













2.  $F_{af} \leftarrow \emptyset, F_{sg}^* \leftarrow F_{sg}$
3. FOR EACH  $v \in I_{sg}$  DO
4.      $f_v \leftarrow [1, e^{w_v}]$
5.      $F_{af}.add(f_v)$
6. END FOR
7. create equations like Eq.(6) and solve it
8. instantiation of the parameters in  $F_{af}$
9.  $F_{sg}^*.add(F_{af})$
10. RETURN  $F_{sg}^*$
11. END

**优化方案.** 在算法 2 中,我们采用子图中的所有变量联合求解近似因子的权重,但当子图较大或已推断变量较多时,求解过程耗时较多.事实上,在求解近似因子的权重时,子图中变量的影响程度与距离有关:距离较近的变量影响较大,距离较远的变量则影响较小.比如在图 5(a)中,变量  $x_2$  与近似因子  $f_1^*$  的距离较近,故它对  $f_1^*$  的权重求解影响较大;变量  $x_6$  与近似因子  $f_1^*$  的距离较远,故它对  $f_1^*$  的权重求解影响较小.因此在求解某近似因子的权重时,我们可适当忽略子图中与它距离较远的变量,从而加快近似因子权重的求解过程.对此,本节提出了分组求解的优化技术.其基本原理为:以每个近似因子相邻的已推断变量为中心,剪枝掉其 2-hop 子图之外的所有变量(且需保证剪枝之后子图中变量节点的数目不能超过  $N_{vars}$ (通常取 20 即可),否则继续剪枝以满足条件),然后,在剪枝后的子图上求解相应近似因子的权重.另外,若两个近似因子均在同一剪枝后的子图中,则同时求解它们的权重.比如,图 5(b)为求解近似因子  $f_1^*$  权重的子图,图 5(c)则为求解近似因子  $f_2^*, f_3^*$  权重的子图.此优化技术的有效性将会在第 4.4 节得到验证.

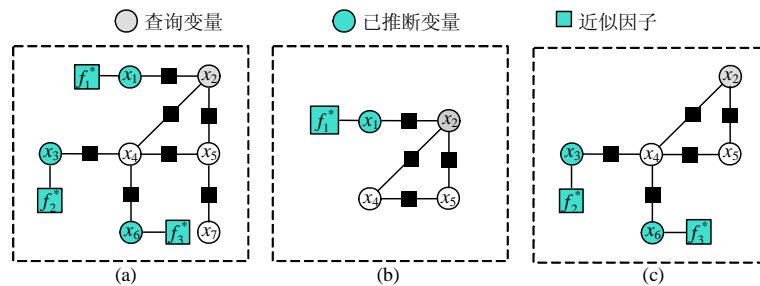


Fig.5 An illustration of grouping  
图 5 分组示意图

### 3.2.3 边缘推断

最后,在 AF 子图上执行边缘推断,进而返回查询变量的边缘概率.目前,大部分概率知识库系统主要采用基于马尔可夫链蒙特卡罗的算法(比如吉布斯抽样)在大规模因子图上执行边缘推断任务.然而当因子图较小时,若想获得较为精确的结果,采样算法通常比精确推理算法(比如团树算法)耗时要多.考虑到本文提取的 AF 子图通常较小,故采用团树算法执行边缘推断.边缘推断的具体过程如算法 3 所示.它以 AF 子图  $F_{sg}^*$  和查询变量  $v^q$  为输入,输出为查询变量的边缘概率  $p(v^q)$ .

**算法 3.** 边缘推断.

输入:AF 子图  $F_{sg}^*$ , 查询变量  $v^q$ .

输出:查询变量的边缘概率  $p(v^q)$ .

1. BEGINE

2.  $p(v^q) = \text{CliqueTree}(F_{sg}^*, v^q)$
3. RETURN  $p(v^q)$
4. END

### 3.3 复杂度分析

本节主要分析 OIAF 算法的复杂度.不妨设子图提取、添加并估算近似因子和边缘推断的时间复杂度分别为  $T_1, T_2, T_3$ .下面分别对它们进行分析.

