Table 8 Heuristic mapping results.

表 8 启发式映射结果

(a) 男士集合

姓名	映射结果	排序后结果
Bob	CGJXY (AD{H~I}XY)	GJCXY (AD{H~I}XY)
Dave	CGHWY (A{D~E}{H~I}XY)	WGHCY (A{D~E}{H~I}XY)

(b) 女士集合

姓名	映射结果	排序后结果
Alice	BDJXY (C{E~F}{I~J}WY)	BDJXY ({E~F}W{I~J}CY)
Carol	CDHXY (CF{H~J}WY)	DHCXY (FWC{H~J}Y)

5 实 验

5.1 实验设置

5.1.1 实验环境

采用 Java 语言实现了本文提出的算法,JDK 为 1.8.0.实验机器的配置为 Xeon E5-2650 2.00GHz CPU,256GB 内存,操作系统为 Windows Server 2008 64 位.

5.1.2 实验数据集

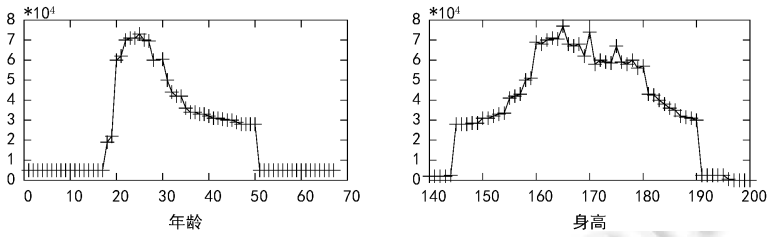
实验中采用了真实数据集 DATING;同时,依据 DATING 中的数据分布规律生成了合成数据集 L-DATING.

- DATING 是交友信息数据集,包括 10 430 条男性交友信息和 9 831 条女性交友信息.这些信息是从交友网站(http://www.eharmony.com/)上爬取的,并进行了去除隐私和归一化等操作.每条记录包括 8 个事实属性和 8 个对应的期望属性,它们分别是居住城市、收入、子女和在表 1 中列举的 5 条属性(不包括阈值).在事实信息中:属性值的数据类型可能是一个数值或一个字符串;期望信息的属性对应的数据类型可能是数值范围或枚举类型(后者可转化为字符串);
- L-DATING 是合成数据集,包括 1 500 000 条记录(男士和女士的信息各 750 000 条).每条记录里包含 12 个事实属性和 12 个期望属性.除了 DATING 原有的属性外,L-DATING 还通过重复年龄、身高、教育和婚姻增加了 4 个额外的虚拟属性.L-DATING 中的数据是根据 DATING 中对应属性上所有数据值的分布情况生成的.

5.2 数据集分析

在进行实验之前,我们先对数据集进行了分析.图 5 展示了数据集中和年龄与身高相关的两个期望属性的分布,我们可以看到,数据不满足均匀分布.对于年龄而言,18 岁和 50 岁处呈现了明显的起伏,期望交友对象的年龄在该区间内的人数显著多于期望交友对象的年龄在区间外的人数.对于身高属性,我们也能观测到类似的情况.

因此,如果我们使用等步长的映射方法,有些符号会对应非常长的倒排列表.与此同时,有些符号所对应的倒排列表又会很短.这种情况会严重影响倒排索引的效率.与此相对的是,启发式映射方法可以利用属性的分布特征来确定划分方式.举例来说,对于年龄属性,启发式映射方法会将 18~50 区间内的值进行细粒度划分,而对于区间外的值会采取粗粒度的划分.这种启发式映射方法使得各个符号所对应的倒排列表的长度相对平均,从而提高倒排索引的效率.



(a) 年龄期望属性 (b) 身高期望属性

Fig.5 Data distribution of attributes

图 5 属性数据分布

5.3 实验结果分析

5.3.1 MFV 算法生成符号记录数量和候选集大小比较

本节通过实验分析比较 MFV 算法映射阶段的 3 种可选映射方法.为方便叙述,用 MFV-I,MFV-O 和 MFV-R 分别代表采用单射的映射-过滤-验证算法、采用等步长映射的映射-过滤-验证算法和采用启发式映射的 MFV 算法.

