COL352 Assignment

Deepanshu Yadav(2019CS10343)
Harikesh(2019CS10355), Sourav (2019CS10404)

March 2022

1 Problem

Claim: A left linear grammar generates a regular language.
Proof: Let $G = (V, \Sigma, S, P)$ be a left linear grammar
Then we can find a NDFA which accepts the language formed by G
$D = (Q, \Sigma, \delta, S, F)$ accepts language generated by G Where
$Q = V$
$\delta(P, a) = Q$ when there is a transition $P \to aQ$ in G
F is the set of all states $A \in V$ such that $A \to \epsilon$
As there exist an NDFA for all left linear grammars, hence the language generated by a left linear grammar is regular Similarly, the language generated by a right linear grammar is also regular.
Let $G = (V, \Sigma, S, P)$ be a non-self-referential grammar.
Define an equivalence class T on V as $T(Q) = (W \in V | \exists a, b, c, d \in (V \cup \Sigma)*$ such that $Q \to *aWb$ and $W \to *cQd)$
Writing $Q \equiv W$ if $W \in T(Q)$ and $Q \in T(W)$
Claim: If $A \in V, B \in V, A \equiv B$ such that $A \to *aBb$ and $B \to *cAd$, then either $a = c = \epsilon$ or $b = d = \epsilon$
Proof:
$A \to *aBb \to *acAbd$
But as we know, the grammar is non referential, hence either $a = c = \epsilon$(left linear grammar) or $b = d = \epsilon$(right linear grammar)
Hence we can reduce the grammar to an equivalent left linear or right linear grammar just by taking all the states reachable to it(as proved above otherwise the grammar would be self referential which is a contradiction)
After reducing the grammar to left or right linear grammar, as we know that the language generated by a left/right linear grammar is regular (proved above)
Hence the self referential grammar is regular.

2 Problem

L_1 is a context free language and L_2 is a regular language.
To Prove: $L_1 \cap L_2$ is context free.
Proof: As L_2 is regular hence there must be a DFA which accepts L_2.
Let $D = (Q_1, \Sigma, \delta_1, q_1, F_1)$ accepts L_2
Let $M = (Q_2, \Sigma, \Gamma, \delta_2, , q_2, F_2)$ be a PDA which accepts L_1.
We can show that $L_1 \cap L_2$ is a context free if we can find a PDA which recognise $L_1 \cap L_2$.
Let we can create a PDA for $L_1 \cap L_2$ as $M_2 = (Q, \Sigma, \Gamma, \delta, q_0, F)$ where
$Q = Q_1$ x Q_2
$q_0 = (q_1, q2)$
$F = F_1$ x F_2
$\delta((p, q), x, a) = \{(p', q', y) | \delta_1(p, a, x) = (p'y)$ and $(q', y) \in \delta_2(q, x, a)\}$
This PDA recognise $L_1 \cap L_2$, hence the language of $L_1 \cap L_2$ is context free.

3 Problem

3.1 Part A

Given that L is a CFL and L' is regular
To Prove: $L||L'$ is context free
Proof:
As L is a CFL, hence there must exist a PDA which accepts L
Let $P = (Q_1, \Sigma, \Gamma, \delta_1, q_1, \perp, F_1)$ accepts L
And a DFA $D = (Q_2, \Sigma, q_2, F_2, \delta_2)$ accepts L'
Then the DFA $P_2 = (Q_1 \times Q_2, \Sigma, \Gamma', \delta, (q_1, q_2), \perp, F_1 \times F_2)$ accepts $L||L'$ where $\Gamma' = \Gamma \cup Z$
$\delta((p_1, p_2), a, Z) = ((p_1, \delta_2(p_2, a)), \epsilon)$
$\delta((p_1, p_2), a, Y) = ((p_3, p_2), Zb) : (p_2, b) \in \delta_1(p_1, a, Y)$ where $Y \neq Z$
Proof: For the proof of correctness we have to show that our newly formed PDA accepts the language $L||L'$. From our definition of our PDA we can clearly see the two of the possible cases on reading the stack alphabet. In the first case directly the transition is made according to the next state possible from the DFA and removes the stack alphabet. In the other case when the top element is not our stack element we will simply make our PDA moving to its next possible state. Also this time we will not remove anything but push the stack alphabet along with the corresponding letter which is there in our regular language. Now to finally show that it will definitely accept the language $L||L'$ we can say that since the acceptance is only when both of the states are final in pair i.e. there are some words let's say w_1 and w_2 such that $w_1||w_2$ when run on our PDA is accepted only when for some transitions the word w_1 reaches a final state f_1 and w_2 reaches a final state f_2. Corresponding to these final states we have (f_1, f_2) being one of the final states of our newly formed PDA. This concludes our proof that our PDA definitely will accept that language.
As there exists a PDA which accepts $L||L'$, hence it is context free.