- (1) 令全局因子图为  $F$ ,最大跳跃步数为  $k$ ,则利用广度优先搜索提取  $k_{cons}$ -hop 子图  $F_{sg}$  的时间复杂度为  $T_1 = O(b^{k+1})$ ([https://en.wikipedia.org/wiki/Breadth-first\\_search](https://en.wikipedia.org/wiki/Breadth-first_search))( $k$  的取值通常较小,默认值为 2),其中  $b$  为图  $F$  的分支系数.
- (2) 估算近似因子的过程主要包括两部分:构建方程组和求解方程组.令在分组求解中估算近似因子的组数为  $n$ ,各组中变量节点的最大阈值为  $N_{vars}$ ,则构建非线性方程组所需的时间为  $O(n2^{N_{vars}})$ ,而求解非线性方程组所需的时间为  $O(nm^3)$ (采用 MINPACK 中的子程序 HYBRD(<http://www.netlib.org/minpack/hybrd.f>)进行求解),其中  $m$  为各方程组中未知变量的最大数目.但当  $N_{vars}$  给定时,估算近似因子的时间复杂度为  $T_2 = O(n(2^{N_{vars}} + m^3)) = O(nm^3)$ .
- (3) 文献[23]指出,概率图模型中的精确推理和近似推理都是 NP 难问题,故本文所采用的团树算法在最坏情况下需要指数时间.然而,实际中提取的  $2_{cons}$ -hop 子图通常较小,故可有效执行推理.另外,该算法的空间复杂度为  $S = O(b^{d+1} + m^2)$ .

## 4 实验及分析

本文第 4.1 节对实验配置(包括运行环境、数据集等)进行详细说明.第 4.2 节为 OIAF 算法和  $k$ -hop 算法的对比实验.第 4.3 节和第 4.4 节分别说明跳跃步数  $k$  和分组优化技术对 OIAF 算法性能的影响.

### 4.1 实验配置

实验所用的编程语言为 Python,且运行环境配置为 Intel(R) Core(TM) i5-6300HQ 2.30GHz 处理器,16GB 内存,Ubuntu 16.04 LTS 64 位操作系统.实验选用了两部分数据集:(1) Poolside 商品知识库<sup>[11]</sup>,主要为为用户提供针对性的推荐服务;(2) Friends & Smokers 知识库<sup>[18]</sup>,主要用于社交网络.数据集的相关统计信息见表 1.

**Table 1** Statistics of the datasets used in the experiment

**表 1** 实验中所用数据集的统计信息

数据集	#规则	#变量节点	#因子节点
Poolside	14	2 671	4 086
Friends & Smokers	2	52 096	81 846

在实验中,若选取因子图中的所有变量作为查询变量,则耗时较多且没有必要.于是,本文分别从 Poolside 和 Friends & Smokers 中随机选取 83 和 153 个变量作为查询节点.已有在线推理算法的精度评估标准为:在线推理结果与全局推断结果的绝对误差.但若仅仅将这些查询变量的平均绝对误差作为精确性的评估标准,则很难得知 OIAF 算法(或  $k$ -hop 算法)在低误差和高误差区间上的具体分布情况.于是,我们将误差区间 $[0,1]$ 切分为 7 部分(如后文图 6 所示):第 1 部分为 $[0,0.005]$ (低误差区间);最后一部分为 $(0.03,1]$ (简记为 others,高误差区间);其余部分为 $(0.005,0.01]$ , $(0.01,0.015]$ , $(0.015,0.02]$ , $(0.02,0.025]$ 和 $(0.025,0.03]$ ;并分别统计了在各误差区间上查询变量所占的比例.同理,时间区间 $[0,\infty)$ (单位为 s)被划分为 6 部分:第 1 部分为 $[0,2]$ (低响应区间);最后一部分为 $[10,+\infty)$ (简记为 others,高响应区间);其余部分为 $(2,4]$ , $(4,6]$ , $(6,8]$ 和 $(8,10]$ .以上数据集中已推断变量的选取比例均为 20%.另外,考虑到精度和时间之间的相互影响(类似于信息检索中精确率和召回率之间的关系),本文对它们的权衡采取类似的准则: $F_\beta = (1 + \beta^2)p_e p_r / (\beta^2 p_e + p_r)$ ,其中  $p_e$  为低误差区间 $[0,0.005]$ 内的百分比(精度的度量), $p_r$  为低响应区间 $[0,2]$ 内的百分比.由于相对于精度,在线算法的时间响应更为重要,因此本文选取 $\beta=2.F_2$ 的取值越大,



