A Probabilistic Evaluation Function for Relaxed Unification

Tony Abou-Assaleh Nick Cercone Vlado Kešelj
Faculty of Computer Science
Dalhousie University
Halifax, Nova Scotia, Canada
{taa, nick, vlado}@cs.dal.ca

Abstract

Classical unification is strict in the sense that it requires a perfect agreement between the terms being unified. In practise, data are seldom error-free and can contain incorrect information. Classical unification fails when the data are imperfect. Relaxed unification is a new formalism that relaxes the rigid constraints of classical unification and enables reasoning under uncertainty and in the presence of inconsistent data. We propose a probabilistic evaluation function to evaluate the degree of mismatches in relaxed terms and illustrate its use with an example.

1. Introduction

The classical unification function [3, 4] takes two terms as input and produces a boolean value indicating whether the unification can be performed successfully. In the case of a result of true, the function also returns a substitution that unifies these two terms. The unification fails if the same feature is assigned different values in the objects being unified. This process places rigid constraints on the data requiring it to be correct and consistent. Since real-world data is seldom perfect, the classical unification fails at the encounter of the slightest error. Erroneous data often contains enough information that one can exploit to overcome the errors. In other cases, it is possible to draw approximate or less certain conclusions.

Relaxed unification [1, 2] provides a method for extracting information from imperfect data. To achieve this functionality, we relax the constraint that the values being unified must be identical. Instead, each value is replaced with a set containing the value as an element. Unifying two sets containing different values results in a new set containing the values from both sets. Since relaxed unification always succeeds, an evaluation function is needed to compute the degree of the mismatch in terms. We present a mechanism of assigning probabilities to edges in a relaxed term, and an evaluation function that computes the probability of correctness of relaxed terms.

2. Probabilistic Relaxed Terms

A probabilistic relaxed term is a rooted, finite, directed, connected, labelled graph defined by the tuple $t = \langle S_t, \overline{s}_t, F_t, \theta_t, \omega_t \rangle$, where S_t is a nonempty set of nodes, $\overline{s}_t \in S_t$ is the root node, F_t is a set of directed edges labelled with function symbols such that every node $s \in S_t$ is accessible from $\overline{s}_t$, $\theta_t : S_t \to \{\textit{attribute}, \textit{value}\}$ identifies some nodes as attribute nodes and others as value nodes, and $\omega_t : F_t \to [0, 1]$ assigns weights to edges subject to

$$\forall s \in S_t : \sum_{f \in F_t, Source(f)=s} \omega_t(f) = 1,$$

where $Source(f)$ is the source node of the edge f. Edges outgoing from the same node must have distinct labels. We represent weights as superscripts to the function symbols labelling the edges. For simplicity, we omit weights of 1.

3. Probabilistic Evaluation Function

We construct an evaluation function suitable for computing the correctness of probabilistic relaxed terms. The intuition behind our construction is that every path in a relaxed term is a way of accessing some information. A random walk of the term t starting at the root $\overline{s}_t$ imposes a probability distribution over Π_t, the set of all paths in t. Each path $\pi_t(s) \in \Pi_t$ from the root $\overline{s}_t$ to a node $s \in S_t$ has probability of being selected, $P(\pi_t(s))$, and a probability of being correct, $P_c(\pi_t(s))$.

During a random walk, we visit nodes and edges. When we reach a node s, we have the choice of stopping at s with probability $P_{stop}(s)$ or following one of $|s|$ edges outgoing from s with probability $1 - P_{stop}(s)$; the probability of selecting an edge f outgoing from s is given by $\omega_t(f)$. The

value of $P_{stop}(s)$ must satisfy the following condition:

$$\forall s \in S_t : \left\{ \begin{array}{rcl} P_{stop}(s) = 1 & \iff & s = \emptyset \\ 0 < P_{stop}(s) < 1 & \iff & s \neq \emptyset \end{array} \right\}. \quad (1)$$

The value of $P_{stop}(s)$ can either be a constant, such as 0.5, or depend on $|s|$, e.g., $P_{stop}(s) = 1/(|s| + 1)$. Thus, the probability of a path $\pi_t(s) = (f_1, f_2, \ldots, f_n), n \geq 0$ is defined as

$$P(\pi_t(s)) = P_{stop}(s) \prod_{i=1}^{n} (1 - P_{stop}(s_{f_i}))\omega_t(f_i), \quad (2)$$

where s_{f_i} is the source node of edge f_i.