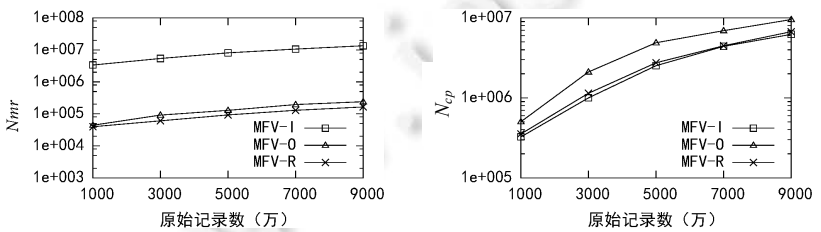
图 6 展示了采用 3 种映射方法的结果:生成的符号记录数量(用 N_{mr} 表示)、最终的候选结果对数量(用 N_{cp} 表示).根据第 4.2.1 节提出的优化目标,可以得到如下两个结论.

(1) N_{mr} 越小,建立索引结构所需要的时间开销就会越小,同时,在过滤阶段的搜索空间也相应减小;

(2) N_{cp} 越小,表明在过滤阶段的过滤效果越好.

如图 6(a)所示,对于 3 种映射方法,N_{mr} 随着原始数据量的增多而变大.一般来说,MFV-I 会比 MFV-O 和 MFV-R 产生多得多的生成记录数量.这将导致 MFV-I 花费大量时间建立索引和利用索引结构进行搜索.这也使得 MFV-I 算法将花费比嵌套循环算法更多的时间来获得最终结果.

图 6(b)显示了 N_{cp} 和原始记录数量的关系.与图 6(a)展示的关系类似,N_{cp} 随着原始记录数量的增多也存在上涨的整体趋势.从图中可以发现,N_{cp} 是原始记录的若干倍.但是需要注意的是:在没有映射的情况下,N_{cp} 的值是原始记录数量的平方,这个数量远多于采用 3 种映射方法的映射-过滤-验证算法生成的 N_{cp} 的数量.更进一步讲,图中也显示了 MFV-I 产生的最终候选结果集的规模最小.这是因为在 3 种映射方法中,单射对于不匹配的数据对有着最强的识别力.MFV-R 产生的候选结果数量与 MFV-I 相差无几.从产生最终候选结果集的规模这个评判指标上看,MFV-O 则是最差的.



(a) N_{mr}, DATING (b) N_{cp}, DATING

Fig.6 Records and candidate pairs comparison

图 6 生成记录数和候选集大小比较

5.3.2 时间开销

图 7 显示了不同算法的时间开销与原始记录数的关系.本实验设置原始记录数量从 5 000 增长到 9 000,步长为 1 000.

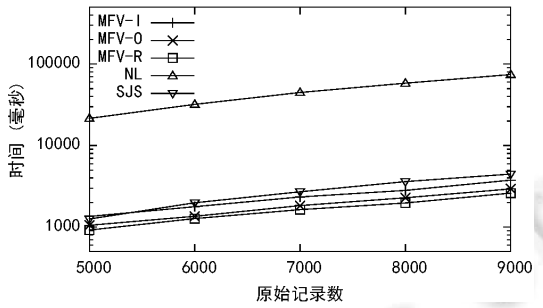


Fig.7 Time cost comparison on DATING

图 7 DATING 数据集时间开销对比

以上实验结果表明:SJS 算法和 3 个 MFV 算法在运行时间上都显著优于嵌套循环(NL)算法.这表明我们提出的用于解决泛化双向相似连接的算法都是非常有效的.在 3 种 MFV 算法中,MFV-R 又表现最好,这个结果显示了优化过的启发式映射方法比单射和等步长映射方法具有更好的效果.

