Accurate prediction of facts using logical rules

Luis Galárraga

January 9, 2015

The inherent incompleteness of web-extracted knowledge bases (KBs), was one of the major motivations for the work presented in AMIE [4] and its successor AMIE+[1]. The AMIE article introduces (a) an algorithm for fast mining of Horn rules on large KBs and (b) a mining model that accounts for incompleteness under the Open World Assumption (OWA). AMIE mines closed Horn rules such as

$$\begin{aligned} hasChild(x,y), isMarriedTo(y,z) &\Rightarrow hasChild(x,z) \\ livesIn(x,y) &\Rightarrow isCitizenOf(x,y) \end{aligned}$$

According to the first rule, if somebody is married and has a child, then her spouse is also a parent of this child. While this rule holds quite often, it will err if a person has children from different marriages. Thus, rules are assigned a confidence score that measures the ratio of cases for which the rule draws correct conclusions. In AMIE we propose the confidence under the Partial Completeness Assumption (PCA). Unlike the Closed World Assumption, the PCA admits that missing facts in KBs are not necessarily false. For instance, if the second rule in our example concludes the nationality of a person for whom the KB does not know the nationality, the standard CWA confidence counts this as a miss, whereas the PCA confidence does not. On the hand, if the KB knows the nationality of the person and rule predicts a different nationality, both the PCA and the standard confidence penalize the rule.

Our example suggests that under the PCA, the gaps in the data become a smaller impediment for rule mining. We have shown already the applicability of such rules in automatic data engineering tasks [3, 2], even though our original motivation was to apply the rules for data inference, i.e., to predict facts beyond the KB. Such facts could, for example, be added to the KB or suggested as likely answers to queries. To investigate the potential of our mining model for data inference, we conducted a naive experiment that shows the suitability of the PCA confidence for ranking logical rules as an alternative to the standard CWA confidence. We ran AMIE on YAGO2 and took the top 30 most confident rules according to the standard and the PCA confidence. We then use the rules to infer statements that were not in the YAGO2. Our results show that the PCA confidence identifies rules that produce many true predictions. Nevertheless, our precision was in the range 30-45% at 300K unique predictions. This result clearly suggests that if we want to use logical rules for effective data inference, we should not naively produce predictions as we did. Notice also that our experimental setup disregards the fact that a prediction can be inferred by multiple rules,

[1] Article under review

Old dataset	Rules	Total hits	Unique Hits
YAGO2	135	14K	12K
YAGO2 (const)	19028	79K	17K
DBpedia 2.0	117163	237536	149308

Table 1: AMIE for predicting facts on YAGO and DBpedia

because we measured precision on the set of unique predictions. This case is, however, very common as Table 1 shows. Out of 14K predictions drawn from YAGO2, 12K were unique. The difference becomes even more obvious when we allow constant arguments in the atoms of rules, e.g., $\mathit{livesIn}(x, USA) \Rightarrow \mathit{speaks}(x, \mathit{English})$[2]. This implies that some facts were indeed deduced by multiple rules.

1 Rules for data inference

In a very simple experiment, we took the 135 rules mined by AMIE on YAGO2 and output all their predictions. For each prediction I kept track of the rules that inferred it and verified its correctness on YAGO2s. For example the correct prediction $\mathit{isCitizenOf}(\mathit{Roy\ Gibson}, UK)$ was inferred from two logical rules:

- R_1: $isLocatedIn(f, b), livesIn(a, f) \Rightarrow isCitizenOf(a, b)$ (0.48)
- R_2: $isLocatedIn(f, b), wasBornIn(a, f) \Rightarrow isCitizenOf(a, b)$ (0.57)

where the number in parentheses is the rule's PCA confidence. Intuitively, different rules concluding the same fact should increase the confidence about the correctness of that fact. We can therefore quantify the confidence of a prediction as a function of the PCA confidence of its generating rules. Predictions with high confidence can be then suggested as possible answers to queries or as candidates to populate the KB.

Given a KB $\mathcal{K}$ and a statement $f := r(x, y)$, f is a *fact* if $f \in \mathcal{K}$, otherwise it is a *prediction*. Let us define the boolean random variable $\widehat{f} = \phi(f)$ where ϕ is an interpretation function that determines whether a statement is true in the real world, otherwise it is false or unknown. If we assume that KBs are a correct representation of the real world, then $\forall\ f \in \mathcal{K} : \widehat{f} = \mathit{true}$. This also implies that the probability distribution of variable $\widehat{f}$ becomes trivial: $P(\widehat{f} = \mathit{true}) = 1$ and $P(\widehat{f} \neq \mathit{true}) = 0$. Hereinafter, I shall use $\widehat{f}$ to denote $\widehat{f} = \mathit{true}$ and $\neg\widehat{f}$ to express $\widehat{f} \neq \mathit{true}$. Computing a confidence score for a prediction $q \notin \mathcal{K}$ is equivalent to calculate the probability $P(\widehat{q} \mid \widehat{E})$, where $\widehat{E}$ is the evidence, i.e., a set of preconditions in the form of concrete assignments to random variables. For our example prediction $q := \mathit{isCitizenOf}(\mathit{Roy\ Gibson}, UK)$, $\widehat{E} := \{\widehat{f_1}, \widehat{f_2}, \widehat{f_3}, \widehat{f_4}\}$ with:

- $f_1 := \mathit{isLocatedIn}(\mathit{London}, UK)$
- $f_2 := \mathit{livesIn}(\mathit{Roy\ Gibson}, \mathit{London})$

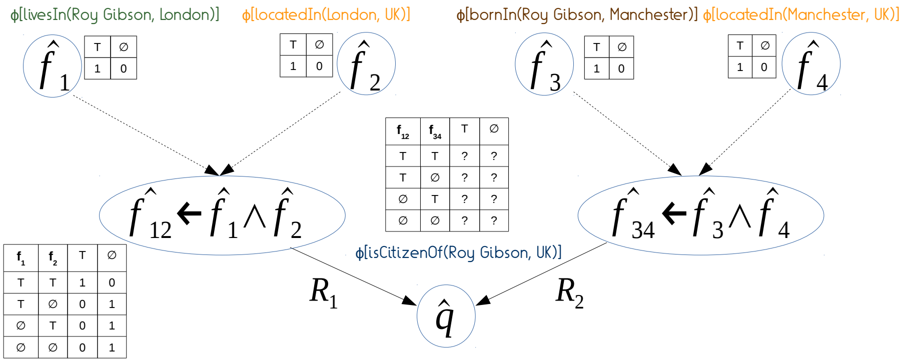
[2] As we will show later, this phenomenon is also aggravated from the dependencies among rules

