The Thirty-Second International Florida
Artificial Intelligence Research Society Conference (FLAIRS-32)

Integrating Typed Model Counting into First-Order Maximum Entropy Computations and the Connection to Markov Logic Networks

Marco Wilhelm,[1] **Gabriele Kern-Isberner,**[1] **Marc Finthammer,**[2] **Christoph Beierle**[2]
[1]Dept. of Computer Science, TU Dortmund, Dortmund, Germany
[2]Dept. of Computer Science, University of Hagen, Hagen, Germany

Abstract

The principle of maximum entropy (MaxEnt) provides a well-founded methodology for commonsense reasoning based on probabilistic conditional knowledge. We show how to calculate MaxEnt distributions in a first-order setting by using typed model counting and condensed iterative scaling. Further, we discuss the connection to Markov Logic Networks for drawing inferences.

Introduction

The *principle of maximum entropy* (MaxEnt) provides a well-founded methodology for commonsense reasoning based on probabilistic knowledge (Paris 2006). Primarily covered in a propositional setting (Kern-Isberner 2001), the MaxEnt principle was successfully adapted to deal with first-order probabilistic conditionals (Thimm and Kern-Isberner 2012). Although it shows expressive power when modeling uncertain knowledge about properties of and interactions among individual objects, the MaxEnt approach has not yet attracted as much interest as competing approaches in the field of *relational probabilistic programming* (Getoor and Taskar 2007; Raedt et al. 2008), mainly due to its computational intransparency. A systematic investigation of probabilistic first-order reasoning gave birth to *(symmetric) weighted first-order model counting* (WFOMC, (Van den Broeck et al. 2011)) as a central methodology.

In this paper, we develop efficient methods for MaxEnt reasoning in a first-order setting by lifting the ideas of WFOMC to conditional reasoning at maximum entropy. We discuss *typed model counting* (TMC, (Wilhelm et al. 2017)) as a variant of WFOMC which serves as a convenient framework for extracting the relevant information from a knowledge base that is necessary to determine the MaxEnt distribution. We present *condensed interative scaling* (CIS) as an optimization algorithm for calculating the MaxEnt distribution which builds upon the results of typed model counting. And we prove that knowledge bases can be compiled into Markov Logic Networks (MLNs, (Richardson and Domingos 2006)) with weights depending on the solution of the dual MaxEnt optimization problem. Hence, the MaxEnt approach can benefit from the sophisticated inference techniques that exist for MLNs (cf. Figure 1).

The rest of the paper is organized as follows: After a brief presentation of the relevant preliminaries on first-order probabilistic logic, the aggregating semantics, and on first-order maximum entropy reasoning, we discuss the connections of maximum entropy to Markov Logic Networks, condensed iterative scaling, and typed model counting, respectively. Eventually, we conclude and address future work.

Preliminaries

We consider a function-free first order language FOL over the signature $\Sigma = (\mathsf{Pred}, \mathsf{Const})$ consisting of finite sets of predicates Pred and constants Const. Formulas in FOL are built by using the common connectives $(\wedge, \vee, \neg)$ and quantifiers $(\exists, \forall)$. We abbreviate $A \wedge B$ with AB, $\neg A$ with $\overline{A}$, and $A \vee \overline{A}$ with $\top$. Variables are denoted by uppercase letters. If p/n is a predicate and $c_1, \ldots, c_n$ are constants, the formula $p(c_1, \ldots, c_n)$ is called a *ground atom*. A *ground literal* is a ground atom or its negation. Formulas can be *ground instantiated* by substituting each free variable by a constant (e.g., if $a \in \mathsf{Const}$, then $\forall X\ r(X, a)$ is a ground instance of $\forall X\ r(X, Y)$). The set of all ground instances of $A \in \mathsf{FOL}$ is denoted by $\mathsf{Gr}(A)$ and the set of the free variables in A by $\mathsf{Var}(A)$. Formulas without free variables are *closed*.

A *conditional* $\mathfrak{c} = (B|A)[\xi]$ with $A, B \in \mathsf{FOL}$ and $\xi \in [0, 1]$ is a formalization of the statement "If A holds, then B follows with probability ξ". It is called *non-deterministic* iff $\xi \notin \{0, 1\}$. A *ground instance* of $(B|A)[\xi]$ is obtained by ground instantiating A and B such that free variables mentioned in both A and B are substituted with the same constant (e.g., $(r(a, b)|p(a))[\xi]$ and $(r(a, a)|p(a))[\xi]$ are ground instances of $(r(X, Y)|p(X))[\xi]$ if $a, b \in \mathsf{Const}$ but $(r(a, b)|p(b))[\xi]$ is not). The set of all ground instances of $\mathfrak{c}$ is denoted by $\mathsf{Gr}(\mathfrak{c})$ and the set of the free variables in $\mathfrak{c}$, i.e. in A or B, by $\mathsf{Var}(\mathfrak{c})$. One has $|\mathsf{Gr}(\mathfrak{c})| = |\mathsf{Const}|^{|\mathsf{Var}(\mathfrak{c})|}$.

A *knowledge base* $\mathcal{K} = (\mathcal{F}, \mathcal{C})$ consists of a finite set of closed formulas $\mathcal{F}$ (representing the *factual knowledge* of an agent) and a finite set of non-deterministic conditionals $\mathcal{C}$ (representing her *conditional beliefs*). For the rest of the paper, let $\mathcal{C} = \{\mathfrak{c}_1, \ldots, \mathfrak{c}_n\}$ with $\mathfrak{c}_i = (B_i|A_i)[\xi_i]$.

Example 1. *Let* $\mathsf{Pred} = \{\mathsf{bird}/1, \mathsf{fly}/1\}$ *and* $\mathsf{coco} \in \mathsf{Const}$.

$\mathcal{K}_{\mathsf{brd}} = (\{\mathsf{bird}(\mathsf{coco})\},$

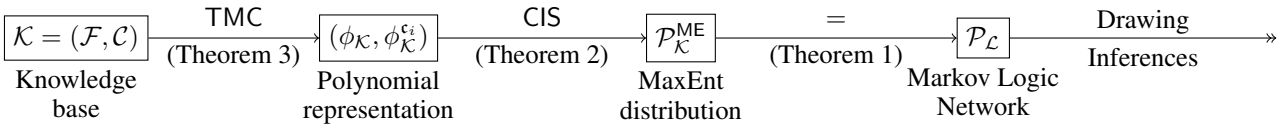

Figure 1: Drawing inferences from first-order probabilistic conditional knowledge at maximum entropy via MLNs.

$$\{(\mathsf{fly}(X)|\mathsf{bird}(X))[0.8], (\mathsf{fly}(\mathsf{coco})|\top)[0.1]\})$$

states that birds typically can fly, here with probability 0.8*, while Coco is assumed to be an abnormal bird which is able to fly with probability* 0.1 *only. Note that probabilities are understood as a reasoner's degree of belief.*

The semantics of knowledge bases is given by probability distributions over *possible worlds*. A possible world ω is a complete conjunction of ground literals, i.e., every ground atom occurs in ω exactly once, either negated or positive. The set of all possible worlds is denoted by Ω, and $\Omega_\mathcal{F} = \{\omega \in \Omega \mid \forall F \in \mathcal{F} : \omega \models F\}$ denotes the set of possible worlds in which all formulas in $\mathcal{F}$ are true. As conditionals may mention free variables, they cannot be interpreted as simple conditional probabilities. Instead, we make use of the *aggregating semantics* (Thimm and Kern-Isberner 2012).

