Lifted Weight Learning of Markov Logic Networks Revisited (Appendix)

Ondřej Kuželka
Czech Technical University in Prague
Dept of Computer Science, KU Leuven

Vyacheslav Kungurtsev
Czech Technical University in Prague

A COMPUTING POLYTOPES FOR 2-VARIABLE FORMULAS

In this section we describe an algorithm for constructing relational marginal polytopes given by sets of first-order formulas, each with at most 2 logical variables. The algorithm described in this section is largely inspired by the WFOMC algorithm from [1]. In what follows in this section, we will denote by Ω_{Φ_0} the set of possible worlds over domain $\Delta = \{c_1, \dots, c_{|\Delta|}\}$ which satisfy a given set Φ_0 of universally quantified first-order logic sentences.[1]

We need an algorithm which can compute the set $\mathcal{K}(\Phi, \Omega_{\Phi_0})$ defined in Section 5. Let $\mathcal{U}$ be the set of all unary predicates in the considered first-order language $\mathcal{L}$ and $\mathcal{B}$ be the set of all binary predicates (for 2-variable formulas, we may assume w.l.o.g.[2] that $\mathcal{L}$ does not contain any literals of arity higher than 2). In the following, we will use the notion of *cells*, which was also used in [1]. Given a possible world ω, we say that two constants $c, c' \in \Delta$ are in the same *cell* if for all $u \in \mathcal{U}$ we have $\omega \models u(c)$ iff $\omega \models u(c')$; each cell can then be identified by a subset of $\mathcal{U}$ naturally.

Remark 1. *Suppose that $\mathcal{B} = \emptyset$ (i.e. we only have unary predicates) and that Φ_0 and Φ are constant-free. Then we can construct the set $\mathcal{K}(\Phi, \Omega_{\Phi_0})$ in polynomial time as follows. First, we construct an auxiliary set of all integer partitions of $|\Delta|$:*

$$\mathcal{J} = \left\{ (j_1, \dots, j_{|2^{\mathcal{U}}|}) \middle| \sum_{k=1}^{|2^{\mathcal{U}}|} j_k = |\Delta| \wedge \forall k : j_k \geq 0 \right\}$$

The intention is that the i-th entry of a vector $J \in \mathcal{J}$ should represent the number of constants $c \in \Delta$ that are in the i-th cell (here the cells will be ordered arbitrarily in some order). We can then use the set $\mathcal{J}$ to define a set of possible worlds $\Omega_R \subseteq \Omega_{\Phi_0}$ which will be representative of all the possible worlds in the sense that $\mathcal{K}(\Phi, \Omega_{\Phi_0}) = \{(Q_\omega(\alpha_1), \dots, Q_\omega(\alpha_l)) | \omega \in \Omega_R\}$. We define the set Ω_R as follows. First we order (arbitrarily) the constants in Δ and we do the same with the sets in $2^{\mathcal{U}}$; we denote by c_i the i-th constant and similarly, by U_i, the i-th subset of $\mathcal{U}$. For every $J = (j_1, \dots, j_{|2^{\mathcal{U}}|}) \in \mathcal{J}$ we construct:

$$\omega_J = \bigcup_{i=1}^{j_1} \bigcup_{R \in U_1} \{R(c_i)\} \cup \bigcup_{i=j_1+1}^{j_1+j_2} \bigcup_{R \in U_2} \{R(c_i)\} \cup \dots$$
$$\dots \cup \bigcup_{i=j_1+\dots+j_{|2^{\mathcal{U}}|-1}+1}^{|\Delta|} \bigcup_{R \in U_{|2^{\mathcal{U}}|}} \{R(c_i)\}$$

Then we define $\Omega_R = \{\omega_J | J \in \mathcal{J}\}$. Notice that $|\Omega_R|$ is polynomial in $|\Delta|$. Finally, it is easy to show that we can do the following in polynomial time (i.e. polynomial in $|\Delta|$): (i) to filter out possible worlds that do not satisfy Φ_0 and (ii) to compute $(Q_\omega(\alpha_1), \dots, Q_\omega(\alpha_l))$.

In the next example we illustrate the construction from the above remark.

