INT201 Decision, Computation and Language

Lecture 4 – Regular Language

Dr Yushi Li

Xi'an Jiaotong-Liverpool University
西交利物浦大学

Definition

Previous: A language is regular if it is recognized by some **DFA**

Now: A language is regular if and only if some **NFA** recognizes it.

Some operations on languages: Union, Concatenation and Kleene star

Closed under operation

A collection S of objects is **closed** under operation f if applying f to members of S always returns an object still in S.

Regular languages are indeed closed under the regular operations (e.g. union, concatenation, star …)

Regular Languages Closed Under Union

The set of regular languages is closed under the union operation.

i.e. A and B are regular languages over the same alphabet Σ, then $A \cup B$ is also a regular language.

Proof:

- Since A and B are regular languages, there are finite automata $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ and $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ that accept A and B, respectively.
- In order to prove that $A \cup B$ is regular, we have to construct a finite automaton M that accepts $A \cup B$. In other words, M must have the property that for every string $w \in \Sigma^*$:

Regular Languages Closed Under Union

Proof

Given M_1 and M_2 that $A = L(M_1)$ and $B = L(M_2)$, we can define $M = (Q, \Sigma, \delta, q, F)$:

- $Q = Q_1 \times Q_2 = \{(q_1, q_2): q_1 \in Q_1 \text{ and } q_2 \in Q_2\}$
- Σ is same as the alphabet of A and B
- $q = (q_1, q_2)$
- $F = \{(q_1, q_2): q_1 \in Q_1 \text{ or } q_2 \in Q_2\}$
- $\delta : Q \times \Sigma \rightarrow Q$

  $\delta((q_1, q_2), a) = (\delta(q_1, a), \delta(q_2, a)), a \in \Sigma$

Regular Languages Closed Under Union

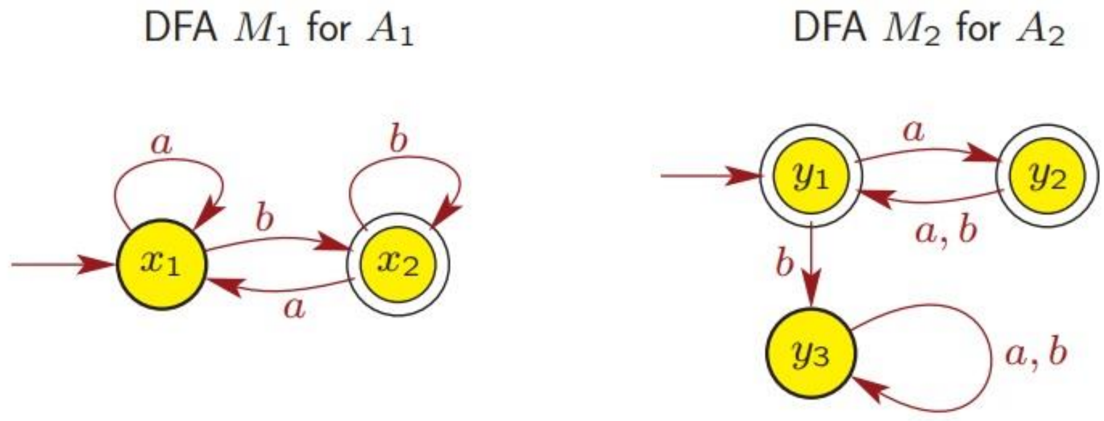
Proof

Example

Consider the following DFAs and languages over $\Sigma = \{a, b\}$:

DFA M_1 recognizes $A_1 = L(M_1)$

DFA M_2 recognizes $A_2 = L(M_2)$



DFA M for $A_1 \cup A_2$?

Example

Regular Languages Closed Under Union

How to prove this from the perspective of NFA?

