CS310 : Automata Theory 2019

Lecture 11: Applications of pumping lemma

Instructor: Ashutosh Gupta

IITB, India

Compile date: 2019-01-28

Contrapositive of pumping lemma

Recall

Theorem 11.1

Let L be a language. L is not regular if, for each n, there is a word $w \in L$ such that $|w| \geq n$ and for each breakup of w into three words $w = xyz$ such that

1. $y \neq \epsilon$ *and*
2. $|xy| \leq n$,

then there is a $k \geq 0$ such that $xy^k z \notin L$.

How to use the pumping lemma?

In the theorem, there are two exists quantifiers, namely w and k.

Proving non regularity boils down to the following two quantifier instantiations.

- Choose a word w for each n
- Find k for each breakup of the w

The instantiations are the creative steps!

Proving a language non regular

Consider language L

- For each n, we propose a word w in L longer than n
- We define parameterized word x and non-empty word y such that
  - $xyz = w$ for some z and
  - parameter space covers all x and y such that $|xy| \leq n$.
- For each split x and y, we choose a k such that $xy^kz \notin L$.

We have proven that L is not regular

Example 1: using pumping lemma

Example 11.1

Consider again $L_{eq} = \{0^n1^n | n \geq 0\}$.

- For each n we need a word. Let it be $w = 0^n1^n$.
- The first n characters of w are 0^n. The breaks x and y are to be from within 0^n.
- Let $x = 0^i$ and $y = 0^j$, where $i + j \leq n$ and $j \neq 0$.

$$w = \underbrace{0^i}_{x}\underbrace{0^j}_{y}\underbrace{0^{n-j-i}1^n}_{z}$$

i and *j* are encoding all possible breaks of 0^n.

- Now we choose $k = 0$ for each i and j. The corresponding word is

$$0^i(0^j)^0 0^{n-j-i}1^n = 0^{n-j}1^n.$$

- Clearly, $0^{n-j}1^n \notin L_{eq}$. Therefore, L_{eq} is not regular.

Example 2: using pumping lemma

Example 11.2

Consider again $L = \{w | w \text{ has equal number of } 0 \text{ and } 1\}$.

- For each n we need a word. Let it be $w = 0^n1^n$.
- The first n characters of w are 0^n. The breaks x and y are to be from within 0^n.
- Let $x = 0^i$ and $y = 0^j$, where $i + j \leq n$ and $j \neq 0$.

$$w = \underbrace{0^i}_{x}\underbrace{0^j}_{y}\underbrace{0^{n-j-i}1^n}_{z}$$

i and j are encoding all possible breaks of 0^n.

- Now we choose $k = 0$ for each i and j. The corresponding word is

$$0^i(0^j)^0 0^{n-j-i}1^n = 0^{n-j}1^n.$$

- Clearly, $0^{n-j}1^n \notin L$. Therefore, L is not regular.

Example 3: using pumping lemma

Let word w^R be reverse of word w.

Example 11.3

Consider $L_{rev} = \{ww^R | w \in \{0,1\}^*\}$.

- For each n, let $w = 0^n110^n$
- The first n characters of w are 0^n
- Let $x = 0^i$ and $y = 0^j$, where $i + j \leq n$ and $j \neq 0$.

$$w = \underbrace{0^i}_{x}\underbrace{0^j}_{y}\underbrace{0^{n-j-i}110^n}_{z}$$

- Let $k = 2$ for each i and j.

$$0^i(0^j)^2 0^{n-j-i}110^n = 0^{n+j}110^n \notin L_{rev}$$

Example 4: using pumping lemma

Example 11.4

Consider $L_{prime} = \{1^p | p \text{ is a prime number.}\}$.

- For each n, let $w = 1^p$ such that $p > n + 2$
- The first n characters of w are 1^n.
- Let $x = 1^i$ and $y = 1^j$, where $i + j \leq n$ and $j \neq 0$.

$$w = \underbrace{1^i}_{x}\underbrace{1^j}_{y}\underbrace{1^{p-j-i}}_{z}$$

- So, $|xy^kz| = (p - j) + kj$.

Choose k such that the term becomes composite?

- Let $k = p - j$. Therefore, $|xy^kz| = (p - j)(1 + j)$.
- Since both $(p - k) > 1_{(\text{why?})}$and $1 + j > 1_{(\text{why?})}$, $xy^kz \notin L_{prime}$.

Example 5: using pumping lemma

Example 11.5

Consider $L = \{1^{p^2} | p \geq 0\}$.

- For each n, let $w = 1^{n^2}$
- The first n characters of w are 1^n.
- Let $x = 1^i$ and $y = 1^j$, where $i + j \leq n$ and $j \neq 0$.

$$w = \underbrace{1^i}_{x}\underbrace{1^j}_{y}\underbrace{1^{n^2-j-i}}_{z}$$

- So, $|xy^kz| = (n^2 - j) + kj$.

Choose k such that the term is not a perfect square?

- Let $k = 2$. Therefore, $|xy^kz| = n^2 + j$.
- Since $0 < j \leq n_{\text{(why?)}}$, $n^2 < n^2 + j < n^2 + 2n + 1_{\text{(why?)}}$.
  Therefore, $xy^2z \notin L$.

Need for infinite memory

Feels like all non-regular languages needed to remember infinite memory.

Example 11.6

In $\{0^n1^n | n \geq 0\}$ *we need to remember the number of seen* $0s$ *and count the* $1s$ *to match.*

Finite number of states cannot count unboundedly increasing number.

More generalized pumping lemma

We have been looking for evidence of bad pumping in the prefixes of the words.

We can look for such evidence for any subword of length greater than n.

Theorem 11.2

Let L be a language. L is not regular if,

- *for each n,*
- *there are words u, and w such that $uw \in L$ and $|w| \geq n$*
- *for each breakup of w into three words $xyz = w$ such that $y \neq \epsilon$ and $|xy| \leq n$ then*
- *there is a $k \geq 0$ such that $uxy^kz \notin L$.*

In our earlier version of pumping lemma, $u = \epsilon$.

Converse does not hold!

Pumping lemma holds for the following language but is not regular.

$$L = \underbrace{\{ca^n b^n | n \geq 1\}}_{L_1} \cup \underbrace{\{c^n w | n \neq 1 \text{ and } w \in \{a, b\}^*\}}_{L_2}$$

Application of pumping lemma:

- $n = 1$
- Two cases
- **Case** take word $ca^j b^j \in L_1$
  - Let $x = \epsilon$, $y = c$, and $z = a^j b^j$.
  - For $k \neq 1$, $c^k a^j b^j \in L_2$, and for $k = 1$, $c^k a^j b^j \in L_1$
- **Case** take word $c^j w \in L_2$ for $j \neq 1$
  - ......

Exercise 11.1

Complete the above application of pumping lemma

End of Lecture 11