Towards Distributed MCMC Inference in Probabilistic Knowledge Bases

Mathias Niepert, Christian Meilicke, Heiner Stuckenschmidt
Universität Mannheim
Mannheim, Germany
{firstname}@informatik.uni-mannheim.de

Abstract

Probabilistic knowledge bases are commonly used in areas such as large-scale information extraction, data integration, and knowledge capture, to name but a few. Inference in probabilistic knowledge bases is a computationally challenging problem. With this contribution, we present our vision of a distributed inference algorithm based on conflict graph construction and hypergraph sampling. Early empirical results show that the approach efficiently and accurately computes a-posteriori probabilities of a knowledge base derived from a well-known information extraction system.

1 Introduction

In recent years, numerous applications of probabilistic knowledge bases have emerged. For instance, large-scale information extraction systems (Weikum and Theobald, 2010) aim at building knowledge bases by applying extraction algorithms to very large text corpora. Examples of such projects include KNOWITNOW (Cafarella et al., 2005), TEXTRUNNER (Etzioni et al., 2008), YAGO (Suchanek et al., 2007; Hoffart et al., 2011; Hoffart et al., 2010), and NELL (Carlson et al., 2010a; Carlson et al., 2010b). These systems face challenges of scalability both in terms of the degree of uncertainty and the sheer size of the resulting knowledge bases. Most of these projects combine pattern learning and matching approaches with some form of logical reasoning, with the majority of the systems employing weighted or unweighted first-order Horn clauses (Suchanek et al., 2007; Carlson et al., 2010a). More recently, random walk algorithms were applied to NELL's knowledge base to infer novel facts (Lao et al., 2011) and both pattern matching and reasoning algorithms were distributed on the HADOOP platform to enrich YAGO (Nakashole et al., 2011).

Similar to the distributed processes building indices for web search engines, there are distributed algorithms continuously building indices for structured knowledge (Carlson et al., 2010a). A combination of learned and manually specified common-sense rules is an important factor for the quality of the indexed knowledge. For the inference component of a large-scale information extraction system we propose a sampling approach consisting of two continuously running processes. The first process aggregates minimal conflict sets where each such set contradicts one or more of the common-sense rules. These conflicts are generated with relational queries and pattern-based approaches. The second component of the system is a sampling algorithm that operates on hypergraphs built from the minimal conflict set. The hypergraph is first decomposed into smaller disconnected sub-hypergraphs to allow distributed processing. Theoretical results on sampling independent sets from hypergraphs are leveraged to construct an ergodic Markov chain for probabilistic knowledge bases. The Markov chains are continuously run on the various connected components of the conflict hypergraph to compute a-posteriori

Proc. of the Joint Workshop on Automatic Knowledge Base Construction & Web-scale Knowledge Extraction (AKBC-WEKEX), pages 1–6,
NAACL-HLT, Montréal, Canada, June 7-8, 2012.

probabilities of individual logical statements which are in turn stored in a large relational index. While this is still work in progress, we have developed the theory, implemented the respective algorithms, and conducted first experiments.

2 Related Work

The presented representational framework is related to that of Markov logic (Richardson and Domingos, 2006) as the semantics is based on log-linear distributions. However, in this work we make the notion of consistency explicit by defining a log-linear distribution over consistent knowledge bases, that is, knowledge bases without logical contradictions. Moreover, the semantics of the knowledge bases is that of description logics which are commonly used for knowledge representation and exchange. There is existing work on distributing large-scale information extraction algorithms. For instance, pattern matching and reasoning algorithms were distributed on the HADOOP platform to enrich YAGO (Nakashole et al., 2011). However, these algorithms are not MCMC based and do not compute a-posteriori probabilities of individual statements. GraphLab (Low et al., 2010) is a recently developed parallel framework for distributing machine learning algorithms similar to MapReduce but better suited for classical learning algorithms. GraphLab was used to implement two parallel Gibbs samplers (Gonzalez et al., 2011). The approach is similar in that it identifies components of the graph (using graph coloring algorithms) from which one can sample in parallel without losing ergodicity. While not a distributed algorithm, query aware MCMC (Wick and McCallum, 2011) is a related approach in that it exploits the locality of the query to make MCMC more efficient.

