也就说当两单词距离大于 12 时, 我们将其特征 LW 重新赋值为 12.

2 基于证据理论度量单词语义相似度

人类对单词进行语义相似度判定, 始于对单词的形象化, 并不断寻找二者的相同点和不同点, 然后衡量共同特征占总特征的比重. 由此, 我们将单

词语义相似度度量问题转化为判定单词对的语义相同程度和不同程度问题. 而正如上文所述证据理论在不确定性度量方面具有独特的优势. 因此我们借助证据理论度量单词语义相似度, 并定义识别框架 $\theta = \{U, N\}$, 其中 U 代表 "相同", N 代表 "不同". 定义了识别框架 θ 后, 还需要为每一证据赋以基本信任分配函数 (Basic probability assignment, BPA), 使用 BPA 完成信息融合.

BPA 生成是运用证据理论度量单词语义相似度的关键环节, 它提供了基于证据理论的不确定性推理所需的初始信度依据. 然而, 目前并没有流程完整、理论性能优良、实现复杂性可接受的 BPA 生成方案. 部分研究将证据理论当作一种数学方法应用于具体问题, 通常采用专家指定信度的人工方法作为 BPA 生成的基础; 已有研究虽然提出了具有一般性的 BPA 计算公式或利用传感器信息自动生成 BPA 的方法, 但没有给出完整的输入处理方法, 或其输入处理方法局限于具体问题而无法推广. 鉴于此, 我们基于训练集, 采用统计技术和分段线性插值方法, 生成面向单词语义相似度度量的 BPA.

2.1 基本信任分配函数

2.1.1 基于统计策略生成 BPA

针对训练样本集中出现的证据 $LW = i$ ($i \in LV_i$) 和 $DW = j$ ($j \in DV_j$), 我们采用统计策略计算其基本信任分配函数. 分别查询训练样本集中 $LW = i$ 和 $DW = j$ 的所有样本, 读取结果样本的 S 值得到集合 $S_i = \{s_{i1}, s_{i2}, \cdots, s_{in}\}$ 和 $S_j = \{s_{j1}, s_{j2}, \cdots, s_{jm}\}$. 并计算 S_i 和 S_j 中所有元素的均值 $LM(i)$ 和 $DM(j)$.

$$LM(i) = \frac{1}{n}\sum_{k=1}^{n} s_{ik} \tag{3}$$

$$DM(j) = \frac{1}{m}\sum_{k=1}^{m} s_{jk} \tag{4}$$

证据 $LW = i$ 对事件 U 的信任程度是用来描述人们依据特征 $LW = i$ 判定单词语义相似的程度, 而 $LM(i)/4$ 为部分群体依据特征 $LW = i$ 量化的单词语义相似度, 因此将证据 $LW = i$ 对事件 U 的信任程度赋值为 $LM(i)/4$; 又识别框架中仅有 U 和 N 两个元素, 因此将证据 $LW = i$ 对事件 N 的信任程度赋值为 $1-LM(i)/4$, 对其他事件的信任程度赋值为 0. 同理, 证据 $DW = j$ 对事件 U 的信任程度赋值为 $DM(j)/4$, 对事件 N 的信任程度赋值为 $1-DM(j)/4$, 对其他事件的信任程度也赋值为 0. 换句话说, 证据 $LW = i$ 和 $DW = j$ 建立的信任程度的初始分配我们用基本信任分配函数 $m_{LW=i}(A)$ 和 $m_{DW=j}(A)$ 表示.

$$m_{LW=i}(A) = \begin{cases} \dfrac{LM(i)}{4}, & A{=}U \\ \dfrac{4-LM(i)}{4}, & A{=}N \\ 0, & \text{其他} \end{cases} \tag{5}$$

$$m_{DW=j}(A) = \begin{cases} \dfrac{DM(j)}{4}, & A{=}U \\ \dfrac{4-DM(j)}{4}, & A{=}N \\ 0, & \text{其他} \end{cases} \tag{6}$$

2.1.2 基于分段线性插值策略生成 BPA

分段线性插值是在每个区间 $[x_i, x_{i+1}]$ 上用 1 阶多项式逼近 $f(x)$. 分段线性插值是数学、计算机图形学等领域广泛使用的一种简单插值方法. 针对训练样本集中未出现的非边界证据 $LW = i$ ($i \in LV_i$) 和 $DW = j$ ($j \in DV_j$), 我们采用分段线性插值策略计算其基本信任分配函数. 遍历训练样本集, 确定插值区间 $[h, k]$ 和 $[f, g]$, 对任意 $x \in (h,k)$ 且 $x \in \mathbf{Z}$ 训练样本集中无样本满足条件 $LW = x$, 对任意 $y \in (f,g)$ 且 $y \in \mathbf{Z}$ 训练样本集中也无样本满足条件 $DW = y$, 且训练样本集中存在满足条件 $LW = h$、$LW = k$、$DW = f$ 和 $DW = g$ 的样本. 并在区间 $[h, k]$ 和 $[f, g]$ 上使用分段线性插值技术计算函数值 $LM(i)$ 和 $DM(j)$.

$$LM(i) = \frac{i-k}{h-k} \times LM(h) + \frac{i-h}{k-h} \times LM(k) \tag{7}$$

$$DM(j) = \frac{j-g}{f-g} \times DM(f) + \frac{j-f}{g-f} \times DM(g) \tag{8}$$

证据 $LW{=}i$ 和 $DW{=}j$ 建立的信任程度的初始分配我们仍用基本信任分配函数 $m_{LW=i}(A)$ 和 $m_{DW=j}(A)$ 表示, 见式 (5) 和式 (6).

由于样本数量有限, 若训练样本集中不存在 $DW{=}16$ 的样本, 因此证据 $DW{=}16$ 建立的信任程度的初始分配我们不能利用统计策略进行量化; 又由于 16 为 DW 值域的边界, 即证据 $DW{=}16$ 为边界证据, 我们也不能利用分段线性插值策略量化其建立的信任程度的初始分配. 考虑到具有特征 $DW{=}16$ 的两个单词必同属于 WordNet 中的某同义词集, 也就是说它们一定具有相同的语义, 我们规定用如下基本信任分配函数表示证据 $DW{=}16$ 建立的信任程度的初始分配.

$$m_{DW=16}(A) = \begin{cases} 1, & A{=}U \\ 0, & \text{其他} \end{cases} \tag{9}$$

同理, 若训练样本集中不存在 $LW = 0$ 的样本, 则规定用如下基本信任分配函数表示证据 $LW{=}0$ 建

立的信任程度的初始分配.

$$m_{LW=0}(A)=\begin{cases}1, & A=U\\ 0, & \text{其他}\end{cases} \tag{10}$$

同时, 若训练样本集中不存在 LW=12 和 DW=0 的样本, 我们则规定用如下基本信任分配函数表示证据 LW=12 和 DW=0 建立的信任程度的初始分配.

