Two-level probabilistic grammars for natural language parsing
Infante Lopez, G.G.

Citation for published version (APA):

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: http://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.
Abstract

Natural language is a very complex phenomenon. Undoubtedly, the sentences we utter are organized according to a set of rules or constraints. In order to communicate with others, we have to stick to these rules up to a certain degree. This set of rules, which is language dependent, is well-known to all speakers of a given language, and it is this common knowledge that makes communication possible. Every sentence has a clear organization: words in an utterance glue together to describe complex objects and actions. This hidden structure, called syntactic structure, is to be recovered by a parser. A parser is a program that takes a sentence as input and tries to find its syntactic organization. A parser searches for the right structure among a set of possible analyses, which are defined by a grammar. The language model decides what the syntactic components of the sentence are and how they are related to each other, depending on the required level of detail.

Designing and building language models is not a trivial task; the design cycle usually comprises designing a model of syntax, understanding its underlying mathematical theory, defining its probability distribution, and finally, implementing the parsing algorithm. The building of a new language model has to complete at least these steps, and each of them is very complex and a line of research in itself. To help handling the intrinsic complexity of this cycle a good level of abstraction is required.

Our view is that state-of-the-art natural language models lack abstraction; their design is often ad hoc, and they mix many features that, at least conceptually, should be kept separated. In this thesis, we explore new levels of abstraction for natural language models. We survey state-of-the-art probabilistic language models looking for characteristic features, and we abstract away from these features to produce a general language model. We investigate three state-of-the-art language models and discover that they have one very noticeable feature: the set of rules they use for building trees is built on the fly, meaning that the set of rules is not defined a priori. The formalisms we
review have two different levels of derivations even though this is not explicitly stated. One level is for generating the set of rules to be used in the second step, and the second step is for building the set of trees that characterize a given sentence. Our formalism, based on Van Wijngaarden grammars (W-grammars), makes these two levels explicit. Our approach to parsing comes from a formal language perspective: we identify features that are used by state-of-the-art language models and take a formalism off the shelf and modify it to incorporate the necessary features.

From a theoretical point of view, general models help us to clarify the set of parameters a particular instance has fixed, and to make explicit assumptions that underlie a particular instance. When analyzing the necessary features from the formal language perspective, the need for probabilities and their role in parsing are the first issue to address. We answer many questions regarding the role of probabilities in probabilistic context free grammars. We focus on these grammars because they are central to the formalism we present.

From a computational point of view, general models for which a clear parsing algorithm and a relatively fast implementation can be defined, produce fast and clear implementations for all particular instances.

General models do not add anything *per se*. Their importance is rather in the set of instances they can capture and the new directions they are able to suggest. We show that bilexical grammars, Markovian context free grammars and stochastic tree substitution grammars are instances of our general model. Our model has well-established consistency properties which we use to derive consistency properties of these three formalisms. The new research directions suggested by a general formalism are a consequence of instantiating the model's parameters in different ways or by re-thinking the set of assumptions the particular instances have made. A brief description of the directions explored in this thesis follows.

Markov models are heavily used in parsing models and they can be replaced by probabilistic regular languages. Since our formalism is not bound to Markov models, we can use any algorithm for inducing probabilistic automata. We explore this idea. We define a type of grammar that uses probabilistic automata for building the set of rules. We compare two classes of grammars that differ in the type of algorithm they use for learning the probabilistic automata. One of them is based on $n$-grams, and the other one is based on the minimum divergence algorithm (MDI). We show that the MDI algorithm produces both smaller and better performing grammars.

The fact that probabilistic automata replace Markov chains in the definition of our model allows us to think of a regular language as the union of smaller, more specific sublanguages. Our intuition is that the sublanguages are easier to induce and that the combination of them fully determines the whole language. We explore this idea by splitting the training material before inducing the probabilistic automata, then inducing
one automaton for each component, and, finally, combining them into one grammar. We show that in this way, a measure that correlates well with parsing performance can be defined over grammars.

Our formalism allows us to isolate particular aspects of parsing. For example, the linear order in which arguments appear in a parse tree is a fundamental feature used by language models. We investigate which sequences of information better predict sequences of dependents. We compare sequences of part-of-speech tags to sequences of non-terminal labels. We show that part-of-speech tags are better predictors of dependents.