The Weak Generative Capacity of Parenthesis-Free Categorial Grammars

Joyce Friedman, Dawei Dai and Weiguo Wang

Computer Science Department, Boston University
111 Cummington Street, Boston, Massachusetts 02215, U. S. A.

Abstract: We study the weak generative capacity of a class of parenthesis-free categorial grammars derived from those of Ades and Steedman by varying the set of reduction rules. With forward cancellation as the only rule, the grammars are weakly equivalent to context-free grammars. When a backward combination rule is added, it is no longer possible to obtain all the context-free languages. With suitable restriction of the forward partial rule, the languages are still context-free and a push-down automaton can be used for recognition. Using the unrestricted rule of forward partial combination, a context-sensitive language is obtained.

INTRODUCTION

The system of categorial grammars, developed in modern times from the work of Ajdukiewicz (1935), has recently been the attention of renewed interest. Inspired by the use of categorial notions in Montague grammar, more recent systems, such as GPSG, have developed related concepts and notations. This in turn leads to a resurgence of interest in pure categorial systems.

Classically, a categorial grammar is a quadruple $G(VT, VA, S, F)$, where VT is a finite set of morphemes, and VA is a finite set of atomic categories, one of which is the distinguished category S. The set CA of categories is formed from VA as follows: (1) VA is a subset of CA, (2) if X and Y are in CA, then (X/Y) is in CA. The grammar also contains a lexicon F, which is a function from words to finite subsets of CA. A categorial grammar lacks rules; instead there is a cancellation rule implicit in the formalism: if X and Y are categories, then $(X/Y)\ Y \rightarrow X$.

The language of a categorial grammar is the set of terminal strings with corresponding category symbol strings reducible by cancellation to the sentence symbol S.

In [1] Ades and Steedman offer a form of categorial grammar in which some of the notations and concepts of the usual categorial grammar are modified. The formalism at first appears to be more powerful, because in addition to the cancellation rule there are several other metarules. However, on closer examination there are other reasons to suspect that the resulting language class differs sharply from that of the traditional grammars. Among the new rules, the forward partial rule (FP rule) is most interesting, since one may immediately conclude that this rule leads to a very large number of possible parsings of any sentence (almost equal to the number of different binary trees of n leaves if the length of the sentence is n). But its effects on the generative power of categorial grammar are not really obvious and immediate. Ades and Steedman raised the question in the footnote 7 in [1] and left it unanswered. We will first formally define categorial grammar and the associated concepts. Then we analyze the generative power of the categorial grammars with different interesting combinations of the reduction rules.

The categorial grammars considered here consist of both a categorial component and a set of reduction rules. The category symbols differ from the traditional ones because they are parenthesis-free. The categorial component consists as before of a set VA of atomic categories including a distinguished symbol S, and a lexical function F mapping words to finite sets of categories. However, the definition of category differs: (1) VA is a subset of CA, (2) if X is in CA, and A is in VA, then X/A is in CA. Notice that the category symbols are parenthesis-free; the implicit parenthesization is left-to-right. Thus the symbol $(A/(B/C))$ of traditional categorial grammar is excluded, since $A/B/C$ abbreviates $((A/B)/C)$. However, some of the rules treat $A/B/C$ as though it were, in fact, $(A/(B/C))$.

DEFINITIONS

Notation. We use A, B, C to denote atomic category symbols, and U, V, X, Y to denote arbitrary (complex) category symbols. The number of occurrences of atomic category symbols in X is $|X|$. Strings of category symbols are denoted by x, y. Morphemes are denoted by a, b; morpheme strings by u, v, w.

A *categorial grammar* under certain reduction rules is a quadruple $G_R = (VT, VA, S, F)$, where: VT is a finite set of morphemes, VA a finite set of atomic categories, $S \in VA$ a distinguished element, F a function from VT to 2^{CA} such that for every $a \in VT$, $F(a)$ is finite, where CA is the *category set* and is defined as: i) if $A \in VA$, then $A \in CA$, ii) if $X \in CA$ and $A \in VA$, then $X/A \in CA$, iii) nothing else is in CA.

