INT201 Decision, Computation and Language

Lecture 4 – Regular Language

Dr Yushi Li

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

Definition

Previous: A language is regular if it is recognized by some **DFA** 如果一种语言被DFA识别，则是常规的

Now: A language is regular if and only if some **NFA** recognizes it. 当且仅当某些NFA识别出一种语言时，才是常规的

Some operations on languages: Union, Concatenation and Kleene star

Closed under operation

集合S是封闭的，如果对其使用操作f后仍然在S中

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

$a \in S,\ b \in S,\quad f(a,b) = c,\quad c \in S$

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

正则语言在常规操作下是封闭的

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. 如果A,B是正则语言,那么A∪B也是在相同字母集Σ上.

Proof:

A,B都是正则语言, 有限自动机 M_1和M_2 分别接受A和B

- 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 $M = (Q, \Sigma, \delta, q, F)$:

- $Q = Q_1 \times Q_2 = \{(q_1, q_2): q_1 \in Q_1 \text{ and } q_2 \in Q_2\}$ 要包含所有的组合可能
- Σ is same as the alphabet of A and B
- $q = (q_1, q_2)$

  $M_1: \to (q_1) \xrightarrow{a} \ldots$

  $M_2: \to (q_2) \xrightarrow{a} \ldots$
- $F = \{(q_1, q_2): q_1 \in \underset{F_1}{Q_1} \text{ or } q_2 \in \underset{F_2}{Q_2}\}$

  只要组合中包含原本M_1或M_2中的F就可以被标记为accept status
- $\delta : Q \times \Sigma \to Q$

  $\delta((q_1, q_2), a) \to (q_1 q_2) \xrightarrow{a} \ldots$

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

δ^* sequence of transition functions

Regular Languages Closed Under Union

DFA证明过程

Proof

- $\delta^*((q_1, q_2), w) = (\delta^*(q_1, w)\ ,\ \delta^*(q_2, w))$
- $\delta^*((q_1, q_2), w) \in F \Leftrightarrow \delta^*(q_1, w) \in F_1$ or $\delta^*(q_2, w) \in F_2$

并

- M accepts $w \Leftrightarrow \delta^*(q_1, w) \in F_1$ or $\delta^*(q_2, w) \in F_2$
- M accepts $w \Leftrightarrow$ M_1 accepts w or M_2 accepts w

Proved

Example

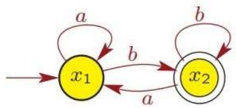
Consider the following DFAs and languages over $\Sigma = \{a, b\}$:

DFA M_1 recognizes $A_1 = L(M_1)$

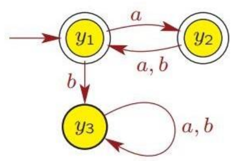
$M = ((q_1, q_2), (a, b), \delta, q, F)$

DFA M_2 recognizes $A_2 = L(M_2)$

DFA M_1 for A_1

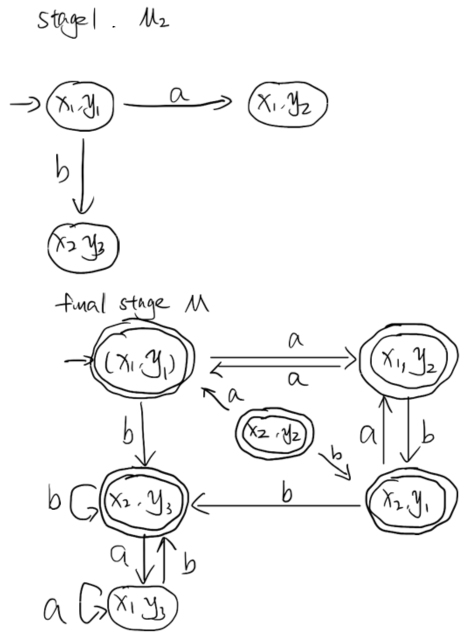


DFA M_2 for A_2



DFA M for $A_1 \cup A_2$?

Example



Example

Regular Languages Closed Under Union

如何从NFA的角度来证明

How to prove this from the perspective of NFA?