Proof

Consider the following NFAs:

NFA $\mathrm{M}_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognizes $\mathrm{A}_1 = \mathrm{L}(\mathrm{M}_1)$

NFA $\mathrm{M}_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ recognizes $\mathrm{A}_2 = \mathrm{L}(\mathrm{M}_2)$

We assume that $Q_1 \cap Q_2 = \emptyset$

We will construct an NFA $\mathrm{M} = (Q, \Sigma, \delta, q, F)$

Regular Languages Closed Under Union

Proof

- $Q = \{q_0\} \cup Q_1 \cup Q_2$
- q_0 is the start state of M
- $F = F_1 \cup F_2$
- $\delta : Q \times \Sigma_\epsilon \rightarrow P(Q)$ is defined as: For any $r \in Q$ and for any $a \in \Sigma_\epsilon$

Regular Languages Closed Under Union

Proof

Regular Languages Closed Under Concatenation

The concatenation of A_1 and A_2 is defined as:

$$\mathrm{A}_1\,\mathrm{A}_2 = \{ww' : w \in \mathrm{A}_1 \text{ and } w' \in \mathrm{A}_2\}$$

Proof

Consider the following NFAs:

NFA $\mathrm{M}_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognizes $\mathrm{A}_1 = \mathrm{L}(\mathrm{M}_1)$

NFA $\mathrm{M}_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ recognizes $\mathrm{A}_2 = \mathrm{L}(\mathrm{M}_2)$

We will construct an NFA $\mathrm{M} = (Q, \Sigma, \delta, q, F)$ for $\mathrm{A}_1\,\mathrm{A}_2$

Regular Languages Closed Under Concatenation

Proof

- $Q = Q_1 \cup Q_2$
- M has the same start state as M_1 : q_1
- Set of accept states of M is same as M_2 : F_2
- $\delta : Q \times \Sigma_\epsilon \rightarrow P(Q)$ is defined as: For any $r \in Q$ and for any $a \in \Sigma_\epsilon$

Regular Languages Closed Under Concatenation

Proof

Regular Languages Closed Under Kleene star

The star of A is defined as:

$$\mathrm{A}^* = \{u_1 u_2 \ldots u_k : k \geq 0 \text{ and } u_i \in \mathbf{A} \text{ for all } i = 1, 2, \ldots, k\}$$

Proof

Consider the following NFA:

NFA $\mathrm{M}_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognizes $\mathrm{A} = \mathrm{L}(\mathrm{M}_1)$

We will construct an NFA $\mathrm{M} = (Q, \Sigma, \delta, q, F)$ for A^*

Regular Languages Closed Under Kleene star

Proof

- $Q = \{q_0\} \cup Q_1$
- q_0 is the start state of M
- $F = \{q_0\} \cup F_1$
- $\delta : Q \times \Sigma_\epsilon \rightarrow P(Q)$ is defined as: For any $r \in Q$ and for any $a \in \Sigma_\epsilon$

Regular Languages Closed Under Kleene star

Proof

Regular Languages Closed Under Complement and Interaction

The set of regular languages is closed under the complement and interaction operations:

- If A is a regular language over the alphabet Σ, then the complement:

$$\overline{\mathrm{A}} = \{w \in \Sigma^*: w \notin \mathrm{A}\}$$

  is also a regular language.

- If A_1 and A_2 are regular languages over the same alphabet Σ, then the interaction:

$$\mathrm{A}_1 \cap \mathrm{A}_2 = \{w \in \Sigma^*: w \in \mathrm{A}_1 \text{ and } w \in \mathrm{A}_2\}$$

  is also a regular language.

Regular Expressions

Regular expressions are means to describe certain languages.

Example

Consider the expression:

$$(0 \cup 1)01^*$$

The language described by this expression is the set of all binary strings satisfy:

- that start with either 0 or 1 (this is indicated by $(0 \cup 1)$),
- for which the second symbol is 0 (this is indicated by 0),
- that end with zero or more 1s (this is indicated by 1^*).

Example

The language $\{w : w \text{ contains exactly two } 0\text{s}\}$ is described by the expression:

$$1^* 01^* 01^*$$

The language $\{w : w \text{ contains at least two } 0\text{s}\}$ is described by the expression:

$$(0 \cup 1)^* 0 (0 \cup 1)^* 0 (0 \cup 1)^*$$

The language $\{w : 1011 \text{ is a substring of } w\}$ is described by the expression:

$$(0 \cup 1)^* 1011 (0 \cup 1)^*$$

Formal Definition of regular expressions

Let Σ be a non-empty alphabet.

1. ϵ is a regular expression.
2. $\emptyset$ is a regular expression.
3. For each $a \in \Sigma$, a is a regular expression.
4. If R_1 and R_2 are regular expressions, then $R_1 \cup R_2$ is a regular expression.
5. If R_1 and R_2 are regular expressions, then $R_1 R_2$ is a regular expression.
6. If R is a regular expression, then R^* is a regular expression.