The set of *reduction rules* R can include any combination of the following:

(1) (F Rule) If $U/A \in CA$, $A \in VA$, the string $U/A\ A$ can be replaced by U. We write: $U/A\ A \rightarrow U$;

(2) (FP Rule) If U/A, $A/V \in CA$, where $A \in VA$, the string $U/A\ A/V$ can be replaced by U/V. We write: $U/A\ A/V \rightarrow U/V$;

(3) (FP_2 Rule) If U/A, $A/B \in CA$, where A, $B \in VA$, the string $U/A\ A/B$ can be replaced by U/B. We write: $U/A\ A/B \rightarrow U/B$;

(4) (FP_S Rule) Same as (2) except that U/A must be headed by S;

(5) (B Rule) If $U/A \in CA$, $A \in VA$, the string $A\ U/A$ can be replaced by U. We write: $A\ U/A \rightarrow U$;

(6) (B_S Rule) Same as (5) except that U/A must be headed by S.

When it won't cause confusion, we write G_R to denote a categorial grammar with rule set R, and specify a categorial grammar by just specifying its lexicon F.

The *reduce relation* $=>$ on $CA^* \times CA^*$ is defined as: for all $\alpha, \beta \in CA^*$ and all $X,Y,Z \in CA$, $\alpha XY\beta => \alpha Z\beta$ if $XY \rightarrow Z$. Let $=>^*$ denote the reflexive and transitive closure of relation $=>$.

A *morpheme string* $w = w_1 w_2 \cdots w_n$, where $w_i \in VT$, $i = 1,2, \cdots n$, is *accepted* by $G_R = (VT, VA, S, F)$ if there is $X_i \in F(w_i)$ for $i = 1,2, \cdots n$, such that $X_1 X_2 \cdots X_n =>^* S$.

The *language accepted* by $G_R = (VT, VA, S, F)$, $L(G_R)$ is the set of all morpheme strings that are accepted by G_R.

The *categorial grammar recognition problem* is: given a categorial grammar $G_R = CG_R(VT, VA, S, F)$ and a morpheme string $w \in VT^*$, decide whether $w \in L(G_R)$.

The *derivable category set* $DA \subseteq CA$ under a set R of reduction rules is the set of categories including all the primary categories designated by F, and all the reachable categories under that set of reduction rules. It is formally defined as: i) X is in DA if there is an $a \in VT$ such that $X \in F(a)$, ii) For all X, $Y \in DA$ and $Z \in CA$, if $X\ Y \rightarrow Z$ by some rule in R then $Z \in DA$, and iii) Nothing else is in DA.

GRAMMARS WITH FORWARD CANCELLATION ONLY

We begin by looking at the most restricted form of the reduction rule set $R = \{\mathrm{F}\}$. The single cancellation rule is the forward combination rule. It is well-known that traditional categorial grammars are equivalent to context-free grammars. We examine the proof to see that it still goes through for categorial grammars G_R with $R = \{\mathrm{F}\}$.

Theorem The categorial grammars G_R, $R = \{\mathrm{F}\}$, generate exactly the context-free languages.

Proof (1) Let G_R be a categorial grammar with $R = \{\mathrm{F}\}$. G_R becomes a traditional categorial grammar once parentheses are restored by replacing them from left to right, so that, e.g., $A/B/C$ becomes $((A/B)/C)$. Hence, its language is CF.

(2) To show that every context-free language can be obtained, we begin with the observation that every context free language has a grammar in Greibach 2-form, that is, with all rules of the three forms A -> aBC, A -> aB, and A -> a, where A, B, C are in VN and a is in VT [6]. A corresponding classical categorial grammar can be immediately constructed: $F(a) \supseteq \{((A/C)/B), (A/B), A\}$. These are the categories $A/C/B$, A/B, and A of a parenthesis-free categorial grammar. The details of the proof can be easily carried out to show that the two languages generated are the same.

GRAMMARS WITH BACKWARDS CANCELLATION

The theorem shows that with $R = \{\mathrm{F}\}$ exactly the context-free languages are obtained. What happens when the additional metarules are added? We examine now parenthesis-free categorial grammars with $R = \{\mathrm{F}, \mathrm{B}\}$ and $R = \{\mathrm{F}, \mathrm{B}_{\mathrm{S}}\}$. Rule B_{S} is the version adopted in [1]; B is an obvious generalization. In either case we are adding the equivalent of context-free rules to a grammar; the result must therefore still yield a context-free language. So one guess might be that categorial grammars of these types will still yield exactly the context-free languages, perhaps with more structures for each sentence. An alternative conjecture would be that fewer languages are obtained, for we have now added some "involuntary" context-free rules to every grammar.

