Formal Languages, Automata and Computation

Pumping Lemma

Properties of CFLs

Carnegie Mellon University in Qatar

SUMMARY

- Context-free Languages and Context-free Grammars
- Pushdown Automata
- PDAs accept all languages CFGs generate.
- CFGs generate all languages that PDAs accept.
- There are languages which are NOT context free.

Pumping Lemma for CFLs

Lemma

If L is a CFL, then there is a number p (the pumping length) such that if s is any string in L of length at least p, then s can be divided into 5 pieces $s = uvxyz$ satisfying the conditions:

1. $|vy| > 0$
2. $|vxy| \leq p$
3. *for each* $i \geq 0$, $uv^ixy^iz \in L$

- The pumping length is determined by the number of variables the grammar for L has.

Application of the Pumping Lemma

- Just as for regular languages we employ the pumping lemma in a two-player game setting.
- If a language violates the CFL pumping lemma, then it can not be a CFL.
- Two Player Proof Strategy:
  - Opponent picks p, the pumping length
  - Given p, we pick s in L such that $|s| \geq p$. We are free to choose s as we please, as long as those conditions are satisfied.
  - Opponent picks $s = uvxyz$ - the decomposition subject to $|vxy| \leq p$ and $|vy| \geq 1$.
  - We try to pick an i such that $uv^i xy^i z \notin L$
  - If for all possible decompositions the opponent can pick, we can find an i, then L is not context-free.

USING PUMPING LEMMA – EXAMPLE-1

- Consider the language $L = \{a^n b^n c^n \mid n \geq 0\}$
- Opponent picks p.
- We pick $s = a^p b^p c^p$. Clearly $|s| \geq p$.
- Opponent may pick the string partitioning in a number of ways.
- Let's look at each of these possibilities:

Using Pumping Lemma–Example 1

- Cases 1,2 and 3: *vxy* contains symbols of only one kind
  1. Only *a*'s: $\underbrace{a\cdots a}_{u}\underbrace{a\cdots a}_{vxy}\underbrace{a\cdots ab\cdots bc\cdots c}_{z}$
  2. Only *b*'s: $\underbrace{a\cdots ab}_{u}\underbrace{b\cdots b}_{vxy}\underbrace{b\cdots bc\cdots c}_{z}$
  3. Only *c*'s: $\underbrace{a\cdots ab\cdots bc}_{u}\underbrace{c\cdots c}_{vxy}\underbrace{c\cdots c}_{z}$
- Pumping *v* and *y* will introduce more symbols of one type into the string.
- The resulting strings will not be in the language.

USING PUMPING LEMMA–EXAMPLE 1

- Cases 4 and 5: *vxy* contains two symbols – crosses symbol boundaries.
  1. Only *a*'s and *b*'s: $\underbrace{a\cdots a}_{u}\underbrace{a\cdots ab\cdots b}_{vxy}\underbrace{b\cdots bc\cdots c}_{z}$
  2. Only *b*'s and *c*s: $\underbrace{a\cdots ab\cdots b}_{u}\underbrace{b\cdots c}_{vxy}\underbrace{c\cdots c}_{z}$
- Note that *vxy* has length at most *p* so can not have 3 different symbols.
- Pumping *v* and *y* will both upset the symbol counts and the symbol patterns.
- The resulting strings will not be in the language.

USING PUMPING LEMMA–EXAMPLE 2

- Consider the language $L = \{a^n \mid n \text{ is prime}\}$
- Opponent picks p.
- We pick $s = a^p$. Clearly $|s| \geq p$.
- Opponent may pick any partitioning $s = uvxyz$.
  - Let $m = |uxz|$ for the partitioning selected, that is, the length of everything else but v and y.
  - Any pumped string uv^ixy^iz will have length $m + i(p - m)$.
  - We choose $i = p + 1$.
  - The pumped string has length $m + (p + 1)(p - m)$. But:

$$\begin{aligned} m + (p+1)(p-m) &= m + p^2 - pm + p - m \\ &= p^2 + p - pm \\ &= p(p - m + 1) \end{aligned}$$

which is not prime since both p and $p - m + 1$ are greater than 1. (Note $0 \leq m \leq p - 1$)

Closure Properties of Context-free Languages

- Context-free languages are closed under
  - Union
  - Concatenation
  - Star Closure
  - Intersection with a regular language
- We will provide very informal arguments for these.

