Context-Free and Noncontext-Free Languages

Examples:

a*b* is regular.

$A^nB^n = \{a^nb^n : n \geq 0\}$ is context-free but not regular.

$A^nB^nC^n = \{a^nb^nc^n : n \geq 0\}$ is not context-free

The Regular and the CF Languages

Theorem: The regular languages are a proper subset of the context-free languages.

Proof: In two parts:

Every regular language is CF.

There exists at least one language that is CF but not regular.

The Regular and the CF Languages

Lemma: Every regular language is CF.

Proof: Every FSM is (trivially) a PDA:

Given an FSM $M = (K, \Sigma, \Delta, s, A)$ and elements of δ of the form:

(p, c, q)

old state, input, new state

Construct a PDA $M' = (K, \Sigma, \{\varnothing\}, \Delta, s, A)$. Each (p, c, q) becomes:

((p, c, ε), (q, ε))

old state, input, don't look at stack, new state don't push on stack

In other words, we just don't use the stack.

The Regular and the CF Languages

Lemma: There exists at least one language that is CF but not regular

Proof: $\{a^n b^n, n \geq 0\}$ is context-free but not regular.

So the regular languages are a proper subset of the context-free languages.

Showing that L is Context-Free

Techniques for showing that a language *L* is context-free:

1. Exhibit a context-free grammar for *L*.

2. Exhibit a PDA for *L*.

3. Use the closure properties of context-free languages.

Unfortunately, these are weaker than they are for regular languages.

Showing that *L* is Not Context-Free

Pumping Theorem for Context-free Languages

If *L* is a context-free language, then

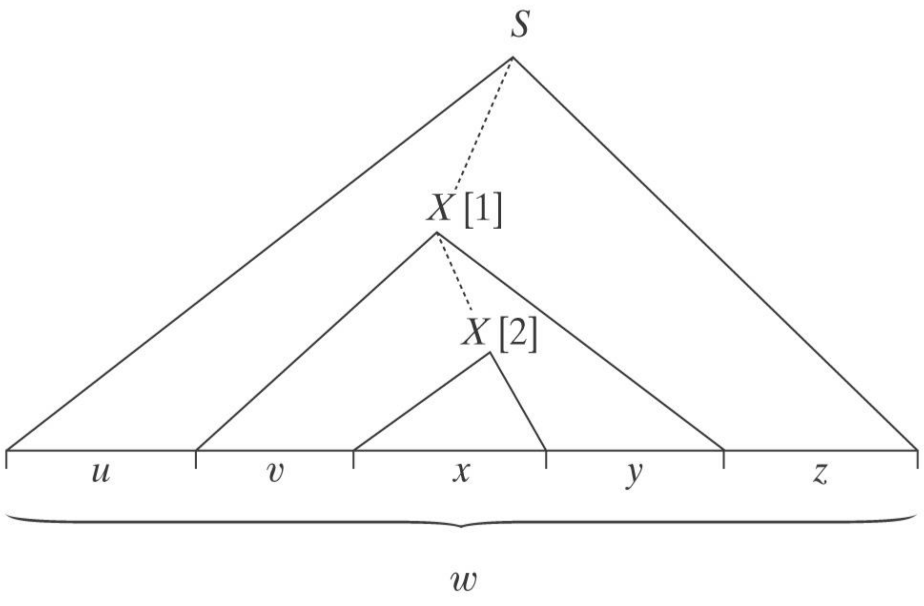
$\exists k \geq 1$ ($\forall$ strings $w \in L$, where $|w| \geq k$

($\exists u, v, x, y, z$ ($w = uvxyz$,

$vy \neq \varepsilon$,

$|vxy| \leq k$ and

$\forall q \geq 0$ (uv^qxy^qz is in L)))).