Example: Consider the standard context-free language $L_1 = \{ a^n b^n \mid n > 0\}$. The easiest grammar is S -> aSb, S -> ab. The Greibach 2-form grammar is S -> aSB, B -> b, S -> aB. The constructed categorial grammar G_R then has $F(a) = \{S/B, S/B/S\}$ and $F(b) = \{B\}$. If $R = \{\mathrm{F}\}$, this yields exactly L_1. However, with $R = \{\mathrm{F}, \mathrm{B}\}$ or $R = \{\mathrm{F}, \mathrm{B}_{\mathrm{S}}\}$, here equivalent, G_R yields a language $L_2 = \{ab, ba, aabb, abab, bbaa, baba, baab, \ldots\}$, which contains L_1 and other strings as well. It is the language of the context-free grammar with rule set $\{S$ -> bC, S -> Cb, C -> aS, C -> Sa, C -> $a\}$.

Reversible languages. Let x^R be the reverse of string x. That is, if $x = a_1 a_2 \cdots a_n$ $(a_i \in VT)$, then $x^R = a_n \cdots a_2 a_1$. Call a language L *reversible* if $x \in L$ iff $x^R \in L$.

Examples: The set of all strings on $\{a, b\}$ with equal numbers of a's and b's is a reversible CF language. $\{a^n b \mid n > 0\}$ is not a reversible language.

Theorem The languages of categorial grammars G_R with $R = \{\mathrm{F}, \mathrm{B}\}$ are reversible.

Proof If $x \Rightarrow {}^*S$, then $x^R \Rightarrow {}^*S$ by a reduction whose tree is the mirror image of the one for x in which rules F and B have been interchanged.

Theorem Let G_R be a categorial grammar with R contains $\{\mathrm{F}, \mathrm{B}\}$ or $\{\mathrm{F}, \mathrm{B}_{\mathrm{S}}\}$. R may or may not also contain some form of FP rules. If $L(G_R)$ contains any sentence of length greater than one, then it contains at least one sentence $w = uv$ such that vu is also in $L(G_R)$.

Proof Let w be a sentence of $L(G_R)$ of length greater than one. Suppose the final step of the reduction to S uses rule F. Then $w = u\,v$ where $u \Rightarrow{}^* S/A$ and $v \Rightarrow{}^* A$. But then $v\,u \Rightarrow{}^* A\ S/A \Rightarrow S$ by rule B or B_{S}. No form of FP can be used as the final step of the reduction to S, so its presence does not affect the result.

Corollary There are context-free languages that cannot be obtained by any categorial grammar G_R, where R contains $\{\mathrm{F}, \mathrm{B}\}$ or $\{\mathrm{F}, \mathrm{B}_{\mathrm{S}}\}$.

CATEGORIAL GRAMMAR IS CONTEXT-FREE IF THE FP RULE IS RESTRICTED

The method that had been used to construct a context-free grammar G equivalent to a classical categorial grammar can be formally described as following:

(1) For each $a \in VT$, if $X \in F(a)$, then put X -> a in G;

(2) For each derivable category X/Y, put $X \rightarrow X/Y\ Y$ in G.

This method remains valid when B_{S} rule is added. We just need to put an additional rule X -> $Y\ X/Y$ in G whenever X is headed by S. But this doesn't work when the FP rule is allowed. We might put in the CF rule U/V -> $U/A\ A/V$ for each derivable category U/V and for each atomic category A, but in case there is a category like $A/B/A$, then any category symbol headed by A followed by B's and ended by A is a derivable category. There are infinitely many of them, so by using this construction method, we might have to put in an infinite number of CF rules. Therefore, this method does not always find a finite context-free grammar equivalent to a category grammar with the FP rule. As we shall see, there may be no such context-free grammar.

Let's now enforce some restrictions on the FP rule so that it won't cause an infinite number of derivable categories. Actually, using the FP rule sometimes violates the parenthesis convention, e.g. applying FP on A/B $B/C/D$ implies that $B/C/D$ is interpreted as $(B/(C/D))$. However, by the parenthesis convention, $B/C/D$ is the abbreviation of $((B/C)/D)$. Notice, however, when the second category symbol has exactly two atomic symbols, i.e., is in form A/B, the FP rule does not violate the convention. Coincidentally, if the FP rule is accordingly restricted as to FP_2, the derivable category set becomes finite.