- $f_3 := \mathit{hasCapital}(UK, London)$
- $f_4 := \mathit{wasBornIn}(Roy\ Gibson, Manchester)$

This formulation was designed with two goals in mind. First, to define a connection between a prediction q and all the facts f_i used to infer q from a set of logical rules. Second, to allow for iterative inference so that previously inferred predictions could become part of the preconditions for further rounds of inference. Unlike the preconditions in our running example where $P(\widehat{f_i}) = 1$ (as $f_i \in \mathcal{K}$), the probability distribution for the correctness of a prediction is not trivial.

Once we have formulated the correctness of statements w.r.t. the real world as random variables, we can borrow the concepts from Bayesian Networks to construct a DAG that models the dependencies between the variables.

A *Bayesian Network* (BM) is a statistical graphical model designed to encode the joint probability distribution of a set of variables as a directed acyclic graph. Each node x in the graph represents a random variable. An edge from node x to node y denotes statistical dependence between the variables represented by the nodes, i.e., $P(y \mid x) \neq P(y)$. Figure 1 shows a BN-like representation of our inference example.



ϕ[livesIn(Roy Gibson, London)]
ϕ[locatedIn(London, UK)]
ϕ[bornIn(Roy Gibson, Manchester)]
ϕ[locatedIn(Manchester, UK)]
$\hat{f}_1$
$\hat{f}_2$
$\hat{f}_3$
$\hat{f}_4$
$\hat{f}_{12} \leftarrow \hat{f}_1 \wedge \hat{f}_2$
$\hat{f}_{34} \leftarrow \hat{f}_3 \wedge \hat{f}_4$
ϕ[isCitizenOf(Roy Gibson, UK)]
R_1
R_2
$\hat{q}$

T	∅
1	0

T	∅
1	0

T	∅
1	0

T	∅
1	0

f_{12}	f_{34}	T	∅
T	T	?	?
T	∅	?	?
∅	T	?	?
∅	∅	?	?

f_1	f_2	T	∅
T	T	1	0
T	∅	0	1
∅	T	0	1
∅	∅	0	1

Figure 1: A Bayesian Network representing the inference of fact from multiple Horn rules.

Each node in the BN is associated to a random variable that represents the truth of a statement or a conjunction of statements. For each instantiation of a rule $\boldsymbol{B} \Rightarrow r(x, y)$, we draw an edge from the node associated to the antecedent to the node associated to the succedent. If the antecedent consists of multiple atoms, then it binds to multiple statements. We model this situation by introducing an additional variable representing the truth of a conjunction of the statements, e.g., $\widehat{f_{12}}$ and $\widehat{f_{34}}$. Furthermore, each node is associated to a conditional probability table (CPT). This table stores the probability distribution of the node given its ancestors. The CPT is trivial for facts, since they do not have ancestors and are assumed as ground truth. The CPT of a conjunction of statements is built according to the rules of deterministic logical conjunction. Figure 1 illustrates the CPT for $\widehat{f_{12}}$. The table for $\widehat{f_{34}}$ follows the same principle. The CPT of a prediction depends on the statements and rules that implied the prediction. Our goal is to define such CPT. We propose to calculate it as follows for our example:

$$P(\widehat{q} \mid \widehat{f}_{12}, \widehat{f}_{34}) := 1 - (1 - PCA(R_1))(1 - PCA(R_2)) \tag{1}$$

$$P(\widehat{q} \mid \widehat{f}_{12}, \neg\widehat{f}_{34}) := PCA(R_1) \tag{2}$$

$$P(\widehat{q} \mid \neg\widehat{f}_{12}, \widehat{f}_{34}) := PCA(R_2) \tag{3}$$

$$P(\widehat{q} \mid \neg\widehat{f}_{12}, \neg\widehat{f}_{34}) := \alpha \tag{4}$$

If the antecedents of all the predicting rules hold, the probability of $\widehat{q}$ is 1 minus the probability that none of the rules hold. Such probability is defined as 1 minus the PCA confidence of the rules. Note that the product of complementary probabilities assumes that the rules are independent. Whereas this assumption seems feasible at a first glance, we will later show some cases where it does not hold. If only one of the rules fires, then the probability depends solely on the PCA confidence of that rule. When none of the rules hold, we can define a default probability α. A value of 0 means that the rules in our model are the only explanation for the prediction. A value greater than 0 models the case our prediction can be true from external reasons not captured in our model. Recall that $P(\neg\widehat{q} \mid E) = 1 - P(\widehat{q} \mid E)$

1.1 Iterative inference

Consider a KB $\mathcal{K}$ containing the facts $f_1 := \textit{isMarriedTo}(\textit{Roy Gibson}, \textit{Alice})$ and $f_2 := \textit{isCitizenOf}(\textit{Roy Gibson}, \textit{England})$ and the logical rules:

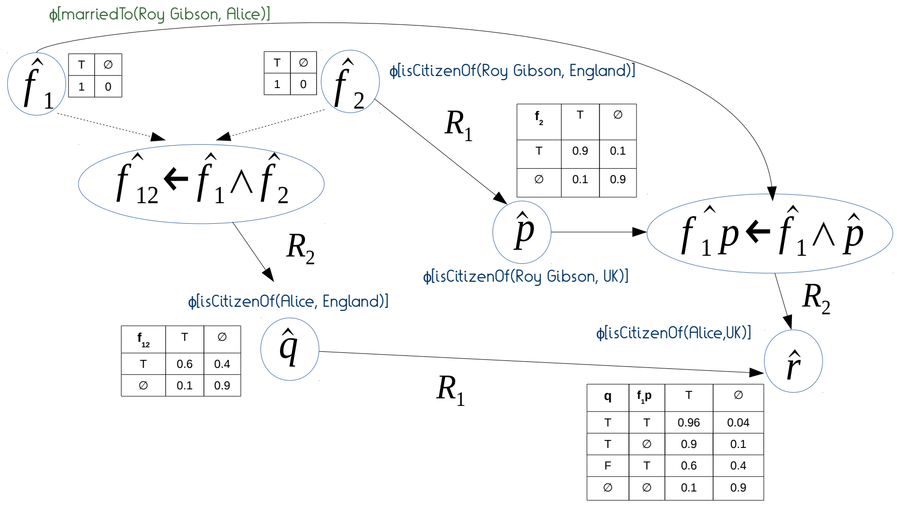
- R_1: $isCitizenOf(a, England) \Rightarrow isCitizenOf(a, UK)$ (0.9)
- R_2: $isMarriedTo(a, b), isCitizenOf(a, c) \Rightarrow isCitizenOf(b, c)$ (0.6)

By applying R_1 on f_1 we can deduce $p := \textit{isMarriedTo}(\textit{Roy Gibson}, UK)$. f_1, f_2 and R_2 allow us to deduce $q := \textit{isCitizenOf}(\textit{Alice}, \textit{England})$. From p and q, both rules infer $r := \textit{isMarriedTo}(\textit{Alice}, UK)$. Figure 2 depicts the BN representing the two rounds of inference on $\mathcal{K}$ using the rules above. Note that we use the formulas introduced in the previous section in the CPT for $\widehat{r}$.

2 Answering queries with Bayesian Networks

2.1 Confident predictions

Identifying predictions that are likely true has a great value for data maintanance. For instance, if the predictions are sent to human evaluators to populate the KB, pruning the non-promising candidates can save a lot of work. With this application in mind, we propose to build a BN as described in the previous section in order to answer queries of the form $P(\widehat{q} \mid \widehat{\mathcal{K}})$, where $q \notin \mathcal{K}$ and the evidence are the facts in $\mathcal{K}$. Since probably only a few facts will take part in the deduction of q, the expression can be rewritten as $P(\widehat{q} \mid \widehat{E})$, where $\widehat{E} \subseteq \widehat{\mathcal{K}}$ is a set of random variables associated to the facts that took part in the deduction of q. In the example presented in Figure 2, $\widehat{E} = \{\widehat{f}_1, \widehat{f}_2\}$. We can calculate this value as follows (recall that $P(\widehat{f}_1) = P(\widehat{f}_2) = P(\widehat{f}_{12}) = 1$):


ϕ[marriedTo(Roy Gibson, Alice)]
$\hat{f}_1$
$\hat{f}_2$
ϕ[isCitizenOf(Roy Gibson, England)]
R_1
$\hat{f}_{12} \leftarrow \hat{f}_1 \wedge \hat{f}_2$
$\hat{p}$
$\hat{f}_1 p \leftarrow \hat{f}_1 \wedge \hat{p}$
R_2
ϕ[isCitizenOf(Roy Gibson, UK)]
R_2
ϕ[isCitizenOf(Alice, England)]
ϕ[isCitizenOf(Alice,UK)]
$\hat{q}$
$\hat{r}$
R_1

T	∅
1	0

T	∅
1	0

f_2	T	∅
T	0.9	0.1
∅	0.1	0.9

f_{12}	T	∅
T	0.6	0.4
∅	0.1	0.9

q	f_1p	T	∅
T	T	0.96	0.04
T	∅	0.9	0.1
F	T	0.6	0.4
∅	∅	0.1	0.9

Figure 2: A Bayesian Network representing an iterative inference process.

$$\begin{aligned} P(\widehat{r} \mid \widehat{f_1}, \widehat{f_2}) = \; & \tfrac{P(\widehat{r}, \widehat{f_1}, \widehat{f_2})}{P(\widehat{f_1}, \widehat{f_2})} = P(\widehat{r}, \widehat{f_1}, \widehat{f_2}) = \\ & P(\widehat{r} \mid \widehat{f_1 p}) P(\widehat{f_1}) P(\widehat{p} \mid \widehat{f_2}) P(\widehat{f_2}) P(q \mid \widehat{f_{12}}) P(\widehat{f_1}) P(\widehat{f_2}) = \\ & 0.96 \times 0.9 \times 0.6 = 0.5184 \end{aligned}$$

More formally, given $\mathcal{K}$ and a set of rules $\mathcal{R}$ with different PCA confidence scores, our goal is to infer a set of predictions $\mathcal{P}$ such that $\forall p \in \mathcal{P} : \; (\mathcal{K}, \mathcal{P}) \models p \; \wedge \; P(\widehat{p} \mid \widehat{\mathcal{K}}) \geq \delta$. Here $\delta \in (0, 1]$ is a confidence threshold. This models the scenario when we are only interested in the predictions above a given confidence threshold.

2.2 Arbitrary conjunctive queries

Coming back to the example in Figure 2, imagine we want to include our predictions in the result of the query $Q(y) = y \; : \; isCitizenOf(Alice, y)$. In a non-probabilistic setting, the query would return an empty set, as no nationality for Alice is known in the KB. We could, however, provide our predictions about Alice's nationality as part of the answer, ranked by their likelihood. In this example the answer would be: *England* (0.6) and *UK* (0.5184), where the number in parentheses is $P(isCiti\widehat{zenOf(}Alice, y) \mid \widehat{\mathcal{K}})$ for each value of y. Recall that the semantics of this ranking use only the KB as evidence, thus the probability that Alice is a citizen of UK does not consider the fact that she could be citizen of England (as this was also deduced). If for instance the user has certainty about any of the answers in the ranking, she could add them to the evidence and recompute the ranking for the remaining answers. If for instance, the user adds $q := isCitizenOf(Alice, England)$, the probability of Alice being a citizen of UK ($r := isMarriedTo(Alice, UK)$) becomes 0.96 equivalent to $P(\widehat{r} \mid \widehat{\mathcal{K}} \cup \{\widehat{q}\})$.