5.3.3 全局阈值和独立阈值比较

直觉上说,独立阈值更能体现用户的个性化需求,例如,能更加准确地描述一个人挑剔的个性或者其相对包容的个性.本节尝试通过客观实验来分析验证设置独立阈值的必要性.

图 8 展示了泛化双向相似连接算法分别采用全局阈值(所有的记录采用相同的阈值)和独立阈值(每条记录都根据自身情况制定独有的阈值)时的对比结果.公平起见,采用独立阈值时,所有阈值的平均值为 0.8;相应地,也将全局阈值设置为 0.8.在 DATING 数据集上,采用独立阈值的结果集包含将近 200 000 对(199 428 对)结果;采用全局阈值的结果集也包括将近 200 000 对(197 958 对)结果.

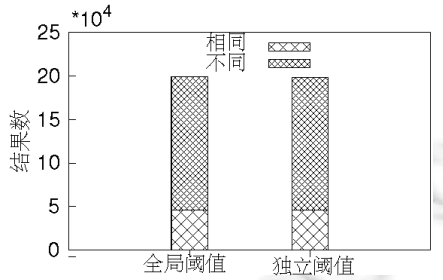


Fig.8 Unified threshold vs. individual thresholds

图 8 全局阈值和独立阈值结果比较

从结果集规模上看,这两个结果集的区别很小,独立阈值似乎可以被全局阈值所取代.值得注意的是,两个结果集中仅有 46 532 对结果是相同的,这个数量还不到全部结果集规模的 1/4.这表明采用两种不同的阈值设定方法会产生多达 150 000 对不同的结果.也就是说,采用全局阈值得到的结果集与采用独立阈值得到的结果集存在相当大的差异.

因此,为了满足真实世界的用户需求,简单地采用全局阈值代替独立阈值是不合适的.这也进一步验证了本文的观点:在相似连接问题中,为了满足不同用户的不同偏好,采用根据用户个人情况制定出的独立阈值是值得尝试的.

5.3.4 模拟数据实验结果分析

由于真实数据集的规模有限,所以我们采用合成数据进行实验以评价所提算法在较大规模数据集上的可扩展性.在合成数据集上的实验结果与在真实数据集上的实验结果有着相似的性质.

图 9 展示了论文所提算法在 L-DATING 数据集上的时间开销.因为前面的实验结果已经表明,MFV-R 算法

在 3 种过滤验证算法中效果最好,所以我们在图中没有展示另两种 MFV 算法.从图中可以看出:实验结果同第 5.3.2 节中的结果一致,SJS 算法和 MFV 算法在时间开销上均远小于 NL 算法.

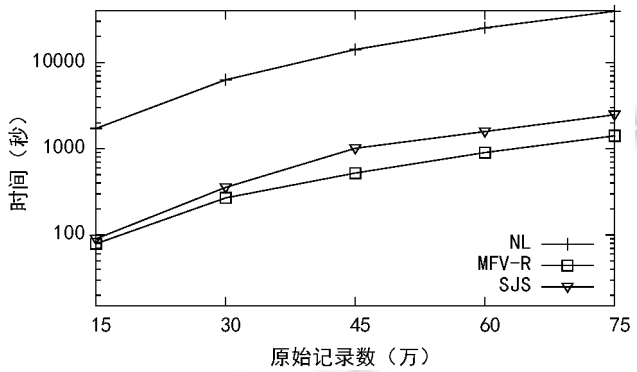
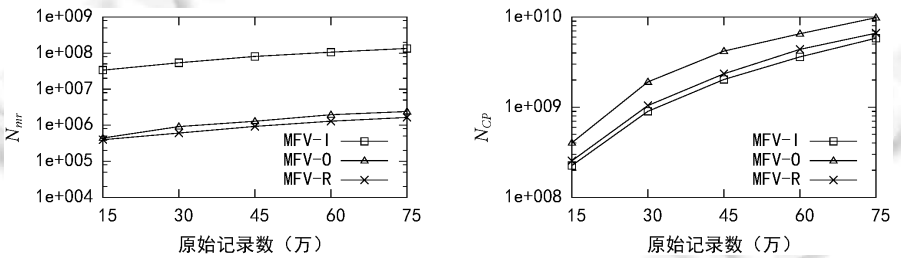


