

ECE 664
Lecture 18
Avi Kak

PROPERTIES OF REGULAR LANGUAGES

In Lecture 17, we learned the following amazing fact: That despite the notion of choice that is embedded in an ndfa, every language that is accepted by some ndfa will also be accepted by some dfa. Speaking colloquially, we could state that both nondeterministic and deterministic automata have the same power when it comes to defining the set of acceptable languages.

- As we will now see, regular languages are closed under a large number of operations. But before we get into these properties, let's define what we mean by nonrestarting automata.
- A dfa is called nonrestarting if there is no pair (q, s) for which $S(q, s) = q_i$ (initial state).
- Any given dfa M can be converted into a nonrestarting dfa \tilde{M} such that $L(\tilde{M}) = L(M)$.

Proof:

M	:	Q	q_i	F	δ	A
\tilde{M}	:	\tilde{Q}	q_i	\tilde{F}	$\tilde{\delta}$	A

with $\tilde{Q} = Q \cup \{q_{n+1}\}$

$\tilde{\delta}(q, s) = \delta(q, s)$ if $s(q, s) \neq q_i$
 $= q_{n+1}$ otherwise

$\tilde{F} = F$ if $q_i \notin F$
 $= F \cup \{q_{n+1}\}$ otherwise

$\tilde{\delta}(q_{n+1}, s) = \tilde{\delta}(q_i, s)$

\tilde{M} does not transition into the initial state q_i

Closure Properties of Regular Languages:

① If L and \tilde{L} are regular, so is $L \cup \tilde{L}$

see text for why we can assume L and \tilde{L} have the same alphabet

② If $L \subseteq A^*$ is regular, so is $A^* - L$

Let M accept L . Construct \tilde{M} like M but with $\tilde{F} = Q - F$. \tilde{M} will accept $A^* - L$.

③ If L_1 and L_2 are regular, so is $L_1 \cap L_2$

$L_1 \cap L_2 = A^* - ((A^* - L_1) \cup (A^* - L_2))$

④ \emptyset is a regular language

⑤ $\{0\}$ is a regular language

accepted by any automaton whose set of acceptance states is empty

M	:	Q	q_i	F	δ	A
		$\{q_1, q_2\}$		$\{q_1\}$		$\{a\}$

$s(q_1, a) = q_2$
 $s(q_2, a) = q_2$

⑥ Let $u \in A^*$. Then $\{u\}$ is a regular language

Let $u = a_1 a_2 \dots a_n$
 Make an ndfa M with $Q = \{q_0, q_1, \dots, q_{n+1}\}$
 $s(q_i, a_i) = \{q_{i+1}\}$
 $s(q_i, a) = \emptyset$ for $a \in A - \{a_i\}$

⑦ EVERY FINITE SUBSET OF A^* CONSTITUTES A REGULAR LANGUAGE

Obviously, a very comforting thought with regard to the parsing of all finite languages.

Definition : Let $L_1 \subseteq A^*$ and $L_2 \subseteq A^*$
 We write $L_1 \cdot L_2 = L_1 L_2 = \{uv \mid u \in L_1, \text{ and } v \in L_2\}$

Definition : Let $L \subseteq A^*$, we write
 $L^* = \{u_1 u_2 \dots u_n \mid n \geq 0 \text{ and } u_1, u_2, \dots, u_n \in L\}$

The dot operator allows us to concatenate two words from two different languages. The star operator allows us to concatenate any number of words, including possibly zero, of the same language.

8) If L and \tilde{L} are regular languages, then $L \cdot \tilde{L}$ is also a regular language.

9) If L is a regular language, so is L^* .

Let M be a nonstarting dfa that accepts L .

$M: Q, q_1, F, \delta, A$
 Now construct an ndfa \tilde{M} as follows:

$\tilde{M}: Q, q_1, F \cup \{q_1\}, \delta, A$

with $\delta^*(q, s) = \begin{cases} \{\delta(q, s)\} & \text{if } s(q, s) \notin F \\ \{\delta(q, s)\} \cup \{q_1\} & \text{otherwise} \end{cases}$

dfa $M: Q, q_1, F, \delta, A$
 $\tilde{M}: \tilde{Q}, \tilde{q}_1, \tilde{F}, \tilde{\delta}, A$
 Now construct ndfa \tilde{M} :
 $\tilde{M}: \tilde{Q}, \tilde{q}_1, \tilde{F}, \tilde{\delta}, A$ $\tilde{Q} = Q \cup \tilde{Q}$
 with $\tilde{\delta}(q, s) = \begin{cases} \{\delta(q, s)\}, & q \in Q - F \\ \{\delta(q, s)\} \cup \{\tilde{\delta}(q, s)\} & \text{for } q \in F \\ \{\tilde{\delta}(q, s)\}, & q \in \tilde{Q} \end{cases}$

The discussion on the front side of this 'scroll' shows that every finite language is regular. [Therefore, it is trivial to conceive of parsers for such languages.]
 Now, the star operator (and also the dot operator) tells us that infinite languages can also be regular. The following theorem goes to the heart of the structure of infinite regular languages.

Kleene's Theorem : Every regular language can be obtained from finite languages by applying the three operations $\cup, \cdot, *$ a finite number of times.

Note that a single application of ' $*$ ' results in an infinite language

The proof of this theorem is based on the existence of $R_{i,j}^k$ subsets of A^* . Let M be a dfa on A with $Q = \{q_1, q_2, \dots, q_n\}$. We define

$$R_{i,j}^k = \left\{ x \in A^* \mid \delta^*(q_i, x) = q_j \text{ and } M \text{ passes through no state } q_l \text{ with } l > k \text{ as } M \text{ moves across } x \right\}$$

The $R_{i,j}^k$ admit the following recursive definition:

$$R_{i,j}^0 = \{a \in A \mid \delta(q_i, a) = q_j\}$$

$$R_{i,j}^{k+1} = R_{i,j}^k \cup \left[R_{i,k+1}^k \cdot \left(R_{k+1,k+1}^k \right)^* \cdot R_{k+1,j}^k \right]$$

Kleene's theorem follows from the observation that for any L accepted by a given M :

$$L(M) = \bigcup_{q_i \in F} R_{1,i}^n$$

n is the number of states in the dfa \rightarrow

Kleene's theorem is critical to understanding regular expressions.