说明精度和时间之间的权衡越好。

OIAF 算法的默认设置为  $k_{cons}$ -hop 子图中的跳跃步数  $k$  为 2,采用分组优化技术求解近似因子的权重。

### 4.2 OIAF算法 vs. $k$ -hop算法

本节主要通过 OIAF 算法和  $k$ -hop 算法在 Poolside 和 Friends & Smokers 数据集上的对比实验来说明 OIAF 算法可以在精度和时间上取得较好的权衡。

图 6 为两种算法在 Poolside 数据集上误差和时间的对比结果,表 2 为此对比实验在各误差和时间区间内所占百分比的详细信息,表 3 为误差和时间的相关统计信息(比如均值和标准差)。

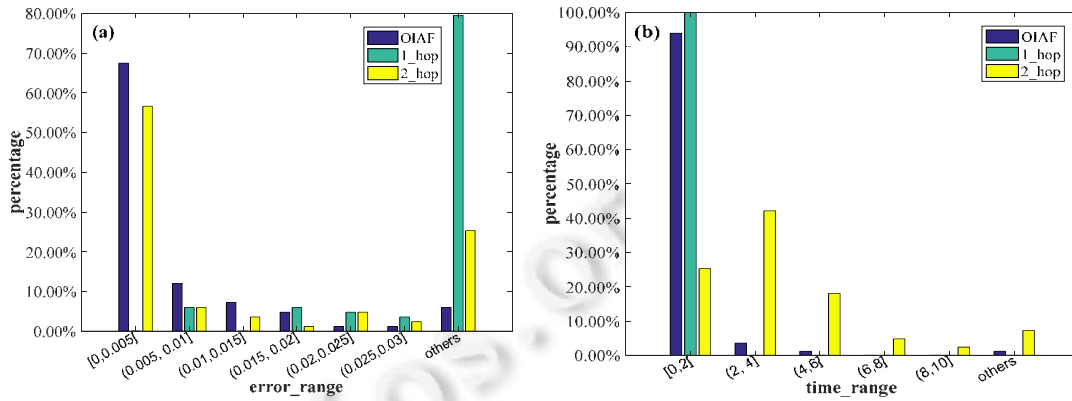


Fig.6 Comparison results of error and time on Poolside (OIAF vs.  $k$ -hop)

图 6 Poolside 上各误差和时间的对比结果(OIAF vs.  $k$ -hop)

Table 2 Percentage of each error and time interval on Poolside (OIAF vs.  $k$ -hop)

表 2 Poolside 上各误差和时间区间内所占的百分比(OIAF vs.  $k$ -hop)

误差区间	OIAF (%)	1-hop (%)	2-hop (%)	时间区间(s)	OIAF (%)	1-hop (%)	2-hop (%)
[0,0.005]	67.47	0.0	56.63	[0,2]	93.98	100	25.3
(0.005,0.01]	12.05	6.02	6.02	(2,4]	3.61	0.0	42.17
(0.01,0.015]	7.23	0.0	3.61	(4,6]	1.2	0.0	18.07
(0.015,0.02]	4.82	6.02	1.2	(6,8]	0.0	0.0	4.82
(0.02,0.025]	1.2	4.82	4.82	(8,10]	0.0	0.0	2.41
(0.025,0.03]	1.2	3.61	2.41	Others	1.2	0.0	7.23
Others	6.02	79.52	25.3	-	-	-	-