Pumping Theorem for Context-free Languages

Proof: *L* is generated by some CFG $G = (V, \Sigma, R, S)$ with *n* nonterminal symbols and branching factor *b*. Let *k* be b^{n+1}. The longest string that can be generated by *G* with no repeated nonterminals in the resulting parse tree has length b^n. Assuming that $b \geq 2$, it must be the case that $b^{n+1} > b^n$. So let *w* be any string in $L(G)$ where $|w| \geq k$. Let *T* be any smallest parse tree for *w*. *T* must have height at least $n + 1$. Choose some path in *T* of length at least $n + 1$. Let *X* be the bottom-most repeated nonterminal along that path. Then *w* can be rewritten as *uvxyz*. The tree rooted at [1] has height at most $n + 1$. Thus its yield, *vxy*, has length less than or equal to b^{n+1}, which is *k*. $vy \neq \varepsilon$ since if *vy* were ε then there would be a smaller parse tree for *w* and we chose *T* so that that wasn't so. *uxz* must be in *L* because $rule_2$ could have been used immediately at [1]. For any $q \geq 1$, uv^qxy^qz must be in *L* because $rule_1$ could have been used *q* times before finally using $rule_2$.

An Example of Pumping: $A^nB^nC^n$

$A^nB^nC^n = \{a^n b^n c^n, n \geq 0\}$

Choose $w = a^k b^k c^k$

1 | 2 | 3

If either *v* or *y* spans regions, then let *q* = 2 (i.e., pump in once). The resulting string will have letters out of order and thus not be in $A^nB^nC^n$.

If both *v* and *y* each contain only one distinct character then set *q* to 2. Additional copies of at most two different characters are added, leaving the third unchanged. There are no longer equal numbers of the three letters, so the resulting string is not in $A^nB^nC^n$.

An Example of Pumping: $\{a^{n^2} : n \geq 0\}$

$L = \{a^{n^2}, n \geq 0\}$. If $n = k^2$, then $n^2 = k^4$. Let $w = a^{k^4}$.

$vy = \mathtt{a}^p$, for some nonzero p.

Set q to 2. The resulting string, s, is a^{k^4+p}. It must be in L. But it isn't because it is too short:

w:	next longer string in L:
$(k^2)^2$ $\mathtt{a}$'s	$(k^2 + 1)^2$ $\mathtt{a}$'s
k^4 $\mathtt{a}$'s	$k^4 + 2k^2 + 1$ $\mathtt{a}$'s

For s to be in L, $p = |vy|$ would have to be at least $2k^2 + 1$.

But $|vxy| \leq k$, so p can't be that large. Thus s is not in L and L is not context-free.

Example: WcW = {*w*c*w* : *w* ∈ {a, b}*}

Let $w = a^k b^k c a^k b^k$.

```
aaa … aaabbb …  bbbcaaa … aaabbb … bbb
|     1     |     2    |3|   4    |    5    |
```

Call the part before c the left side and the part after c the right side.

- If *v* or *y* overlaps region 3, set *q* to 0. The resulting string will no longer contain a c.
- If both *v* and *y* occur before region 3 or they both occur after region 3, then set *q* to 2. One side will be longer than the other.
- If either *v* or *y* overlaps region 1, then set *q* to 2. In order to make the right side match, something would have to be pumped into region 4. Violates $|vxy| \le k$.
- If either *v* or *y* overlaps region 2, then set *q* to 2. In order to make the right side match, something would have to be pumped into region 5. Violates $|vxy| \le k$.

Example: WcW = $\{wcw : w \in \{a, b\}^*\}$

Corollary:

Java is not context-free

Closure Theorems for Context-Free Languages

The context-free languages are closed under:

- Union
- Concatenation
- Kleene star
- Reverse
- Letter substitution

Closure Under Union

Let $G_1 = (V_1, \Sigma_1, R_1, S_1)$, and
$G_2 = (V_2, \Sigma_2, R_2, S_2)$.

Assume that G_1 and G_2 have disjoint sets of nonterminals, not including S.

Let $L = L(G_1) \cup L(G_2)$.