For more complex queries, there are further considerations. For example, if we want to integrate our predictions into the answers of the query $Q(x) = x : isCitizenOf(x, y) \land isLocatedIn(y, Europe)$, we need to calculate the probability that Alice is a citizen of a country located in Europe. One may be tempted to calculate it as $1 - (1 - P(\widehat{q} \mid \widehat{\mathcal{K}}))(1 - P(\widehat{r} \mid \widehat{\mathcal{K}}))$[3]. However, $\widehat{q}$ and $\widehat{r}$ are not independent, therefore we have account for such dependency when computing $P(\widehat{q} \lor \widehat{r} \mid \mathcal{K})$ from the BN representation.

2.3 Independence of rules

Consider a KB containing the facts $f_1 := \mathit{isCitizenOf}(\mathit{The\ Queen}, \mathit{England})$, $f_2 := \mathit{hasOfficialLanguage}(\mathit{England}, \mathit{English})$ and the following logical rules:

- R_1: $isCitizenOf(f, b), hasOfficialLanguage(b, c) \Rightarrow \mathit{speaks}(f, c)$ (0.80)
- R_2: $isCitizenOf(f, England) \Rightarrow \mathit{speaks}(f, \mathit{English})$ (0.80)

Both rules would conclude that $p := \mathit{speaks}(\mathit{The\ Queen}, \mathit{English})$, however it is clear that R_1 subsumes R_2, therefore *the rules are not independent.* This is certainly a problem when defining the CPT for $\widehat{p}$, because our definition for $P(\widehat{p} \mid \widehat{f_1}, \widehat{f_2})$ assumes independence between rules. The independence assumption overestimates the probability $P(\widehat{p} \mid \widehat{f_1}, \widehat{f_2})$ because it counts the same statistical evidence twice.

While a syntactic query containment test can identify subsumption dependencies between rules, e.g., $\vec{B} \land r(x, y) \Rightarrow r_h(x', y') \sqsubseteq \vec{B} \Rightarrow r_h(x', y')$, arbitrary correlations can stand even between rules with apparently no syntactic connection. In this case, only an independence test on the actual data can spot such anomalies. If for two rules R, R' it holds that $R \sqsubseteq R'$, some alternatives are (a) to focus on precision and take the most specific rule, (b) to focus on recall and take the most general rule, (c) take the rule with highest gain g where

$$g(R) = \mathit{recall}(R) \times PCA(R)$$

where $\mathit{recall}(R)$ is the number of predictions drawn from rule R. This approach focuses only in pairwise correlations, even though correlations could occur among arbitrary sets of rules.

2.4 Integrity constraints

Horn rules encode regularities that hold often in the data but not necessarily always. They are *soft constraints.* In contrast a *hard constraint* is a rule that holds (or should hold) always. Violations of hard-constraints in KBs are considered bugs in the data. We are interested in functional and cardinality hard constraints, e.g., a person must have at most one place of birth. While these types of constraints could be mined from KBs using our machinery, the gaps and noise in the data makes very hard to differentiate confident soft constraints from hard constraints. On the other hand, probabilistic approaches for inference are not affected by this phenomenon as they do not need to treat hard constraints in a special way. Nevertheless, the BN model per se is in principle

[3] As in the context of tuple independent probabilistic databases

unable to represent inference using Horn rules and cardinality constraints. To see why, consider a KB $\mathcal{K}$ containing the time-agnostic[4] facts

- $f_1 := \mathit{hasChild}(Liz\ Taylor, Maria)$
- $f_2 := \mathit{isMarriedTo}(Liz\ Taylor, Richard\ Burton)$
- $f_3 := \mathit{isMarriedTo}(Liz\ Taylor, John\ Warner)$

and the rules:

- $R_1 : hasChild(x, y), isMarriedTo(y, z) \Rightarrow hasChild(x, z)$ (0.60)
- $R_2 : hasChild(x, y) \Rightarrow \pi_{count(x)}\gamma_y(hasChild(x, y)) \leq 2$ (1.0)

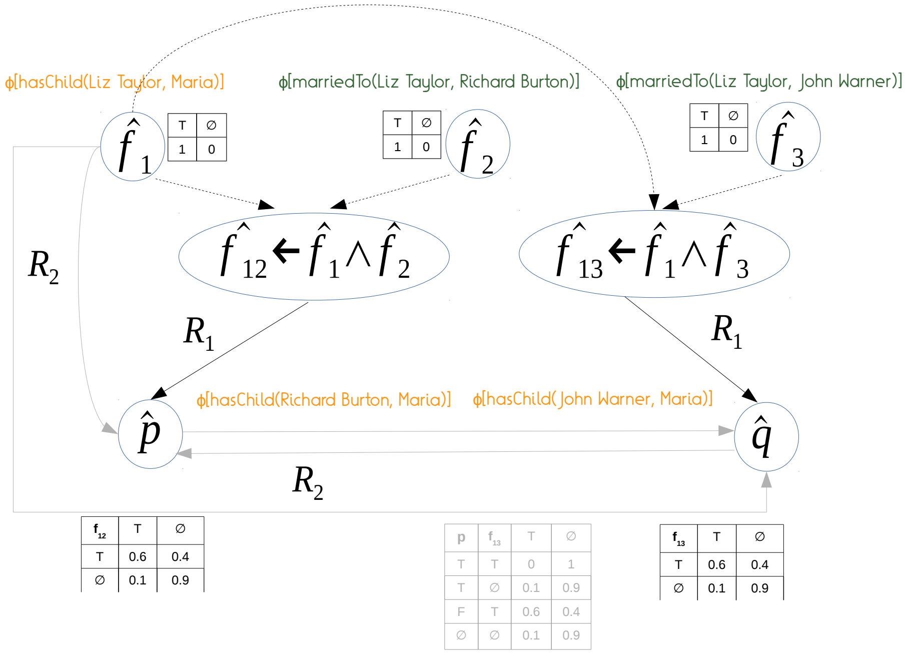
The second rule is a hard cardinality constraint stating that a person can have at most two parents. In this example, R_1 allows us to infer:

- $p := \mathit{hasChild}(Richard\ Burton, Maria)$
- $q := \mathit{hasChild}(John\ Warner, Maria)$

Now imagine we want to plug R_2 into our statistical graphical model. [1] proposes two different semantics for iterative data inference in Datalog in the presence of datalog rules and functional constraints (FD). Under the fact-at-a-time semantics, each iteration produces a single fact. If the fact violates a functional constraint, it is rejected. These semantics are referred in [1] as non-deterministic fact-at-a-time (*nfact*) because the result of the process depends on the order of selection of the rules and facts for inference. In our previous example, two rounds of inference[5] under the *nfact* semantics would lead to two possible outcomes: $\{f_1, f_2, f_3, p\}$ or $\{f_1, f_2, f_3, q\}$, depending on whether f_2 or f_3 are used for deduction. Under the set-at-a-time semantics (*nsat*), each stage produces a maximal subset of derivable facts that respects the FDs and is entailed both from $\mathcal{K}$ and the facts inferred so far, i.e., $\mathcal{K} \cup \mathcal{P}$. We adapt these ideas to our setting from the observation that cardinality constraints are a generalization of functional constraints, i.e., a FD is a 1-cardinality constraint. In our previous example, inference under the *nsat* would not produce any new facts since it is impossible to add two extra parents to *Maria* without violating rule R_2. A prediction p is called *possible* if $P(\widehat{p} \mid \widehat{\mathcal{K}} \cup \widehat{\mathcal{P}}) > 0$ at the end of the inference process. [1] shows that any possible prediction p under *nsat* is also possible under *nfat*. The converse statement does not hold as our example shows. We propose a different approach: apply deduction under the *nsat* semantics ignoring any functional or n-cardinality constraints. If a set of predictions $\mathcal{P}$ can potentially violate a constraint, we represent this as clique, that models the fact that the belief in any of the predictions in the clique does affect the belief in the other predictions. Moreover, if a prediction in $\mathcal{P}$ may conflict with a fact in $\mathcal{K}$, the flow of influence goes from the certain to the uncertain statement. Figure 3 illustrates this idea applied to the BN for our last example. The edges in gray are derived from the cardinality constraint R_2.

[4] To the best of my knowledge no inference approach on web-extracted KB uses the temporal dimension of facts.

[5] The second round would conclude that is not possible to infer anything else.



ϕ[hasChild(Liz Taylor, Maria)]
ϕ[marriedTo(Liz Taylor, Richard Burton)]
ϕ[marriedTo(Liz Taylor, John Warner)]
$\hat{f}_1$
$\hat{f}_2$
$\hat{f}_3$
$\hat{f}_{12} \leftarrow \hat{f}_1 \wedge \hat{f}_2$
$\hat{f}_{13} \leftarrow \hat{f}_1 \wedge \hat{f}_3$
R_2
R_1
R_1
ϕ[hasChild(Richard Burton, Maria)]
ϕ[hasChild(John Warner, Maria)]
$\hat{p}$
$\hat{q}$
R_2

T	∅
1	0

T	∅
1	0

T	∅
1	0

f_{12}	T	∅
T	0.6	0.4
∅	0.1	0.9

p	f_{13}	T	∅
T	T	0	1
T	∅	0.1	0.9
F	T	0.6	0.4
∅	∅	0.1	0.9

f_{13}	T	∅
T	0.6	0.4
∅	0.1	0.9

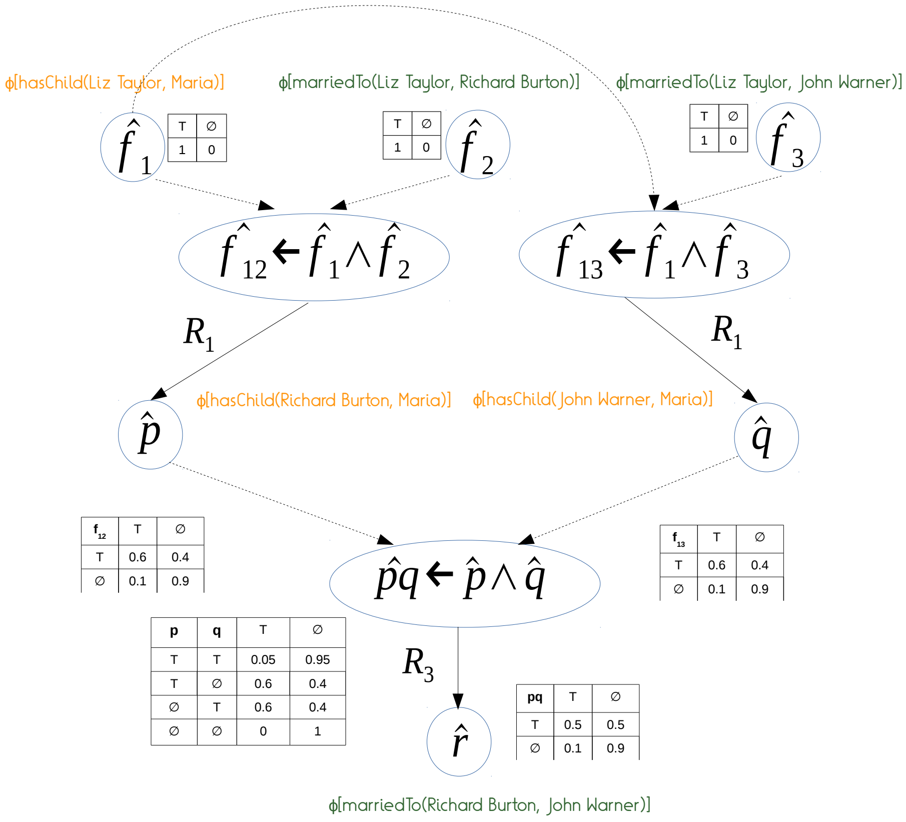
Figure 3: A Bayesian-like Network representing an inference process with a 2-cardinality constraint.