Table 3 Statistics of error and time on Poolside (OIAF vs.  $k$ -hop)

表 3 Poolside 上误差和时间的统计信息(OIAF vs.  $k$ -hop)

	统计量	OIAF	1-hop	2-hop
误差	平均值	0.007 2	0.115 5	0.025 3
	标准差	0.012 6	0.098 8	0.042 6
时间	平均值	1.179 4	0.734 4	20.488 1
	标准差	1.568 1	0.182 4	94.325 7

误差对比结果如图 6(a)所示,其横坐标为误差区间,纵坐标为在每个误差区间上查询变量所占的百分比.分析得知:OIAF 算法在低误差区间[0,0.005]上的比例高达 67.47%,而在高误差区间内的比例较低(仅为 6.02%);1-hop 算法则正好相反,它在低误差区间上的比例极低,而在高误差区间上的比例极高(高达 79.52%);2-hop 算法相对于 1-hop 算法,精确性虽然有所提升,但它在低误差区间和高误差区间上的表现均没有 OIAF 算法好.时间对比结果如图 6(b)所示,其横坐标为时间区间,纵坐标为在每个时间区间上查询变量所占的百分比.结果表明:1-hop 算法的推理速度最快(集中在低响应区间[0,2]上的比例为 100%);OIAF 算法的推理速度次之,它在低响应

区间上的比例达到 93.98%;而 2-hop 算法的推理速度相对较慢.分析得知:1-hop 算法的推理速度较快但误差较大;2-hop 算法虽然在准确性上有所提升但推理相对较慢,而 OIAF 算法在时间和精度上的表现均较好.另外,在精度和时间的权衡方面,OIAF,1-hop 和 2-hop 算法的  $F_2$  值分别为 0.871 3,0.0 和 0.284 5.综合分析可知,OIAF 算法可在时间和精度上取得较好的权衡.

图 7 为两种算法在 Friends & Smokers 数据集上的误差和时间对比结果,表 4 为此对比实验在各误差和时间区间内所占百分比的详细信息,表 5 为误差和时间的统计信息(比如均值和标准差).

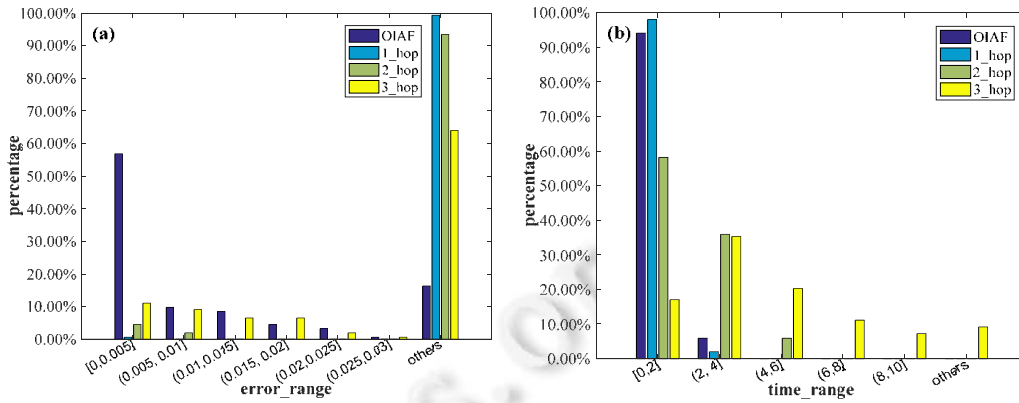


Fig.7 Comparison results of error and time on Friends & Smokers (OIAF vs.  $k$ -hop)

图 7 Friends & Smokers 上误差和时间的对比结果(OIAF vs.  $k$ -hop)

Table 4 Percentage of each error and time interval on Friends & Smokers (OIAF vs.  $k$ -hop)