We can show that L is CF by exhibiting a CFG for it:

$$G = (V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, R_1 \cup R_2 \cup \{S \rightarrow S_1, S \rightarrow S_2\}, S)$$

Closure Under Concatenation

Let $G_1 = (V_1, \Sigma_1, R_1, S_1)$, and
$G_2 = (V_2, \Sigma_2, R_2, S_2)$.

Assume that G_1 and G_2 have disjoint sets of nonterminals, not including S.

Let $L = L(G_1)L(G_2)$.

We can show that L is CF by exhibiting a CFG for it:

$$G = (V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, \; R_1 \cup R_2 \cup \{S \rightarrow S_1 \, S_2\}, S)$$

Closure Under Kleene Star

Let $G = (V, \Sigma, R, S_1)$.

Assume that G does not have the nonterminal S.

Let $L = L(G)^*$.

We can show that L is CF by exhibiting a CFG for it:

$$G = (V_1 \cup \{S\}, \Sigma_1, \; R_1 \cup \{S \to \varepsilon, S \to S\, S_1\}, S)$$

Closure Under Reverse

$L^R = \{w \in \Sigma^* : w = x^R$ for some $x \in L\}$.

Let $G = (V, \Sigma, R, S)$ be in Chomsky normal form.

Every rule in G is of the form $X \rightarrow BC$ or $X \rightarrow a$, where X, B, and C are elements of $V - \Sigma$ and $a \in \Sigma$.

- $X \rightarrow a$: $L(X) = \{a\}$. $\{a\}^R = \{a\}$.
- $X \rightarrow BC$: $L(X) = L(B)L(C)$. $(L(B)L(C))^R = L(C)^R L(B)^R$.

Construct, from G, a new grammar G', such that $L(G') = L^R$:

$G' = (V_G, \Sigma_G, R', S_G)$, where R' is constructed as follows:

- For every rule in G of the form $X \rightarrow BC$, add to R' the rule $X \rightarrow CB$.
- For every rule in G of the form $X \rightarrow a$, add to R' the rule $X \rightarrow a$.

Closure Under Intersection

The context-free languages are not closed under intersection:

The proof is by counterexample. Let:

$L_1 = \{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^m: n, m \geq 0\}$ /* equal a's and b's.
$L_2 = \{\mathtt{a}^m\mathtt{b}^n\mathtt{c}^n: n, m \geq 0\}$ /* equal b's and c's.

Both L_1 and L_2 are context-free, since there exist straightforward context-free grammars for them.

But now consider:

$$L = L_1 \cap L_2 = \{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^n: n \geq 0\}$$

Closure Under Complement

The context-free languages are not closed under complement:

The proof :

$$\textit{We know that } L_1 \cap L_2 = \neg(\neg L_1 \cup \neg L_2)$$

The context-free languages are closed under union, so if they were closed under complement, they would be closed under intersection (which they are not).

The Intersection of a Context-Free Language and a Regular Language is Context-Free

$L = L(M_1)$, a PDA = $(K_1, \Sigma, \Gamma_1, \Delta_1, s_1, A_1)$.
$R = L(M_2)$, a deterministic FSM = $(K_2, \Sigma, \delta, s_2, A_2)$.

We construct a new PDA, M_3, that accepts $L \cap R$ by simulating the parallel execution of M_1 and M_2.

$M = (K_1 \times K_2, \Sigma, \Gamma_1, \Delta, (s_1, s_2), A_1 \times A_2)$.

Insert into Δ:

For each rule $((q_1,\ a, \beta), (p_1,\ \gamma))$ in Δ_1,
and each rule $(q_2,\ a,\ p_2)$ in δ,
$(((q_1, q_2), a, \beta), ((p_1, p_2), \gamma))$.

For each rule $((q_1,\ \varepsilon, \beta), (p_1,\ \gamma))$ in Δ_1,
and each state q_2 in K_2,
$(((q_1, q_2), \varepsilon, \beta), ((p_1, q_2), \gamma))$.