Closure properties of CFLs

Closure ?

- The CFL are closed under some operation means after performing that particular operation on those CFLs, The resultant language is CFL.

Closure properties of CFLs

The context free languages **are closed** under the following operations:-

- Union
- Concatenation
- Closure and positive closure
- Homomorphism

Closure properties of CFLs

The context free languages **are not closed** under the following operations:-

- Intersection
- Complement

Closed under Union

- If L1 and L2 are Context free languages then L=L1 U L2 is also Context free.

Proof:-

- Let L1 and L2 be CFLs generated by the CFGs

G1=(V1,T1,P1,S1)

G2=(V2,T2,P2,S2)

- We assume that V1 and V2 are disjoint
- Also S3 is not in V1 or V2

- Construct a grammar

G3=(V3,T3,P3,S3)

V3=V1 U V2 U {S3},

T3=T1 U T2

P3 =P1 U P2 U { S3->S1|S2 }

Closed under Union

- If w is in L1, then the derivation
- S3=>S1=>w is a derivation in G3, As every production of G1 is a production of G3

- Similarly every word in L2 has a derivation in G3 beginning with S3=>S2.

- **L(G3) contains those strings that are derivable from S1 as well as derivable from S2**

- Thus L1 U L2 c L(G3)

- **As all the strings of G1 and G2 can be derived from G3 and can be represented in VTPS format, Thus it is Closed under Union**

Closed under Concatenation

- If L1 and L2 are Context free languages then L=L1.L2 is also Context free.

Proof:-

- Let L1 and L2 be CFLs generated by the CFGs

G1=(V1,T1,P1,S1)

G2=(V2,T2,P2,S2)

- We assume that V1 and V2 are disjoint
- Also S4 is not in V1 or V2

- For L1.L2,
- Construct a grammar

G4=(V1 U V2 U {S4},T1 U T2,P4,S4)

P4=P1 U P2 U {S4->S1S2}

- The proof that L(G4)=L(G1)L(G2) is similar to the proof for union

Closed under Concatenation

For L1.L2,

- Construct a grammar

G4=(V1 U V2 U {S4},T1 U T2,P4,S4)

P4=P1 U P2 U {S4->S1S2}

- The proof that L(G4)=L(G1)L(G2) is similar to the proof for union
- **All the strings S1 followed by strings s2 can be generated by the grammar G4**
- **G4 can be represented in VTPS format**
- **Thus, G4 is closed under Concatenation**

Closed under Concatenation

- If L1 and If L2 are two context free languages, their concatenation L1.L2 will also be context free.

For example,

- L1 = { $a^n b^n$ | n >= 0 }
- L2 = { $c^m d^m$ | m >= 0 }
- L3 = L1.L2 = {$a^n b^n c^m d^m$ | m >= 0 and n >= 0}

- L1 says number of a's should be equal to number of b's
- L2 says number of c's should be equal to number of d's.

- Concatenation says first number of a's should be equal to number of b's, then number of c's should be equal to number of d's.

- We can create a PDA which will first push for a's, pop for b's, push for c's then pop for d's.
- So it can be accepted by pushdown automata, hence context free.
- Thus, CFL are closed under Concatenation.

Closed under Kleene Closure

- If L1 is Context free languages then L=L1* is also Context free.

Proof:-

- Let L1 be CFL generated by the CFG

G1=(V1,T1,P1,S1)

- Also S5 is not in V1

For L1*,

- Construct a grammar

G5=(V1 U {S5},T1 ,P5,S5)

P5=P1 U {S5->S1S5|ε}

- The proof that L(G5)=L(G1)* can be done similarly

Closed under Concatenation

- For $L1^*$,
- Construct a grammar G5=(V1 U {S5},T1 ,P5,S5)

where P5 is P1 plus the production

- S5->S1S5|ε

- **The proof that $L(G5)=L(G1)^*$ can be done similarly**
- **All the strings for $S1^*$ can be generated by the grammar G5**
- **G5 can be represented in VTPS format**
- **Thus, G5 is closed under Kleene Closure**

Not Closed under Intersection

- Let us take two languages L1 and L2
- L1=$\{a^i b^i c^j \mid i>=1 \text{ and } j>=1\}$ and
- L2=$\{a^i b^j c^j \mid i>=1 \text{ and } j>=1\}$

- L1 says number of a’s should be equal to number of b’s
- L2 says number of b’s should be equal to number of c’s.
- Both are CFLs

- A PDA to recognize L1 stores a’s on its stack and cancels them against b’s then accepts its input after reading one or more c’s.

Not Closed under Intersection

- L1 can be generated as
- S->AB
- A->aAb|ab
- B->cB|c
- where A generates $a^i b^i$ and B generates c^j

- L2 can be generated as
- S->CD
- C->aC|a
- D->bDc|bc
- where D generates $b^j c^j$ and C generates a^i

Not Closed under Intersection

- However, L1 ∩ L2=L
- L={$a^i b^i c^i$|i>=1 }
- L1 says number of a’s should be equal to number of b’s
- L2 says number of b’s should be equal to number of c’s.
- A string in both the languages must have equal numbers of all three symbols
- But push down automata can compare only two symbols. So it cannot be accepted by pushdown automata, hence not context free.
- Thus, L is not CFL
- Hence, proved

Not Closed under Complementation

- Let L1, L2 are 2 CFLs
- Lets assume that complement of a CFL is a CFL itself.
- Thus, L1’,L2’ both are CFLs
- Then L=(L1’ U L2’) is also a CFL
- L1 ∩ L2 = (L1’ U L2’)
- we know that intersection is not closed
- Thus, L’ may or may not be CFL
- Hence, Not Closed under Complementation

Intersection with Regular Sets

- CFL are closed under intersection with regular sets
- If L is a CFL and R is a regular set then $L \cap R$ is always a CFL