$$m_{LW=12}(A)=\begin{cases}1, & A=N\\ 0, & \text{其他}\end{cases} \tag{11}$$

$$m_{DW=0}(A)=\begin{cases}1, & A=N\\ 0, & \text{其他}\end{cases} \tag{12}$$

2.2 证据合成

对于具有特征 $LW=i$ 和 $DW=j$ 的特定单词对, 我们可以通过查询基本信任分配函数集获得证据 $LW=i$ 和 $DW=j$ 建立的信任程度的初始分配函数 $m_{LW=i}(A)$ 和 $m_{DW=j}(A)$. 一方面, $m_{LW=i}(A)$ 和 $m_{DW=j}(A)$ 通常是不同的, 为了得到全局的信任程度的分配函数, 我们需要将 $m_{LW=i}(A)$ 和 $m_{DW=j}(A)$ 合并成一个概率分配函数. 另一方面, 悖论问题的存在直接影响到推理的正确性和可靠性, 而许多悖论问题是由证据间的冲突或不一致造成的. 因此在进行证据合成前, 要先进行证据冲突判别和冲突处理. 而证据 $LW=i$ 和 $DW=j$ 的重要度也不尽相同, 为了获得更合理的合成结果, 我们在处理冲突的同时结合证据的重要度对信度进行了适当调整.

2.2.1 证据冲突判别

我们通过计算证据 $LW=i$ 和 $DW=j$ 对命题 U 和 N 的命题稀释度进行冲突判别. 命题稀释度可体现不同证据间的相互影响, 一方面表示命题支持度的损失, 另一方面表示命题未知度的增加. 当某一命题的稀释度较大时, 各证据对该命题的冲突必然较大; 反之亦然. 设定命题稀释度阈值 β, 当命题 A 的稀释度 $R_A\geq\beta$ 时, 各证据对命题 A 的支持程度存在较大的冲突; 当命题 A 的稀释度 $R_A<\beta$ 时, 各证据对命题 A 的支持程度具有相对一致性.

定义 8 (命题稀释度[33]). 为各证据对同一命题支持度的标准差. 已知证据集 $S=\{S_1,S_2,\cdots,S_t\}$, 相应基本信任分配函数集 $M=\{m_1,m_2,\cdots,m_t\}$, 则 S 中各证据对命题 A 的命题稀释度为

$$R_A=\sqrt{\frac{1}{t}\sum_{i=1}^{t}(m_i(A)-\frac{1}{t}\sum_{i=1}^{t}m_i(A))^2} \tag{13}$$

由命题稀释度的定义可知, 通过以下计算可获得证据 $LW=i$ 和 $DW=j$ 对命题 U 的命题稀释度 R_U. 1) 计算 $LW=i$ 和 $DW=j$ 对命题 U 的支持度 $m_{LW=i}(U)$ 和 $m_{DW=j}(U)$ 的均值 $V=(m_{LW=i}(U)+m_{DW=j}(U))/2$. 2) 计算信度 $m_{LW=i}(U)$ 和 $m_{DW=j}(U)$ 与均值 V 的差 V_1 和 V_2: $V_1=m_{LW=i}(U)-V$, $V_2=m_{DW=j}(U)-V$. 3) 计算 $m_{LW=i}(U)$ 和 $m_{DW=j}(U)$ 的方差 $V'=(V_1\times V_1+V_2\times V_2)/2$. 4) 计算 $m_{LW=i}(U)$ 和 $m_{DW=j}(U)$ 的标准差并赋值给 R_U.

$$R_U=\sqrt{V'} \tag{14}$$

因为 $m_{LW=i}(U)$ 和 $m_{LW=i}(N)$ 的和等于 1 且 $m_{DW=j}(U)$ 和 $m_{DW=j}(N)$ 的和也等于 1, 所以证据 $LW=i$ 和 $DW=j$ 对命题 N 的命题稀释度 $R_N=R_U$. 面向单词语义相似度度量, 我们设定命题稀释度阈值 $\beta=0.5$. 当 $R_U\geq 0.5$ 时, 我们认为证据 $LW=i$ 和 $DW=j$ 对命题 U 和 N 的支持程度存在较大的冲突; 当 $R_U<0.5$ 时, 我们认为证据 $LW=i$ 和 $DW=j$ 对命题 U 和 N 的支持程度具有相对一致性.

2.2.2 证据相对重要度分配

LW 与 DW 对于度量单词语义相似度的影响力不尽相同, 这一影响力也应当体现在合成过程中. 我们通过为证据 LW 和 DW 分配不同相对重要度, 表达不同证据对于度量单词语义相似度的影响力. 1) 利用基本信任分配函数集 $m_{LW=i}(A)|0\leq i\leq 12\cup m_{DW=j}(A)|0\leq j\leq 16$ (本节的基本信任分配函数是训练所得) 度量样本集 (训练集与测试集的并集) 中所有单词对的语义相似度. 遍历样本集量化各单词对的特征, 依据特定单词对的特征 $LW=i$ 和 $DW=j$ 提取基本信任分配函数 $m_{LW=i}(A)$ 和 $m_{DW=j}(A)$ 并将该单词对的语义相似度赋值为 $(m_{LW=i}(U)\times m_{DW=j}(U))/(1-m_{LW=i}(U)\times m_{DW=j}(N)-m_{LW=i}(N)\times m_{DW=j}(U))$, 将所有单词对的语义相似度表示为向量 $v=\langle v_1,v_2,\cdots,v_n\rangle$ (n 为样本数). 2) 利用基本信任分配函数子集$m_{LW=i}(A)|0\leq i\leq 12$ 度量样本集中所有单词对的语义相似度. 遍历样本集量化各单词对的距离特征, 依据特定单词对的特征 $LW=i$ 提取基本信任分配函数 $m_{LW=i}(A)$ 并将该单词对的语义相似度赋值为 $m_{LW=i}(U)$, 将所有单词对的语义相似度表示为向量 $v_1=\langle v_{11},v_{12},\cdots,v_{1n}\rangle$. 3) 利用基本信任分配函数子集$m_{DW=j}(A)|0\leq j\leq 16$ 度量样本集中所有单词对的语义相似度. 遍历样本集量化各单词对的深度特征, 依据特定单词对的特征 $DW=j$ 提取基本信任分配函数 $m_{DW=j}(A)$, 并将该单词对的语义相似度赋值为 $m_{DW=j}(U)$, 将

所有单词对的语义相似度表示为向量 v_2=⟨v_{21},v_{22}, $\cdots$,v_{2n}⟩. 4) 计算证据 LW 和 DW 对于度量单词语义相似度的重要度 δ_L=1−$(v\cdot v_1)/(|v|\times|v_1|)$ 和 δ_D=1−$(v\cdot v_2)/(|v|\times|v_2|)$. 5) 依据 δ_L 和 δ_D 度量证据 LW 和 DW 对于度量单词语义相似度的相对重要度 $\lambda_L=\delta_L/\delta_L+\delta_D)$ 和 $\lambda_D=\delta_D/(\delta_L+\delta_D)$.