Lemma For a categorial grammar $G_R(VA, VT, S, F)$, let $R_1 = \{\mathrm{F}, \mathrm{FP}_2\}$, $R_2 = \{\mathrm{F}, \mathrm{FP}_2, \mathrm{B}_{\mathrm{S}}\}$; and $R_3 = \{\mathrm{F}, \mathrm{FP}_2, \mathrm{B}\}$, then $DA_{R_1} = DA_{R_2} = DA_{R_3}$.

Proof From the definition ii) of DA, we can see that any new category Z added to DA by a form of the B rule can be added by the F rule. The lemma follows.

□

Lemma The derivable category set DA of a categorial grammar G_R with $R = \{\mathrm{F}, \mathrm{FP}_2\}$ is finite and constructible.

Sketch of Proof We begin with the observation that none of the reduction rules in R increases the length of category symbols, and the initial lexical category symbols are all of finite length. This implies that the length of all the derivable category symbols are bounded. So there are only finitely many of them.

We now give an algorithm for computing DA, to show that it is constructible.

```
Algorithm: Compute $DA$ of a $G_R$ with $R = \{\mathrm{F}, \mathrm{FP}_2\}$.
Input: A categorial grammar $G_R(VT, VA, S, F)$ with
$R = \{\mathrm{F}, \mathrm{FP}_2\}$.
Output: $DA$ of $G_R$.
Method:
  Let $DA = \bigcup_{a \in VT} F(a)$;
  Repeat
    For all non-atomic categories $U/A \in DA$
      (1) If $A \in DA$ Then $DA = DA \cup \{U\}$;
      (2) For all non-atomic categories $A/B \in DA$
```

```
        $DA = DA \cup \{U/B\}$;
    Until $DA$ was not updated during the last iteration.
    Return $DA$.
```

□

Theorem For every categorial grammar $G_R(VT, VA, S, F)$, with $R = \{\text{F}, \text{FP}_2, \text{B}_\text{S}\}$, there is a context-free grammar $G(VT, VN, S, P)$ such that $L(G_R) = L(G)$.

Sketch of Proof Since DA is finite, the method for converting CG to CFG described in last section works.

□

Remark The theorem remains true for R being $\{\text{F}, \text{FP}_2\}$ and $\{\text{F}, \text{FP}_2, \text{B}\}$, and can be similarly proved. We choose $R = \{\text{F}, \text{FP}_2, \text{B}_\text{S}\}$ to state the theorem because it is closest to Ades and Steedman's model [1].

THE FP RULE IS USEFUL ONLY ON S-HEADED CATEGORIES

Now the next question is what if the FP rule is not restricted to $U/A \; A/B \rightarrow U/B$. Intuitively, we can see that the application of the FP rule on a category which is not headed by S is not crucial in the sense that it can be replaced by an application of the F rule, because whenever $U/A \; A/V$ appears in a valid derivation to a sentence, the V part must be cancelled out sooner or later, so we can make a new derivation that cancels the V part first and get $U/A \; A$ on which we can apply the F rule instead of the FP rule. But this doesn't hold if U/A is headed by S. For example, when we have $A \; S/B \; B/A$, we can't do backward combination on A and S/A if we don't combine S/B and B/A first. So, we expect that the weak generative power of categorial grammar wouid remain unchanged if the FP rule is restricted to be used only on categories which are headed by S. This in fact follows as our next theorem.

Lemma Given a categorial grammar $G_R(VT, VA, S, F)$ with $R = \{\text{F}, \text{FP}, \text{B}_\text{S}\}$, for any $w \in CA^*$ and $A \in VA$, if there is a reduction $w \Rightarrow^* A$, then there is a reduction of w to C using FP rule only on categories which are headed by S.

Sketch of Proof Formalize the idea illustrated above. □

As an almost immediate consequence, we have:

Theorem The language accepted by categorial grammar $G_R(VT, VA, S, F)$ with $R = \{\text{F}, \text{FP}, \text{B}_\text{S}\}$ is the same as that accepted with $R = \{\text{F}, \text{FP}_\text{S}, \text{B}_\text{S}\}$.

Proof It trivially follows the lemma. □

Corollary FP rule is useless if there is no form of the B rule, i.e., any $G_R(VT, VA, S, F)$ with $R = \{\text{F}, \text{FP}\}$ will generate the same language as that generated with $R = \{\text{F}\}$.