Definition 1. *A (probability) distribution* $\mathcal{P} : \Omega \to [0, 1]$ *is a* model *of a closed formula* F *iff* $\omega \not\models F$ *implies* $\mathcal{P}(\omega) = 0$*, and of a conditional* $(B|A)[\xi]$ *iff*

$$\frac{\sum_{(B'|A')[\xi] \in \mathsf{Gr}(\mathfrak{c})} \mathcal{P}(A'B')}{\sum_{(B'|A')[\xi] \in \mathsf{Gr}(\mathfrak{c})} \mathcal{P}(A')} = \xi, \tag{1}$$

where $\mathcal{P}(A) = \sum_{\omega \models A} \mathcal{P}(\omega)$*.* $\mathcal{P}$ *is a model of a knowledge base* $\mathcal{K} = (\mathcal{F}, \mathcal{C})$ *iff it models every* $F \in \mathcal{F}$ *and every* $\mathfrak{c} \in \mathcal{C}$*.*

The aggregating semantics captures the definition of conditional probabilities by summing up the probabilities of the ground instances of the conditional, thereby providing a declarative semantics for first-order knowledge bases.

Example 2. *Recall* $\mathfrak{c} = (\mathsf{fly}(X)|\mathsf{bird}(X))[0.8]$ *from Example 1. A distribution* $\mathcal{P} : \Omega \to [0, 1]$ *models* $\mathfrak{c}$ *iff*

$$\frac{\sum_{c \in \mathsf{Const}} \mathcal{P}(\mathsf{bird}(c)\mathsf{fly}(c))}{\sum_{c \in \mathsf{Const}} \mathcal{P}(\mathsf{bird}(c))} = 0.8.$$

Example 2 suggests to understand the aggregating semantics in the following way: The relative frequency of the flying birds measured against all birds, in which the single events (an individual c is a bird or a flying bird) are weighted by the degree of belief with which the agent assumes that the certain events happen, has to equal 0.8. If $\mathcal{P}$ is the Dirac distribution assigning the probability 1 to a single possible world ω, i.e., the agent is certain that ω represents the real world, then (1) means counting relative frequencies in ω. In the contrary, if $\mathcal{P}$ is the uniform distribution on $\Omega_\mathcal{F}$ which means that the agent is maximally unconfident with her beliefs, then (1) means counting relative frequencies spread over all possible worlds. If $|\mathsf{Gr}(\mathfrak{c})| = 1$, then (1) equals conditional probabilities.

Consistent knowledge bases, i.e. knowledge bases with at least one model, typically have infinitely many models, and reasoning based on the whole set of models leads to monotonic and often uninformative inferences. Hence, for reasoning tasks, it is expedient to select a certain model among them. The choice that fits best to common sense is the unique model which maximizes entropy, according to (Paris 2006).

Definition 2. *Let* $\mathcal{K}$ *be a consistent knowledge base. Then,*

$$\mathcal{P}^{\mathsf{ME}}_\mathcal{K} = \arg\max_{\mathcal{P} \models \mathcal{K}} - \sum_{\omega \in \Omega} \mathcal{P}(\omega) \cdot \log \mathcal{P}(\omega) \tag{2}$$

is called the maximum entropy distribution *for* $\mathcal{K}$*. In (2) the convention* $0 \cdot \log 0 = 0$ *applies.*

Under the aggregating semantics, (2) is a convex optimization problem. Thus, $\mathcal{P}^{\mathsf{ME}}_\mathcal{K}$ for a consistent $\mathcal{K}$ is guaranteed to exist and is unique. As the dimension of problem (2) is exponential in $|\mathsf{Const}|$, (2) is usually reduced to its $n+1$-dimensional *dual* optimization problem by applying the method of Lagrange multipliers (Kern-Isberner 2001) in order to determine $\mathcal{P}^{\mathsf{ME}}_\mathcal{K}$. One obtains

$$\mathcal{P}^{\mathsf{ME}}_\mathcal{K}(\omega) = \begin{cases} \alpha_0 \cdot \prod_{i=1}^n \alpha_i^{f_i(\omega)}, & \omega \in \Omega_\mathcal{F} \\ 0, & \omega \in \Omega \setminus \Omega_\mathcal{F} \end{cases}, \tag{3}$$

where $f_i(\omega) = \mathsf{ver}_{\mathfrak{c}_i}(\omega) - \xi_i \cdot \mathsf{app}_{\mathfrak{c}_i}(\omega)$ is the *feature function* of the i-th conditional in $\mathcal{C}$ and

$$\mathsf{app}_\mathfrak{c}(\omega) = |\{(B'|A')[\xi] \in \mathsf{Gr}(\mathfrak{c}) \mid \omega \models A'\}|,$$
$$\mathsf{ver}_\mathfrak{c}(\omega) = |\{(B'|A')[\xi] \in \mathsf{Gr}(\mathfrak{c}) \mid \omega \models A'B'\}|,$$

are the numbers of ground instances of the conditional $\mathfrak{c}$ that are *applicable* ($\mathsf{app}_\mathfrak{c}$) resp. *verified* ($\mathsf{ver}_\mathfrak{c}$) in the possible world ω. Further, $\vec{\alpha}^{\mathsf{ME}}_\mathcal{K} = (\alpha_0, \alpha_1, \ldots, \alpha_n) \in \mathbb{R}^{n+1}_{>0}$ are exponentials of the Lagrange multipliers and have to satisfy

$$\sum_{\omega \in \Omega_\mathcal{F}} f_i(\omega) \cdot \prod_{j=1}^n \alpha_j^{f_j(\omega)} = 0, \quad i = 1, \ldots, n, \tag{4}$$

$$\sum_{\omega \in \Omega_\mathcal{F}} \prod_{j=1}^n \alpha_j^{f_j(\omega)} = \alpha_0. \tag{5}$$

Equations (4) are the aggregating semantics for $\mathfrak{c}_i \in \mathcal{C}$ with $\mathcal{P} = \mathcal{P}^{\mathsf{ME}}_\mathcal{K}$ and (5) assures $\sum_{\omega \in \Omega} \mathcal{P}^{\mathsf{ME}}_\mathcal{K}(\omega) = 1$.