2.2.3 信度调整

对判定为存在冲突的证据 $LW=i$ 和 $DW=j$, 我们依据其对命题 U 的命题稀释度 R_U 调整信度函数 $m_{LW=i}(A)$ 和 $m_{DW=j}(A)$. 并且为了进一步降低冲突对合成结果的影响, 我们在调整信度函数时引入了相对重要度.

1) 调整证据 $LW=i$ 和 $DW=j$ 对命题 U 的支持程度:

$$m'_{LW=i}(U)=\lambda_L\times(1-R_U)\times m_{LW=i}(U) \quad (15)$$

$$m'_{DW=j}(U)=\lambda_D\times(1-R_U)\times m_{DW=j}(U) \quad (16)$$

2) 调整证据 $LW=i$ 和 $DW=j$ 对命题 N 的支持程度:

$$m'_{LW=i}(N)=\lambda_L\times(1-R_U)\times m_{LW=i}(N) \quad (17)$$

$$m'_{DW=j}(N)=\lambda_D\times(1-R_U)\times m_{DW=j}(N) \quad (18)$$

3) 补充命题 θ 并将命题 U 和 N 损失的信度分配给识别框架 θ, 表示结论必是 U 和 N 中的一个命题, 但不能确定是哪一个命题, 即结论完全不确定. 量化证据 $LW=i$ 和 $DW=j$ 对命题 θ 的支持程度:

$$m'_{LW=i}(\Theta)=1-\sum_{A\neq\Theta}\lambda_L(1-R_U)m_{LW=i}(A) \quad (19)$$

$$m'_{DW=j}(\Theta)=1-\sum_{A\neq\Theta}\lambda_D(1-R_U)m_{DW=j}(A) \quad (20)$$

2.2.4 BPA 合成

我们使用 D-S 合成规则对证据 $LW=i$ 和 $DW=j$ 建立的信任程度的初始分配函数或调整后的 $m_{LW=i}(A)$ 和 $m_{DW=j}(A)$ 进行正交和运算实现证据合成, 生成全局信度函数 $m(A)$.

定义 9 (D-S合成规则[34]). 假设识别框架 θ 下的两个证据 E_1 和 E_2, 其相应的基本信任分配函数为 m_1 和 m_2, 焦元分别为 A_i 和 B_j, 设 K=$\sum A_i\bigcap B_j=\phi$ $m_1(A_i)m_2(B_j)<1$, 则 D-S 合成规则为

$$m(A)=\begin{cases}\dfrac{\sum\limits_{A_i\bigcap B_j=A}m_1(A_i)m_2(B_j)}{1-K}, & A\neq\phi\\ 0, & A=\phi\end{cases} \quad (21)$$

系数 $1/(1-K)$ 称为正则化因子.

D-S 合成规则是反映证据联合作用的一个法则. 给定几个同一识别框架上基于不同证据的信任函数, 如果这几批证据不是完全冲突的 (K<1), 那么就可以利用 D-S 合成规则计算出一个新的信任函数, 而这个信任函数就可以作为在那几批证据的联合作用下产生的全局信度函数 $m(A)$. 因此在使用 D-S 合成规则获得证据 LW=i 和 DW=j 联合作用下产生的全局信度函数 $m(A)$ 前, 我们将论证证据 LW=i 和 DW=j 不是完全冲突的.

结论 1. 证据 LW=i 和 DW=j 建立的信任程度的初始分配函数 $m_{LW=i}(A)$ 和 $m_{DW=j}(A)$ 满足条件 $K<1$.

$$\begin{aligned}&\because K=m_{LW=i}(U)\times m_{DW=j}(N)+\\&\quad m_{LW=i}(N)\times m_{DW=j}(U);\\&\text{又 } 0<m_{DW=j}(N)<1 \text{ 且 } 0<m_{DW=j}(U)<1;\\&\therefore K<m_{LW=i}(U)+m_{LW=i}(N);\\&\text{又 } m_{LW=i}(U)+m_{LW=i}(N)=1;\\&\therefore K<1\end{aligned}$$

结论 2. 证据 LW=i 和 DW=j 建立的信任程度的初始分配函数经信度调整后的 $m_{LW=i}(A)$ 和 $m_{DW=j}(A)$ 仍满足条件 $K<1$.

$$\begin{aligned}&\because K=m_{LW=i}(U)\times m_{DW=j}(N)+\\&\quad m_{LW=i}(N)\times m_{DW=j}(U);\\&\text{又 } 0<m_{DW=j}(N)<1 \text{ 且 } 0<m_{DW=j}(U)<1;\\&\therefore K<m_{LW}=i(U)+m_{LW}=i(N);\\&\text{又 } m_{LW=i}(U)+m_{LW=i}(N)<1;\\&\therefore K<1\end{aligned}$$

经过如上论证可知, 使用 D-S 合成规则获得证据 LW=i 和 DW=j 联合作用下产生的全局信度函数 $m(A)$ 在理论上是可行的. 具体步骤如下: 1) 计算正则化因子 $\alpha=1/(1-m_{LW=i}(U)\times m_{DW=j}(N)-m_{LW=i}(N)\times m_{DW=j}(U))$; 2) 计算证据 LW=i 和 DW=j 对命题 U 的联合信度 $m(U)$=$\alpha\times(m_{LW=i}(U)\times m_{DW=j}(U)+m_{LW=i}(U)\times m_{DW=j}(\theta)+m_{LW=i}(\Theta)\times m_{DW=j}(U))$; 3) 计算证据 $LW=i$ 和 $DW=j$ 对命题 N 的联合信度 $m(N)$=$\alpha\times(m_{LW=i}(N)\times$

$m_{DW=j}(N) + m_{LW=i}(N) \times m_{DW=j}(\theta) + m_{LW=i}(\theta) \times m_{DW=j}(N))$; 4) 计算证据 $LW{=}i$ 和 $DW{=}j$ 对命题 θ 的联合信度 $m(\theta){=}\alpha{\times}m_{LW=i}(\theta) \times m_{DW=j}(\theta)$.

2.3 单词语义相似度度量

我们以证据 $LW{=}i$ 和 $DW{=}j$ 对命题 U 的联合信度 $m(U)_{ij}$ 量化单词蕴含的共同信息, 以证据 $LW{=}i$ 和 $DW{=}j$ 对命题 N 的联合信度 $m(N)_{ij}$ 量化单词蕴含的不同信息, 而证据 $LW{=}i$ 和 $DW{=}j$ 对命题 θ 的联合信度 $m(\Theta)_{ij}$ 则视为不确定信息量. 如上文所述, 人类对单词进行语义相似度判定, 始于对单词的形象化, 并不断寻找二者的相同点和不同点, 然后衡量共同特征占总特征的比重. 由此, 我们将单词语义相似度 Sim 建模如下:

$$Sim(w_1, w_2) = \frac{m(U)_{ij} + \frac{m(\Theta)_{ij}}{2}}{m(U)_{ij} + m(N)_{ij} + m(\Theta)_{ij}} \quad (22)$$

其中, i 和 j 分别为单词对 w_1 和 w_2 的 LW 和 DW 值.

