Monash University
Faculty of Information Technology

FIT2014 Theory of Computation

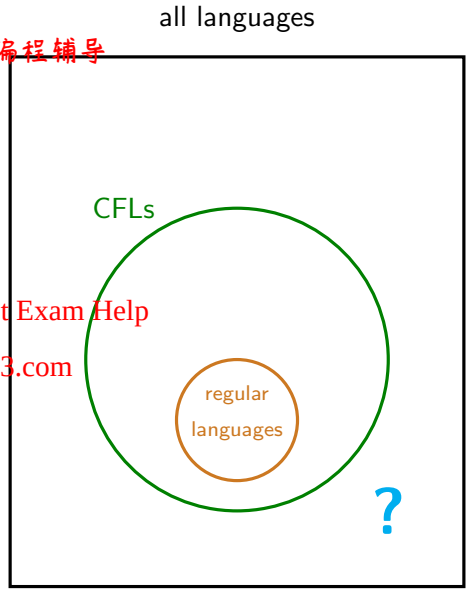
Lecture 17

The Pumping Lemma for Context-Free Languages

slides by Graham Farr

Overview

- Pumping Lemma for CFLs
- Proof
- application:
  showing that some languages
  are *not* context-free



Pumping Lemma for Regular Languages (paraphrased)

Recall:

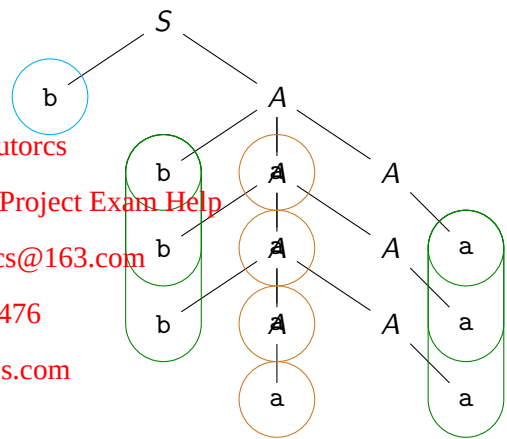
If a Finite Automaton
with N states
accepts
a sufficiently long string,
then the path taken by the string
in the FA
contains a repeated state.

This enables us to "pump" the string
by repeating one substring —
to generate an infinite family
of members of the language.

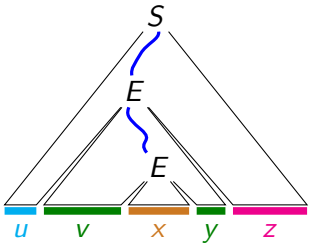
EQUAL

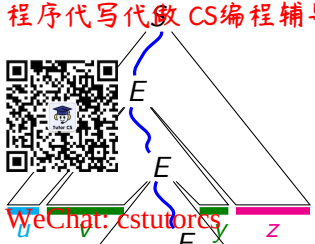
1. $S \to \varepsilon$
2. $S \to \mathtt{b}A$
3. $S \to \mathtt{a}B$
4. $A \to \mathtt{a}$
5. $A \to \mathtt{a}S$
6. $A \to \mathtt{b}AA$
7. $B \to \mathtt{b}$
8. $B \to \mathtt{b}S$
9. $B \to \mathtt{a}BB$

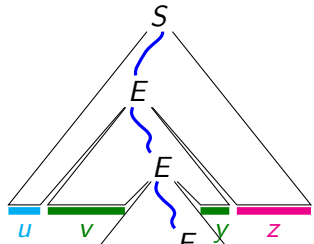
Parse tree for bbbaaaaa



Nonterminal repetition in parse tree paths

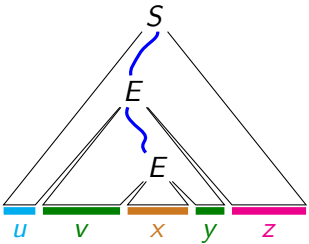


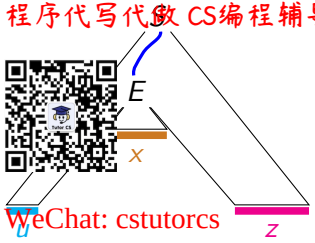




$uvxyz$ uv^2xy^2z uv^3xy^3z

Nonterminal repetition in parse tree paths





$uvxyz$

uxz

Nonterminal repetition in parse tree paths

Nonterminals: S, E, T, F

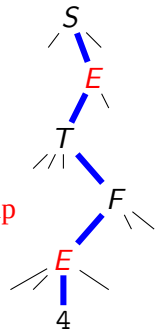
In a parse tree:

If

length of a root-to-leaf path $>$ # nonterminals

then

some nonterminal appears twice on that path.



Nonterminal repetition in parse tree paths

How can we *ensure* that this happens?

How to guarantee that the parse tree for a *sufficiently long* string has a path with a repeated stat

Consider:

length of a path from root to leaf $=$ # non-leaf nodes in that path.

Each non-leaf node has a nonterminal symbol.

