Non-Context-free Languages: Pumping on Steroids and Closure Revisited

Is Every L a CFL?

Again, just "counting" says no:

Fixed an alphabet Σ

Let $\Gamma = \Sigma \cup \{\varepsilon, \rightarrow, |, ;, A, {}_0, {}_1\}$

I can encode every *grammar* over Σ as a *single* string over the somewhat larger finite alphabet Γ, e.g. :

"$A_{01} \rightarrow aA_1bA_{01} \mid \varepsilon;\ A_1 \rightarrow A_{01}$"

Since Γ^* is countably infinite, but the set of languages $L \subseteq \Sigma^*$ is uncountably infinite, non-context-free languages must exist.

(I could encode every grammar as a single string of bits, too, so the dependence on Σ above is unnecessary, but avoids some technical details.)

What are some concrete examples of non-CFLs?

Examples

Which are CFLs?

$\{a^i b^j c^k \mid i=j \text{ or } i=k\}$ CFL

$\{a^n b^n c^n \mid n \geq 0\}$ nonCFL

$\{ww^R \mid w \in \{a,b\}^*\}$ CFL

$\{ww \mid w \in \{a,b\}^*\}$ nonCFL

Q: How might we prove such facts?
A: Via a CFL-specific form of the "Pumping Lemma."

The Pumping Lemma for Context-free Languages

$\forall$ CFL A $\exists p$ st $\forall s \in A$
if $|s| \geq p$ then $\exists u,v,x,y,z \in \Sigma^*$
st
(o) $s = u \cdot v \cdot x \cdot y \cdot z$
(i) $\forall i \geq 0 \quad u v^i x y^i z \in A$
(ii) $|vy| > 0$
(iii) $|vxy| \leq p$

Example

$L = \{ a^n b^n c^n \mid n \geq 0 \}$ is not a CFL

$\forall$ CFL A $\exists p$ st $\forall s \in A$
if $|s| \geq p$ then $\exists u,v,x,y,z \in \Sigma^*$
st
(0) $s = u \cdot v \cdot x \cdot y \cdot z$
(i) $\forall i \geq 0$ $u v^i x y^i z \in A$
(ii) $|vy| > 0$
(iii) $|vxy| \leq p$

Suppose L were a CFL. Let p be the constant from the pumping lemma & let $s = a^p b^p c^p$. By the pumping lemma there are strings u, v, x, y, z such that…

Since $|vxy| \leq p$, vxy cannot include both a and c.

Case 1: vxy does not contain a "c". Then uv^0xy^0z has p c's, but fewer a's or b's (or both), hence is not in L

Case 2: vxy does not contain an "a". Then uv^0xy^0z has p a's, but fewer b's or c's (or both), hence is not in L.

Contradiction. Thus L is not a CFL

To prove the pumping lemma, this fact about trees will be useful:

Lemma: a b-ary tree of height h has $\leq b^h$ leaves

$\bullet \leq 3^0$ $\leq 3^1$ $\leq 3^2$ …

Conversely, $\geq b^h$ leaves implies height $\geq h$

$\forall$ CFL A $\exists p$ st $\forall s \in A$
if $|s| \geq p$ then $\exists u,v,x,y,z \in \Sigma^*$
st
(0) $s = u \cdot v \cdot x \cdot y \cdot z$
(i) $\forall i \geq 0$ $u v^i x y^i z \in A$
(ii) $|vy| > 0$
(iii) $|vxy| \leq p$

