CS5371
Theory of Computation
Lecture 9: Automata Theory VII
(Pumping Lemma, Non-CFL)
Objectives

• Introduce Pumping Lemma for CFL

• Apply Pumping Lemma to show that some languages are non-CFL
Pumping Lemma for CFL

Theorem: If \( L \) is a CFL, then there is a number \( p \) (pumping length) where, if \( w \) is any string in \( L \) of length at least \( p \), we can find \( u,v,x,y,z \) with \( w = uvxyz \) and
- for each \( i \geq 0 \), \( uv^i xy^i z \) is in \( L \)
- \( |vy| > 0 \), and
- \( |vxy| \leq p \)
Proof of Pumping Lemma

• Let $b$ be the maximum branching factor in the parse tree of any string in $L$
  - that is, the right side of any rule has at most $b$ terminals and variables)

• We shall use $p = b^{|V|} + 1$ to prove the lemma

• Observation: What is the minimum height of the parse tree for a string $w$ with length at least $p$?
Proof of Pumping Lemma (2)

• Height of the parse tree \( \geq |V| + 1 \)
  \( \Rightarrow \) some path in tree \( \geq |V|+2 \) nodes
• Only one such node can be a terminal
  \( \Rightarrow \) at least \( |V|+1 \) variable on the path
• What does that mean?
  Some variable appears at least twice
Proof of Pumping Lemma (3)

- Let \( R \) be a variable that appears at least twice.
- Then, the parse tree of the string \( w \) looks something like:

\[
\begin{align*}
S & \quad R \\
R & \quad R \\
 R & \\
\quad w = u v x y z
\end{align*}
\]

So, \( uv^i xy^i z \) is in \( L \) for any \( i \geq 0 \) (why??)
$uv^i x y^i z$ is in $L$ for any $i \geq 0$

- **Facts:** $R$ derives $x$, $R$ derives $vxy$
- Since $S$ derives $uRz$, and $R$ derives $x$, $S$ can derive $uxz$

- Since $S$ derives $uvRyz$ and $R$ derives $vxy$, $S$ can derive $uvvyzz$
Proof of Pumping Lemma (5)

- To complete the prove, we need to show $|vy| > 0$ and $|vxy| \leq p$
- The current construction cannot, but we can do so if we further restrict:
  1. parse tree is the smallest among all that can generate the string $w$
  2. $R$ is chosen from the lowest $|V|+1$ variables in the longest root-to-leaf path
\[|vy| > 0\]

- Suppose on the contrary that \(|vy| = 0\)
  - \(\Rightarrow\) Both \(v\) and \(y\) are empty strings
- Then in the parse tree, we replace “Subtree of \(R\) that generates \(vxy\)” by “Subtree of \(R\) that generates \(x\)”
- Resulting parse tree will also generate \(w\) (why?), but it has fewer nodes
  - \(\Rightarrow\) contradiction occurs
\[ |vxy| \leq p \]

- \( R \) is chosen from the lowest \(|V| + 1\) variables in the longest root-to-leaf path
- Consider subtree of \( R \) that generates \( vxy \)
  - Its height is at most \(|V|+1\) (why?)
  - It has at most \( b^{|V|+1} \) leaves
  - Thus, \( vxy \) has at most \( p \) characters
    (as \( p = b^{|V|+1} \))

Recall: \( b = \text{maximum branching factor} \)
Non-CFL (example 1)

Theorem: The language

\[ A = \{a^n b^n c^n \mid n \geq 0\} \]

is not a context-free language.

How to prove?
By contradiction, using pumping lemma
First thing: Assume that \( A \) is CFL
Proof (example 1)

• Let \( p \) be the pumping length
• Let \( w = a^p b^p c^p \) in \( A \), and consider partition \( w \) into any \( u,v,x,y,z \) such that \( w = uvxyz \)
• Two possible cases:
  
  - Case 1: Both \( v \) and \( y \) have only one type of char
  - Case 2: \( v \) or \( y \) has more than one type of char
• In both cases, \( uvvxyyz \) is not in \( A \) (why?)
• Thus, we find a string at least \( p \) long in \( A \) that does not satisfy pumping lemma
  
  \( \Rightarrow \) contradiction occurs
Theorem: The language 
\[ B = \{a^ib^jc^k \mid 0 \leq i \leq j \leq k\} \]
is not a context-free language.