Eventually, the maximum entropy distribution yields the *nonmonotonic inference relation* $\mathcal{K} \mathrel{|\!\sim_{\mathsf{ME}}} \mathfrak{c}$ iff $\mathcal{P}^{\mathsf{ME}}_\mathcal{K} \models \mathfrak{c}$ between the knowledge base $\mathcal{K}$ and a query conditional $\mathfrak{c}$ (Kern-Isberner 2001). Using the conditional $\mathfrak{c} = (F|\top)[1]$ where F is a closed formula, the relation $\mathrel{|\!\sim_{\mathsf{ME}}}$ also affords one to infer the factual knowledge F.

MaxEnt and Markov Logic Networks

Markov Logic Networks (MLNs, (Richardson and Domingos 2006)) constitute a popular approach in the field of statistical relational learning (cf. (Getoor and Taskar 2007)) by

combining probabilistic graphical models, namely Markov Random Fields, and first-order logic. For MLNs there exist well-investigated techniques for both exact and approximative inference. We show that the MaxEnt distribution $\mathcal{P}_{\mathcal{K}}^{\mathsf{ME}}$ can be compiled into an MLN and therefore benefits from the inference techniques for MLNs.

A *Markov Logic Network* $\mathcal{N}$ is a set of pairs (F_i, ν_i) consisting of formulas $F_i \in \mathsf{FOL}$ and weights $\nu_i \in \mathbb{R}$ which, in the context of a finite set of constants, define a probability distribution $\mathcal{P}_{\mathcal{L}} : \Omega \to [0,1]$ by

$$\mathcal{P}_{\mathcal{L}}(\omega) = \frac{1}{\zeta} \cdot \exp\left(\sum_i \nu_i \cdot \mathsf{cnt}(F_i, \omega)\right),$$

where ζ is a normalization constant and

$$\mathsf{cnt}(F_i, \omega) = |\{F_i' \in \mathsf{Gr}(F_i) \mid \omega \models F_i'\}|.$$

In order to represent hard constraints, i.e. factual knowledge, it is convenient to admit infinite weights.

Theorem 1. *Let $\mathcal{K} = (\mathcal{F}, \mathcal{C})$ be a consistent knowledge base, let $\vec{\alpha}_{\mathcal{K}}^{\mathsf{ME}} = (\alpha_0, \alpha_1, \ldots, \alpha_n)$, and let $\mathcal{L}$ be the* MLN *defined by $(A_i B_i, (1-\xi_i) \cdot \log \alpha_i), (A_i \overline{B_i}, -\xi_i \cdot \log \alpha_i) \in \mathcal{L}$ for $\mathfrak{c}_i \in \mathcal{C}$ and $(\overline{F}, -\infty) \in \mathcal{L}$ for $F \in \mathcal{F}$. Then, $\mathcal{P}_{\mathcal{L}} = \mathcal{P}_{\mathcal{K}}^{\mathsf{ME}}$.*

Proof. Let $\omega \in \Omega \setminus \Omega_{\mathcal{F}}$. There is at least one $F' \in \mathcal{F}$ with $\omega \not\models F'$ but $\omega \models \overline{F'}$, hence $\mathsf{cnt}(\overline{F'}, \omega) = 1$, and

$$\begin{aligned}
\mathcal{P}_{\mathcal{L}}(\omega) = {} & \lim_{\nu \to -\infty} \tfrac{1}{\zeta} \cdot \exp\big(\textstyle\sum_{F \in \mathcal{F}} \nu \cdot \mathsf{cnt}(\overline{F}, \omega) \\
& + \textstyle\sum_{\mathfrak{c}_i \in \mathcal{C}} (1 - \xi_i) \cdot \log(\alpha_i) \cdot \mathsf{cnt}(A_i B_i, \omega) \\
& + \textstyle\sum_{\mathfrak{c}_i \in \mathcal{C}} -\xi_i \cdot \log(\alpha_i) \cdot \mathsf{cnt}(A_i \overline{B_i}, \omega)\big) \\
= {} & \lim_{\nu \to -\infty} \tfrac{1}{\zeta} \cdot \exp(\nu \cdot \mathsf{cnt}(\overline{F}, \omega)) = 0 = \mathcal{P}_{\mathcal{K}}^{\mathsf{ME}}(\omega).
\end{aligned}$$

Now, let $\omega \in \Omega_{\mathcal{F}}$. Then, $\mathsf{cnt}(\overline{F}, \omega) = 0$ for all $F \in \mathcal{F}$, and

$$\begin{aligned}
\mathcal{P}_{\mathcal{L}}(\omega) = {} & \tfrac{1}{\zeta} \cdot \exp\big(\textstyle\sum_{\mathfrak{c}_i \in \mathcal{C}} (1 - \xi_i) \cdot \log(\alpha_i) \cdot \mathsf{cnt}(A_i B_i, \omega) \\
& + \textstyle\sum_{\mathfrak{c}_i \in \mathcal{C}} -\xi_i \cdot \log(\alpha_i) \cdot \mathsf{cnt}(A_i \overline{B_i}, \omega)\big) \\
= {} & \tfrac{1}{\zeta} \cdot \textstyle\prod_{i=1}^{n} \alpha_i^{(1-\xi_i) \cdot \mathsf{cnt}(A_i B_i, \omega)} \cdot \prod_{i=1}^{n} \alpha_i^{-\xi_i \cdot \mathsf{cnt}(A_i \overline{B_i}, \omega)} \\
= {} & \tfrac{1}{\zeta} \cdot \textstyle\prod_{i=1}^{n} \alpha_i^{\mathsf{cnt}(A_i B_i, \omega) - \xi_i \cdot (\mathsf{cnt}(A_i B_i, \omega) + \mathsf{cnt}(A_i \overline{B_i}, \omega))} \\
= {} & \tfrac{1}{\zeta} \cdot \textstyle\prod_{i=1}^{n} \alpha_i^{\mathsf{ver}_{r_i}(\omega) - \xi_i \cdot \mathsf{app}_{r_i}(\omega)} = \mathcal{P}_{\mathcal{K}}^{\mathsf{ME}}(\omega)
\end{aligned}$$

with the normalizing constant $\zeta = \alpha_0$. □

Usually, the weights of MLNs are learned from relational databases by maximum likelihood estimation without being declarative. Theorem 1 provides a semantically meaningful choice of the weights. In particular, the probabilities of the given conditionals are established. Note that the weights in Theorem 1 are not simply these probabilities but require to solve the dual MaxEnt optimization problem in order to get $\vec{\alpha}_{\mathcal{K}}^{\mathsf{ME}}$. We focus on computing $\vec{\alpha}_{\mathcal{K}}^{\mathsf{ME}}$ in the rest of the paper.