Proof idea

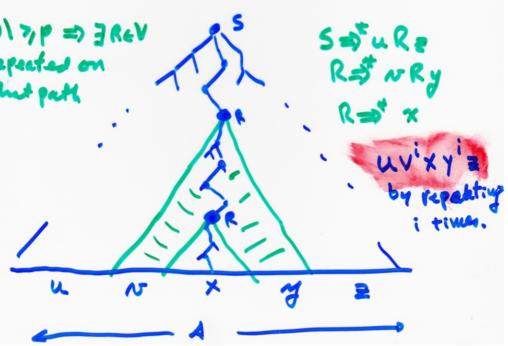
G: a cfg for A

b = length of longest r.h.s. of a rule in G

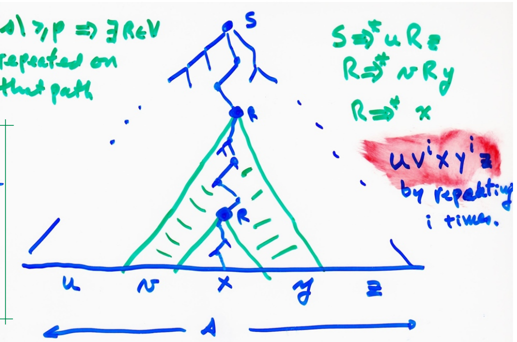
$p = b^{n+1}$ where $n = |V|$, # of vars in G

$s \in L(G)$ with $|s| \geq p$

Pick a smallest parse tree for s and a longest path in that tree



$\forall$ CFL A $\exists p$ st $\forall s \in A$
if $|s| \geq p$ then $\exists u,v,x,y,z \in \Sigma^*$
st
(0) $s = u \cdot v \cdot x \cdot y \cdot z$
(i) $\forall i \geq 0$ $u v^i x y^i z \in A$
(ii) $|vy| > 0$
(iii) $|vxy| \leq p$



Why a repeat? Pigeon-Hole Principle, again

$> b^{|V|}$ leaves $\Rightarrow$ $> |V|$ path length

$\Rightarrow$ some variable R repeated.

Why $vy \neq \varepsilon$?

because it was smallest tree

Why $|vxy| \leq p$?

Pick repeat nearest leaf

Example

$L = \{ ww \mid w \in \{a,b\}^* \}$

~~$s = a^p b a^p b$~~

$s = a^p b^p a^p b^p$

original middle

a a⋯a a a b b⋯b b b a a⋯a a b⋯b b

$k/2$

new middle

$|vxy| \le p$ ∴ confined to at most 2 adjacent blocks of a's & b's.

case 1 $|uvxy| \le 2p$

uv^0xy^0z removed k letters from left half $1 \le k \le p$

Last letter of (new) left half is a, but last of right half is b.

∴ $\notin L$

case 2 vxy in right half: similar

case 3 vxy straddles middle.

$uv^0xy^0z = a^p b^i a^j b^p$

for some $i \le p, j \le p$

not both $i = j = p$

$i < j$ new left half ends with a, right half with b

$j < i$ new right half starts with b, left half with a

$i = j < p$ $a^p b^i \ne a^i b^p$

"Corollary"

$\{ ww \mid w \in \{a,b\}^* \}$ not CFL "⇒" Java not CFL

"ww" is representative of programming languages that require variables to be declared (1st w) before use (2nd w).

None of these languages (C, C++, Java,...) are CFLs at this level.

But CFGs are still very useful in compilers! The parse tree defines the structure of the program:
"this is a variable name in a declaration"
"this is a variable name in an expression"

Details like "is this name declared somewhere" are easily tacked on: store in dictionary at decl; look up in expr.

Some closure & non-closure results

$L_1 = \{a^m b^m c^n \mid m,n \ge 0\}$ is a CFL

$L_2 = \{a^m b^n c^n \mid m,n \ge 0\}$ is a CFL

$L_1 \cap L_2 = \{a^n b^n c^n \mid n \ge 0\}$ is *not* a CFL

Therefore, the set of CFLs is *not* closed under intersection

Therefore, *not* closed under complementation, either

Fact: if L is CFL & R is regular, then $L \cap R$ is CFL

Ex: L_3 = {w|w has equal numbers of a's, b's, & c's} is not a CFL, since $L_3 \cap a^*b^*c^* = \{a^n b^n c^n \mid n \ge 0\}$, which is not CFL

Summary

There are many non-context-free languages (uncountably many, again)

Famous examples: $\{ ww \mid w \in \Sigma^* \}$ and $\{ a^n b^n c^n \mid n \geq 0 \}$

"Pumping Lemma": uv^ixy^iz ; v-y pair comes from a repeated var on a long tree path

Unlike the class of regular languages, the class of CFLs is *not* closed under intersection, complementation; *is* closed under intersection with regular languages (and various other operations; see exercises in text).