On Probability Distributions for Trees: Representations, Inference and Learning

François Denis, Amaury Habrard, Rémi Gilleron, Marc Tommasi, Édouard Gilbert

On Probability Distributions for Trees: Representations, Inference and Learning

François Denis and **Amaury Habrard**
Laboratoire d'Informatique Fondamentale de Marseille (L.I.F.)
UMR CNRS 6166 — http://www.lif.univ-mrs.fr

Rémi Gilleron and **Marc Tommasi**
INRIA Futurs and Lille University, LIFL, Mostrare Project
http://www.grappa.univ-lille3.fr/mostrare

Édouard Gilbert
ÉNS de Cachan, Brittany extension
INRIA Futurs and Lille University, LIFL, Mostrare Project

We study probability distributions over free algebras of trees. Probability distributions can be seen as particular *(formal power) tree series* [BR82; EK03], i.e. mappings from trees to a semiring K. A widely studied class of tree series is the class of *rational* (or *recognizable*) tree series which can be defined either in an algebraic way or by means of multiplicity tree automata. We argue that the algebraic representation is very convenient to model probability distributions over a free algebra of trees. First, as in the string case, the algebraic representation allows to design learning algorithms for the whole class of probability distributions defined by rational tree series. Note that learning algorithms for rational tree series correspond to learning algorithms for weighted tree automata where both the structure and the weights are learned. Second, the algebraic representation can be easily extended to deal with unranked trees (like XML trees where a symbol may have an unbounded number of children). Both properties are particularly relevant for applications: nondeterministic automata are required for the inference problem to be relevant (recall that Hidden Markov Models are equivalent to nondeterministic string automata); nowadays applications for Web Information Extraction, Web Services and document processing consider unranked trees.

1 Representation Issues

Trees, either ranked or unranked, arise in many application domains to model data. For instance XML documents are unranked trees; in natural language processing (NLP), syntactic structure can often be considered as treelike. From a machine learning perspective, dealing with tree structured data often requires to design probability distributions over sets of trees. This problem has been addressed mainly in the NLP community with tools like probabilistic context free grammars [MS99].

Weighted tree automata and tree series are powerful tools to deal with tree structured data. In particular, probabilistic tree automata and stochastic series, which both define probability distributions on trees, allow to generalize usual techniques from probabilistic word automata (or hidden markov models) and series.

Tree Series and Weighted Tree Automata In these first two paragraphs, we only consider the case of ranked trees. A tree series is a mapping from the set of trees into some semiring K. Motivated by defining probability distributions, we mainly consider the case $K = \mathbb{R}$. A *recognizable tree series* [BR82] S is defined by a finite dimensional vector space V over K, a mapping μ which maps every symbol of arity p into a multilinear mapping from V^p into V (μ uniquely extends into a morphism from the set of trees into V), and a linear form λ. $S(t)$ is defined to be $\lambda(\mu(t))$. Tree series can also be defined by *weighted tree automata* (WTA). A WTA A is a tree automaton in which every rule is given a weight in K. For every run r on a tree t (computation of the automaton according to rules over t), a weight $A(t, r)$ is computed multiplying weights of rules used in the run and the final weight of the state at the root of the tree. The weight $A(t)$ is the sum of all $A(t, r)$ for all runs r over t.

For commutative semirings, recognizable tree series in the algebraic sense and in the automata sense coincide because there is an equivalence between summation at every step and summation over all runs. It can be shown, as in the string case, that the set of recognizable tree series defined by deterministic WTA is strictly included in the set of recognizable tree series. A Myhill-Nerode Theorem can be defined for WTA over fields [Bor03].

Probability Distributions and Probabilistic Tree Automata A probability distribution S over trees is a tree series such that, for every t, $S(t)$ is between 0 and 1, and such that the sum of all $S(t)$ is equal to 1. Probabilistic tree automata (PTA) are WTA verifying normalization conditions over weights of rules and weights of final states. They extend probabilistic automata for strings and we recall that nondeterministic probabilistic string automata are equivalent to hidden Markov models (HMMs). As in the string case [DEH06], not all probability distributions defined by WTA can be defined by PTA. However, we have proved that any distribution defined by a WTA *with non-negative coefficients* can defined by a PTA, too.

While in the string case, every probabilistic automaton defines a probability distribution, this is no longer true in the tree case. Similarly to probabilistic context-free grammars [Wet80], probabilistic automata may define inconsistent (or improper) probability distributions: the probability of all trees is less than one. We have defined a sufficient condition for a PTA to define a probability distribution and a polynomial time algorithm for checking this condition.

