COMPSCI 501: Formal Language Theory

Lecture 6: Myhill-Nerode Theorem

Marius Minea
marius@cs.umass.edu

University of Massachusetts Amherst

4 February, 2019

Regular and Nonregular Languages

- We've defined regular languages as accepted by automata
- Then, equivalently, using language closure properties
  (regular expression: union, concatenation, star)
- A **necessary** condition for regular languages: Pumping Lemma
  prove by contradiction that a language is *not* regular
- A **necessary** and **sufficient** condition?

A nonregular language that can be pumped

Consider $\Sigma = \{a, b, c\}$ and
$L = \{ca^nb^n | n \geq 1\} \cup \{c^kw | k \neq 1, w \in \{a, b\}^*\}$

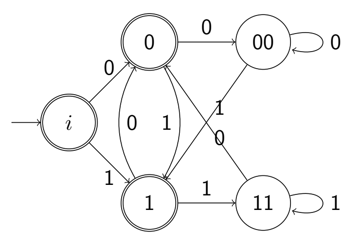
This is the disjoint union of two parts:
$L_1 = \{ca^nb^n | n \geq 1\}$ is not regular
$L_2 = \{c^kw | k \neq 1, w \in \{a, b\}^*\}$ is regular

Pumping L_1 (up or down with c) gives a string in L_2

L_2 is regular so it can be pumped:
with the first symbol, if it is not c
with cc, if it starts with cc

DFA: State = Prefix

Example: binary strings that do **not** end with same two symbols



State i reached by ε, 0 reached by 0 (also 10, 110, ...), etc.

If two strings reach the same state, no suffix will further distinguish them.

$\delta^*(q_0, u) = \delta^*(q_0, v) = q \Rightarrow \delta^*(q_0, uw) = \delta^*(q_0, vw) = \delta^*(q, w)$ for any w where δ^* is the transition function for strings.

L-distinguishable strings

Let $x, y \in \Sigma^*$ be any strings and L be any language.

We say that x and y are **distinguishable** by L if there exists a string z such that exactly one of the strings xz and yz is in L (the other one is not).

Otherwise, if for all $z \in \Sigma^*$, $xz \in L \Leftrightarrow yz \in L$, we say that x and y are **indistinguishable** by L, and write $x \equiv_L y$.

L-indistinguishability is an equivalence relation

because L-indistinguishability is defined as an equivalence

- Reflexive: clearly, $xw = xw$ for any w (same string)
- Symmetric: if $xw \in L \Leftrightarrow yw \in L$ then $yw \in L \Leftrightarrow xw \in L$
- Transitive: if $xw \in L \Leftrightarrow yw \in L$ and $yw \in L \Leftrightarrow zw \in L$ then $xw \in L \Leftrightarrow zw \in L$

What does L-(in)distinguishability tell us ?

Can check if a *particular* DFA is suitable for recognizing a language

Let L be any language and M be any DFA. If for two L-distinguishable strings u and v, we have $\delta^*(q_0, u) = \delta^*(q_0, v)$, then $L(M) \neq L$.

If u and v are distinguished by L, then there is some string w so uw is accepted and vw is not (or the reverse).

But $\delta^*(q_0, u) = \delta^*(q_0, v) \Rightarrow \delta^*(q_0, uw) = \delta^*(q_0, vw)$, and the latter state can not be both accepting and not accepting (q.e.d.).

Myhill - Nerode Theorem

Define the **index** of a language L the maximum number of strings so that any two are pairwise distinguishable by L.

Theorem: L is recognized by a DFA with k states iff it has index at most k.

- If L is recognized by a DFA with k states, L has index at most k
- If L has a finite index k, it is recognized by a DFA with k states (and this is the minimal DFA)

Corollary: If L has infinite index, it is not regular

Examples: languages with infinite index

$\{0^n 1^n \mid n \geq 0\}$

Choose set of strings $\{0^n \mid n \geq 0\}$

0^i distinguishable from 0^j $(i \neq j)$:
$0^i 1^i$ accepted, $0^j 1^i$ not accepted

L-equivalence is defined over *all strings* in Σ^*
For pumping lemma, we choose string *from language*.
For distinguishability, we choose *any family of strings*.

Example: Balanced Parentheses

$\Sigma = \{L, R\}$

Language of strings with equal number of L and R, no prefix has more R than L.

L^i distinguishable from L^j $(i \neq j)$:
$L^i R^i$ accepted, $L^j R^i$ not accepted

Proof of Myhill-Nerode (1)

- If L is recognized by a DFA with k states, L has index at most k

Proof: by contradiction.

Assume L has index greater than k, so at least $k+1$ strings are pairwise L-distinguishable.

Then by the pigeonhole principle, there are two strings x and y tht take the DFA to the same state: $\delta^*(q_0, x) = \delta^*(q_0, y)$.

Then, for any suffix, $\delta^*(q_0, xw) = \delta^*(q_0, yw)$, so both strings are either accepted or not
$\Rightarrow x$ and y are not distinguishable (contradiction)

Proof of Myhill-Nerode (2)

- If L has a finite index k, it is recognized by a DFA with k states

We construct the DFA M. Consider a set $\{s_1, s_2, \ldots s_k\}$ of L-distinguishable strings. We'll have one state q_i for each string s_i.

For any string s_i and $a \in \Sigma$, $s_i a$ must be L-equivalent to some s_j: $s_i a \equiv_L s_j$ (else we'd have one more equivalence class, index $> k$).

Choose $\delta(q_i, a) = q_j$.

Take as initial state the q_i with $s_i \equiv_L \varepsilon$.

Let $F = \{q_i \mid s_i \in L\}$ (the states for strings in L)

Are we done?
Need to prove that for all w, $\delta^*(q_0, w) = q_i$ such that $w \equiv_L s_i$
by induction over string length

Example: Prime Lengths

$\Sigma = \{1\}$, language: $\{1^p \mid pisprime\}$

Choose any two strings 1^i and 1^j, $i < j$, and a prime $p > i, j$.

For any suffix 1^k, lengths of $1^i 1^k$ and $1^j 1^k$ differ by $j - i$.

Choose sequence of strings with lengths
$p, p + (j - i), p + 2(j - i), \ldots p + p(j - i)$

Consecutive strings have length difference $j - i$, so are obtained from 1^i and 1^j with same suffix.

p is prime, but $p + p(j - i)$ is not (divisible by p).
Thus, there must be a consecutive pair (prime, not prime), and that pair is distinguishable.

Minimizing DFAs by Partition Refinement

Start by partitioning states in $(F, Q \setminus F)$ (accept or not)

If for all partitions X, all states $q, r \in X$ and all symbols $a \in \Sigma$, we have $\delta(q, a)$ and $\delta(r, a)$ in the same partition, stop.
(states in partition are not distinguishable)

Otherwise, refine partition X and repeat.

Example: binary strings, accept if divisible by 6