结论 3. $Sim(w_1, w_2) = m(U)_{ij}{+}m(\theta)_{ij}/2$. 已知式 (22) 成立, 而 $m(U)_{ij}{+}m(N)_{ij}{+}m(\theta)_{ij}{=}1$, 所以 $Sim(w_1, w_2){=}\ m(U)_{ij}{+}m(\theta)_{ij}/2$.

结论 4. 单词语义相似度具有对称性. 因为量化 $Sim(w_1, w_2)$ 和 $Sim(w_2, w_1)$ 均以 $LW{=}i$ 和 $DW{=}j$ 为证据经证据合成获得, 即 $Sim(w_1, w_2){=}m(U)_{ij}{+}m(\Theta)_{ij}/2$ 且 $Sim(w_2, w_1) {=}m(U)_{ij}{+}m(\theta)_{ij}/2$, 所以 $Sim(w_1, w_2){=}Sim(w_2, w_1)$.

综合上述分析, 基于证据理论度量单词语义相似度: 首先, 依据训练集生成基本信任分配函数集获得证据 $LW{=}i(0{\le}i \le 12)$ 和 $DW{=}j\ (0 \le j \le 16)$ 对事件的初始信任度, 并为特征 LW 和 DW 分配不同的权重; 然后利用基本信任分配函数计算证据 $LW{=}i\ (0 \le i \le 12)$ 和 $DW{=}j\ (0 \le j \le 16)$ 对命题 U 的命题稀释度; 最后, 对给定单词 w_1 和 w_2 依次提取特征 LW 和 DW、命题稀释度和基本信任分配函数, 依据命题稀释度识别冲突并调整基本信任分配函数, 并合成基本信任分配函数量化单词 w_1 和 w_2 的语义相似度. 伪代码见算法 Sim_{D-S}.

算法 1. Sim_{D-S}.
输入. 单词 w_1, w_2; 样本数据集
输出. Sim
1. For i=1 to n
(LW, DW)←*computeFeature*(w_{i1},w_{i2});
End for
2. For i = 0 to 12
$m_{\ LW=i}(A)$←*computeBPA*((LW, DW, v)*, $LW{=}i$);
End for
3. For j=0 to 16
$m_{DW=j}\ (A)$←*computeBPA*((LW, DW, v)*, $DW{=}j$);
End for
4. (λ_L, λ_D) ← *computeRI*((LW, DW, v)*, m^*);
5. R_U^* ←*computePD*(m^*);
6. (i, j) ← *computeFeature*(w_1, w_2);
7. R_U ←*getPD*($LW{=}i$, $DW{=}j$, R_U^*);
8. ($m_{LW=i}(A)$,$m_{DW=j}(A)$)←
getBPA($LW{=}i$, $DW{=}j$, m^*);
9. If R_U >0.5 then
$m_{LW=i}(A)$←*adjustBPA*($m_{LW=i(A)}$, R_U, λ_L);
$m_{DW=j}(A)$←*adjustBPA* ($m_{DW=j}(A)$, R_U, λ_D);
End if
10. $m(A)$←*synthesisBPA*($m_{LW=i}(A)$, $m_{DW=j}(A)$);
11. Sim←$m\ (U)_{ij}{+}m(\theta)_{ij}/2$;
12. Return Sim

3 实验与结果分析

面向单词语义相似度度量, 多位研究人员给出了人工标注数据集. Rubenstein 和 Goodenouth 于 1965 年选取了 65 对单词由来自 51 个学科的两组本科生进行人为判定, 学生为每对单词分配一个 0 到 4 的值作为自己对相应单词对的语义相似度的判定结果, 0 表示 “无关”, 4 表示 “同义”, 获得数据集 R&G. Miller 等[35] 于 1991 年对 R&G 数据集中的 30 对单词的语义相似度重新标注, 获得数据集 M&C, 标注结果和 R&G 标注值的相关度为 0.97. Resnik[9] 于 1995 年对 M&C 数据集中的所有单词对的语义相似度重新标注, 标注结果和 M&C 标注值的相关度为 0.96. 最后 Pirró[20] 于 2009 年对 R&G 数据集中的所有单词对的语义相似度重新标注, 标注结果和 R&G 标注值的相关度为 0.97. 以上数据显示过去的 40 多年中人们对单词语义相似度的认知比较稳定, 数据集 R&G 作为单词语义相似度度量的基准数据集是合理有效的, 数据集的重复标注理论上对实验结果的影响应不大, 为了更好地说明这一点我们在选用 R&G 数据集进行对比实验分析的同时将给出算法在数据集 M&C 上的实验结果. 考虑到 R&G 数据集不大, 我们亦在数据集 WordSim353[36] 上进行了实验分析.

在数据集 R&G 上, 我们进行 5 折交叉验证, 样本数据集 R&G 被分割成 5 个子样本集, 一个单独的子样本集被保留作为验证模型的数据, 其他 4 个子样本集的并集为训练集, 交叉验证重复 5 次, 每个子样本集验证一次, 5 次交叉验证的实验结果见表 2 ∼ 表 6, Sample 列数据为测试样本, R&G 列数据为 R&G 人工标注值, Sim_{D-S} 列数据是算法 Sim_{D-S} 的量化结果. 在数据集 M&C 上, 我们进行 3 折交叉验证, 样本数据集 M&C 被分割成 3 个子样本集, 一个单独的子样本集被保留作为验证模型的数据, 其他 2 个子样本集的并集为训练集, 交叉验证重复 3 次, 每个子样本集验证一次, 3 次交叉验证

的实验结果见表 7～表 9, Sample 列数据为测试样本, M&C 列数据为 M&C 人工标注值, Sim_{D-S} 列数据是算法 Sim_{D-S} 的量化结果. 在数据集 WordSim353 上, 由于单词 live、eat、earning、defeating 和 Maradona 没有名词词义, 因此除去样本 stock-live, Maradona-football, drink-eat, investor-earning, fighting-defeating, 我们提取了 60 个样本为测试样本, 288 个样本为训练样本 (标注值乘以 0.4 将其由 [0, 10] 映射到 [0, 4]), 实验结果见表 10, Sample 列数据为测试样本, WS 列数据为人工标注值, Sim_{D-S} 列数据是算法 Sim_{D-S} 的量化结果.