Towards unranked trees Until this point, we only have considered ranked trees. However, unranked trees can be expressed by ranked ones using an isomorphism defined by an algebraic formulation ([CDG+97], chapter 8). It consists in using the right adjonction operator defined by $f(t_1, \ldots, t_{n-1})@t_n = f(t_1, \ldots, t_n)$; any tree can then be written as an expression whose only operator is @, and thus as a binary tree: e.g., $b(a, a, c(a, a))$ corresponds to $@(@(@(b, a), a), @(@(c, a), a))$. WTA for unranked trees can be defined as WTA for ranked trees applied to the algebraic formulation. We call such automata weighted stepwise tree automata (WSTA).

Hedge automata are automata for unranked trees. Each rule of a hedge automaton [CDG+97] is written $f(L) \rightarrow q$ where L is a regular language of word with the set of states of the automata as its alphabet. For weighted hedge automata (WHA), the weight of the rule $f(u) \rightarrow q$ is the product of a weight given to the whole rule $f(L) \rightarrow q$ and the weight of u according to a weighted word automata associated to $f(L) \rightarrow q$. When K is commutative, WSTA and WHA define the same weight distributions on unranked trees.

Probabilistic hedge automata can be defined by adding the same kind of summation conditions than on WHA, but it has yet to be shown that they can be expressed by PTA through algebraic formulation. We don't know yet weither defining series on unranked trees directly is possible, although it can be achieved using the algebraic formulation.

2 Learning Probability Distributions

Inference and Training PTA can be considered as generative models for trees. The two classical **inference problems** are : given a PTA A and given a tree t, compute $p(t)$ which is defined to the sum over all of all $p(t, r)$; and given a tree t, find the most likely (or Viterbi) labeling (run) $\hat{r}$ for t, i.e. compute $\hat{r} = \arg\max_r p(r|t)$. It should be noted that the inference problems are relevant only for nondeterministic PTA. The **training problem** is: given a sample set S of trees and a PTA, learn the best real-valued parameter vector (weights assigned to rules and to states) according to some criteria. For instance, the likelihood of the sample set or the likelihood of the sample over Viterbi derivations. Classical algorithms for inference (the message passing algorithm) and learning (the Baum-Welch algorithm) can be designed for PTA over ranked trees and unranked trees.

Learning Weighted Automata The **learning problem** extends over the training problem. Indeed, for the training problem, the structure of the PTA is given by the set of rules and only weights have to be found. In the learning problem, the structure of the target automaton is unknown. The learning problem is: given a sample set S of trees drawn according to a target rational probability distribution, learn a WTA according to some criteria. If the probability distribution is defined by a deterministic PTA, a learning algorithm extending over the unweighted case has been defined in [COCR01]. However, this algorithm works only for deterministic PTA. We recall that the class of probability distributions defined by deterministic PTA is strictly included in the class of probability distributions defined by PTA [Bor03].

Learning Recognizable Tree Series and thus learning WTA can be achieved thanks to an algorithm proposed by Denis and Habrard [DH07]. This algorithm, which benefits from the existence of a canonical linear representation of series, can be applied to series which take their values in $\mathbb{R}$ or $\mathbb{Q}$ to learn stochastic tree languages. It should be noted that the algebraic view allows to learn probability distributions defined by nondeterministic WTA. Learning probability distributions for unranked trees is ongoing work.

References

[Bor03] Björn Borchardt. The myhill-nerode theorem for recognizable tree series. In Zoltán Ésik and Zoltán Fülöp, editors, *Developments in Language Theory*, volume 2710 of *Lecture Notes in Computer Science*, pages 146–158. Springer Verlag, 2003.

[BR82] Jean Berstel and Christophe Reutenauer. Recognizable formal power series on trees. *Theoretical Computer Science*, 18:115–148, 1982.

[CDG+97] H. Comon, M. Dauchet, R. Gilleron, F. Jacquemard, D. Lugiez, S. Tison, and M. Tommasi. Tree automata techniques and applications. Available on: http://www.grappa.univ-lille3.fr/tata, 1997.

[COCR01] Rafael C. Carrasco, José Oncina, and Jorge Calera-Rubio. Stochastic inference of regular tree languages. *Machine Learning*, 44(1/2):185–197, 2001.

[DEH06] François Denis, Yann Esposito, and Amaury Habrard. Learning rational stochastic languages. In Gabor Lugosi and Hans Ulrich Simon, editors, *Learning theory*, Lecture Notes in Computer Science. Springer Verlag, 2006.

[DH07] François Denis and Amaury Habrard. Learning rational stochastic tree languages. In Markus Hutter, Rocco A. Servedio, and Eiji Takimoto, editors, *Algorithmic Learning Theory, 18th International Conference*, volume 4754 of *Lecture Notes in Artificial Intelligence*, pages 242–256. Springer Verlag, 2007.

[EK03] Z. Esik and W. Kuich. Formal tree series. *Journal of Automata, Languages and Combinatorics*, 8:219 – 285, 2003.

[MS99] C. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing.* MIT Press, Cambridge, 1999.

[Wet80] C. S. Wetherell. Probabilistic languages: A review and some open questions. *ACM Comput. Surv.*, 12(4):361–379, 1980.