3 Log-Linear Knowledge Bases

We believe that the common-sense rules should be stated in a representation language whose syntax and semantics is well-understood and standardized so as to support data and rule exchange between systems. Description logics are a commonly used representation for knowledge bases. There are numerous tools and standards for representing and reasoning with knowledge using description logics. The description logics framework allows one to represent both facts about individuals (concept and role assertions) as well as axioms expressing schema information. Log-linear description logics integrate description logics with probabilistic log-linear models (Niepert et al., 2011). The syntax of log-linear DLs is equivalent to that of the underlying DL except that it is possible to assign weights to general concept inclusion axioms (GCIs), role inclusion axioms (RIs), and assertions. We will use the term axiom to denote GCIs, RIs, and concept and role assertions. A log-linear knowledge base $\mathcal{K} = (\mathcal{K}^{\mathsf{D}}, \mathcal{K}^{\mathsf{U}})$ is a pair consisting of a deterministic knowledge base $\mathcal{K}^{\mathsf{D}}$ and an uncertain knowledge base $\mathcal{K}^{\mathsf{U}} = \{(c, w_c)\}$ with each c being an axiom and w_c a real-valued weight assigned to c. The deterministic KB contains axioms that are known to hold and the uncertain knowledge base contains the uncertain axioms. The greater the a-priori probability of an uncertain axiom the greater its weight. A set of axioms $\mathcal{A}$ is *inconsistent* if it has no model. A set of axioms $\mathcal{A}'$ is a *minimal inconsistency preserving subset* if it is inconsistent and every strict subset $\mathcal{A}'' \subset \mathcal{A}'$ is consistent.

The semantics of log-linear knowledge bases is based on probability distributions over consistent knowledge bases – the distribution assigns a non-zero probability only to consistent sets of axioms. For a log-linear knowledge base $\mathcal{K} = (\mathcal{K}^{\mathsf{D}}, \mathcal{K}^{\mathsf{U}})$ and a knowledge base $\mathcal{K}'$ with $\mathcal{K}^{\mathsf{D}} \subseteq \mathcal{K}' \subseteq \mathcal{K}^{\mathsf{D}} \cup \{c : (c, w_c) \in \mathcal{K}^{\mathsf{U}}\}$, we have that

$$\Pr_{\mathcal{K}}(\mathcal{K}') = \begin{cases} \frac{1}{Z} \exp\left(\sum_{\{c \in \mathcal{K}' \setminus \mathcal{K}^{\mathsf{D}}\}} w_c\right) & \text{if } \mathcal{K}' \textit{ consistent}; \\ 0 & \text{otherwise} \end{cases}$$

where Z is the normalization constant of the log-linear distribution $\Pr_{\mathcal{K}}$.

The weights of the axioms determine the log-linear probability (Koller and Friedman, 2009; Richardson and Domingos, 2006). The marginal probability of an axiom c given a log-linear knowledge base is the sum of the probabilities of the consistent knowledge bases containing c. Please note that an axiom with weight 0, that is, an a-priori probability of 0.5, which is not in conflict with other axioms has the a-posteriori probability of 0.5. Given these definitions, a Monte Carlo algorithm must sample consistent knowledge bases according to the distribution $\Pr_{\mathcal{K}}$. This seems daunting at first due to the reasoning complexity, the size of web-extracted knowledge bases, and the presence of

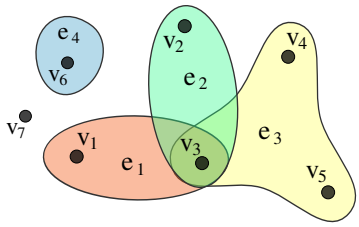

Figure 1: Hypergraph with 7 vertices (axioms) and 4 edges (conflict sets). Both the maximum degree of the hypergraph and the size of the largest edge are 3.

deterministic dependencies. However, we describe an approach with two separate distributable components. One that generates *minimal conflict sets* and one that leverages these conflict sets to build parallel Markov chains whose global unique stationary distribution is $\Pr_{\mathcal{K}}$.

4 Independent Sets in Hypergraphs

A hypergraph $G = (V, E)$ consists of a vertex set V and a set E of edges, where each edge is a subset of V. Let $m = \max\{|e| : e \in E\}$ be the size of the largest edge and let $\Delta = \max\{|\{e \in E : v \in e\}| : v \in V\}$ be the maximum degree of the graph. An independent set X in the hypergraph G is a subset of the vertex set V with $e \not\subseteq X$ for all $e \in E$. Let v be a vertex, let e be an edge with $v \in e$, and let $X \subseteq V$. If $v \notin X$ but, for all $u \in (e \setminus \{v\})$, we have that $u \in X$, then v is said to be *critical* for the edge e in X. Figure 1 depicts a hypergraph with 7 vertices and 4 edges.