表 4 Friends & Smokers 上各误差和时间区间内所占的百分比(OIAF vs.  $k$ -hop)

误差区间	OIAF (%)	1-hop (%)	2-hop (%)	3-hop (%)	时间区间(s)	OIAF (%)	1-hop (%)	2-hop (%)	3-hop (%)
[0,0.005]	56.86	0.65	4.58	11.11	[0,2]	94.12	98.04	58.17	16.99
(0.005,0.01]	9.8	0.0	1.96	9.15	(2,4]	5.88	1.96	35.95	35.29
(0.01,0.015]	8.5	0.0	0.0	6.54	(4,6]	0.0	0.0	5.88	20.26
(0.015,0.02]	4.58	0.0	0.0	6.54	(6,8]	0.0	0.0	0.0	11.11
(0.02,0.025]	3.27	0.0	0.0	1.96	(8,10]	0.0	0.0	0.0	7.19
(0.025,0.03]	0.65	0.0	0.0	0.65	Others	0.0	0.0	0.0	9.15
Others	16.34	99.35	93.46	64.05	-	-	-	-	-

Table 5 Statistics of error and time on Friends & Smokers (OIAF vs.  $k$ -hop)

表 5 Friends & Smokers 上误差和时间的统计信息(OIAF vs.  $k$ -hop)

	统计量	OIAF	1-hop	2-hop	3-hop
误差	平均值	0.018 5	0.357 9	0.263 3	0.159 8
	标准差	0.039 3	0.155 8	0.223 1	0.219 1
时间	平均值	1.253 3	0.805 0	2.011 9	5.040 7
	标准差	0.471 9	0.326 8	0.983 6	4.496 0

图 7(a)和图 7(b)表明:相对于  $k$ -hop 算法( $k$  分别取 1,2,3),OIAF 算法在低误差区间上的比例明显较高,且推理速度也相对较快(仅慢于 1-hop 算法).另外,在评测精度和时间之间的权衡时,OIAF,1-hop,2-hop 和 3-hop 算法的  $F_2$  值分别为 0.832 1,0.031 7,0.174 2 和 0.153 6.由此可进一步说明,相对于  $k$ -hop 算法,OIAF 算法可在精度和时间上取得较好的权衡.

### 4.3 跳跃步数 $k$ 对 OIAF 算法的影响

本节实验主要比较不同跳跃步数  $k$  对 OIAF 算法的影响.在上述实验中, $k$  的默认取值为 2.图 8 测评了  $k$  取不同值(1,2,3)时,OIAF 算法在 Friends & Smokers 数据集上误差和时间的对比结果,表 6 为此对比实验在各误差和时间区间内所占百分比的详细信息.

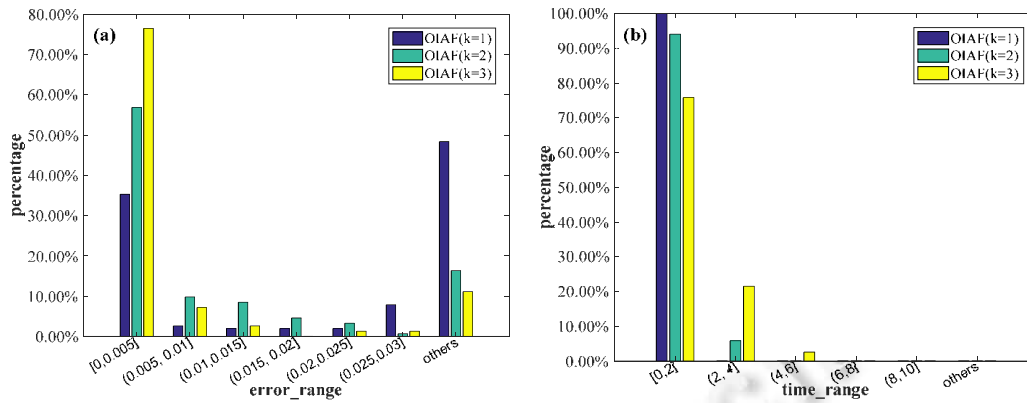


Fig.8 Comparison results of error and time for OIAF ( $k=1,2,3$ ) on Friends & Smokers