Example

Given $(0 \cup 1)^* 101 (0 \cup 1)^*$, prove it is a regular expression (note: $\Sigma = \{0, 1\}$).

Formal Definition of regular expressions

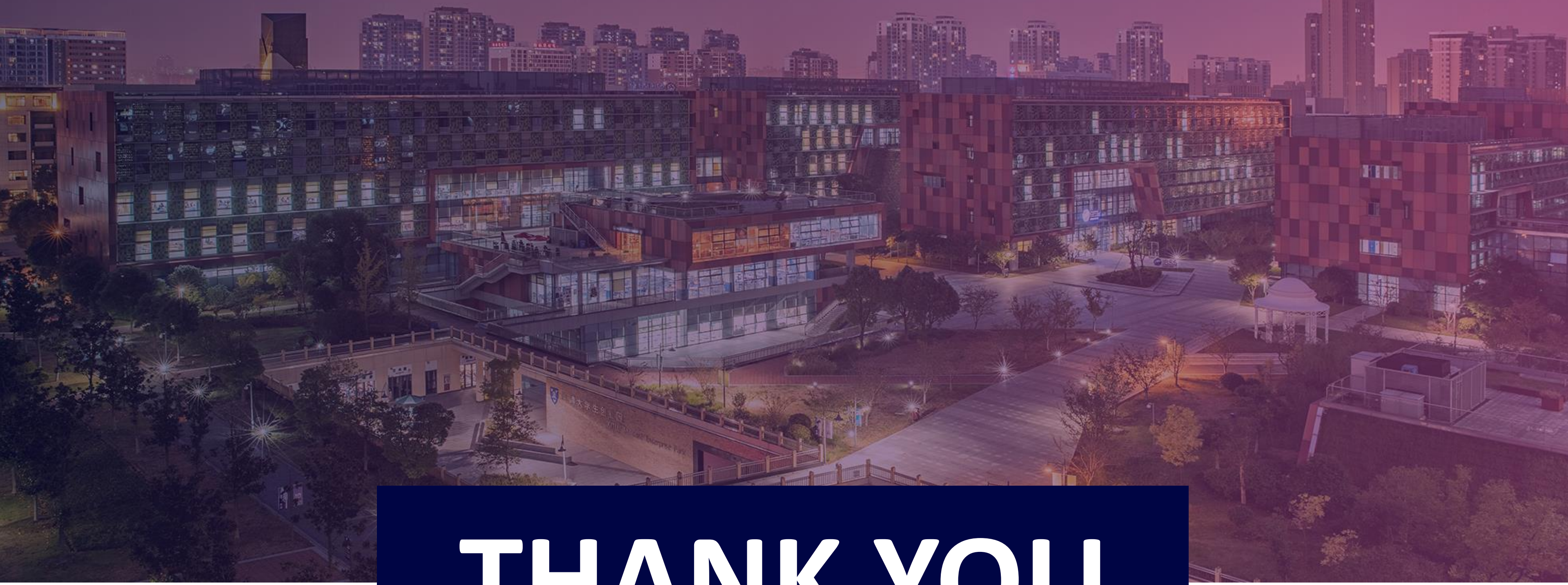
If R is a regular expression, then $\mathrm{L}(R)$ is the **language** generated (or described or defined) by R.

Let Σ be a non-empty alphabet.

1. The regular expression ϵ describes the language $\{\epsilon\}$.
2. The regular expression ∅ describes the language ∅.
3. For each $a \in \Sigma$, the regular expression a describes the language $\{a\}$.
4. Let R_1 and R_2 be regular expressions and let L_1 and L_2 be the languages described by them, respectively. The regular expression $R_1 \cup R_2$ describes the language $\mathrm{L}_1 \cup \mathrm{L}_2$.
5. Let R_1 and R_2 be regular expressions and let L_1 and L_2 be the languages described by them, respectively. The regular expression R_1R_2 describes the language $\mathrm{L}_1\mathrm{L}_2$.
6. Let R be a regular expression and let L be the language described by it. The regular expression R^* describes the language L^* .

Example

Given a regular expression $(0 \cup \epsilon)^{*}1$, it describes the language:



THANK YOU

Xi'an Jiaotong-Liverpool University
西交利物浦大学

XJTLU | SCHOOL OF FILM AND TV ARTS