Let $\mathcal{I}(G)$ be the set of all independent sets in the hypergraph G and let $\lambda \in \mathbb{R}_+$ be a positive parameter. The distribution π on $\mathcal{I}(G)$ is defined as

$$\pi(X) = \lambda^{|X|} / \sum_{X' \in \mathcal{I}(G)} \lambda^{|X'|}.$$

The problem of counting independent sets in graphs and hypergraphs (Dyer and Greenhill, 2000) was initially motivated by problems in statistical physics. While NP-hard in general, approximately counting independent sets in graphs is possible in polynomial time using the Markov Chain Monte Carlo method whenever a rapidly mixing Markov chain is available (Jerrum and Sinclair, 1996). Leveraging the theory of sampling independent sets from hypergraphs for efficient inference in probabilistic knowledge bases is straight-forward once the connection between consistent knowledge bases and independent sets in conflict hypergraphs is made.

5 Sampling Consistent Knowledge Bases

The set of inconsistency preserving subsets of a log-linear KB is denoted by $\mathcal{S}(\mathcal{K})$. This set is iteratively computed over the entire knowledge base consisting of *both* the known and the uncertain axioms. The conflict hypergraph is the projection of the minimal conflict sets onto the set of uncertain axioms.

Definition 1. *Let $\mathcal{K} = (\mathcal{K}^{\mathsf{D}}, \mathcal{K}^{\mathsf{U}})$ be a log-KB base and let $\mathcal{S}(\mathcal{K})$ be the set of all minimal conflict sets in $\mathcal{K}$. The* conflict hypergraph *$G = (V, E)$ of $\mathcal{K}$ is constructed as follows. For each axiom c in $\{c : (c, w_c) \in \mathcal{K}^{\mathsf{U}}\}$ we add one vertex v_c to V. For each minimal conflict set $S \in \mathcal{S}(\mathcal{K})$ we add the edge $\{v_c : c \in S \cap \{c : (c, w_c) \in \mathcal{K}^{\mathsf{U}}\}\}$ to E.*

Example 2. *Let* $\mathsf{Student}$ *and* $\mathsf{Professor}$ *be concepts,* $\mathsf{hasAdvisor}$ *an object property; and* Peter, Anna, *and* Bob *be distinct individuals. Now, let* $\mathcal{K}^{\mathsf{D}} = \{v_0 := \mathsf{Range}(\mathsf{hasAdvisor}) \sqcap \mathsf{Student} \sqsubseteq \bot,\ v_0' := \{\mathsf{Anna}\} \sqcap \mathsf{Student} \sqsubseteq \bot\}$ *and*

$$\mathcal{K}^{\mathsf{U}} = \left\{ \begin{array}{l} v_1 := \langle \mathsf{hasAdvisor}(\mathsf{Anna}, \mathsf{Peter}), 0.8 \rangle, \\ v_2 := \langle \mathsf{hasAdvisor}(\mathsf{Bob}, \mathsf{Peter}), 0.5 \rangle, \\ v_3 := \langle \mathsf{Student}(\mathsf{Peter}), 0.1 \rangle, \\ v_4 := \langle \mathsf{Student} \sqcap \mathsf{Professor} \sqsubseteq \bot, 0.9 \rangle, \\ v_5 := \langle \mathsf{Professor}(\mathsf{Peter}), 1.0 \rangle, \\ v_6 := \langle \mathsf{Student}(\mathsf{Anna}), 0.1 \rangle, \\ v_7 := \langle \mathsf{Professor}(\mathsf{Bob}), 0.4 \rangle \end{array} \right\}$$

Axiom v_0 expresses that advisors cannot be students and axiom v_0' expresses that Anna is not a student. Here, we have that $\mathcal{S}(\mathcal{K}) = \{\{v_0, v_1, v_3\}, \{v_0', v_6\}, \{v_0, v_2, v_3\}, \{v_3, v_4, v_5\}\}$. Figure 1 depicts the corresponding conflict hypergraph.