图 8 OIAF( $k=1,2,3$ )在误差和时间上的对比结果(Friends & Smokers)

Table 6 Percentage of each error and time interval for OIAF ( $k=1,2,3$ ) on Friends & Smokers

表 6 OIAF( $k=1,2,3$ )在各误差和时间区间内所占的百分比(Friends & Smokers)

误差区间	k=1 (%)	k=2 (%)	k=3 (%)	时间区间(s)	k=1 (%)	k=2 (%)	k=3 (%)
[0,0.005]	35.29	56.86	76.47	[0,2]	100	94.12	75.82
(0.005,0.01]	2.61	9.8	7.19	(2,4]	0.0	5.88	21.57
(0.01,0.015]	1.96	8.5	2.61	(4,6]	0.0	0.0	2.61
(0.015,0.02]	1.96	4.58	0.0	(6,8]	0.0	0.0	0.0
(0.02,0.025]	1.96	3.27	1.31	(8,10]	0.0	0.0	0.0
(0.025,0.03]	7.84	0.65	1.31	Others	0.0	0.0	0.0
Others	48.37	16.34	11.11	-	-	-	-

图 8(a)表明:在低误差区间[0,0.005]上, $k=1$  时比例最低, $k=2$  时次之, $k=3$  时最高.图 8(b)表明:整体来看, $k=1$  时推理速度最快, $k=2$  时次之, $k=3$  时推理相对较慢.另外,当  $k$  取 1,2,3 时,计算得知,OIAF 算法的  $F_2$  值分别为 0.731 7,0.832 1 和 0.759 5.综合分析得知:当  $k=2$  时,OIAF 算法在精度和时间上的均衡较好.

#### 4.4 分组求解对OIAF算法的影响

本节实验主要说明分组求解优化技术对 OIAF 算法的影响.图 9 给出了在数据集 Friends & Smokers 上,分组优化技术对 OIAF 算法的影响,表 7 为此对比实验在各误差和时间区间内所占百分比的详细信息.

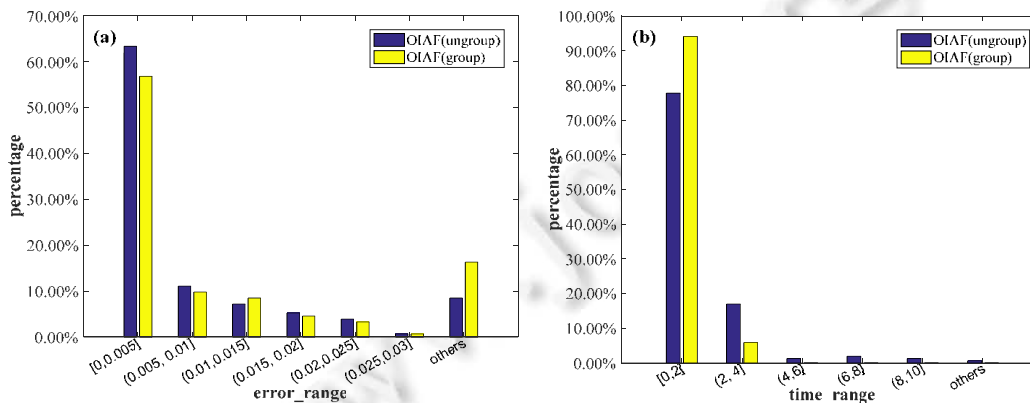


Fig.9 Performance for the OIAF approach with the optimization technique on Friends & Smokers

图 9 分组求解优化技术对 OIAF 算法的影响(Friends & Smokers)

**Table 7** Percentage of each error and time interval for OIAF (ungroup/group) on Friends & Smokers**表 7** OIAF(ungroup/group)在各误差和时间区间内所占的百分比(Friends & Smokers)