MaxEnt and Condensed Iterative Scaling

In principal, the dual MaxEnt optimization problem (3)-(5) can be solved by any method of non-linear convex optimization. In our situation, establishing the input of these methods in the naïve way depends exponentially on $|\mathsf{Const}|$. To avoid this, it is necessary to evaluate the sums in (4) and (5) without iterating over the whole set Ω but in a more condensed way, and to have an optimization algorithm for the dual MaxEnt optimization problem that utilizes this condensed form of (4) and (5). We counter the task of determining (4) and (5) efficiently by a variant of first-order weighted model counting (Van den Broeck et al. 2011) in the next section and focus on the optimization algorithm now.

Input: Consistent knowledge base $\mathcal{K}$, precision ϵ
Output: Approximation $\overrightarrow{\alpha}_{\mathcal{K}}^{\star}$ of $\vec{\alpha}_{\mathcal{K}}^{\mathsf{ME}}$

1. $G = \sum_{i=1}^{n} |\mathsf{Gr}(\mathfrak{c}_i)|$
2. $k = 0$
3. **FOR** $i = 1..n$ **:** $\alpha_i^k = 1$
4. **REPEAT**
   (a) $k = k + 1$
   (b) **FOR** $i = 1..n$ **:** $\alpha_i^k = \alpha_i^{k-1} \cdot \left(1 + \frac{\Phi_{\mathcal{K}}^{\mathfrak{c}_i, k-1}}{p_i |\mathsf{Gr}(\mathfrak{c}_i)| \cdot \Phi_{\mathcal{K}}^{k-1}}\right)^{-1/G}$
   UNTIL $|\alpha_i^k - \alpha_i^{k-1}| < \epsilon$ for $i = 1..n$ **HOLDS**
5. $\alpha_0^k = (\Phi_{\mathcal{K}}^k)^{-1}$
6. **RETURN** $\overrightarrow{\alpha}_{\mathcal{K}}^{\star} = (\alpha_0^k, \alpha_1^k, \ldots, \alpha_n^k)$

Here, $\Phi_{\mathcal{K}}^k = \phi_{\mathcal{K}}(\alpha_1^k, \ldots, \alpha_n^k, (\alpha_1^k)^{-\xi_1}, \ldots, (\alpha_n^k)^{-\xi_n})$ and $\Phi_{\mathcal{K}}^{\mathfrak{c}_i, k} = \phi_{\mathcal{K}}^{\mathfrak{c}_i}(\alpha_1^k, \ldots, \alpha_n^k, (\alpha_1^k)^{-\xi_1}, \ldots, (\alpha_n^k)^{-\xi_n}, -\xi_i)$.

Figure 2: Algorithm CIS.

Let $\mathcal{K}$ be a knowledge base, and let $\mathfrak{c}$ be a conditional. We define the polynomials $\Phi_{\mathcal{K}}$ and $\Phi_{\mathcal{K}}^{\mathfrak{c}}$ in the polynomial ring $\mathbb{Z}[x_1, \ldots, x_n, y_1, \ldots, y_n, z]$ by

$$\begin{aligned}
\phi_{\mathcal{K}}^{\mathfrak{c}} &= \sum_{\omega \in \Omega_{\mathcal{F}}} (\mathsf{ver}_{\mathfrak{c}}(\omega) + z \cdot \mathsf{app}_{\mathfrak{c}}(\omega)) \cdot \prod_{j=1}^{n} x_j^{\mathsf{ver}_{\mathfrak{c}_j}(\omega)} \cdot y_j^{\mathsf{app}_{\mathfrak{c}_j}(\omega)}, \\
\phi_{\mathcal{K}} &= \sum_{\omega \in \Omega_{\mathcal{F}}} \prod_{j=1}^{n} x_j^{\mathsf{ver}_{\mathfrak{c}_j}(\omega)} \cdot y_j^{\mathsf{app}_{\mathfrak{c}_j}(\omega)}. \qquad (6)
\end{aligned}$$

From these polynomials (4) and (5) can be obtained by substituting $\mathfrak{c}$ with $\mathfrak{c}_i$ and plugging in α_j for x_j, $\alpha_j^{-\xi_j}$ for y_j, both for $j = 1, \ldots, n$, and $-\xi_i$ for z. Based on this connection, our algorithm CIS (cf. Figure 2) solves the dual MaxEnt optimization problem given (6). CIS is a variation of *generalized iterative scaling* (GIS, (Darroch and Ratcliff 1972)) but, as CIS solves the dual optimization problem, it does not suffer from the expensive iterations over possible worlds like common GIS algorithms. Actually, $|\mathsf{Const}|$ has an influence on CIS only as a parameter if the same holds for $\phi_{\mathcal{K}}$ and $\phi_{\mathcal{K}}^{\mathfrak{c}_i}$. The correctness of CIS is stated in the following theorem.

Theorem 2. *The sequence $(\alpha_0^k, \alpha_1^k, \ldots, \alpha_n^k)_{k \in \mathbb{N}_0}$ as defined in* CIS *converges to $\vec{\alpha}_{\mathcal{K}}^{\mathsf{ME}}$ in any suitable vector norm $\|\cdot\|$.*

Proof. If one applies standard generalized iterative scaling to the primal MaxEnt problem (2), one iteratively scales the

probabilities of the possible worlds $\omega \in \Omega_{\mathcal{F}}$ via

$$\mathcal{P}^k(\omega) = \tfrac{1}{\eta} \cdot \mathcal{P}^{k-1}(\omega) \cdot \textstyle\prod_{j=1}^{n+1} (\beta_j^k)^{\hat{f}_j(\omega)}, \quad (7)$$

where k is the iteration index, η a normalizing constant, and

$$\beta_i^k = \frac{\hat{\epsilon}_i}{\sum_{\omega \in \Omega_{\mathcal{F}}} \mathcal{P}^{k-1}(\omega) \cdot \hat{f}_i(\omega)}, \quad i = 1, \ldots, n+1, \quad (8)$$

are scaling factors (cf. (Finthammer and Beierle 2014), also for the definitions of $\hat{\epsilon}_i$ and $\hat{f}_i(\omega)$). From this one can derive $\hat{\alpha}_i^k = \hat{\alpha}_i^{k-1} \cdot \beta_i^k$ with $\hat{\alpha}_0^k = 1$. Further, $\alpha_i^k = (\hat{\alpha}_i^k \cdot (\hat{\alpha}_{n+1}^k)^{-1})^{1/G}$ since $\hat{\alpha}_i^k$ is α_i^k up to normalization, such that

$$\alpha_i^k = \alpha_i^{k-1} \cdot \left(\frac{\beta_i^k}{\beta_{n+1}^k} \right)^{1/G}, \quad i = 1, \ldots, n. \quad (9)$$