表 2 子样本 1 的实验结果 (R&G)

Table 2 Results of Sample 1 (R&G)

Word	R&G	Sim_{D-S}
glass-magician	0.44	0.0363
asylum-cemetery	0.79	0.0613
coast-forest	0.85	0.0166
monk-oracle	0.91	0.1782
lad-wizard	0.99	0.2240
forest-graveyard	1	0.1314
journey-car	1.55	0.0136
cemetery-mound	1.69	0.0941
crane-implement	2.37	0.4355
oracle-sage	2.61	0.1782
autograph-signature	3.59	0.9218
car-automobile	3.92	0.9748
midday-noon	3.94	0.9748

表 3 子样本 2 的实验结果 (R&G)

Table 3 Results of Sample 2 (R&G)

Word	R&G	Sim_{D-S}
automobile-wizard	0.11	0.0196
grin-implement	0.18	0.0264
asylum-monk	0.39	0.0286
graveyard-madhouse	0.42	0.0562
grin-lad	0.88	0.0264
boy-sage	0.96	0.1572
cemetery-woodland	1.18	0.1543
glass-jewel	1.78	0.1739
sage-wizard	2.46	0.1572
food-fruit	2.69	0.3835
journey-voyage	3.58	0.9134
coast-shore	3.6	0.9523
tool-implement	3.66	0.9523

表 4 子样本 3 的实验结果 (R&G)

Table 4 Results of Sample 3 (R&G)

Word	R&G	Sim_{D-S}
rooster-voyage	0.04	0.0341
autograph-shore	0.06	0.0341
asylum-fruit	0.19	0.0463
monk-slave	0.57	0.2968
bird-woodland	1.24	0.1486
coast-hill	1.26	0.5600
magician-oracle	1.82	0.9819
bird-crane	2.63	0.9366
asylum-madhouse	3.04	0.9844
furnace-stove	3.11	0.7460
glass-tumbler	3.45	0.9328
cock-rooster	3.68	0.9925
gem-jewel	3.94	0.9968

表 5 子样本 4 的实验结果 (R&G)

Table 5 Results of Sample 4 (R&G)

Word	R&G	Sim_{D-S}
cord-smile	0.02	0.0204
mound-stove	0.14	0.0491
boy-rooster	0.44	0.0094
cushion-jewel	0.45	0.0491
mound-shore	0.97	0.4304
shore-voyage	1.22	0.0204
hill-woodland	1.48	0.1569
lad-brother	2.41	0.2517
bird-cock	2.63	0.9416
cord-string	3.41	0.9622
grin-smile	3.46	0.9896
forest-woodland	3.65	0.9335
boy-lad	3.82	0.9622

表 6 子样本 5 的实验结果 (R&G)

Table 6 Results of Sample 5 (R&G)

Word	R&G	Sim_{D-S}
noon-string	0.04	0.0216
fruit-furnace	0.05	0.0565
shore-woodland	0.9	0.1891
automobile-cushion	0.97	0.4080
food-rooster	1.09	0.0216
furnace-implement	1.37	0.2670
crane-rooster	1.41	0.6875
brother-monk	2.74	0.9576
magician-wizard	3.21	0.9861
hill-mound	3.29	0.9929
serf-slave	3.46	0.6986
cushion-pillow	3.84	0.9630
cemetery-graveyard	3.88	0.9991

表 7 子样本 1 的实验结果 (M&C)
Table 7 Results of Sample 1 (M&C)

Word	M&C	Sim_{D-S}
cord-smile	0.13	0.0074
coast-forest	0.42	0.0562
lad-wizard	0.42	0.3360
lad-brother	1.66	0.3360
crane-implement	1.68	0.3692
journey-car	1.16	0.0074
monk-oracle	1.1	0.1955
brother-monk	2.82	0.9795
bird-cock	3.05	0.9795
magician-wizard	3.5	0.9889

表 8 子样本 2 的实验结果 (M&C)
Table 8 Results of Sample 2 (M&C)

Word	M&C	Sim_{D-S}
rooster-voyage	0.08	0.0263
glass-magician	0.11	0.0446
monk-slave	0.55	0.2121
coast-hill	0.87	0.5231
tool-implement	2.95	0.9790
bird-crane	2.97	0.8460
asylum-madhouse	3.61	0.9961
boy-lad	3.76	0.9790
midday-noon	3.42	0.9987
furnace-stove	3.11	0.4584

表 9 子样本 3 的实验结果 (M&C)
Table 9 Results of Sample 3 (M&C)

Word	M&C	Sim_{D-S}
noon-string	0.08	0.0130
shore-woodland	0.63	0.0254
forest-graveyard	0.84	0.0426
food-fruit	3.08	0.1406
journey-voyage	3.84	0.9224
car-automobile	3.92	0.9789
cemetery-woodland	0.95	0.0426
food-rooster	0.89	0.0130
gem-jewel	3.84	0.9642
coast-shore	3.7	0.9604

表 10 基于 WordSim353 的实验结果
Table 10 Results based on WordSim353

Word	WS	Sim_{D-S}
coast-hill	4.38	0.48
lad-brother	4.46	0.48
image-surface	4.56	0.56
peace-plan	4.75	0.52
space-chemistry	4.88	0.54
journal-association	4.97	0.55
doctor-personnel	5	0.55
death-inmate	5.03	0.5
energy-laboratory	5.09	0.52
impartiality-interest	5.16	0.54
doctor-liability	5.19	0.55
death-row	5.25	0.5
exhibit-memorabilia	5.31	0.53
glass-metal	5.56	0.49
OPEC-country	5.63	0.62
money-laundering	5.65	0.58
lobster-wine	5.7	0.52
planet-people	5.75	0.56
journey-car	5.85	0.62
deployment-withdrawal	5.88	0.61
reason-criterion	5.91	0.57
energy-crisis	5.94	0.57
grocery-money	5.94	0.62
game-round	5.97	0.57
shower-flood	6.03	0.57
bread-butter	6.19	0.65
skin-eye	6.22	0.61
disaster-area	6.25	0.6
train-car	6.31	0.67
shower-thunderstorm	6.31	0.69
oil-stock	6.34	0.6
governor-office	6.34	0.59
gender-equality	6.41	0.49
street-place	6.44	0.61
government-crisis	6.56	0.62
fertility-egg	6.69	0.59
cup-tableware	6.85	0.76
tiger-mammal	6.85	0.68
game-defeat	6.97	0.62
doctor-nurse	7	0.72
summer-drought	7.16	0.65
bird-crane	7.38	0.72
book-library	7.46	0.76
day-dawn	7.53	0.76
money-property	7.57	0.76

续表 10　基于 WordSim353 的实验结果

Table 10　(Continued) Results based on WordSim353

Word	WS	Sim_{D-S}
rock-jazz	7.59	0.81
boxing-round	7.61	0.69
money-deposit	7.73	0.86
cell-phone	7.81	0.81
liquid-water	7.89	0.81
news-report	8.16	0.85
money-wealth	8.27	0.85
murder-manslaughter	8.53	0.9
king-queen	8.58	0.94
asylum-madhouse	8.87	0.89
street-avenue	8.88	0.85
car-automobile	8.94	0.98
gem-jewel	8.96	0.94
type-kind	8.97	0.89
football-soccer	9.03	0.94