There is a one-to-one correspondence between independent sets of the hypergraph and consistent knowledge bases. Hence, analogous to sampling independent sets from hypergraphs we can now sample conflict-free knowledge bases from the *conflict hypergraph*. The difference is that each vertex v_c is weighted with w_c. Let $G = (V, E)$ be the conflict hypergraph and let m be the size of the largest edge in G. The following Markov chain $\mathcal{M}^{\mathbf{w}}(\mathcal{I}(G))$ samples independent sets from the conflict hypergraph

taking into account the weights of the axioms. If the chain is in state $X^{(t)}$ at time t, the next state $X^{(t+1)}$ is determined according to the following process:

- Choose a vertex $v_c \in V$ uniformly at random;
- If $v_c \in X^{(t)}$ then let $X^{(t+1)} = X^{(t)} \setminus \{v_c\}$ with probability $1/(\exp(w_c) + 1)$;
- If $v_c \in X^{(t)}$ and v_c is not critical in $X^{(t)}$ for any edge then let $X^{(t+1)} = X^{(t)} \cup \{v_c\}$ with probability $\exp(w_c)/(1 + \exp(w_c))$;
- If $v_c \in X^{(t)}$ and v_c is critical in $X^{(t)}$ for a unique edge e then with probability $(m - 1)\exp(w_c)/(2m(\exp(w_c) + 1))$ choose $w \in e \setminus \{v_c\}$ uniformly at random and let $X^{(t+1)} = (X^{(t)} \cup \{v_c\}) \setminus \{w\}$;
- Otherwise let $X^{(t+1)} = X^{(t)}$.

The following theorem is verifiable by showing that the Markov chain $\mathcal{M}^{\mathbf{w}}(\mathcal{I}(G))$ is aperiodic and irreducible and that $\Pr_{\mathcal{K}}$, projected onto the set of uncertain axioms, is a reversible distribution for the Markov chain.

Theorem 3. *Let $\mathcal{C} = (\mathcal{K}^{\mathsf{D}}, \mathcal{K}^{\mathsf{U}})$ be a log-linear knowledge base with conflict hypergraph G. Let $\Pr : \wp(\{c : (c, w_c) \in \mathcal{K}^{\mathsf{U}}\}) \to [0, 1]$ be a probability distribution. Then, $\Pr(U) = \Pr_{\mathcal{K}}(\mathcal{K}^{\mathsf{D}} \cup U)$ for every $U \subseteq \{c : (c, w_c) \in \mathcal{K}^{\mathsf{U}}\}$ if and only if $\Pr$ is the unique stationary distribution of $\mathcal{M}^{\mathbf{w}}(\mathcal{I}(G))$.*

The first component of the proposed approach accumulates minimal inconsistency preserving subsets. These minimal conflict sets can be efficiently computed with relational queries and pattern-based approaches and, therefore, are distributable. For instance, consider the common-sense rule "Students cannot be PhD advisors." To compute the sets of statements contradicting said rule, we process the conjunctive query "$\mathsf{hasAdvisor}(x, y) \wedge \mathsf{Student}(y)$." Each returned tuple corresponds to a minimal inconsistency preserving subset, that is, a set of statements that together contradicts the known rule. For instance, let us assume we execute the query "$\mathsf{hasAdvisor}(x, y) \wedge \mathsf{Student}(y)$" for the knowledge base in Example 2. The returned tuples are $(\mathsf{Anna}, \mathsf{Peter})$ and $(\mathsf{Bob}, \mathsf{Peter})$ corresponding to the minimal conflict sets $\{v_0, v_1, v_3\}$ and $\{v_0, v_2, v_3\}$. Again, since we can iteratively accumulate these sets of conflicts using relational joins we can distribute the process, for instance using a MAPREDUCE platform.

In order to facilitate distributed processing, the global conflict hypergraph is decomposed into its connected components. For instance, the conflict hypergraph in Figure 1 can be decomposed into the sub-hypergraphs induced by the partition of the nodes $\{\{v_1, v_2, v_3, v_4, v_5\}, \{v_6\}, \{v_7\}\}$. Markov chain for independent sets of hypergraphs are continuously run on the various conflict sub-hypergraphs to (re-)compute the a-posteriori probabilities of the statements.

6 Experiments

To assess the practicality of the approach, we conducted preliminary experiments focusing on the data and common-sense rules of the PROSPERA system due to the availability of recent results[1] (Nakashole et al., 2011). Each logical rule of the PROSPERA system was translated to a relational database query returning the minimal conflict sets *violating* said rule. For instance, for the common-sense PROSPERA rule "A student can have only one alma mater that she/he graduated from (with a doctoral degree)," the following relational query is executed: $\mathsf{graduatedFrom}(x, y) \wedge \mathsf{graduatedFrom}(x, y') \wedge \neg(y = y')$. For the rule "The advisor of a student must be older than her/his student" the query is $\mathsf{hasAdvisor}(x, y) \wedge \mathsf{bornOn}(x, y') \wedge \mathsf{bornOn}(y, y'') \wedge (y' > y'')$. Analogously, these queries can be formulated for the type constraints used by the PROSPERA system. Figure 2 depicts a subset of the minimal conflict sets in the academic domain of PROSPERA involving the object Albert Einstein.