As a direct consequence of both the convergence of GIS and the strong duality between the primal and the dual MaxEnt problem, it follows that $(\alpha_0^k, \alpha_1^k, \ldots, \alpha_n^k)_{k \in \mathbb{N}_0}$ converges to $\vec{\alpha}_{\mathcal{K}}^{\mathsf{ME}}$. It remains to show that the iteration specification (9) can be reformulated to Step 4b of CIS (the converges of $(\alpha_0^k)_{k \in \mathbb{N}_0}$ to α_0 is trivial then). For this, recursively plug the predecessor $\mathcal{P}^{j-1}(\omega)$ of $\mathcal{P}^j(\omega)$ into (7) and use that $\hat{\alpha}_i^0 = 1$ for $i = 1, \ldots, n$ and $\mathcal{P}^0(\omega) = |\Omega_{\mathcal{F}}|^{-1}$ for $\omega \in \Omega_{\mathcal{F}}$ (GIS starts iterating from the uniform distribution) which leads to

$$\mathcal{P}^k(\omega) = \tfrac{1}{\eta} \cdot \textstyle\prod_{j=1}^{n+1} (\hat{\alpha}_j^k)^{\hat{f}_j(\omega)}. \quad (10)$$

Now, insert (10) into (8) which removes the probabilities $\mathcal{P}^k(\omega)$ from the iteration specification for $\hat{\alpha}_i^k$ and get

$$\alpha_i^k = \alpha_i^{k-1} \cdot \left(\frac{\hat{\epsilon}_{n+1} \cdot \sum_{\omega \in \Omega_{\mathcal{F}}} \hat{f}_i(\omega) \cdot \prod_{j=1}^{n+1} (\hat{\alpha}_j^k)^{\hat{f}_j(\omega)}}{\hat{\epsilon}_i \cdot \sum_{\omega \in \Omega_{\mathcal{F}}} \hat{f}_{n+1}(\omega) \cdot \prod_{j=1}^{n+1} (\hat{\alpha}_j^k)^{\hat{f}_j(\omega)}} \right)^{-1/G}.$$

Finally, one gets the equality in Step 4b of CIS by plugging $\hat{\epsilon}_i$ and $\hat{f}_i(\omega)$ into the last equation. □

If the polynomials (6) are set up appropriately, CIS is able to outperform current algorithms for calculating $\mathcal{P}_{\mathcal{K}}^{\mathsf{ME}}$ significantly (cf. Table 1). In the next section, we discuss *typed model counting* as a formal framework for determining (6) in a very condensed way by exploiting the conditional logical structure of $\mathcal{K}$. With typed model counting it is possible to overcome the exponential dependence on $|\mathsf{Const}|$ for many classes of knowledge bases entirely.

MaxEnt and Typed Model Counting

First-order typed model counting (TMC, (Wilhelm et al. 2017)) is a variant of weighted first-order model counting (Van den Broeck et al. 2011) that aims to extend model counting by the ability to classify models into different *types*. These types are represented by elements of an algebraic structure that are directly incorporated into the formulas. In particular, TMC allows for representing conditionals as structured formulas, and for an efficient evaluation of their conditional logical structure. We recall the basic notations and definitions from TMC while adjusting them for the specific purpose of calculating the polynomials (6). This enables one to rewrite the side conditions (4) and (5) of the dual MaxEnt optimization problem in a more condensed way and to solve the optimization problem efficiently via CIS. We start with a simple example that illustrates our objective.

	Set-up		**Runtimes** in sec.		
$\mathcal{R}$	$\lvert\mathsf{Const}\rvert$	$\lvert\Omega_{\mathcal{F}}\rvert$	GIS^{α}	iGIS^{α}	CIS
$\mathcal{K}_{\mathsf{brd}}$	200	2^{399}	6.37	< 0.01	< 0.01
$\mathcal{K}_{\mathsf{brd}}$	300	2^{599}	19.91	< 0.01	< 0.01
$\mathcal{K}_{\mathsf{brd}}$	400	2^{799}	33.03	< 0.01	< 0.01
$\mathcal{K}_{\mathsf{mky}}$	30	2^{1830}	> 600	0.37	< 0.01
$\mathcal{K}_{\mathsf{mky}}$	60	2^{7260}	> 600	5.74	< 0.01
$\mathcal{K}_{\mathsf{mky}}$	90	2^{16290}	> 600	28.72	< 0.01
$\mathcal{K}_{\mathsf{smk}}$	30	2^{930}	> 600	> 600	0.29
$\mathcal{K}_{\mathsf{smk}}$	60	2^{3660}	> 600	> 600	3.27
$\mathcal{K}_{\mathsf{smk}}$	90	2^{8190}	> 600	> 600	6.89

Table 1: Runtime comparison of CIS with the algorithms GIS^{α} (Finthammer and Beierle 2014) and iGIS^{α} (Wilhelm et al. 2018) with respect to $\mathcal{K}_{\mathsf{brd}}$ from Example 1, $\mathcal{K}_{\mathsf{mky}} = (\emptyset, \{(f(X,Y)|h(Y))[0.2], (f(X,Y)|h(Y)r(X,Y))[0.9]\})$, and $\mathcal{K}_{\mathsf{smk}} = (\emptyset, \{(s(Y)|s(X)f(X,Y))[0.7]\})$. The termination condition was uniformly chosen as $k = 100$ such that all algorithms computed an output of the same precision. Runs were executed on an Intel Core i5-6200U processor with two cores and were canceled after 600 seconds.