Each edge f has a probability of being correct $P_c(f)$ that depends on the source node $s_f = Source(f)$. If $\theta_t(s_f) =$ *attribute* then $P_c(f) = 1$. Otherwise, $\theta_t(s_f) =$ *value* and $P_c(f) = 1/|s_f|$. The correctness probability of of a path is defined as

$$P_c(\pi_t(s)) = \prod_{i=1}^{n} P_c(f_i). \quad (3)$$

If the root node has no outgoing edges, i.e., $\overline{s}_t = \emptyset$, then Π_t contains a single path, the empty path $()$, and its expected correctness is 1. Otherwise, the correctness value $\delta_t(t)$ of a term t is an expectation value of the correctness function P_c in the discrete random variable $\pi_t(s)$; i.e.,

$$\begin{array}{rcl} \delta_t(t) & = & \langle P_c(\pi_t(s)) \rangle \\ & = & \sum_{\pi_t(s) \in \Pi_t} P_c(\pi_t(s)) \times P(\pi_t(s)). \end{array} \quad (4)$$

Proposition 1 *The correctness value $\delta(t)$ from equation 4 satisfies $\forall t : 0 \leq \delta(t) \leq 1$. This property holds even when t contains cycles. Cycles in the term create infinite paths. Since each edge f along the path is multiplied by $1 - P_{stop}(s_f)$, which, according to condition 1, is less than 1, the sum of the geometric series for the correctness value of paths converges.*

Proposition 2 *The correctness value $\delta(t)$ from equation 4 satisfies $\forall t, u : \delta(t) > \delta(u) \iff t$ is more accurate than u. In particular, if all the value nodes in S_t contain at most one outgoing edge, then $\delta(t)$ is equal to 1. In other words, relaxed terms that can be derived directly from classical terms have a correctness of 1. Similarly, any term containing a value node with more than one outgoing edge will have a correctness value less than 1. Thus, setting the pruning threshold to 1 enables relaxed unification to act as classical unification.*

4. Example

Initially, we are presented with two terms, t_1 and t_2, that represent two employees, John and Bob, respectively, and a query q for the name of the employee with ID 123 and Age 21. The terms and the query are represented as follows:

$$\begin{array}{l} t_1 = \{\mathrm{ID}\{123\}, \mathrm{Name}\{\mathrm{John}\}, \mathrm{Age}\{22\}\} \\ t_2 = \{\mathrm{ID}\{124\}, \mathrm{Name}\{\mathrm{Bob}\}, \mathrm{Age}\{21\}\} \\ q = \{\mathrm{ID}\{123\}, \mathrm{Name}\{\}, \mathrm{Age}\{21\}. \end{array}$$

We decide that 'ID' is a more important attribute than the other two, and that 'Name' is more important than 'Age'. Accordingly, we choose to associate a weight of 0.6 with the attribute 'ID', 0.3 with 'Name', and 0.1 with 'Age'. Observe that the sum of the weights for these three attributes is 1. We assign a weight of 1 to all the other edges. The new representation of the terms and the query is

$$\begin{array}{rl} t_1 & = \{\mathrm{ID}^{0.6}\{123\}, \mathrm{Name}^{0.3}\{\mathrm{John}\}, \mathrm{Age}^{0.1}\{22\}\} \\ t_2 & = \{\mathrm{ID}^{0.6}\{124\}, \mathrm{Name}^{0.3}\{\mathrm{Bob}\}, \mathrm{Age}^{0.1}\{21\}\} \\ q & = \{\mathrm{ID}^{0.6}\{123\}, \mathrm{Name}^{0.3}\{\}, \mathrm{Age}^{0.1}\{21\}\}. \end{array}$$

To answer the query q, we relax unify it with each of the terms t_1 and t_2, giving

$$\begin{array}{l} q \sqcup_R t_1 = \\ \quad \{\mathrm{ID}^{0.6}\{123\}, \mathrm{Name}^{0.3}\{\mathrm{John}\}, \mathrm{Age}^{0.1}\{21^{0.5}, 22^{0.5}\}\} \\ q \sqcup_R t_2 = \\ \quad \{\mathrm{ID}^{0.6}\{123^{0.5}, 124^{0.5}\}, \mathrm{Name}^{0.3}\{\mathrm{Bob}\}, \mathrm{Age}^{0.1}\{21\}\}. \end{array}$$

We have two candidate answers for the 'Name' attribute: John and Bob. To determine which answer is more accurate, we apply the evaluation function δ from equation 4 to $q \sqcup_R t_1$ and $q \sqcup_R t_2$, with $P_{stop}(s) = 0.5$ for nonempty nodes, which produces

$$\begin{array}{rcl} \delta(q \sqcup_R t_1) & = & 0.9875 \\ \delta(q \sqcup_R t_2) & = & 0.925. \end{array}$$

The evaluation shows that $q \sqcup_R t_1$ is greater than $q \sqcup_R t_2$, implying that $q \sqcup_R t_1$ has a higher probability of being the correct answer than $q \sqcup_R t_2$; therefore, we conclude that John is the name that we are looking for.

Acknowledgement The first author is supported by the NSERC Post Graduate Scholarship (PGS B).

References

[1] T. Abou-Assaleh. Theory of relaxed unification – proposal. Master's thesis, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, December 2002.

[2] T. Abou-Assaleh, N. Cercone, and V. Kešelj. An overview of the theory of relaxed unification. In *Proceedings of the International Conference on Advances in the Internet, Processing, Systems, Interdisciplinary Research, IPSI-2003*, Sveti Stefan, Montenegro, Former Yugoslavia, October 2003.

[3] K. Knight. Unification: A multidisciplinary survey. *ACM Computing Surveys*, 21(1):93–124, 1989.

[4] J. Robinson. A machine-oriented logic based on the resolution principle. *Journal of the ACM*, 12(1):23–41, 1965.