For the preliminary experiments we used the academic domain facts that were extracted by PROSPERA without reasoning, and employed the common-sense rules mentioned in the description of PROSPERA[1] (Nakashole et al., 2011). The knowledge base has 384,816 bornOn, 59,933 facultyAt, 154,874 graduatedFrom, and 5,606 hasAcademicAdvisor assertions. Each assertion was assigned an a-priori probability of 0.5 except for bornOn assertions contained in YAGO which were

[1] http://www.mpi-inf.mpg.de/yago-naga/prospera/

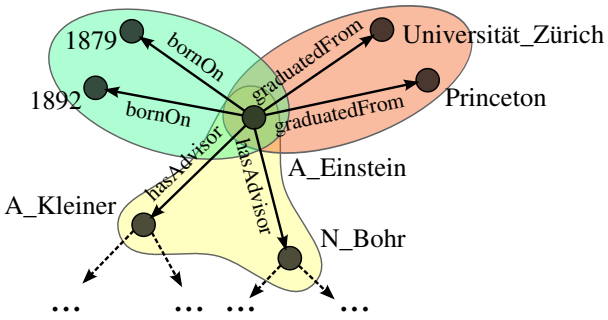

Figure 2: A knowledge base fragment with object A_Einstein and its properties. Some of the minimal conflicts between property assertions (edges in the graph) are indicated by hyperedges.

assigned an a-priori probability of 0.75. To build a gold standard for the evaluation, we selected 50 academics randomly for which the actual PhD advisor or the alma mater was present in the data. To compute the minimal conflict sets, we processed the join queries using a relational database system. After the construction of the conflict hypergraphs we ran the Markov chains for 10^5 iterations on the individual connected components.

In order to evaluate the marginal a-posteriori probabilities we computed the mean reciprocal rank measure (MRR) of the ranking induced by the computed marginal probabilities and compared it to the expected value of the MRR when no sampling is performed. The MRR measure (for example, see (Lao et al., 2011)) is defined as the inverse rank of the highest ranked correct result in a set of results. More formally, for a set of queries Q we have

$$\text{MRR} = \frac{1}{|Q|} \sum_{q \in Q} \frac{1}{\text{rank of first correct answer}}.$$

Table 1 list the averaged results of 100 experiments each with 50 queries. The columns $|E|$, m, and Δ are the averaged properties of the conflict hypergraphs the Markov chain was run on. t_c is the time needed to execute the relational queries for one connected component. The increase in MRR and precision@1 of the ranking induced by the a-posteriori probabilities over the initial ranking without sampling is statistically significant (paired t-test, $p < 0.01$). These results are encouraging and we are optimistic that they can be improved when individual a-priori weights of assertions are available.

sampling	$\lvert E\rvert$	m	Δ	t_s	MRR	p@1
no	-	-	-	-	0.35	0.24
yes	102.3	2.5	13.4	1.3	0.88	0.82

sampling	$\lvert E\rvert$	m	Δ	t_s	MRR	p@1
no	-	-	-	-	0.60	0.37
yes	250.2	2.4	27.0	2.2	0.86	0.81

Table 1: Empirical results for the probabilistic query $\mathsf{graduatedFrom}(\mathsf{Individual}, x)$ (top) and $\mathsf{hasAcademicAdvisor}(\mathsf{Individual}, x)$ (bottom). The values are averaged over 100 repetitions of the 50 probabilistic queries. t_s: seconds to compute samples for one connected component; MRR: mean reciprocal rank measure values; p@1: precision @ 1.