Fig.9 Time cost comparison on L-DATING

图 9 合成数据集时间消耗

图 10 展示了 3 种 MFV 算法在 L-DATING 上的生成符号记录数量和候选集大小.MFV-I 生成符号记录数量最多,过滤效果最好.MFV-O 和 MFV-R 生成符号记录数量相对较少.这二者的区别是:MFV-O 牺牲了很多过滤效果,而 MFV-R 过滤效果接近 MFV-I.



(a) N_{mr}, L-DATING (b) N_{cp}, L-DATING

Fig.10 Records and candidate pairs comparison on L-DATING

图 10 合成数据集上 MFV 算法的生成记录数和候选集大小比较

6 结束语

本文提出一种新的相似连接——泛化双向相似连接.它是对现有的大量相似连接工作的扩展,适用于更加广泛的应用场景,比如交友和求职招聘.同时提出了子连接集算法和映射-过滤-验证(MFV)算法来处理泛化双向相似连接查询,并对于 MFV 算法提出了等步长映射方法和启发式映射方法,以进一步提高算法效率.在真实和合成数据集上的大量实验结果,表明了论文所提算法的正确性和有效性.此外,对实验结果的深入比较分析显示,采用用户独立阈值比采用全局统一阈值更符合用户需求.这也证明了在泛化双向相似连接中考虑独立阈值的必要性.今后将继续改进算法,同时提高算法的可扩展性.

References:

[1] Xiao C, Wang W, Lin XM, Yu JX, Wang GR. Efficient similarity joins for near-duplicate detection. ACM Trans. on Database Systems (TODS), 2011,36(3). [doi: 10.1145/2000824.2000825]

[2] Papapetrou P, Athitsos V, Kollios G, Gunopulos D. Reference-Based alignment in large sequence databases. Proc. of the VLDB Endowment, 2009,2(1):205−216. [doi: 10.14778/1687627.1687651]

[3] Li YN, Patel JM, Terrell A. Wham: A high-throughput sequence alignment method. ACM Trans. on Database Systems (TODS), 2012,37(4):28. [doi: 10.1145/2389241.2389247]

[4] Chaudhuri S, Ganti V, Kaushik R. A primitive operator for similarity joins in data cleaning. In: Proc. of the 22nd Int'l Conf. on Data Engineering (ICDE 2006). 2006. [doi: 10.1109/ICDE.2006.9]

[5] Liu XL, Wang HZ, Li JZ, Gao H. Similarity join algorithm based on entity. Ruan Jian Xue Bao/Journal of Software, 2015,26(6): 1421−1437 (in Chinese with English abstract). http: //www.jos.org.cn/1000-9825/4610.htm [doi: 10.13328/j.cnki.jos.004610]

[6] Arasu A, Ganti V, Kaushik R. Efficient exact set-similarity joins. In: Proc. of the 32nd Int'l Conf. on Very Large Data Bases. 2006. 918−929.

[7] Zhang ZJ, Hadjieleftheriou M, Ooi BC, SrivastavaD. Bed-Tree: An all-purpose index structure for string similarity search based on edit distance. In: Proc. of the 2010 ACM SIGMOD Int'l Conf. on Management of Data. 2010. 915−926. [doi: 10. 1145/1807167. 1807266]

[8] Zhao X, Xiao C, Lin XM, Wang W. Efficient graph similarity joins with edit distance constraints. In: Proc. of the 28th Int'l Conf. on Data Engineering (ICDE 2012). 2012. 834−845. [doi: 10.1109/ICDE.2012.91]

[9] Wang JN, Feng JH, Li GL. Trie-Join: Efficient trie-based string similarity joins with edit-distance constraints. Proc. of the VLDB Endowment, 2010,3(1-2):1219−1230.