Example 3. *Consider* $\mathcal{K}_{\mathsf{ex}} = (\emptyset, \{(\mathsf{fly}(X)|\mathsf{bird}(X))[0.8]\})$. *The naïve way of determining* $\phi_{\mathcal{K}_{\mathsf{ex}}}$ *(cf. (6)) requires to count the number of birds (*$\mathsf{app}_{\mathfrak{c}_1}(\omega)$*) as well as the number of flying birds (*$\mathsf{ver}_{r_1}(\omega)$*) in every single possible world* $\omega \in \Omega$ *independently which ends up in a sum with* $4^{|\mathsf{Const}|}$ *summands (the number of possible worlds).*

When applying some combinatorial arguments, $\phi_{\mathcal{K}_{\mathsf{ex}}}$ *can be represented much more compactly: If exactly* k *individuals are birds, there are* $\binom{|\mathsf{Const}|}{k}$ *many possible sets of individuals that could cover these birds. And if exactly* m *of the birds are able to fly, there are* $\binom{k}{m}$ *many possible combinations of them. The flying behavior of all other individuals (*$2^{|\mathsf{Const}|-k}$ *many) is irrelevant for the evaluation of* $\mathfrak{c}_1$*. Each combination of the mentioned characteristics corresponds to a possible world, and hence*

$$\phi_{\mathcal{K}_{\mathsf{ex}}} = \sum_{k=0}^{|\mathsf{Const}|} \sum_{m=0}^{k} \binom{|\mathsf{Const}|}{k} \binom{k}{m} \cdot 2^{|\mathsf{Const}|-k} \cdot x_1^m \cdot y_1^k.$$

This way, the number of summands in $\phi_{\mathcal{K}_{\mathsf{ex}}}$ *and evaluating* $\phi_{\mathcal{K}_{\mathsf{ex}}}$ *at a given point depend polynomially on* $|\mathsf{Const}|$.

An even more efficient way of setting up $\phi_{\mathcal{K}_{\mathsf{ex}}}$ *is to realize that the ground instances of* $\mathfrak{c}_1$ *can be evaluated independently, as the ground instances of* $\mathfrak{c}_1$ *correspond to different individuals. Every individual is possibly a flying bird (represented by the product* $x_1 \cdot y_1$*), a non-flying bird (*y_1*) or not a bird at all. In the last case it is irrelevant for the evaluation of* $\mathfrak{c}_1$ *whether the individual is able to fly or not (which results in a factor* 2*). As there are* $|\mathsf{Const}|$ *many individuals,*

$$\phi_{\mathcal{K}_{\mathsf{ex}}} = (x_1 \cdot y_1 + y_1 + 2)^{|\mathsf{Const}|}.$$

Here, $|\mathsf{Const}|$ *has an influence on* $\phi_{\mathcal{K}_{\mathsf{ex}}}$ *only as a parameter.*

Let $\mathbb{Z}[\mathcal{X}]$ be the polynomial ring over the set of variables $\mathcal{X} = \{x_1, \ldots, x_n, y_1, \ldots, y_n, z\}$. To anticipate the meaning

of the variables, note that x_i resp. y_i indicate the verification resp. the applicability of the i-th conditional of a knowledge base, and z serves as a placeholder for probabilities. Further, let $[\mathcal{X}]$ denote the set of monomials in $\mathbb{Z}[\mathcal{X}]$ (products of powers of the variables with nonnegative integer exponents).

Definition 3. *The* structured language $\mathsf{FOL}^{\mathsf{s}}$ *is defined by*

$$\Phi ::= A \mid \Phi \wedge \Phi \mid \Phi \vee \Phi \mid \exists X\ \Phi \mid \forall X\ \Phi \mid x \circ \Phi,$$

where $A \in \mathsf{FOL}$, $X \in \mathsf{Var}(B)$, $x \in [\mathcal{X}]$ *and* $\circ$ *is an outer operation between* $[\mathcal{X}]$ *and* $\mathsf{FOL}^{\mathsf{s}}$. $\circ$ *shall bind strongest.*[1]

The language $\mathsf{FOL}^{\mathsf{s}}$ consists of all formulas in FOL and additionally allows one to concatenate monomials from $[\mathcal{X}]$ to any part of a formula as long as they are not in the scope of negations. For example, if $p/1 \in \mathsf{Pred}$, $\forall X\ z \circ \neg p(X)$ is in $\mathsf{FOL}^{\mathsf{s}}$ but $\forall X\ \neg z \circ p(X)$ is not.

Definition 4. *For every* $\omega \in \Omega$, *the* structured interpretation $\mathcal{I}^{\mathsf{s}}_\omega$ *is a mapping from the set of closed structured formulas in* $\mathsf{FOL}^{\mathsf{s}}$ *to* $[\mathcal{X}] \cup \{0\}$ *and is recursively defined by*

1. $\mathcal{I}^{\mathsf{s}}_\omega(A) = \begin{cases} 1 & \text{if } \omega \models A \\ 0 & \text{otherwise} \end{cases}$,
2. $\mathcal{I}^{\mathsf{s}}_\omega(B \wedge C) = \mathcal{I}^{\mathsf{s}}_\omega(B) \cdot \mathcal{I}^{\mathsf{s}}_\omega(C)$,
3. $\mathcal{I}^{\mathsf{s}}_\omega(B \vee C) = \begin{cases} \mathcal{I}^{\mathsf{s}}_\omega(B) & \text{if } \mathcal{I}^{\mathsf{s}}_\omega(C) = 0 \\ \mathcal{I}^{\mathsf{s}}_\omega(C) & \text{if } \mathcal{I}^{\mathsf{s}}_\omega(B) = 0 \\ \mathcal{I}^{\mathsf{s}}_\omega(B) \cdot \mathcal{I}^{\mathsf{s}}_\omega(C) & \text{otherwise} \end{cases}$,
4. $\mathcal{I}^{\mathsf{s}}_\omega(\exists X\ B) = \mathcal{I}^{\mathsf{s}}_\omega(\bigvee_{c \in \mathsf{Const}} B[X/c])$,
5. $\mathcal{I}^{\mathsf{s}}_\omega(\forall X\ B) = \mathcal{I}^{\mathsf{s}}_\omega(\bigwedge_{c \in \mathsf{Const}} B[X/c])$,
6. $\mathcal{I}^{\mathsf{s}}_\omega(x \circ B) = x \cdot \mathcal{I}^{\mathsf{s}}_\omega(B)$,

where $A \in \mathsf{FOL}$, $B, C \in \mathsf{FOL}^{\mathsf{s}}$, $x \in [\mathcal{X}]$, $X \in \mathsf{Var}(B)$, *and* $B[X/c]$ *is* B *after substituting every occurrence of* X *by* c.

$\mathcal{I}^{\mathsf{s}}_\omega$ interprets closed formulas $A \in \mathsf{FOL}$ in the same way as classical interpretations do, whereby ω determines the truth assignment (i.e., $\mathcal{I}^{\mathsf{s}}_\omega(A) = 1$ iff $\omega \models A$). Only for formulas that mention elements from $[\mathcal{X}]$ the interpretation is multi-valued, registered in its *type*. The following example illustrates this by highlighting how both the presence and the position of elements from $[\mathcal{X}]$ affect the interpretation.