We point out two observations. First and obvious, this representation does not meet the definition of a BN because it contains cycles. Second, these dependencies matter only if we use the predictions in $\mathcal{P}$ either as evidence for queries or as preconditions for further rounds of inference. To grasp this notion, imagine we halt the inference routine at this point and omit the undirected edges in our example, so that $\widehat{p}$ and $\widehat{q}$ become leaves in the directed graph. In the simplest scenario, if we want to rank the potential parents of Maria by likelihood given the KB, i.e., rank the answers to $Q(x) := y : \ p_y := hasChild(y, Maria)$ by $P(\widehat{p_y} \mid \widehat{\mathcal{K}})$, we do not need to model the interdependencies between the predictions because the probability associated to each answer is not suppose to consider the other answers as evidence. In contrast, imagine the user knows that $p := hasChild(Richard\ Burton, Maria)$. If she adds this to the evidence, the score for $q := hasChild(John\ Warner, Maria)$, i.e., $P(\widehat{q} \mid \widehat{\mathcal{K}} \cup \{\widehat{p}\})$ becomes zero in compliance with the cardinality constraint R_2. This is equivalent to add a directed edge from $\widehat{p}$ to $\widehat{q}$ on the fly and write zero to the entry $P(\widehat{q} \mid \widehat{p}, \widehat{f_{13}})$ (table in gray in Figure 3).

Now imagine we have an additional rule:

$$R_3 : hasChild(x, y), hasChild(z, y) \Rightarrow isMarriedTo(x, z) \ \ (0.50)$$

and the rule lets us infer $r := isMarriedTo(Richard\ Burton, John\ Warner)$. Since p and q cannot be true without violating R_2, an inference approach could refrain itself from inferring r. This is possible under hard cardinality constraints, however it is unlikely to mine hard cardinality constraints on noisy and incomplete web-extracted KBs. Unless someone defines the integrity constraints manually, those learned by a statistical rule mining approach will not have confidence 1.0. If we relax R_2 by assigning it confidence 0.95 in our previous example, we could still get a small probability for prediction r by redefining the construction of the CPT for $\widehat{pq}$ as shown in Figure 4. This CPT does not implement the rules of logical conjunction, instead it encodes the "compatibility" between the different truth values of the individual random variables according to constraint R_2.


ϕ[hasChild(Liz Taylor, Maria)]
ϕ[marriedTo(Liz Taylor, Richard Burton)]
ϕ[marriedTo(Liz Taylor, John Warner)]
$\hat{f}_1$
$\hat{f}_2$
$\hat{f}_3$
$\hat{f}_{12} \leftarrow \hat{f}_1 \wedge \hat{f}_2$
$\hat{f}_{13} \leftarrow \hat{f}_1 \wedge \hat{f}_3$
R_1
R_1
ϕ[hasChild(Richard Burton, Maria)]
ϕ[hasChild(John Warner, Maria)]
$\hat{p}$
$\hat{q}$
$\hat{p}\hat{q} \leftarrow \hat{p} \wedge \hat{q}$
R_3
$\hat{r}$
ϕ[marriedTo(Richard Burton, John Warner)]

T	∅
1	0

T	∅
1	0

T	∅
1	0

f_{12}	T	∅
T	0.6	0.4
∅	0.1	0.9

f_{13}	T	∅
T	0.6	0.4
∅	0.1	0.9

p	q	T	∅
T	T	0.05	0.95
T	∅	0.6	0.4
∅	T	0.6	0.4
∅	∅	0	1

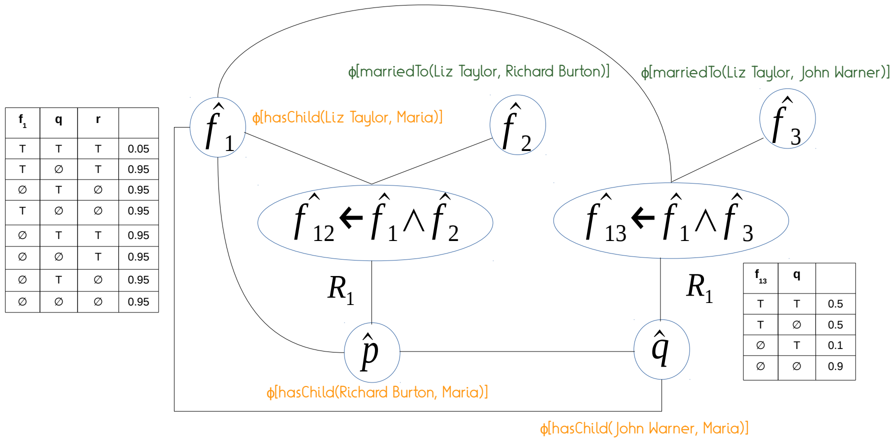
pq	T	∅
T	0.5	0.5
∅	0.1	0.9

Figure 4: Encoding cardinality constraints in the CPT for conjunctions of statements.

The ideas presented in this section are an attempt to extend the framework of Bayesian Networks to cope with cardinality constraints in our scenario. However, Markov Logic Networks were originally designed to model situations when for a given pair of random variables, influence flows in both directions.

3 Modelling the problem with Markov Logic Networks

A Markov Logic Network (MLN) is a graphical model that can represent the joint distribution of a set correlated variables. Like Bayesian Networks, they rely on a graph representation, but instead of modeling pairwise correlations as directed edges, they use undirected edges with factors that encode the level of agreement or compatibility between arbitrary groups of random variables. More formally, a factor $\Phi(\widehat{X})$, with $\widehat{X} := \{\widehat{x_1}, \ldots, \widehat{x_m}\}$ is a function $\Phi : dom(\widehat{x_1}) \times \cdots \times dom(\widehat{x_m}) \to \mathbb{R}$. The $\widehat{x_i}$ are random variables and $dom(\widehat{x_i})$ is the domain of $\widehat{x_i}$, that is, the set of possible values that $\widehat{x_i}$ can take. In our scenario $dom(\widehat{x_i}) = \{T, \emptyset\}$. $\Phi(\widehat{X})$ models the level of agreement for each possible combination of values for the variables in $\widehat{X}$, where higher values denote higher compatibility. Unlike BNs, there is no 1-1 mapping from factors to nodes in MLNs. Factors, also called *potential functions*, factorize over subgraph cliques. To illustrate this concept, consider the MLN in Figure 5 that models our last example.