Proof

Consider the following NFAs:

NFA $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognizes $A_1 = L(M_1)$

NFA $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ recognizes $A_2 = L(M_2)$

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

Regular Languages Closed Under Union

NFA 证明过程

Proof

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

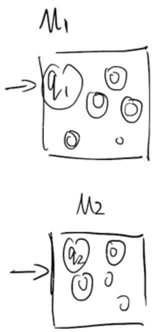
$$\delta(r,a) = \begin{cases} \delta_1(r,a) & \text{if } r \in Q_1, \\ \delta_2(r,a) & \text{if } r \in Q_2, \\ \{q_1, q_2\} & \text{if } r = q_0 \text{ and } a = \epsilon, \\ \emptyset & \text{if } r = q_0 \text{ and } a \neq \epsilon. \end{cases}$$

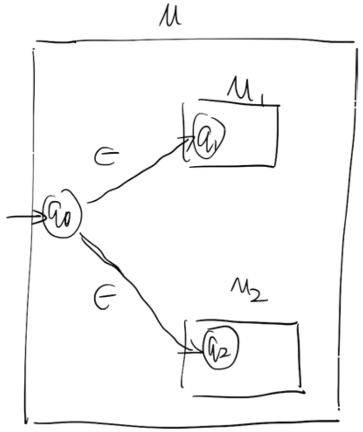
Basic idea   Only need to know how to transfer the function into graph

Regular Languages Closed Under Union

no need to memorize the internal concept.

Proof





closed under regular operation

Regular Languages Closed Under Concatenation ?

The concatenation of A_1 and A_2 is defined as: 串联

$$A_1 A_2 = \{ww' : w \in A_1 \text{ and } w' \in A_2\}$$

Proof

Consider the following NFAs:

NFA $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognizes $A_1 = L(M_1)$

NFA $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ recognizes $A_2 = L(M_2)$

We will construct an NFA $M = (Q, \Sigma, \delta, q, F)$ for $A_1 A_2$

Regular Languages Closed Under ~~Union~~ Concatenation

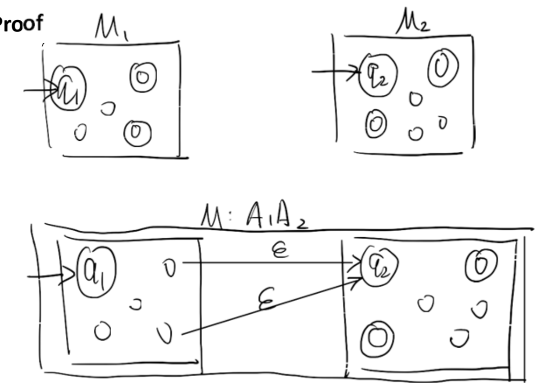
Proof

- $Q = Q_1 \cup Q_2$
- 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 $a \in \Sigma_\epsilon$

$$\delta(r,a) = \begin{cases} \delta_1(r,a) & \text{if } r \in Q_1 \text{ and } r \notin F_1, \\ \delta_1(r,a) & \text{if } r \in F_1 \text{ and } a \neq \epsilon, \\ \delta_1(r,a) \cup \{q_2\} & \text{if } r \in F_1 \text{ and } a = \epsilon, \\ \delta_2(r,a) & \text{if } r \in Q_2. \end{cases}$$

Regular Languages Closed Under ~~Union~~ Concatenation

Proof



Convert the acceptance states of M_1 => middle state

Regular Languages Closed Under Kleene star

The star of A is defined as: string

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

Proof

Consider the following NFA:

NFA $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ recognizes $A = L(M_1)$

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

Regular Languages Closed Under ~~Union~~ Kleene star

Proof

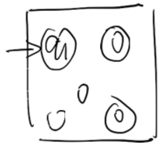
- $Q = \{q_0\} \cup Q_1$
- q_0 is the start state of M
- $F = \{q_0\} \cup F_1$
- $\delta : Q \times \Sigma_\epsilon \to P(Q)$ is defined as: For any $r \in Q$ and for any $a \in \Sigma_\epsilon$