Example 4. *Let* $\omega \in \Omega$ *be a possible world which satisfies* $\omega \models p(c)$ *for all* $c \in \mathsf{Const}$. *Then,* $\mathcal{I}^{\mathsf{s}}_\omega(\forall X\ p(X)) = 1$, $\mathcal{I}^{\mathsf{s}}_\omega(z \circ \forall X\ p(X)) = z$, *and* $\mathcal{I}^{\mathsf{s}}_\omega(\forall X\ z \circ p(X)) = z^{|\mathsf{Const}|}$.

Definition 5. *A structured interpretation* $\mathcal{I}^{\mathsf{s}}_\omega$ *is a* model of a closed structured formula $A \in \mathsf{FOL}^{\mathsf{s}}$ *iff* $\mathcal{I}^{\mathsf{s}}_\omega(A) \neq 0$. *It is a* model of type $x \in [\mathcal{X}]$ *iff* $\mathcal{I}^{\mathsf{s}}_\omega(A) = x$. *The* typed model counting *task is calculating* $\mathsf{TMC}(A) = \sum_{\omega \in \Omega} \mathcal{I}^{\mathsf{s}}_\omega(A)$.

Theorem 3 below links the typed model counting task to determining the polynomials (6). For this, a compilation of knowledge bases $\mathcal{K}$ into structured formulas is needed and given by

$$\Psi_\mathcal{K} = \Psi_\mathcal{F} \wedge \Psi_\mathcal{C}, \tag{11}$$

where $\Psi_\mathcal{F} = \bigwedge_{F \in \mathcal{F}} F$ and

$$\Psi_\mathcal{C} = \bigwedge_{j=1}^{n} \forall X_{j,1} \ldots \forall X_{j,m_j} (y_j \circ A_j \wedge (x_j \circ B_j \vee \overline{B}_j) \vee \overline{A}_j),$$

where again $\mathsf{Var}(\mathfrak{c}_j) = \{X_{j,1}, \ldots, X_{j,m_j}\}$ for $j = 1, \ldots, n$.

The idea behind the formula $\Psi_\mathcal{C}$ is to build a conjunction over the conditionals in the knowledge base. In every conjunct, all ground instances of the respective conditional $\mathfrak{c}_j$ are considered by universal quantification over $\mathsf{Var}(\mathfrak{c}_j)$. For every ground instance, the possible ways of evaluating its logical part ((non-)applicability and verification) are represented, basically, by disjunction. The applicability is indicated by the variable y_j and the verification by x_j. By calculating the structured interpretation $\mathcal{I}^{\mathsf{s}}_\omega$ of the formula $\Psi_\mathcal{K}$, it is possible to read out from the exponents of x_j and y_j how often the conditional $\mathfrak{c}_j$ is applicable resp. verified in ω. Therewith, calculating $\mathsf{TMC}(\Psi_\mathcal{K})$ constitutes a formal method for calculating the counting functions $\mathsf{app}_{\mathfrak{c}_i}(\omega)$ and $\mathsf{ver}_{\mathfrak{c}_i}(\omega)$ for $i = 1, \ldots, n$ and all $\omega \in \Omega_\mathcal{F}$ *simultaneously*. Thereby, $\Psi_\mathcal{F}$ ensures that only those structured interpretations are considered that satisfy the facts in $\mathcal{F}$.

Example 5. *Recall the knowledge base* $\mathcal{K}_{\mathsf{brd}}$ *from Example 1. With* $b = \mathsf{bird}$, $f = \mathsf{fly}$, $c = \mathsf{coco}$, *one has*

$$\Psi_{\mathcal{K}_{\mathsf{brd}}} \equiv b(c) \wedge \forall X [y_1 b(X) \wedge (x_1 f(X) \vee \overline{f(X)}) \vee \overline{b(X)}] \wedge\ y_2\ (x_2 f(c)\ \vee \overline{f(c)}).$$

Applying the definitions of $\Psi_\mathcal{K}$ and TMC according to (11) and Definition 5 yields the following theorem.

Theorem 3. *Let* $\mathcal{K} = (\mathcal{F}, \mathcal{C})$ *be a consistent knowledge base and let* $\mathfrak{c}$ *be a conditional. The following equations hold.*[2]

$$\phi_\mathcal{K} = \mathsf{TMC}(\Psi_\mathcal{K}),$$

$$\phi^{\mathfrak{c}}_\mathcal{K} = \sum_{(B|A)[\xi] \in \mathsf{Gr}(\mathfrak{c})} (\mathsf{TMC}(\Psi_\mathcal{K} \wedge A \wedge B) + z \cdot \mathsf{TMC}(\Psi_\mathcal{K} \wedge A)).$$

To sum up so far, Theorem 3 enables one to compile knowledge bases into structured formulas while preserving the information about the conditional logical structure of the knowledge base. TMC on the structured formula in turn unveils this information. In order to perform TMC efficiently, it is necessary to compile the formula into an equivalent formula in normal form first. Closed formulas $A, B \in \mathsf{FOL}^{\mathsf{s}}$ are *equivalent* iff $\mathcal{I}^{\mathsf{s}}_\omega(A) = \mathcal{I}^{\mathsf{s}}_\omega(B)$ for all $\omega \in \Omega$. A normal form which is suitable for the typed model counting task is called sd-DNNF$^{\mathsf{s}}$ (Wilhelm et al. 2017) which is the structured counterpart to sd-DNNFs for classical first order formulas (Van den Broeck et al. 2011). As formulas in sd-DNNF, a structured formula F in sd-DNNF$^{\mathsf{s}}$ allows one to count models recursively due to the following properties:

- Every conjunction $A \wedge B$ in F is *decomposable*, i.e., A and B do not share[3] any ground atom, and therefore $\mathsf{TMC}(A \wedge B) = \mathsf{TMC}(A) \cdot \mathsf{TMC}(B)$ holds.
- Every disjunction $A \vee B$ in F is *deterministic*, i.e., A and B are mutually exclusive ($\mathcal{I}^{\mathsf{s}}_\omega(A) \cdot \mathcal{I}^{\mathsf{s}}_\omega(B) = 0$ for $\omega \in \Omega$), and therefore $\mathsf{TMC}(A \vee B) = \mathsf{TMC}(A) + \mathsf{TMC}(B)$ holds.
- Every universal/existential quantification in F is decomposable/deterministic over *isomorphic instances*, i.e., every two instances are equivalent up to a permutation of

[1] We sometimes omit the symbol $\circ$ in structured formulas.

[2] For the first equation, see also (Wilhelm et al. 2017).

[3] We say that the formulas A and B share the ground atom G iff there are $A' \in \mathsf{Gr}(A)$ and $B' \in \mathsf{Gr}(B)$ that both mention G.

constants. Hence, $\mathsf{TMC}(\forall X\ A) = \mathsf{TMC}(A[X/c])^{|\mathsf{Const}|}$ and $\mathsf{TMC}(\exists X\ A) = |\mathsf{Const}| \cdot \mathsf{TMC}(A[X/c])$, $c \in \mathsf{Const}$.