ϕ[marriedTo(Liz Taylor, Richard Burton)]
ϕ[marriedTo(Liz Taylor, John Warner)]
ϕ[hasChild(Liz Taylor, Maria)]
$\hat{f}_1$
$\hat{f}_2$
$\hat{f}_3$
$\hat{f}_{12} \leftarrow \hat{f}_1 \wedge \hat{f}_2$
$\hat{f}_{13} \leftarrow \hat{f}_1 \wedge \hat{f}_3$
R_1
R_1
$\hat{p}$
$\hat{q}$
ϕ[hasChild(Richard Burton, Maria)]
ϕ[hasChild(John Warner, Maria)]

f_1	q	r	
T	T	T	0.05
T	∅	T	0.95
∅	T	∅	0.95
T	∅	∅	0.95
∅	T	T	0.95
∅	∅	T	0.95
∅	T	∅	0.95
∅	∅	∅	0.95

f_{13}	q	
T	T	0.5
T	∅	0.5
∅	T	0.1
∅	∅	0.9

Figure 5: A MLN representation for an inference process with a 2-cardinality constraint.

For simplicity we show only two factors. The full MLN for this particular inference process consists of 6 factors, one per maximal subgraph clique: $\Phi(\widehat{f_1}, \widehat{f_{12}})$, $\Phi(\widehat{f_2}, \widehat{f_{12}})$, $\Phi(\widehat{f_1}, \widehat{f_{13}})$, $\Phi(\widehat{f_3}, \widehat{f_{13}})$, $\Phi(\widehat{f_{12}}, \widehat{p})$, $\Phi(\widehat{f_{13}}, \widehat{q})$, $\Phi(\widehat{f_1}, \widehat{p}, \widehat{q})$. We derive some observations:

- Except from $\Phi(\widehat{f_1}, \widehat{q}, \widehat{r})$, all cliques in our factorization contain 2 vertices.
- This factorization is not unique. In our example, we use a maximal clique, i.e., $\Phi(\widehat{p}, \widehat{q}, \widehat{r})$, however nothing prevent us from factorizing over the non-maximal cliques $\Phi(\widehat{p}, \widehat{q})$, $\Phi(\widehat{p}, \widehat{r})$ and $\Phi(\widehat{q}, \widehat{r})$.
- $\Phi(\widehat{f_{13}}, \widehat{q})$ is a rearrangement of the CPT used in BN representation of the process, however the factors are not necessarily probability distributions. Still they are designed to model the confidence we have on a particular combination of values for the variables. For instance, factor $\Phi(\widehat{p}, \widehat{q}, \widehat{r})$

assigns the probability of violating the soft cardinality constraint as score, to the case when Maria has three parents. In contrast, the other cases that do not violate the constraint have a higher compatibility score, the confidence of the cardinality constraint. This decision for the scores is arbitrary.

- The second half of $\Phi(\widehat{f_1}, \widehat{p}, \widehat{q})$ can be omitted since we know that f_1 is certain. The shortened version of the factor is denoted as $\Phi[\widehat{f_1}](\widehat{p}, \widehat{q})$ in the literature.

MLNs model the joint probability distribution of a set of variables as a normalized product of factors. Let us denote the set of variables of the i-th factor as $\widehat{D_i}$ (e.g., $\widehat{D_1} := \{\widehat{f_1}, \widehat{f_{12}}\}$) and the number of factors as n. If $\widehat{\mathcal{X}} = \widehat{\mathcal{K}} \cup \widehat{\mathcal{P}}$ is the whole set of random variables in the network, a MLN formulates the join probability distribution as follows:

$$\Phi(\widehat{\mathcal{X}}) := \frac{1}{Z} \prod_{i=1}^{n} \Phi(\widehat{D_i})$$

where Z is the normalization constant:

$$Z := \sum_{x_1, \ldots, x_{|X|}} \prod_{i=1}^{n} \Phi(D_i)$$

Algorithms such as Belief Propagation or Variable Elimination[5] are iterative methods designed to calculate the stationary probability distribution of the system, that is, the probabilities $P(\widehat{x_i})$. As we discussed in the analysis for Bayesian Networks, one of our goals is to calculate $P(\widehat{q} \mid \widehat{\mathcal{K}})$ for a given prediction $q \notin \mathcal{K}$. From the definition of conditional probability we have:

$$P(\widehat{q} \mid \widehat{\mathcal{K}}) = \frac{P(\widehat{q}, \widehat{\mathcal{K}})}{P(\widehat{\mathcal{K}})} = P(\widehat{q}, \widehat{\mathcal{K}}) = \sum_{\widehat{w} \in \widehat{\mathcal{X}} - \widehat{\mathcal{K}}, \widehat{w} \neq \widehat{q}} P(\widehat{q}, \widehat{w})$$

$P(\widehat{\mathcal{K}}) = \prod_{f_i \in \mathcal{K}} P(\widehat{f_i}) = 1$ due to the assumption that the KB contains only true information and that the truthness of each fact is independent from truthness of the other facts. The sumation over the term $P(\widehat{q}, \widehat{w})$ considers the join probability distribution of $\widehat{q}$ and each $\widehat{w}$ that is neither q nor lies in the evidence set $\widehat{\mathcal{K}}$.

4 Probabilistic Databases

Probabilistic databases are an attractive representation system in our setting. They allow us to represent the uncertainty of predictions as well as to keep track of the sources that influence the belief of a given prediction. For iterative inference, pc-tables are the most suitable representation. Table 6 illustrates a pc-table for the example illustrated in Figure 2.