Example 2. *Let $\mathcal{U} = \{sm/1\}$ and $\Delta = \{Alice, Bob\}$. Then $\mathcal{J} = \{(0, 2), (1, 1), (2, 0)\}$. Now, for every $J \in \mathcal{J}$, we need to construct the respective ω_J. That is, for the ordering of constants $Alice \prec Bob$ and the ordering of cells $\emptyset \prec \{sm/1\}$, we have:*

$$\omega_{(0,2)} = \{sm(Alice), sm(Bob)\},$$
$$\omega_{(1,1)} = \{sm(Bob)\},$$
$$\omega_{(2,0)} = \emptyset.$$

The set of representative possible worlds is $\Omega_R = \{\omega_{(0,2)}, \omega_{(1,1)}, \omega_{(2,0)}\}$.

We now need to explain how to compute the set $\mathcal{K}(\Phi, \Omega_{\Phi_0})$ for the case when $\mathcal{B} \neq \emptyset$. We again show how to construct the set of representative possible worlds but this time also with binary predicates; we denote this set Ω_R^B. We will explain how to construct representatives by extending one possible world $\omega_0 \in \Omega_R$, constructed as in Remark 1. Hence, obviously the

[1] Existential quantifiers can be treated using a form of Skolemization we omit the details here.

[2] We refer to [1] for details.

same procedure will need to be repeated for all possible worlds from Ω_R.

Remark 3. *First, we consider literals of the form $R(c,c)$ where $R \in \mathcal{B}$ and $c \in \Delta$. We can notice that these literals can be added already in the construction of Ω_R (using auxiliary unary predicates), so we will not consider this type of literals here further.*

The next remark will provide us with a simple way to construct the set of representatives.

Remark 4. *Let us suppose that the possible world ω_J, where $J = (j_1, \ldots, j_{|2^{\mathcal{J}}|}) \in 2^{\mathcal{J}}$, is as in Remark 1. We first discuss how we could generate all possible worlds that could be obtained from ω_J. Let $\Delta_q = \{c_{\sum_{k=1}^{q-1} j_k + 1}, \ldots, c_{\sum_{k=1}^{q} j_k}\}$, and $\Delta_r = \{c_{\sum_{k=1}^{r-1} j_k + 1}, \ldots, c_{\sum_{k=1}^{r} j_k}\}$. Next we could assign a subset of binary predicates $\mathcal{B}$ to each element of the set $\{(c,c') \in (\Delta_q \times \Delta_r) | c \neq c'\}$ (note that the condition $c \neq c'$ is only relevant for $r = q$ and note that we have already taken care of literals of the form $R(c,c)$). If for instance, (c_1, c_2) got assigned the predicates friends, teammates then we would include the literals friends(c_1, c_2) and teammates(c_1, c_2) to the constructed possible world, and analogically for all the other tuples. Finally, let us define $\#(B,q,r)$ to be the number of pairs of domain elements from $\Delta_q \times \Delta_r$ which are assigned the subset of binary predicates $B \in 2^{\mathcal{B}}$. We may notice that $Q_\omega(\alpha)$ for any 2-variable quantifier-free formula α will only depend on the numbers $\#_\omega(B,q,r)$ but not on any other details of the possible worlds. The same also holds for the 2-variable universally quantified formulas in Φ_0. Hence, we can construct only representatives with distinct $\#_\omega(B,q,r)$'s using a straightforward generalization of the procedure from Remark 1.*

Finally, we need to show that the number of representatives in the set constructed according to Remark 4 has size polynomial in $|\Delta|$. Using Remarks 1, 3 and 4, we can obtain the rather crude upper bound:

$$|\Omega_R^B| \leq (|\Delta| + 1)^{2^{|\mathcal{U}|+|\mathcal{B}|}} \cdot (|\Delta| + 1)^{2 \cdot 4^{|\mathcal{U}|+|\mathcal{B}|} \cdot 2^{|\mathcal{B}|}}.$$

Here, the first part comes from Remarks 1 and 3 and the second part from Remark 4. Importantly, the bound is polynomial in $|\Delta|$. Since our main aim in this paper is establishing existence of polynomial-time algorithms for weight learning, we will not try to optimize this bound. In practice, one could probably find the vertices defining the polytope faster using a generic SAT solver as an oracle inside a heuristic algorithm iteratively traversing vertices of the polytope, but that would not lead to an algorithm with runtime polynomial in the size of the domain.

References

[1] P. Beame, G. Van den Broeck, E. Gribkoff, and D. Suciu. Symmetric weighted first-order model counting. In *Proceedings of the 34th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pages 313–328. ACM, 2015.