误差区间	未分组(ungroup) (%)	分组(group) (%)	时间区间(s)	未分组(ungroup) (%)	分组(group) (%)
[0,0.005]	63.4	56.86	[0,2]	77.78	94.12
(0.005,0.01]	11.11	9.8	(2,4]	16.99	5.88
(0.01,0.015]	7.19	8.5	(4,6]	1.31	0.0
(0.015,0.02]	5.23	4.58	(6,8]	1.96	0.0
(0.02,0.025]	3.92	3.27	(8,10]	1.31	0.0
(0.025,0.03]	0.65	0.65	Others	0.65	0.0
Others	8.5	16.34	-	-	-

图 9(a)表明:若采用分组优化技术,则在低误差区间的比例只有稍微下降.但从图 9(b)可以看出,分组优化技术明显加快了推理过程.另外,在权衡精度和时间时,OIAF 算法在未分组和分组情形下的  $F_2$  值分别为 0.744 0 和 0.832 1.综合分析得知,分组求解对 OIAF 算法具有一定的有效性.

## 5 总 结

针对概率知识库的在线查询场景,本文提出了一种基于近似因子的在线推理方法.其主要思想为:重复利用已推断结果计算查询变量的概率.该算法通过子图提取和添加近似因子的方式,在含有近似因子的子图上执行边缘推断,进而计算查询变量的概率.相对于已有算法,该算法能够在时间和精度上取得较好的权衡.另外,在对概率知识库进行增量式推理时,需要根据已推断结果来更新节点信息,故本文算法可进一步推广到增量式推理场景中.

## References:

- [1] Suchanek FM, Kasneci G, Weikum G. Yago: A core of semantic knowledge. In: Proc. of the 16th Int'l Conf. on World Wide Web (WWW 2007). Banff: ACM Press, 2007. 697–706. [doi: 10.1145/1242572.1242667]
- [2] Bollacker K, Evans C, Paritosh P, Sturge T, Taylor J. Freebase: A collaboratively created graph database for structuring human knowledge. In: Proc. of the ACM SIGMOD Int'l Conf. on Management of Data (SIGMOD 2008). Vancouver: ACM Press, 2008. 1247–1250. [doi: 10.1145/1376616.1376746]
- [3] Singhal A. Introducing the Knowledge Graph: Things, Not Strings. Official Google Blog, 2012.
- [4] Auer S, Bizer C, Kobilarov G, Lehmann J, Cyganiak R, Ives Z. Dbpedia: A nucleus for a Web of open data. In: Proc. of the 6th Int'l Semantic Web Conf., 2nd Asian Semantic Web Conf. (ISWC2007, ASWC 2007). Busan: Springer-verlag, 2007. 722–735. [doi: 10.1007/978-3-540-76298-0\_52]
- [5] Carlson A, Betteridge J, Kisiel B, Settles B, Hruschka ER, Mitchell TM. Toward an architecture for never-ending language learning. In: Proc. of the 24th AAAI Conf. on Artificial Intelligence (AAAI 2010). Atlanta: AAAI Press, 2010. 1306–1313.
- [6] Richardson M, Domingos P. Markov logic networks. Machine Learning, 2006,62(1):107–136. [doi: 10.1007/s10994-006-5833-1]
- [7] Zhang C. DeepDive: A data management system for automatic knowledge base construction. Madison: University of Wisconsin-Madison, 2015.
- [8] Niu F, Zhang C, Ré C, Shavlik J. Elementary: Large-Scale knowledge-base construction via machine learning and statistical inference. Int'l Journal on Semantic Web and Information Systems, 2012,8(3):42–73. [doi: 10.4018/jswis.2012070103]
- [9] Chen Y, Wang DZ. Knowledge expansion over probabilistic knowledge bases. In: Proc. of the 2014 ACM SIGMOD Int'l Conf. on Management of Data (SIGMOD 2014). Snowbird: ACM Press, 2014. 649–660. [doi: 10.1145/2588555.2610516]
- [10] Wick M, McCallum A, Miklau G. Scalable probabilistic databases with factor graphs and MCMC. Proc. of the VLDB Endowment, 2010,3(1):794–804. [doi: 10.14778/1920841.1920942]
- [11] Zhong P, Li ZH, Chen Q, Wang YY, Wang LP, Ahmed MHM, Fan FF. Poolside: An online probabilistic knowledge base for shopping decision support. In: Proc. of the 26th ACM Int'l Conf. on Information and Knowledge Management (CIKM 2017). Singapore: ACM Press, 2017. 2559–2562. [doi: 10.1145/3132847.3133168]
- [12] Zhou X, Chen Y, Wang DZ. ArchimedesOne: Query processing over probabilistic knowledge bases. Proc. of the VLDB Endowment, 2016,9(13):1461–1464. [doi: 10.14778/3007263.3007284]