In a pc-table each tuple is assigned a boolean expression $\Psi[X]$ on a set of boolean variables X. Like in our representation based on graphical models, each statement is mapped to a boolean variable. We define additional boolean

Statements	Formula ($\Psi[f_i]$)
$f_1 := \mathit{marriedTo}(\mathit{Roy\ Gibson}, \mathit{Alice})$	$\widehat{f_1}$
$f_2 := \mathit{isCitizenOf}(\mathit{Roy\ Gibson}, \mathit{England})$	$\widehat{f_2}$
$p := \mathit{isCitizenOf}(\mathit{Roy\ Gibson}, \mathit{UK})$	$(\widehat{R_1} \wedge \widehat{f_2}) \vee \widehat{p}$
$q := \mathit{isCitizenOf}(\mathit{Alice}, \mathit{England})$	$(\widehat{R_2} \wedge \widehat{f_1} \wedge \widehat{f_2}) \vee \widehat{q}$
$r := \mathit{isCitizenOf}(\mathit{Alice}, \mathit{UK})$	$(\widehat{R_1} \wedge \Psi[q]) \vee (\widehat{R_2} \wedge \Psi[p] \wedge \widehat{f_1}) \vee \widehat{r}$

$\widehat{f_1}$	$P(\widehat{f_1})$
T	1.0
∅	0.0

$\widehat{f_2}$	$P(\widehat{f_2})$
T	1.0
∅	0.0

$\widehat{p}$	$P(\widehat{p})$
T	0.0
∅	1.0

$\widehat{q}$	$P(\widehat{q})$
T	0.0
∅	0.0

$\widehat{r}$	$P(\widehat{r})$
T	0.0
∅	1.0

$\widehat{R_1}$	$P(\widehat{R_1})$
T	0.9
∅	0.1

$\widehat{R_2}$	$P(\widehat{R_2})$
T	0.7
∅	0.3

Figure 6: pc-table representation for our example of iterative inference.

variables $\widehat{R_i}$ for the rules used for inference. The pc-table also stores the probability distribution of the variables. Facts in the KB are certain, whereas deductions are by default non-true. For a logical rule R_i, $P(\widehat{R_i}) = PCA(R_i)$ and $P(\neg\widehat{R_i}) = 1 - PCA(R_i)$. Table 6 shows the probability distribution for all the variables below the table storing the facts. The main table contains the facts and their associated boolean formulas. For example, the formula for prediction r, denoted as $\Psi[r]$ states that r is true if R_1 and q are true or if R_2, p and f_1 are true. Some observations:

1. The representation in this example could be simplified by replacing $\widehat{f_i}$ with *true*.

2. The boolean variable associated to each prediction (at the end of the expressions) can be used to add the prediction to the evidence set for queries of the form $P(\widehat{q} \mid \widehat{E})$. Such probability can be calculated by evaluating $\Psi[q](\widehat{E})$ and calculating the probability of the resulting expression. For example $P(\widehat{r} \mid \widehat{f_1}, \widehat{f_2}, \widehat{q}) = P(\Psi[q](\widehat{f_1}, \widehat{f_2}, \widehat{q})) = P(\widehat{R_1} \vee (\widehat{R_2} \wedge (\widehat{R_1} \vee \widehat{p})) \vee \widehat{r})$. $\widehat{r}$ does not affect the probability score.

3. Rules are assumed to be independent, even though pc-tables do not restrict to single probability distributions. In our example if R_1 and R_2 happened to be correlated, we could have a table storing their join distribution and use it in the evaluation of queries.

4. This model can be extended to support functional and cardinality constraints, however it seems tricky. In this example we could model a hard functional constraint on nationality by making the truth values for r and q mutually exclusive. We can achieve this if we rewrite their corresponding formulas as follows:

$$\Psi'[r] := \Psi[r] \wedge \neg\widehat{q}$$

$$\Psi'[r] := \Psi[q] \wedge \neg\widehat{r}$$

Since p and r are predictions, $P(\neg\widehat{r}) = P(\neg\widehat{p}) = 1$. If for instance one of the facts is added to the evidence, e.g., $\widehat{r}$, we get $P(\widehat{q} \mid \widehat{\mathcal{K}} \cup \{\widehat{r}\}) = P(\Psi'[q](\widehat{\mathcal{K}} \cup \{\widehat{r}\})) = P(\Psi[q](\widehat{\mathcal{K}}) \wedge false) = P(false) = 0$.

Modelling arbitrary functionality constraints increases the complexity of the boolean formulas. In general if we have a set $P = \{p_1, \ldots, p_k\}$ of predictions that may violate a hard n-cardinality constraint ($k > n$), we can enforce the constraint in the boolean formula for a given prediction p_i as

$$\Psi'(p_i) := \Psi(p_i) \wedge \bigvee_{s \in \mathcal{P}_n(\widehat{\mathcal{P}} - \{\widehat{p_i}\})} \neg s \wedge s^c$$

where $\Psi(p_i)$ is the original boolean formula for p_i without cardinality constraints, $\mathcal{P}_n$ stands for power set of size n and s^c is the complement of s, i.e., $s^c = (\widehat{P} - \{\widehat{p_i}\}) - s$. We define $\neg\{x_1, \ldots, x_n\} = \{\neg x_1, \ldots, \neg x_n\}$. Soft n-cardinality constraints are much more tricky to model in a pc-table. They would require us to write down a probability distribution for the combinations of truth values for the potentially contradicting statements, similarly as we did in the context of MLNs.

References

[1] S. Abiteboul, D. Deutch, and V. Vianu. Deduction with Contradictions in Datalog. In *International Conference on Database Theory*, Athens, Greece, 2014.

[2] L. Galárraga, G. Heitz, K. Murphy, and F. M. Suchanek. Canonicalizing open knowledge bases. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, CIKM '14, pages 1679–1688, New York, NY, USA, 2014. ACM.

[3] L. Galárraga, N. Preda, and F. M. Suchanek. Mining rules to align knowledge bases. In *Proceedings of the 2013 Workshop on Automated Knowledge Base Construction*, AKBC '13, pages 43–48, New York, NY, USA, 2013. ACM.

[4] L. Galárraga, C. Teflioudi, K. Hose, and F. Suchanek. Amie: Association rule mining under incomplete evidence in ontological knowledge bases. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 413–422, Republic and Canton of Geneva, Switzerland, 2013. International World Wide Web Conferences Steering Committee.

[5] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning.* The MIT Press, 2009.