If

max root-to-leaf path length $>$ # nonterminal symbols in the grammar

then some nonterminal symbol occurs twice on that path.

How to guarantee that the parse tree for a *sufficiently long* string has a *sufficiently long* root-to-leaf path?

Nonterminal repetition in binary parse tree paths

Let's use *binary* parse trees

Binary parse tree:

$2^{\text{max path length}} \geq \#\text{ leaves} \geq \text{word length}$

If

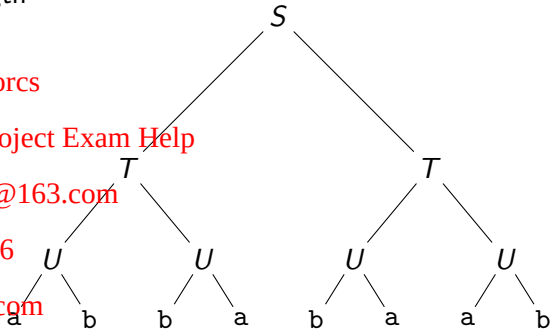
$$\text{word length} > 2^{\#\text{ nonterminals}}$$

then

$$2^{\text{max path length}} > 2^{\#\text{ nonterminals}},$$

$$\therefore \text{ max path length} > \#\text{ nonterminals}$$

and so there will be a repeated symbol
in a root-to-leaf path.



Nonterminal repetition in binary parse tree paths

Let's use *binary* parse trees, **from Chomsky Normal Form grammars!**

Binary **CNF** parse tree:

$2^{\text{max path length}-1} \geq \#\text{ leaves} \geq \text{word length}$

If

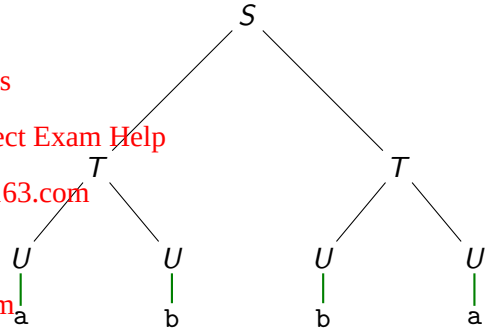
$$\text{word length} > 2^{\#\text{ nonterminals}-1}$$

then

$$2^{\text{max path length}-1} > 2^{\#\text{ nonterminals}-1},$$

$$\therefore \text{ max path length} > \#\text{ nonterminals}$$

and so there will be a repeated symbol
in a root-to-leaf path.



Pumping Lemma for CFLs

Let L be any context-free language that has a CFG in CNF with k non-terminal symbols. Then for every word $w \in L$ with $> 2^{k-1}$ letters,

there exist strings u, v, x, y, z ($\underbrace{vy \neq \varepsilon}_{v \text{ or } y \text{ } \underline{\text{not both}} \text{ } \varepsilon}$)

such that

- $w = uvxyz$
- $|vxy| \leq 2^k$, and
- for all $i \geq 0$, $uv^i xy^i z \in L$

i.e.,

$$uxz,\ uvxyz,\ uvvxyyz,\ \ldots,\ uv^n xy^n z,\ \ldots \in L.$$

Symbolically:

$$\forall w \in L\ :\ |w| > 2^{k-1}\ \Rightarrow\ \big(\exists u, v, x, y, z\ :\ (w = uvxyz) \wedge (vy \neq \varepsilon) \wedge (|vxy| \leq 2^k) \wedge\ (\forall i \geq 0 : uv^i xy^i z \in L)\big)$$

Pumping Lemma for CFLs

Proof. (outline)
Take any word $w \in L$ with > 2 symbols.

Let T be a parse tree for w, using the CNF CFG for L.

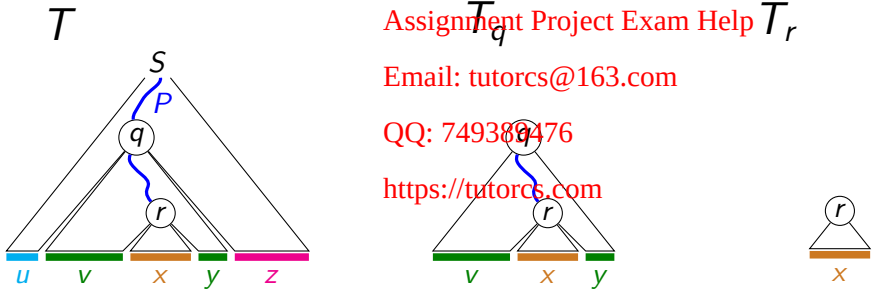
By our earlier Observations on root-to-leaf paths in CNF parse trees,
some root-to-leaf path P in T contains a repeated nonterminal symbol.

Among all pairs of nodes in P containing the same nonterminals,
choose the pair q, r, with q above r, so that q is as far as possible down the path P.
This ensures all nonterminals below q on P are distinct.
(Reason to be revealed later.)