由表 2～表 6 可以看出, 对于人工标注相似度较低的样本, 算法 Sim_{D-S} 给出的相似度值也较小; 人工标注相似度较高的样本, 算法 Sim_{D-S} 给出的相似度值也较大. 基本上 R&G 标注值小于 1 的样本, 算法 Sim_{D-S} 给出的相似度值会小于 0.25, 而 R&G 标注值大于 3 的样本, 算法 Sim_{D-S} 给出的相似度值会大于 0.75; 且经典方法标注偏差较大的样本, Sim_{D-S} 值也有所改善, 如 cemetery-woodland 的 R&G 标注值 1.18, reLHS、simL 和 simR 的标注值都是 0, Sim_{D-S} 值 0.1543; 相较于根据专家经验直接给出建模公式的方法, Sim_{D-S} 通过训练 BPA 可有效利用群体智慧, 更好地模拟人类单词语义相似度判定过程, 证据挖掘越充分训练集越完整, 语义词典越完备, Sim_{D-S} 算法越接近人类单词语义相似度判定过程. 鉴于算法 Sim_{D-S} 输出的单词语义相似度值的取值范围为 [0, 1], 而数据集 R&G 的人工标注单词语义相似度的取值范围为 [0, 4], 以上数据显示算法 Sim_{D-S} 给出的相似度值与 R&G 标注值具有较高的吻合度.

为了直观地展现算法 Sim_{D-S} 给出的相似度值与人工标注值的高相关性, 我们以 R&G 值为横坐标 Sim_{D-S} 给出的相似度为纵坐标, 针对表 2～表 6 分别绘制散点图. 图 4 中从上到下从左向右依次为实验 1～5 的散点图. 5 个散点图中点的分布大体都呈直线, 即除了少数点偏离较大, 多数点在一条直线上或分布在距离直线较近的区域. 离散点越少, 算法 Sim_{D-S} 给出的相似度值与人工标注值的相关性越高, 由图 4 可以看出实验 2 的离散点最少, 点的离散程度也较低.

我们进一步计算 Sim_{D-S} 给出的相似度值与人工标注值 R&G、M&C 和 WS 的相关系数, 以定量分析 Sim_{D-S} 的有效性. 相关系数是衡量两个随机变量之间线性相关程度的指标, 现已广泛地应用于科学的各个领域. 相关系数 (r) 的定义如式 (23) 所示, 取值范围为 $[-1, 1]$, $r>0$ 表示 X 和 Y 正相关, $r<0$ 表示 X 和 Y 负相关, $|r|$ 可表示 X 和 Y 之间相关程度的高低. 特殊地, $r=1$ 称为 X 和 Y 完全正相关, $r=-1$ 称为 X 和 Y 完全负相关, $r=0$

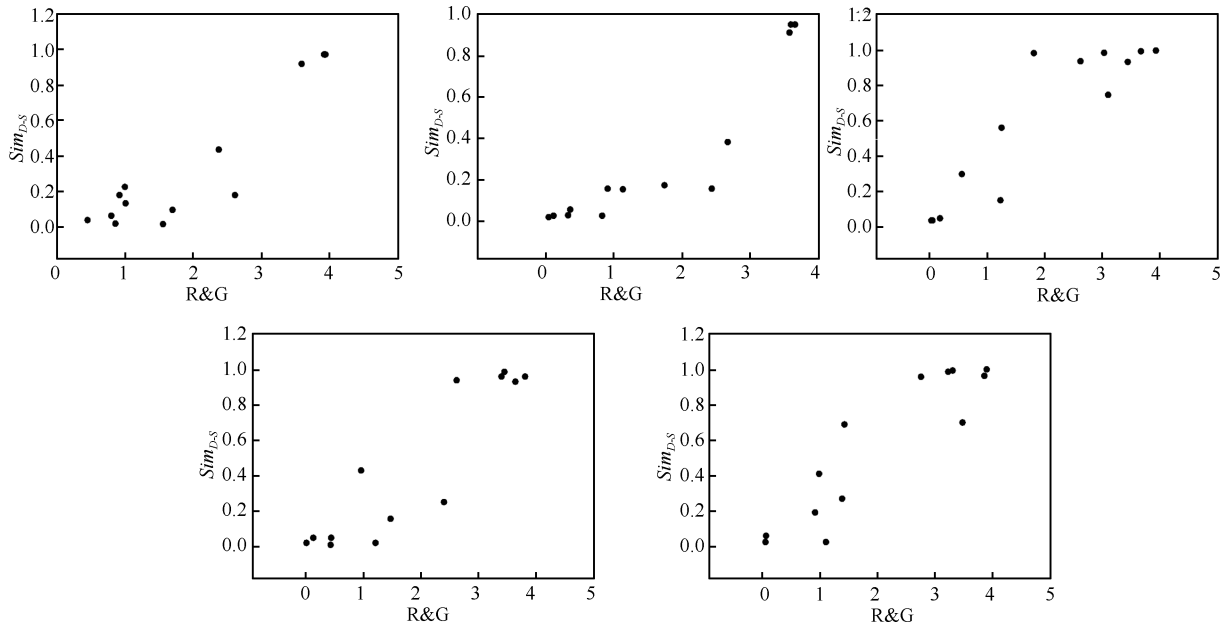


图 4　实验结果散点图

Fig. 4　Scatter plots of example results

称为 X 和 Y 不相关. 通常 $|r|$ 大于 0.8 时, 认为 X 和 Y 有很强的线性相关性.

$$r_{XY}=\frac{\sum_{i=1}^{N}(X_i-\bar{X})(Y_i-\bar{Y})}{\sqrt{\sum_{i=1}^{N}(X_i-\bar{X})^2}\sqrt{\sum_{i=1}^{N}(Y_i-\bar{Y})^2}} \tag{23}$$

基于数据集 R&G 的 5 次交叉实验的输出结果与人工标注值的相关系数依次为 0.913、0.914、0.908、0.913 和 0.912. 由于所做实验是 5 折交叉验证实验, 因此我们取 5 次实验的平均相关系数 0.912 作为基于数据集 R&G 对 Sim_{D-S} 的评估结果. 统计 0.913、0.914、0.908、0.913 和 0.912 的方差, 其值仅为 0.0000044, 表明该方法具有很强的数据适应能力, 性能稳定. 另外, 基于数据集 M&C 的 3 次交叉实验的相关系数依次为 0.922、0.912 和 0.911, 平均相关系数为 0.915; 基于数据集 WordSim353 的相关系数为 0.941. 以上数据显示数据集的重复标注对实验结果影响不大, 且本文所提方法在较大样本环境下亦有效.