A CONTEXT-SENSITIVE LANGUAGE GENERATED USING UNRESTRICTED FP RULE

This section gives a categorial grammar with unrestricted FP rule that generates a language which is not context-free. Consider categorial grammar $G_1 = G_R(VA, VT, S, F)$, where $VT = \{a, b, c\}$, $VA = \{A, C, S\}$, $F(a) = \{A\}$, $F(b) = \{S/A/C/S, S/A/C\}$, $F(c) = \{C\}$, and $R = \{\text{F}, \text{B}_\text{S}, \text{FP}\}$.

Claim 1 $a^i b^i c^i \in L(G_1)$ for $i > 0$.

Proof For any $i > 0$, we can find a corresponding categorial string for $a^i b^i c^i$: $A^i (S/A/C/S)^{i-1} (S/A/C) C^i$. A reduction to S is straightforward. □

Let $\phi_w(a)$ denote the number of occurrences of a in string w.

Claim 2 For all $w \in VT^*$, if $w \in L(G_1)$ then $\phi_w(a) = \phi_w(b) = \phi_w(c)$.

Proof First, it is easy to see that from the lexical categories, we cannot get any complex category headed by either A or C, and we can get atomic category symbol A or C only directly from the lexicon.

Second, each morpheme b would introduce one A and one C within a complex category symbol which must be cancelled out sooner or later in order to reduce the whole string to S. In general, there are two ways for such A and C being cancelled: (1) with an A-headed or C-headed complex category by the FP rule, which is impossible in this example; (2) with a single atomic category A or C by either the F or B_S rule. We have seen that such single A and C can only be introduced by the morpheme a and c, respectively. So $\phi_w(a) = \phi_w(b) = \phi_w(c)$.

□

To show that $L(G_1)$ is not context-free, we take its intersection with the regular language $a^*b^*c^*$. By claim 1 and 2, the intersection is exactly the language $\{a^n b^n c^n \mid n > 0\}$ which is well known to be non context-free. Since the intersection of a context free language with a regular set must be context-free, $L(G_1)$ cannot be context-free.

PROCESSORS

A categorial grammar is certainly no worse than context-sensitive. We can verify this by using a nondeterministic linear bounded automaton to model the reduction process. For even in the case of reduction by the unrestricted FP rule, the category symbol obtained by reduction is shorter than the combined length of the two inputs to the rule.

Ades and Steedman [1] propose a processor that is a pushdown stack automaton and pushdown stack automata are known to correspond to the context-free languages. How can we reconcile this with the context sensitive example above? The contradiction arises because the stack of their processor must be able to contain any derived category symbol of DA, and thus the size of the stack symbols is unlimited. The processor is thus not a pushdown automaton in the usual sense.

Acknowledgment: We would like to thank Ramarathnam Venkatesan and Remko Scha for helpful discussions of the material. This work was supported in part by National Science Foundation Grant No. IST-8317736.

BIBLIOGRAPHY

[1] Ades, A. E., Steedman, M. J. (1982) "On the order of words", *Linguistics and Philosophy*, vol. 4, pp. 517-558.
[2] Ajdukiewicz, K. (1935) "Die syntaktische konnexitat", *Studia Philosophica*, vol. 1, pp. 1-27, translated into English as "Syntactic Connection" in S. McCall, ed., *Polish Logic 1920-1939*, Oxford: Clarendon Press, 1967.
[3] Bar-Hillel, Y. (1950) "On syntactical categories", *Journal of Symbolic Logic*, vol. 15 pp. 1-16, reprinted in Bar-Hillel (1964), pp. 19-37.
[4] Bar-Hillel, Y. (1953) "A quasi-arithmetical notation for syntactic description", *Language*, vol. 29, pp. 47-58, reprinted in Bar-Hillel (1964), pp. 61-74.
[5] Bar-Hillel, Y. (1964) *Language and Information*, Reading, Mass.: Addison-Wesley.
[6] Greibach, S. (1965) "A new normal form theorem for context-free phrase structure grammars", *J. ACM* vol. 12, No. 1, pp. 42-52.
[7] Gaifman, H. (1965) "Dependency Systems and Phrase-Structured Systems", *Information and Control*, vol. 8, No. 3, pp. 304-337.