- F is *smooth*, i.e., every ground atom is mentioned in F, and every two disjuncts of a disjunction/instances of an existential quantification mention the same ground atoms. This guarantees that all models are count.

Example 6. *$\Psi_{\mathcal{K}_{\mathsf{brd}}}$ from Example 5 can be compiled into*[4]

$$F = \forall X_{\neq c}\,[y_1 b(X) \wedge (x_1 f(X) \vee \overline{f(X)}) \vee \overline{b(X)} \\ \wedge (f(X) \vee \overline{f(X)})] \wedge y_1 y_2 b(c) \wedge (x_1 x_2 f(c) \vee \overline{f(c)}).$$

F satisfies all the requirements of sd-DNNF$^{\mathsf{s}}$*s, e.g., it splits into two syntactically independent parts of which the first (universal quantification) deals with the constants other than* coco *while the second mentions* coco *only. Therefore, the conjunction of both parts is decomposable. The universal quantification itself is decomposable, too, since each instance refers to a different (unnamed) constant. Therewith,*[5]

$$\phi_{\mathcal{K}_{\mathsf{brd}}} = \mathsf{TMC}(\Psi_{\mathcal{K}_{\mathsf{brd}}}) = \rho^{m-1} \cdot \gamma_2 \cdot \chi_{2,1}.$$

Analogously, it follows that (we assume $m > 1$)

$$\phi^{r_1}_{\mathcal{K}_{\mathsf{brd}}} = \rho^{m-2} \cdot \gamma_2 \cdot [\rho \cdot (\chi_{2,0} + z \cdot \chi_{2,1}) \\ + (m-1) \cdot \gamma_1 \cdot \chi_{2,1} \cdot (\chi_{1,0} + z \cdot \chi_{1,1})],$$
$$\phi^{r_2}_{\mathcal{K}_{\mathsf{brd}}} = \rho^{m-1} \cdot \gamma_2 \cdot (\chi_{2,0} + z \cdot \chi_{2,1}).$$

In (Van den Broeck et al. 2011) one can find a detailed list of strategies for compiling formulas into sd-DNNF that preserve tractability with respect to the dependence on $|\mathsf{Const}|$ (e.g., skolemization is used to handle existential quantification). All these strategies can be adapted for TMC. Not covered by these strategies is the outer sum in $\Psi^{\mathfrak{c}}_{\mathcal{K}}$ (cf. Theorem 3) which ranges over the ground instances of the conditional $\mathfrak{c}$, as it is beyond the model counting task. This implicitly involves a dependence on $|\mathsf{Const}|$. However, following the principle of preemptive shattering (Poole, Bacchus, and Kisynski 2011), one can avoid this dependence. The basic idea of this technique is that all unnamed individuals, i.e. individuals that are not mentioned in the knowledge base $\mathcal{K}$, have the same influence on $\Psi^{\mathfrak{c}}_{\mathcal{K}}$, and hence are interchangeable and can be merged to a single prototypical individual.

Conclusion

Drawing inferences from first-order probabilistic conditional knowledge at maximum entropy (MaxEnt) involves three aspects: (1) Extracting the relevant information from the knowledge base, (2) calculating the MaxEnt distribution, and (3) drawing the inferences. For lifted inference, all three tasks may depend at most polynomially on the domain size. In this paper, we faced this challenge of first-order reasoning by pursuing the following strategy: (1) Perform typed model counting on formulas that reflect the given knowledge to unveil its conditional structure. (2) Calculate the MaxEnt distribution by condensed iterative scaling. (3) Compile the MaxEnt distribution into a Markov Logic Network for which sophisticated inference techniques exist.

In future work, we want to further investigate and automate typed model counting. Also, we want to discern for which knowledge bases and for which queries lifted inference at maximum entropy is possible.

[4] Here, $\forall X_{\neq c}$ means "for all constants, except for constant c".

[5] We abbreviate $m = |\mathsf{Const}|$ and $\chi_{i,k} = \prod_{j=1}^{i} x_j + k$ as well as $\gamma_i = \prod_{j=1}^{i} y_j$ and $\rho = \gamma_1 \cdot \chi_{1,1} + 2$ to shorten formulas.

Acknowledgments. This research was supported by the German National Science Foundation (DFG) Research Unit FOR 1513 on Hybrid Reasoning for Intelligent Systems.

References

Darroch, J. N., and Ratcliff, D. 1972. Generalized iterative scaling for log-linear models. *Annals of Mathematical Statistics* 43(5):1470–1480.

Finthammer, M., and Beierle, C. 2014. A two-level approach to maximum entropy model computation for relational probabilistic logic based on weighted conditional impacts. In *Proceedings of 8th SUM Conference*, volume 8720 of *LNCS*, 162–175. Springer.

Getoor, L., and Taskar, B., eds. 2007. *Introduction to Statistical Relational Learning*. MIT Press.

Kern-Isberner, G. 2001. *Conditionals in Nonmonotonic Reasoning and Belief Revision*. Springer.

Paris, J. B. 2006. *The Uncertain Reasoner's Companion: A Mathematical Perspective*. Cambridge University Press.

Poole, D.; Bacchus, F.; and Kisynski, J. 2011. Towards completely lifted search-based probabilistic inference. *CoRR* abs/1107.4035.

Raedt, L. D.; Frasconi, P.; Kersting, K.; and Muggleton, S. H., eds. 2008. *Probabilistic Inductive Logic Programming*. Springer.

Richardson, M., and Domingos, P. 2006. Markov logic networks. *Machine Learning* 62(1-2):107–136.

Thimm, M., and Kern-Isberner, G. 2012. On probabilistic inference in relational conditional logics. *Logic Journal of the IGPL* 20(5):872–908.

Van den Broeck, G.; Taghipour, N.; Meert, W.; Davis, J.; and De Raedt, L. 2011. Lifted probabilistic inference by first-order knowledge compilation. In *Proceedings of 22th IJCAI Conference*, 2178–2185. AAAI Press.

Wilhelm, M.; Finthammer, M.; Kern-Isberner, G.; and Beierle, C. 2017. First-order typed model counting for probabilistic conditional reasoning at maximum entropy. In *Proceedings of 11th SUM Conference*, volume 10564 of *LNCS*, 266–279. Springer.

Wilhelm, M.; Kern-Isberner, G.; Finthammer, M.; and Beierle, C. 2018. A generalized iterative scaling algorithm for maximum entropy model computations respecting probabilistic independencies. In *Proceedings of 10th FOIKS Conference*, 379–399. Springer.