当前基于数据集 R&G(65) 做实验结果分析的最好成果取得了 0.908 的相关系数, 而 5 次交叉实验的相关系数均不低于 0.908, 确切地说除了实验 3 与最好成果相当外, 其余 4 次实验均高于最好成果. 对比分析本文所提方法与基于数据集 R&G(65) 做实验结果分析的其他方法 [2, 9, 12, 13, 14, 16, 18, 20] (见图 5), Sim_{D-S} 的实验结果高出现有最好方法 P&S 的实验结果 0.4%, 而相较于经典算法 reLHS、distJC、simLC、simL 和 simR 少则高出 7% 多则高出 13%. 另外, 算法 reLHS、distJC、simLC、simL 和 simR 和 P&S 在数据集 M&C 上的实验结果均低于本文所提方法在数据集 M&C 上的实验结果 (见图 6). 再者, 基于数据集 WordSim353 的 Spearman Rank 相关系数为 0.9117, 远高于文献 [7] 中所提方法的最好结果 0.6094.

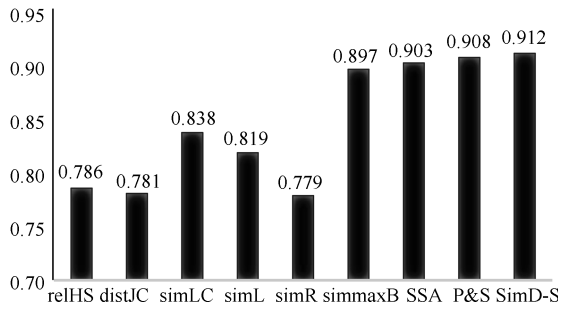


图 5 基于 R&G(65) 算法准确度对比

Fig. 5 Algorithm results comparison chart for R&G(65)

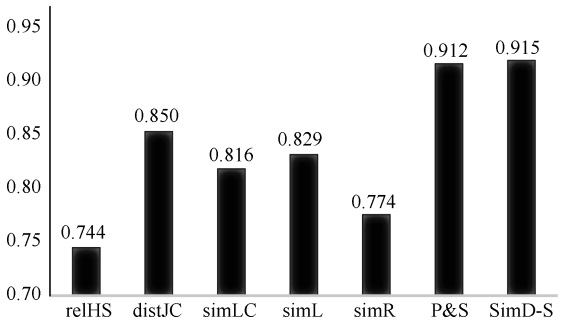


图 6 基于 M&C(30) 算法准确度对比

Fig. 6 Algorithm results comparison chart for M&C(30)

Sim_{D-S} 不仅与人工标注值具有很强的线性相关性, 其计算效率亦很高. 这是由于仅选用了计算量小且区分度高的单词对距离和单词对深度两个量作为证据, 相较于仍考虑其他因素 (如 Part-of 关系) 的方法减少了查询知识库的时间. 考虑到描述当前方法的文献中未给出算法的执行效率的介绍, 而经典算法 disJC、simLC 和 simL 由于执行效率较高多被应用于词义消歧、本体映射等应用. 我们在笔记本 HP Compaq nx6325 上同时实现了经典算法 disJC、simLC 和 simL, 结果见表 11. 实验环境的具体配置: 处理器 AMD Turion 64 X2 Mobile Technology TL-50 1.6 GHz, 内存 2 GB, 硬盘 TOSHIBA 120 GB 5400 rpm, 操作系统 Windows XP, 开发平台 Eclipse, 编程语言 Java. 算法效率的对比结果显示 Sim_{D-S} 的运行速度比算法 disJC 和 simL 的快, 稍低于算法 simLC 的; 也就是说 Sim_{D-S} 较经典算法大幅度提升准确度的同时, 仍维持了较高的执行效率.

表 11 算法效率对比

Table 11 Comparison of computation time

方法	时间 (s)
Leacock-Chodorow	0.48
Jiang-Conrath	0.50
Lin	0.50
Sim_{D-S}	0.49

4 结束语

我们针对单词语义相似度度量问题, 通过分析现有成果并借助散点图选取了计算量小且区分度高的单词对特征, 又利用取自 R&G 数据集的训练样本使用统计和分段线性插值方法计算基本信任分配函数, 最后在引入证据重要度分配和冲突证据处理的基础上利用 D-S 合成规则合成 BPA, 并利用全局 BPA 量化单词对的语义相似度. 在数据集 R&G(65) 上, 对比算法评判结果与人类评判结果的

相关度, 其相关度比当前最优方法 P&S 高出 0.4 %, 比经典算法 relHS、disJC、simLC、simL 和 simR 高出 7 %～13 %; 在数据集 M&C(30) 和 WordSim353 上也取得了比较好的实验结果, 相关度分别为 0.915 和 0.941; 同时 Sim_{D-S} 的执行效率和经典算法的相当. 实验结果显示使用证据理论解决单词语义相似度问题是合理有效的. 本文的主要贡献: 定义了证据重要度分配策略, 并将其引入证据合成过程; 证据理论首次被用于单词语义相似度度量, 不同于当前依赖专家经验直接给出其数学建模的处理方法, Sim_{D-S} 方法可有效利用群体经验.

我们希望, D-S 方法将有益于促进机器翻译的智能性、提高信息检索的准确性及改善其他通常需要计算单词语义相似度任务的研究现状. 下一步, 我们将对可用于度量单词语义相似度的证据进行深入挖掘, 继续完善 D-S 方法; 并将 D-S 方法应用于词义消歧.

References

1 Zhou M, Ding Y, Huang C N. Improving translation selection with a new translation model trained by independent monolingual corpora. *Computational Linguistics and Chinese Language Processing*, 2001, **6**(1): 1−26

2 Leacock C, Chodorow M. *Combining Local Context and WordNet Similarity for Word Sense Identification*. Cambridge: MIT Press, 1998. 265−283

3 Lu Wen-Peng, Huang He-Yan, Wu Hao. Word sense disambiguation with graph model based on domain knowledge. *Acta Automatica Sinica*, 2006, **40**(12): 2836−2850
(鹿文鹏, 黄河燕, 吴昊. 基于领域知识的图模型词义消歧方法. 自动化学报, 2014, **40**(12): 2836−2850)

4 Liu Yu-Peng, Li Sheng, Zhao Tie-Jun. System combination based on WSD using wordnet. *Acta Automatica Sinica*, 2010, **36**(11): 1575−1580
(刘宇鹏, 李生, 赵铁军. 基于 WordNet 词义消歧的系统融合. 自动化学报, 2010, **36**(11): 1575−1580)

5 Hassan H, Hassan A, Emam O. Unsupervised information extraction approach using graph mutual reinforcement. In: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA, USA: Association for Computational Linguistics, 2006. 501−508

6 Li Wen-Qing, Sun Xin, Zhang Chang-You, Feng Ye. A semantic similarity measure between ontological concepts. *Acta Automatica Sinica*, 2012, **38**(2): 229−235
(李文清, 孙新, 张常有, 冯烨. 一种本体概念的语义相似度计算方法. 自动化学报, 2012, **38**(2): 229−235)

7 Cui Q, Gao B, Bian J, Qiu S, Liu T Y. KNET: A General Framework for Learning Word Embedding Using Morphological Knowledge. arXiv: 1407.1687, 2014. 1−16

8 Rada R, Mili H, Bicknell E, Blettner M. Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man, and Cybernetics*, 1989, **19**(1): 17−30