7 Discussion

Log-linear knowledge bases integrate description logics with probabilistic log-linear models. Since it is possible to express knowledge both on the schema and the instance level it allows the *explicit* representation of background knowledge that is already used implicitly by several information extraction systems such as PROSPERA. These systems employ the common-sense rules to ensure a high-quality knowledge base amid a high degree of uncertainty in the extraction process. The presented approach based on the generation of minimal conflict sets and hypergraph sampling is a first step towards a distributed sampling algorithm for structured knowledge extraction. We are also working on incorporating temporal information into the knowledge base (Dylla et al., 2011). We have developed the theory, namely the adaptation of Markov chains for independent sets in hypergraphs so as to incorporate individual node weights, implemented the respective algorithms, and conducted first experiments with the YAGO and PROSPERA datasets and rules. The robust implementation and distribution of the presented algorithms on a HADOOP cluster will be the main objective of future work. Moreover, in many real-world applications, the conflict hypergraph might not be decomposable without the removal of edges. Nevertheless, there are several hypergraph partitioning approaches that one could employ to find an finer-grained decomposition of the conflict hypergraph. We will also compare the presented approach to existing probabilistic inference algorithms such as belief propagation.

References

Michael J. Cafarella, Doug Downey, Stephen Soderland, and Oren Etzioni. 2005. Knowitnow: Fast, scalable information extraction from the web. In *Proceedings of the Conference on Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT-EMNLP)*.

A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E.R. Hruschka Jr, and T.M. Mitchell. 2010a. Toward an architecture for never-ending language learning. In *Proceedings of the 24th Conference on Artificial Intelligence (AAAI)*, pages 1306–1313.

A. Carlson, J. Betteridge, R. C. Wang, E. R. Hruschka, and T. M. Mitchell. 2010b. Coupled semi-supervised learning for information extraction. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining (WSDM 2010)*.

M. Dyer and C. Greenhill. 2000. On markov chains for independent sets. *Journal of Algorithms*, 35(1):17–49.

M. Dylla, M. Sozio, and M. Theobald. 2011. Resolving temporal conflicts in inconsistent rdf knowledge bases. In *14. GI-Fachtagung Datenbanksysteme für Business, Technologie und Web (BTW)*, pages 474–493.

O. Etzioni, M. Banko, S. Soderland, and D.S. Weld. 2008. Open information extraction from the web. *Communications of the ACM*, 51(12):68–74.

Joseph Gonzalez, Yucheng Low, Arthur Gretton, and Carlos Guestrin. 2011. Parallel gibbs sampling: From colored fields to thin junction trees. In *In Artificial Intelligence and Statistics (AISTATS)*.

J. Hoffart, F. M. Suchanek, K. Berberich, and G. Weikum. 2010. YAGO2: A Spatially and Temporally Enhanced Knowledge Base from Wikipedia. Research report, Max-Planck-Institut für Informatik.

J. Hoffart, F. M. Suchanek, K. Berberich, E. Lewis-Kelham, G. de Melo, and G. Weikum. 2011. Yago2: exploring and querying world knowledge in time, space, context, and many languages. In *Proceedings of the 20th international conference on World wide web (WWW)*, pages 229–232.

M. Jerrum and A. Sinclair. 1996. The markov chain monte carlo method: an approach to approximate counting and integration. In *Approximation algorithms for NP-hard problems*, pages 482–520. PWS Publishing.

D. Koller and N. Friedman. 2009. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.

N. Lao, T. Mitchell, and W. W. Cohen. 2011. Random walk inference and learning in a large scale knowledge base. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 529–539.

Yucheng Low, Joseph Gonzalez, Aapo Kyrola, Danny Bickson, Carlos Guestrin, and Joseph M. Hellerstein. 2010. Graphlab: A new parallel framework for machine learning. In *Conference on Uncertainty in Artificial Intelligence (UAI)*.

N. Nakashole, M. Theobald, and G. Weikum. 2011. Scalable knowledge harvesting with high precision and high recall. In *Proceedings of the 4th International Conference on Web Search and Data Mining (WSDM)*, pages 227–236.

M. Niepert, J. Noessner, and H. Stuckenschmidt. 2011. Log-linear description logics. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2153–2158.

M. Richardson and P. Domingos. 2006. Markov logic networks. *Machine Learning*, 62(1-2).

F. M. Suchanek, G. Kasneci, and G. Weikum. 2007. Yago: A Core of Semantic Knowledge. In *Proceedings of the 16th International World Wide Web Conference (WWW)*, pages 697–706.

G. Weikum and M. Theobald. 2010. From information to knowledge: harvesting entities and relationships from web sources. In *Proceedings of the 29th Symposium on the Principles of Database Systems (PODS)*, pages 65–76.

Michael L. Wick and Andrew McCallum. 2011. Query-aware mcmc. In *proceedings of the 25th Conference on Neural Information Processing Systems (NIPS)*, pages 2564–2572.