How to prove?
By contradiction, using pumping lemma
First thing: Assume that \( B \) is CFL
Proof (example 2)

• Let $p$ be the pumping length
• Let $w = a^p b^p c^p$ in $B$, and consider partition $w$ into any $u, v, x, y, z$ such that $w = uvxyz$
• Two possible cases:
  Case 1: Both $v$ and $y$ have only one type of char
  Case 2: $v$ or $y$ has more than one type of char
• We can see that for Case 2, $uvvxyyyz$ cannot be in $B$
• How about Case 1?
Proof (example 2)

• Unfortunately, for Case 1, if \( v = b, y = c \), then the string \( uvvxyyz \) is always in \( B \)...

• So, how to get a contradiction??

• We divide Case 1 into two subcases:
  Subcase 1.1: char \( a \) not appear in both \( v \) and \( y \)
  Subcase 1.2: char \( a \) appears in \( v \) or \( y \)
Proof (example 2)

• For Subcase 1.1 (char a not appear in v and y),
  \(uxz\) cannot be in \(B\) [why?]

• For Subcase 1.2 (char a appears in v or y),
  \(uvvxyyz\) cannot be in \(B\) [why?]

• Thus, we find a string at least \(p\) long in \(B\)
  that does not satisfy pumping lemma
  \(\Rightarrow\) contradiction occurs
Non-CFL (example 3)

Theorem: The language
\[ C = \{ww \mid w \text{ in } \{0,1\}^*\} \]
is not a context-free language.

How to prove?
By contradiction, use pumping lemma on
\[ 0p1p0p1p \]
Proof (example 3)

• When \( w = 0^p 1^p 0^p 1^p = uvxyz \), what can be the corresponding \( vxy \)?
  - Case 1: \( vxy \) appears in the first half
  - Case 2: \( vxy \) appears in the second half
  - Case 3: \( vxy \) includes the middle ‘10’

• For Cases 1 or 2, \( uvvxyzz \) not in \( C \) (why?)

• For Case 3, \( u \) must start with \( 0^p \), and \( z \) must end with \( 1^p \) (because \(|vxy| \leq p \) and \( vxy \) includes the middle ‘10’)

  \( \Rightarrow \) Then, \( uxz \) cannot be in \( C \) (why?)
CFL is closed under all regular operations

- **Union:** We have seen that before

- **Concatenation:**
  Let $G_A$ and $G_B$ be CFGs for two CFLs $A$ and $B$, using different sets of variables.
  Let $S_A$ and $S_B$ be their start variables.
  Combine the rules, add rule $S \rightarrow S_A S_B$

- **Star:** Add rule $S \rightarrow S S_A | \epsilon$
CFL closed under complement?

- What is the complement of \( A = \{a^n b^n c^n \mid n \geq 0\} \)?

- The complement of \( A \) includes:
  - strings containing \( ba \), \( ca \), or \( cb \);
  - strings \( a^i b^j c^k \) with \( i \neq j \) or \( j \neq k \)

  \( \Rightarrow \) the complement of \( A \) is a CFL (why??)

- As \( A \) is not a CFL, what can we conclude?
CFL closed under intersection?

• Is $A = \{a^n b^n c^m \mid n,m \geq 0\}$ a CFL?
• Is $B = \{a^m b^n c^n \mid n,m \geq 0\}$ a CFL?
• What is the intersection of $A$ and $B$? Is it a CFL?
• What can we conclude?
What we have learnt so far?

• PDA = CFG
  - Prove by Construction

• Properties of CFG
  - Ambiguous, Chomsky Normal Form

• Pumping Lemma
  - Prove by Contradiction (using Parse Tree)

• Existence of non-CFL
Language Hierarchy

Set of Languages (\(= \) set of “set of strings”)

\[ \{0^n1^n2^n\} \]

\{\{w \mid w = w^R\}\}

\{\{w \mid w \text{ with even } |w|\}\}

\{\{0 \times 1^y\}\}

Set of Regular Language

Set of Context-Free Free Language
Next Time

- Turing Machine
  - A even more power computer