9 Resnik P. Using information content to evaluate semantic similarity in a taxonomy. In: Proceedings of the 14th International Joint Conference on Artificial Intelligence. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1995. 448−453

10 Wu Z B, Palmer M. Verbs semantics and lexical selection. In: Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics. Stroudsburg, PA, USA: Association for Computational Linguistics, 1994. 133−138

11 Agirre E, Rigau G. A proposal for word sense disambiguation using conceptual distance. In: Proceedings of the 1st International Conference on Recent Advances in Natural Language Processing. Stroudsburg, Cambridge: MIT Press, 1995. 35−43

12 Jiang J J, Conrath D W. Semantic similarity based on corpus statistics and lexical taxonomy. In: Proceedings of the 1997 International Conference on Research in Computational Linguistics. Stroudsburg, PA: ACL, 1997. 19−33

13 Lin D K. An information-theoretic definition of similarity. In: Proceedings of the 15th International Conference on Machine Learning. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998. 296−304

14 Hirst G, St-Onge D. *Lexical Chains as Representations of Context for the Detection and Correction of Malapropisms*. Cambridge: MIT Press, 1998. 305−332

15 Li Y H, Bandar Z A, McLean D. An approach for measuring semantic similarity between words using multiple information sources. *IEEE Transactions on Knowledge and Data Engineering*, 2003, **15**(4): 871−882

16 Yang D Q, Powers D M W. Measuring semantic similarity in the taxonomy of wordnet. In: Proceedings of the 28th Australasian Conference on Computer Science. Darlinghurst, Australia, Australia: Australian Computer Society, Inc., 2005. 315−322

17 Budanitsky A, Hirst G. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 2006, **32**(1): 13−47

18 Alvarez M A, Lim S J. A graph modeling of semantic similarity between words. In: Proceedings of the 2007 International Conference on Semantic Computing. Irvine, CA: IEEE, 2007. 355−362

19 Qin P, Lu Z, Yan Y, Wu F. A new measure of word semantic similarity based on wordnet hierarchy and DAG theory. In: Proceedings of the 2009 International Conference on Web Information Systems and Mining. Shanghai, China: IEEE, 2009. 181−185

20 Pirró G. A semantic similarity metric combining features and intrinsic information content. *Data & Knowledge Engineering*, 2009, **68**(11): 1289−1308

21 Cai S M, Lu Z. An improved semantic similarity measure for word pairs. In: Proceedings of 2010 International Conference on e-Education, e-Business, e-Management and e-Learning. Sanya, China: IEEE, 2010. 212−216

22 Sánchez D, Batet M, Isern D. Ontology-based information content computation. *Knowledge-Based Systems*, 2011, **24**(2): 297−303

23 Sánchez D, Batet M, Isern D, Valls A. Ontology-based semantic similarity: a new feature-based approach. *Expert Systems with Applications*, 2012, **39**(9): 7718−7728

24 Liu H Z, Bao H, Xu D. Concept Vector for semantic similarity and relatedness based on WordNet structure. *Journal of Systems and Software*, 2012, **85**(2): 370−381

25 Dagan I, Lee L, Pereira F C N. Similarity-based models of word cooccurrence probabilities. *Machine Learning*, 1999, **34**(1−3): 43−69

26 Brown P F, Pietra S A D, Pietra V J D, Mercer R L. Word-sense disambiguation using statistical methods. In: Proceedings of the 29th Annual Meeting on Association for Computational Linguistics. Stroudsburg, PA, USA: Association for Computational Linguistics, 1991. 264−270

27 Lee L. Similarity-based Approaches to Natural Language Processing [Ph. D. dissertation], Harvard University, Cambridge, MA, USA, 1997.

28 Liu L, Zhong M S, Lu R Z. Measuring word similarity based on pattern vector space model. In: Proceedings of the 2009 International Conference on Artificial Intelligence and Computational Intelligence. Piscataway, NJ: IEEE, 2009. 72−76

29 Xu T, Qu W G, Tang X R, Ding D X, Li B, Li H. Computing word similarity on large-scale corpus. In: Proceedings of the 4th International Conference on Innovative Computing, Information and Control. Kaohsiung: IEEE, 2009. 1076−1079

30 Radinsky K, Agichtein E, Gabrilovich E, Markovitch S. A word at a time: computing word relatedness using temporal semantic analysis. In: Proceedings of the 20th international conference on World Wide Web. New York, NY, USA: ACM, 2011. 337−346

31 Shafer G. *A Mathematical Theory of Evidence*. Princeton: Princeton University Press, 1976.

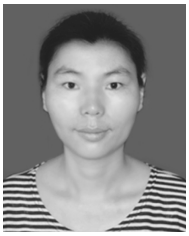
32 Rubenstein H, Goodenough J B. Contextual correlates of synonymy. *Communications of the ACM*, 1965, **8**(10): 627−633

33 Zhou Hao, Li Shao-Hong. New combination algorithm of conflict evidences introduced by GDOP. *Control and Decision*, 2010, **25**(2): 278−281
(周皓, 李少洪. GDOP 引出的冲突证据组合新算法. 控制与决策, 2010, **25**(2): 278−281)

34 Voorbraak F. A Computationally efficient approximation of Dempster-Shafer theory. *International Journal of Man-Machine Studies*, 1989, **30**(5): 525−536

35 Miller G, Charles W. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 1991, **6**(1): 1−28

36 Finkelstein L, Gabrilovich E, Matias Y, Rivlin E, Solan Z, Wolfman G, Ruppin E. Placing search in context: the concept revisited. *ACM Transactions on Information Systems*, 2002, **20**(1): 116−131



王俊华 吉林大学计算机科学与技术学院博士研究生. 2005 年获得东北师范大学传媒科学学院学士学位. 主要研究方向为自然语言处理与 Web 数据挖掘.
E-mail: wangjunhua_1982@126.com
(**WANG Jun-Hua** Ph. D. candidate at the College of Computer Science and Technology, Jilin University. She received her bachelor degree from Northeast Normal University in 2005. Her research interest covers natural language processing and Web mining.)



左祥麟 吉林大学计算机科学与技术学院本科生. 主要研究方向为自然语言处理与 Web 数据挖掘.
E-mail: 295228473@qq.com
(**ZUO Xiang-Lin** Bachelor student at the College of Computer Science and Technology, Jilin University. His research interest covers natural language processing and Web mining.)



左万利 吉林大学计算机科学与技术学院教授. 1982 年获得吉林大学计算机科学与技术学院学士学位. 主要研究方向为信息检索, 自然语言处理, 本体工程与 Web 数据挖掘. 本文通信作者.
E-mail: zuowl@jlu.edu.cn
(**ZUO Wan-Li** Professor at the College of Science and Technology, Jilin University. He received his bachelor degree from Jilin University in 1982. His research interest covers database, Web mining, information retrieval, machine learning, and natural language processing. Corresponding author of this paper.)