[10] Li GL, Deng D, Wang JN, Feng JH. Pass-Join: A partition-based method for similarity joins. Proc. of the VLDB Endowment, 2011, 5(3):253−264.

[11] Li R, Ju L, Peng Z, Yu ZW, Wang CK. Batch text similarity search with MapReduce. In: Proc. of the 13th Asia-Pacific Web Conf. Beijing, 2011. 412−423.

[12] Lu W, Du XY, Hadjieleftheriou MM, Ooi BC. Efficiently supporting edit distance based string similarity search using B+-trees. IEEE Trans. on Knowledge and Data Engineering, 2014,26(12):2983−2996. [doi: 10.1109/TKDE.2014.2309131]

[13] Wang JN, Li GL, Deng D, Zhang Y, Feng JH. Two birds with one stone: An efficient hierarchical framework for top-*k* and threshold-based string similarity search. In: Proc. of the 31st Int'l Conf. on the Data Engineering (ICDE 2015). 2015. 519−530. [doi: 10.1109/ICDE.2015.7113311]

[14] Wang JN, Li GL, Feng JH. Fast-Join: An efficient method for fuzzy token matching based string similarity join. In: Proc. of the 27th Int'l Conf. on the Data Engineering (ICDE 2011). 2011. 458−469. [doi: 10.1109/ICDE.2011.5767865]

[15] Wang W, Qin JB, Xiao C, Lin XM, Shen HT. VChunkJoin: An efficient algorithm for edit similarity joins. IEEE Trans. on Knowledge and Data Engineering, 2013,25(8):1916−1929. [doi: 10.1109/TKDE.2012.79]

[16] Wang CK, Wang JM, Lin XM, Wang W, Wang HX, Li HS, Tian WP, Xu J, Li R. MapDupReducer: Detecting near duplicates over massive datasets. In: Proc. of the 2010 ACM SIGMOD Int'l Conf. on Management of Data. 2010. 1119−1122. [doi: 10.1145/ 1807167.1807296]

[17] Deng D, Li GL, Feng JH. A pivotal prefix based filtering algorithm for string similarity search. In: Proc. of the 2014 ACM SIGMOD Int'l Conf. on Management of Data. 2014. 673−684. [doi: 10.1145/2588555.2593675]

[18] Rong CT, Lu W, Wang XL, Du XY, Chen YG, Tung AKH. Efficient and scalable processing of string similarity join. IEEE Trans. on Knowledge and Data Engineering, 2013,25(10):2217−2230. [doi: 10.1109/TKDE.2012.195]

[19] Deng D, Li GL, Hao S, Wang JN, Feng JH. MassJoin: A MapReduce-based method for scalable string similarity joins. In: Proc. of the 30th Int'l Conf. on the Data Engineering (ICDE 2014). 2014. 340−351. [doi: 10.1109/ICDE.2014.6816663]

[20] Brualdi RA. Introductory Combinatorics. 5th ed., Pearson Education, 2009.

附中文参考文献:

[5] 刘雪莉,王宏志,李建中,高宏.基于实体的相似性连接算法.软件学报,2015,26(6):1421−1437. http://www.jos.org.cn/1000-9825/ 4610.htm [doi: 10.13328/j.cnki.jos.004610]

王昶平(1970－),男,陕西西安人,博士生,主要研究领域为社交网络,数据挖掘.

王朝坤(1976－),男,博士,副教授,博士生导师,CCF 专业会员,主要研究领域为图和社交数据管理,音乐计算,大数据系统.

汪浩(1989－),男,硕士,主要研究领域为相似连接查询处理,时间序列分析.

王萌(1990－),女,硕士,主要研究领域为社交网络分析与挖掘,复杂网络理论.

陈俊(1989－),男,博士生,主要研究领域为数据挖掘,推荐系统,社交网络.