3.2 Part B

If both L and L' are context free, then $L||L'$ may not be context free.
Proof:
It can be showed for one example. Take context free languages $L = \{x_k y_{2k}, k \geq 1\}$ and $L' = \{x_{2k} y_k, k \geq 1\}$
Then $L||L' = \{x_{2k}(xy)_k y_{2k}\}$ is not a context free language as it does not satisfy the pumping lemma for context free languages.
Hence $L||L'$ may not be context free.

4 Problem

Given: for $A \subseteq \Sigma^*$ $Cycle(A) = \{yx|xy \in A\}$
To Prove: If A is CFL (context free language) then so is $cycle(A)$ i.e. A is CFL (Context free language) $\Rightarrow Cycle(A)$ is CFL(Context free language).
Proof: Claim 1: Take any language $L' = \{pref(A)\}$ which contains prefix of all the words of language A. This is CFL(Context free language).
Proof: As A is assumed to be CFL (Context free language), it must follow CNF(Chomsky Normal Free) and hence should have production rules as $P \to \epsilon$ $P \to a$ $P \to QR$.
Now, we define the set of rules as $P \to \epsilon$ $P \to a$ $P \to Q|QR$. and hence it will contain all set of prefix string of A. Hence $pref(A)$ has a CFG(Context free Grammer) thus L' is CFL(Context Free language).
Claim 2: Take any language $L" = \{suffix(A)\}$ which contains suffices of all the words of language A.This is CFL(Context free language).
Proof: As A is element to be CFL(Context free language) it must follow CNF(Chomsky Normal Form) with production rules as $P \to \epsilon$ $P \to a$ $P \to QR$
Now we define the set of rules as $P \to \epsilon$ $P \to a$ $P \to R|QR$ and this will contain all set of suffix strings of A Thus $suff(A)$ has a CFG(Context free grammer) and thus $L"$ is CFL(Context free language).
Claim 3: CFLs(Context free languages) are closed under concatenation operation(Proved in lecture)
Now we can say that $Cycle(A) = \{yx|xy \in A\}$ is clearly obtained by concatenation of $L"$ and L' i.e $Cycle(A) = \{L', L"\}$ and form all above 3 claims we have $L", L'$ are CFL(Context free language) and CFLs(Context free languages) are closed under concatenation.Thus $Cycle(A)$ is a context free language.

5 Problem

Given: $A = \{wtw^R | w, t \in \{0,1\}^*$ and $|w| = |t|\}$
To Prove: A is not CFL (Context free language).
Proof: we will prove this by contradiction (Pumping lemma for CFLs).
Assumption: Let A be CFL(Context free language) and lets take any integer $n \geq 1$. Now lets consider a string $s = 0^{2n}1^n0^n0^{2n}$ where $|s| > n$ and $w = 0^{2n}, t = 1^n0^n, w^R = 0^{2n}$.Now lets take any split of this string s i.e $s = abcde$ such that $|bd| > 0, |bcd| \leq n$.Thus A being CFL (Context free language) $\Rightarrow$ $ab^icd^ie \in A$ for any $i \geq 0$.
Now we will form cases of splitting of the string:
I: bcd is completely from the first 0^{2n} part.
In this case we will choose i such that $3n < |bd| * (i-1) \leq 4n$
Thus length of $|ab^icd^ie| = (i-1)|bd| + 6n \Rightarrow 9n < length(|ab^icd^ie|) \leq 10n$
So either length of $|ab^icd^ie|$ is not a multiple of 3 or if it is then we see that the first $\frac{1}{3}$rd contains only 0's as our string has minimum $5n$ zeros at start and the last $\frac{1}{3}$rd part will definitely have a 1 as $\frac{1}{3}$rd length $> 3n$ and $(3n+1)^{st}$ element is 1.Thus $w = w^R \Rightarrow ab^icd^ie \notin A$ in this case.
II(a): Note even a single letter of bed is from first.
Now in this case also let's take i=2 which gives $|ab^2cd^2e| = |bd| \neq 6n$
Again either $|ab^2cd^2e|$ is not a multiple of 3 i.e $|w| = |t| = |w^R|$ is not satisfied or if it is multiple of 3.We know that $|bcd| \leq n \Rightarrow |bd| \leq n$ and thus $6n \leq |ab^2cd^2e| \leq 7n$.Thus the $\frac{1}{3}$rd will have length $> 2n$ first $\frac{1}{3}$rd will contain a 1 but lost $\frac{1}{3}$rd will not as $(2n+1)$th element from start is 1 but last $3n$ elements are zero So, $w = w^R \Rightarrow ab^2cd^2e \notin A$ in this case.
II(b): Some part of bcd is from the first 0^{2n}. Now in this case we have our bcd from $0^n0^n1^n0^{2n}$ from the underline part.Now, on taking $i = 6n+1$ we get $ab^{n+1}cd^{n+1}e$ which implies that $|ab^{6n+1}cd^{6n+1}e| = 6n + (6n(|bd|))$ and $|bd| > 0$.Thus length of our now string $> 12n$ Thus $\frac{1}{3}$rd of it will be $> 4n$.
Here we ahve taken $6n+1$ a randomly big value.Now we can see that the last $\frac{1}{3}$rd part contains a 1 as $(3n+1)$th element form behind is 1 and also we know that atleast one of the zero is in b as bcd coincides with the first 0^{2n}.Thus atleast first $6n$ are zeros.Thus there us a 1 atleast in last $\frac{1}{3}$rd part but there is no 1 in first $\frac{1}{3}$rd part Hence $ww^R \Rightarrow ab^{6n+1}cd^{6n+1}e \notin A$ in this case.Hence from above cases we can conclude that A is not context free language.