Pumping Lemma for CFLs

Reading the letters of w from left to right from the leaves of the tree, define:

- u be the letters of w to the left of the subtree T_q rooted at node q.
- v be the letters at the leaves of T that are to the left of the subtree T_r rooted at r.
- x be the letters at the leaves of T_r.
- y be the letters of T_q that are to the right of the subtree T_r.
- z be the remaining letters of w, i.e., those to the right of T_q.



Pumping Lemma for CFLs

We have:

- $w = uvxyz$ by constructi
- Since q, r are distinct nodes of the path P with q above r, the tree T_r is a proper subtree of T_q.
- Furthermore, since the grammar is in CNF, q has two children, and only one of them is above r, so T_q has some leaves that do not belong to T_r.
- Therefore $vy \neq \varepsilon$.

Pumping Lemma for CFLs

- By our choice of q and r, [illegible]minals appearing below q on P are distinct.
- Since we have k nonterminals altogether, the subpath of P from q downwards has $\leq k+2$ nodes (being q, then at most k nonterminals, then the leaf).
- Therefore it has length $\leq k+1$. Therefore T_q has $\leq 2^k$ leaves. These leaves are the strings v, x, y, in order.
- Therefore $|vxy| \leq 2^k$.

Pumping Lemma for CFLs

- Replacement of T_q by T_r gives a parse tree for uxz.
- Replacement of T_r by T_q gives a parse tree for $uvvxyyz$.
- The new copy of T_q contains a copy of T_r.
  Replacing that copy of T_r by T_q gives a parse tree for $uvvvxyyyz$.
- Any parse tree with a copy of T_r can be enlarged, to be a parse tree of a longer string, by replacing T_r by T_q.
- These observations can be turned into a full formal proof by induction. □

A Tale of Two Pumping Lemmas

If a Finite Automaton with N states accepts a sufficiently long string, then the path taken by the string in the FA contains a repeated state.

This enables us to "pump" the string — by repeating one substring — to generate an infinite family of members of the language.

If a Context-Free Grammar in CNF with k nonterminals generates a sufficiently long string, then some root-to-leaf path in the parse tree contains a repeated nonterminal.

This enables us to "pump" the string — by repeating two substrings — to generate an infinite family of members of the language.

Pumping Lemma for CFLs: application

Consequence

Using the Pumping Lemma for CFLs we can show there are non-context-free languages.

Method

Assume L is context-free.

Then it has a Context-Free Grammar in CNF.

Let k be the number of nonterminal symbols in this CFG.

Choose a suitable word $w \in L$, of length $\geq 2^k$.

Show that, for any u, v, x, y, z such that $w = uvxyz$ and $vy \neq \varepsilon$ and $|vxy| \leq 2^k$ ...

... there exists $i \geq 0$ s.t. $uv^i xy^i z \notin L$.

Contradiction.

Compare quantifiers above with those in the Pumping Lemma for CFLs.

Non-context-free languages

$L := \{a^n b^n a^n : n \geq 0\} = \{\varepsilon, aba, aabbaa, aaabbbaaa, \ldots\}$.

Theorem.
L is not context-free.

Proof. (by contradiction)
Assume that L is context-free. Then it has a CFG.
Then there is a CFG in Chomsky Normal Form that generates $L \setminus \{\varepsilon\}$.
Let $k = \#$ nonterminals in this CNF CFG.
Take $N > 2^{k-1}/3$.
Choose $w = a^N b^N a^N$.
Consider any u, v, x, y, z such that

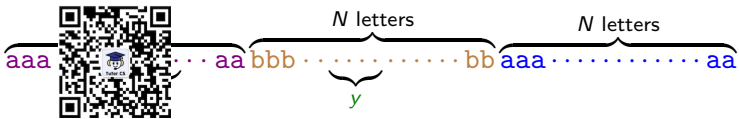
- $vy \neq \varepsilon$,
- $|vxy| \leq 2^k$, and
- $w = uvxyz$.

Think: are $uxz, uvxyz, uvvxyyz, \ldots, uv^i xy^i z, \ldots$ all in L?

Non-context-free languages

Case 1: v and y are each all a's, or all b's, or empty.

For example:



Then $uvvxyyz$ can no longer have three equal-length stretches of a's and b's, since:

- The two strings v, y must each lie entirely within one stretch, and there are three stretches, so one of these stretches is unaltered by pumping.
- But at least one of the other stretches *is lengthened*, because $vy \neq \varepsilon$.

So $uv^2xy^2z \notin L$.

Non-context-free languages

Case 2: Either v or y contains ab.

For example:

Then $uvvxyyz$ has two occurrences of ab.
This cannot happen for strings in L. So $uv^2xy^2z \notin L$.

Case 3: Either v or y contains ba.

For example:

Similar argument to Case 2.

In every possible case, we have found an i such that $uv^ixy^iz \notin L$.
This violates the conclusion of the Pumping Lemma for CFLs.
Contradiction. □

Revision

Reading: Sipser, pp. 125–129.