- [13] Li K, Zhou X, Wang DZ, Grant C, Dobra A, Dudley C. In-Database batch and query-time inference over probabilistic graphical models using UDA—GIST. *The VLDB Journal*, 2017,26(2):177–201. [doi: 10.1007/s00778-016-0446-1]
- [14] Singla P, Kautz H, Luo J, Gallagher A. Discovery of social relationships in consumer photo collections using Markov logic. In: *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPRW 2008)*. Anchorage: IEEE Computer Society, 2008. 1–7. [doi: 10.1109/CVPRW.2008.4563047]
- [15] Singla P, Domingos P. Entity resolution with Markov logic. In: *Proc. of the 6th Int'l Conf. on Data Mining (ICDM 2006)*. Hong Kong: IEEE Computer Society, 2006. 572–582. [doi: 10.1109/ICDM.2006.65]
- [16] Poon H, Domingos P. Joint inference in information extraction. In: *Proc. of the 22nd AAAI Conf. on Artificial Intelligence*. Vancouver: AAAI Press, 2007. 913–918.
- [17] Bishop CM. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. New York: Springer-Verlag, 2006.
- [18] Singla P, Domingos P. Lifted first-order belief propagation. In: *Proc. of the 23th AAAI Conf. on Artificial Intelligence*. Chicago: AAAI Press, 2008. 1094–1099.
- [19] Van den Broeck G, Taghipour N, Meert W, Davis J, De Raedt L. Lifted probabilistic inference by first-order knowledge compilation. In: *Proc. of the 22nd Int'l Joint Conf. on Artificial Intelligence*. Barcelona: IJCAI Press, 2011. 2178–2185. [doi: 10.5591/978-1-57735-516-8/IJCAI11-363]
- [20] Kok S, Singla P, Richardson M, Domingos P. The Alchemy system for statistical relational AI. 2007. <http://www.cs.washington.edu/ai/alchemy>
- [21] Niu F, Ré C, Doan AH, Shavlik J. Tuffy: Scaling up statistical inference in Markov logic networks using an RDBMS. *Proc. of the VLDB Endowment*, 2011,4(6):373–384. [doi: 10.14778/1978665.1978669]
- [22] Shin J, Wu S, Wang F, De SC, Zhang C, Ré C. Incremental knowledge base construction using deepdive. *Proc. of the VLDB Endowment*, 2015,8(11):1310–1321. [doi: 10.14778/2809974.2809991]
- [23] Koller D, Friedman N, Wrote; Wang FY, Han SQ, *Trans. Probabilistic Graph Model*. Beijing: Tsinghua University Press, 2015 (in Chinese).

## 附中文参考文献:

- [23] Koller D, Friedman N, 著;王飞跃,韩素青,译.概率图模型.北京:清华大学出版社,2015.



王艳艳(1991—),女,山西吕梁人,博士生,主要研究领域为概率知识库.



钟评(1985—),男,博士生,主要研究领域为数据质量.



陈群(1976—),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为大数据管理,物联网信息管理.



李战怀(1961—),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为数据库理论与技术.