6 Problem

To Prove: If A is CFL (Context free language) then $\exists\ k$ where if any string s such that $|s| \leq k$ may be divided into 5 pieces $s = uvyxz$ satisfying
for each $i \geq 0, uv^ixy^iz \in A$
$v \notin \epsilon$ and $y \notin \epsilon$ and
$|vxy| \geq k$
Proof: Let there are some production rules of grammer corresponding to our CFL (Context free language) A. Let k denote the rule which is having the maximum size term on RHS. Let w be the size of that term on RHS and h be the height of the parse tree generated. Then the maximum number of leaves in that parse tree is w^h .Now assume that there are nt number of non terminals in the above grammer.Then take for any string having length more than nt.This definitely implies that the number of leaf nodes are greater than nt and hence the depth of our parse tree is more than nt.Hence by pigeon hole principle we can say that there is atleast one-non terminal which is repeating from root to leaf path and let P is that non-terminal symbol.
Now let's take gthe two occurences of P in our string of length more than w^{nt} i.e the first one and last one lets the two trees corresponding to these symbols i.e rooting at there are T and T' Now lets take s as $s = uvxyz$ and where v denotes the string between the first pair of leaves of T and T',y denotes the string between the pair of final leaves of T and T'.On the other hand u is the string formed by the prefix of first leaf of T and z being the suffix of the final leaf of T'.The string x is the string formed by leaves of the subtree T'. Thus this way we have splitted our original string.
As by our assumption initially A occurs more than once, we have that v and y are not empty strings i.e $|v| \geq 1$ and $|y| \geq 1$.Also there is a rule in our grammer to produce p from P, we can easily pump up our string by adding any number of $p's$ to our tree before T' and thus it gives us that any string of the form $uv^ixy^iz \in A\ \forall i \geq 0$ and the final condition can be concluded by the fact that depth of T is at max h and hence $|vxy| \leq w^h \Rightarrow |vxy| \leq k$.Thus there exists such k where any string of length $\geq k$ i.e $\geq w^h$ can be splitted in these 5 parts and will also satisfy the condition.

7 Problem

Let L_1 and L_2 be two languages over $\Sigma = \{a, b, c, d\}$ such that
$L_1 = \{ab^kc^kd^k | k \geq 1\}$ and $L_2 = \{a^k(b + c + d)^* | k \neq 1\}$
$L_3 = L_1 \cup L_2$
Let $L_5 = L_3 \cap L_4 = \{ab^ic^id^i | i \geq 0\}$
Clearly we can see that L_5 is non context free.
Suppose L_3 is context free. As we know L_4 is regular.Hence $L_3 \cap L_4$ must be regular.
$\Rightarrow L_5$ is context free (as intersection of regular context free language is context free).
Contradiction
Hence L_3 is not context free.
Proving that L_3 satisfies pumping lemma for CFLs.
Let the pumping length of L_3 be P.
Taking a string $s = ab^pc^pd^p \ |s| \geq P$
There must exist u, v, x, y, z such that $s = uvxyz$
$uv^kxy^kz \in L_3 \ \forall k \geq 0 \ |vy| \geq 0$ and $|vxy| \leq P$
Taking P=2
Case 1:
$s = a^ib^jc^kd^l \ i \neq 2 \ |s| \geq 2$
Taking $u = \epsilon, v = \epsilon, x = \epsilon$
$y = s[i]$(first symbol of x)
$z =$ remaining x
Then $|vy| = 1 > 0$ Holds true
$|vxy| = 1 \leq 2$ Holds true
and $uv^nxy^nz = y^nz$
$= 0$ or more then i a followed by string of the for $b^*c^*d^*$
$= b^*c^*d^* + aad^*b^*c^*d^* \in L_3$
Hence it lies in L_3
Case 2: $s = a^2b^jc^kd^l \ |s| \geq 2$
Taking $u = \epsilon, v = \epsilon, x = \epsilon$
$y = a^2$
$z = b^jc^kd^l$
$|vy| = 1 > 0$
$|vxy| = 2 \leq 2$
$uv^nxy^nz = a^{2n}b^jc^kd^l \ inL_3$
Hence it also lies in L_3
Hence in both the cases pumping lemma holds.
Hence we can see that L_3 is not a context free language but satisfies the pumping lemma.