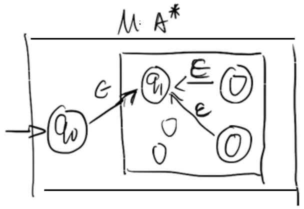
$$\delta(r,a) = \begin{cases} \delta_1(r,a) & \text{if } r \in Q_1 \text{ and } r \notin F_1, \\ \delta_1(r,a) & \text{if } r \in F_1 \text{ and } a \neq \epsilon, \\ \delta_1(r,a) \cup \{q_1\} & \text{if } r \in F_1 \text{ and } a = \epsilon, \\ \{q_1\} & \text{if } r = q_0 \text{ and } a = \epsilon, \\ \emptyset & \text{if } r = q_0 \text{ and } a \neq \epsilon. \end{cases}$$

kleene star

Regular Languages Closed Under ~~Union~~

Proof $M_1: A_1$





a^* : iteration

有可能会问

Regular Languages Closed Under Complement and Interaction

正则表达式在 complement 和 interaction 操作下是封闭的

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

如果A是基于 Σ 字母集的正则语言

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

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

is also a regular language.

如果 A_1 和 A_2 是同一字母集 Σ 上的正则语言

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

交集 $$A_1 \cap A_2 = \{w \in \Sigma^*: w \in A_1 \text{ and } w \in A_2\}$$

is also a regular language.

Regular Expressions

Regular expressions are means to describe certain languages. 正则表达式是描述特定语言的方式

Example

Consider the expression:

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

这是满足以下条件的所有二进制字符串集合

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 $(0 \cup 1)$), 以0，1开头
- for which the second symbol is 0 (this is indicated by 0), 第2个是0
- that end with zero or more 1s (this is indicated by 1^*). 以多个或零个1结尾

Example *可以有也可以无

正好包含2个0

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:

$$(0 \cup 1)^*101(0 \cup 1)^*$$

remember all description

Formal Definition of regular expressions

Let Σ be a non-empty alphabet. Σ 非空字母集

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 $0 \in \Sigma$ $1 \in \Sigma$

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

① $0, 1$ are regular expression

② $0 \cup 1$ is regular expression

③ $(0 \cup 1)^*$ is regular expression

④ $(0 \cup 1)^* 101$ is regular exprecsion

$(0 \cup 1)^* 101 (0 \cup 1)^*$ is regular expression

Formal Definition of regular expressions

如果R是一个正则表达式，那么 L(R) 是由R生成的（或描述的）语言

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. Σ非空字母表

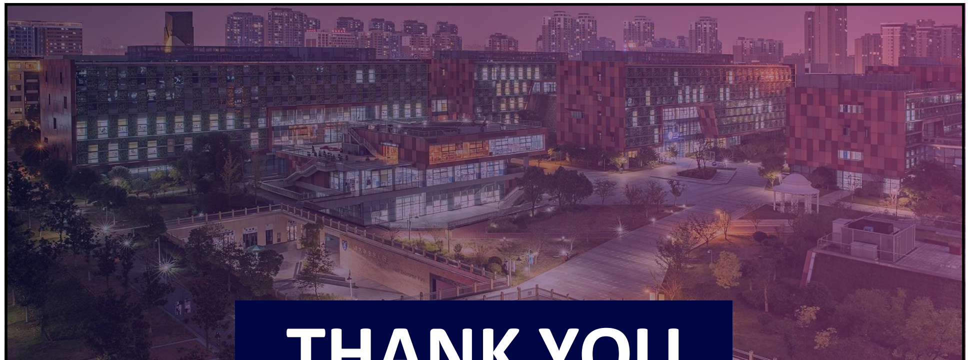
1. The regular expression ϵ describes the language $\{\epsilon\}$.
2. The regular expression $\emptyset$ describes the language $\emptyset$.
3. For each $a \in \Sigma$, the regular expression a describes the language $\{a\}$.
4. Let R_1 and R_2 be regular expressions and let L_1 and L_2 be the languages 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:

$$\{0, 01, 011, 0111, \ldots, \epsilon, 1, 11, 111, \ldots\}.$$

Observe that this language is also described by the regular expression $01^* \cup 1^*$.



THANK YOU

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

XJTLU | SCHOOL OF FILM AND TV ARTS