CLOSURE PROPERTIES OF CFLS-UNION

- Let G_1 and G_2 be the grammars with start variables S_1 and S_2, variables V_1 and V_2, and rules R_1 and R_2.
- Rename the variables in V_2 if they are also used in V_1
- The grammar G for $L = L(G_1) \cup L(G_2)$ has
  - $V = V_1 \cup V_2 \cup \{S\}$ (S is the new start symbol $S \notin V_1$ and $S \notin V_2$
  - $R = R_1 \cup R_2 \cup \{S \rightarrow S_1 \mid S_2\}$

Closure Properties of CFLs – Concatenation

- Let G_1 and G_2 be the grammars with start variables S_1 and S_2, variables V_1 and V_2, and rules R_1 and R_2.
- Rename the variables in V_2 if they are also used in V_1
- The grammar G for $L = \{wv \mid w \in L(G_1), v \in L(G_2)\}$ has
  - $V = V_1 \cup V_2 \cup \{S\}$ (S is the new start symbol $S \notin V_1$ and $S \notin V_2$
  - $R = R_1 \cup R_2 \cup \{S \rightarrow S_1 S_2\}$

Closure Properties of CFLs – Star Closure

- Let G_1 be the grammar with start variable S_1, variables V_1, rules R_1.
- The grammar G for $L = \{w \mid w \in L(G_1)^*\}$ has
  - $V = V_1 \cup \{S\}$ (S is the new start symbol $S \notin V_1$).
  - $R = R_1 \cup \{S \rightarrow S_1 S \mid \epsilon\}$

Closure Properties of CFLs – Intersection with a Regular Language

- Let P be the PDA for the CFL L_{cfl} and M be the *DFA* for the regular language $L_{regular}$
- We have a procedure for building the cross-product PDA from P and M.
  - Very similar to the cross-product construction for DFAs.
  - Details are not terribly interesting. (Perhaps later.)

Closure Properties of CFLs

- CFLs are NOT closed under intersection.
  - $L_1 = \{a^n b^n c^m \mid n, m \geq 0\}$ is a CFL.
  - $L_2 = \{a^m b^n c^n \mid n, m \geq 0\}$ is a CFL.
  - $L = L_1 \cap L_2 = \{a^n b^n c^n \mid n \geq 0\}$ is NOT a CFL.
- CFLs are not closed under complementation.
  - $L = \{ww \mid w \in \Sigma^*\}$ is NOT a CFL (Prove it using pumping lemma!)
  - $\overline{L}$ is actually a CFL and $L = L_1 \cup L_2$
    - $\overline{L}$ has all strings of odd length (L_1)
    - $\overline{L}$ has all strings where at least one pair of symbols $n/2$ apart are different (n length of the string!) (L_2)
    - $S \rightarrow aA \mid bA \mid a \mid b$
      $A \rightarrow aS \mid bS$
      generates L_1

      $S \rightarrow AB \mid BA$
      $A \rightarrow ZAZ \mid a$
      $B \rightarrow ZBZ \mid b$
      $Z \rightarrow a \mid b$
      generates L_2

CFL Closure Properties in Action

- Is $L = \{a^n b^n \mid n \geq 0, n \neq 100\}$ a CFL?
  - $L = \underbrace{\{a^n b^n \mid n \geq 0\}}_{CFL} \cap \underbrace{(L(a^* b^*) - \{a^{100} b^{100}\})}_{RL}$
  - The intersection of a CFL and a RL is a CFL!

- Is $L = \{w \mid w \in \{a, b, c\}^* \text{ and } n_a(w) = n_b(w) = n_c(w)\}$ a CFL?
  - $\underbrace{L}_{CFL?} \cap \underbrace{L(a^* b^* c^*)}_{RL} = \underbrace{\{a^n b^n c^n \mid n \geq 0\}}_{Not\ CFL}$
  - Thus L is NOT a CFL.

Moving beyond the Milky Way

What other kinds of languages are out there?