Two-Dimensional Languages

Jiří Zuzaňák

Brno University of Technology

December 19, 2007

Two-Dimensional Language

- Generalization of formal languages to two dimensions.
- Several models has been proposed in literature.
- Motivation - Pattern recognition, Image processing, Cellular automata studies, . . .

Definition (Two-Dimensional string)

Let Σ be a finite alphabet. A two-dimensional string (or a picture) over alphabet Σ is a two-dimensional rectangular array of elements from Σ.

Definition (Two-Dimensional language)

The set of all two-dimensional strings (or a pictures) from Σ is denoted by Σ^{**}. A two-dimensonal language over Σ is defined as subset of Σ^{**}.

Denotation

- *Given a picture* $p \in \Sigma^{**}$, *let* $\ell_1(p)$ *denote the number of rows, of* p, *and* $\ell_2(p)$, *denote the number of columns of* p.
- *The pair* $(\ell_1(p), \ell_2(p))$ *is called the size of the picture* p.
- *The empty picture has size* $(0, 0)$ *and it will be denoted by* λ.
- *The pictures of size* $(0, n)$ *or* $(n, 0)$ *where* $n > 0$ *are not defined.*
- *The set of all pictures over* Σ *of size* (m, n), *with* $m, n > 0$ *will be indicated by* $\Sigma^{m \times n}$.
- $p(i, j)$ *or equivalently,* $p_{i,j}$ *denotes symbol in* p, *with coordinates* (i, j), *where* $1 \leq i \leq \ell_1(p)$ *and* $1 \leq j \leq \ell_2(p)$.

Example (Two-Dimensional language)

Let $\Sigma = \{a\}$ be a alphabet. The set of pictures over Σ where every picture has 3 columns is two-dimensional language over Σ, which can be formally described as

$$L = \{p | p \in \Sigma^{**} \text{ and } \ell_2(p) = 3\}.$$

Example (Two-Dimensional language)

Let $\Sigma = \{0, 1\}$ be an alphabet. Language L of pictures over Σ whose first column is equal to the last one is formally defined as:

$$L = \{p | p(i, 1) = p(i, \ell_2(p)), i = 1, \ldots, \ell_1(p)\}$$

Definition (Sub-Picture)

Let p be a picture of size (m, n). A block (or a sub-picture) of p is a picture p' that is a sub-array of p. That is, if (m', n') is size of p', then $m' \leq m$ and $n' \leq n$ and there exist integers h, k $(h \leq m - m', k \leq n - n')$ such that $p'(i, j) = p(i + h, j + k)$ for all $0 \leq i \leq m'$ and $0 \leq j \leq n'$.

Definition (Projection of a picture)

Let $p \in \Gamma^{**}$ be a picture. The projection by mapping π of picture p is the picture $p' \in \Sigma^{**}$ such that $p'(i, j) = \pi(p(i, j))$, for all $1 \leq i \leq \ell_1(p)$, $1 \leq j \leq \ell_2(p)$.

Definition (Projection of a language)

Let $L \subseteq \Gamma^{**}$ be a picture language. The projection by mapping π of L is the language $L' = \{p' | p' = \pi(p) \forall p \in L\} \subseteq \Sigma^{**}$.

Definition (Concatenation of pictures)

The column concatenation of p and q (denoted by $p \obar q$) is a partial operation, defined only if $m = m'$ and it is given by:

$$p \obar q = \begin{array}{|ccc|ccc|} \hline p_{11} & \cdots & p_{1n} & q_{11} & \cdots & q_{1n'} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ p_{m1} & \cdots & p_{mn} & q_{m'1} & \cdots & q_{m'n'} \\ \hline \end{array}$$

Similarly, the row concatenation of p and q (denoted by $p \ominus q$) is a partial operation, defined only if $n = n'$, and it is given by:

$$p \ominus q = \begin{array}{|ccc|} \hline p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ p_{m1} & \cdots & p_{mn} \\ \hline q_{11} & \cdots & q_{1n'} \\ \vdots & \ddots & \vdots \\ q_{m'1} & \cdots & q_{m'n'} \\ \hline \end{array}$$

Definition (Concatenation of languages)

Let L_1,L_2 be two-dimensional languages over an alphabet Σ, the column concatenation of L_1 and L_2 (denoted by $L_1 \oplus L_2$) is defined by

$$L_1 \oplus L_2 = \{p \oplus q | p \in L_1 \text{ and } q \in L_2\}$$

Similarly, the row concatenation of L_1 and L_2 (denoted by $L_1 \ominus L_2$) is defined by

$$L_1 \ominus L_2 = \{p \ominus q | p \in L_1 \text{ and } q \in L_2\}$$

Definition (Column concatenation closure)

Let L be a picture language. The column closure of L (denoted by $L^{*\varobar}$) is defined as

$$L^{*\varobar} = \bigcup_{i \geq 0} L^{i\varobar}$$

where $L^{0\varobar} = \lambda, L^{1\varobar} = L,\ L^{n\varobar} = L \varobar L^{(n-1)\varobar}$.

Definition (Row concatenation closure)

Similarly, the row closure of L (denoted by $L^{*\ominus}$) is defined as

$$L^{*\ominus} = \bigcup_{i \geq 0} L^{i\ominus}$$

where $L^{0\ominus} = \lambda, L^{1\ominus} = L,\ L^{n\ominus} = L \ominus L^{(n-1)\ominus}$.

Definition (Rotation)

Let p be a picture. The (clockwise) rotation of p, indicated as p^R, is defined as

$$p^R = \begin{array}{|ccc|} \hline p_{m1} & \cdots & p_{11} \\ \vdots & \ddots & \vdots \\ p_{mn} & \cdots & p_{1n} \\ \hline \end{array}$$

Definition (Row-Column combination)

Let Σ be a finite alphabet and let $S_1, S_2 \subseteq \Sigma^*$ be two string languages over Σ. The row-column combination over S_1 and S_2 is two-dimensional language $L = S_1 \oplus S_2 \subseteq \Sigma^{**}$ such that, a picture $p \in \Sigma^{**}$ belongs to L if and only if the strings corresponding to the rows and to the columns of p belong to S_1 and to S_2, respectively.

Regular expressions

First natural approach is to define picture languages by means of regular expressions.

Definition

A regular expression (RE) over an alphabet Σ is defined as follows:

1. 0 and every letter $a \in \Sigma$ are regular expressions.
2. If α and β are regular expressions, then $(\alpha) \cup (\beta)$, $(\alpha) \cap (\beta)$, ${}^{c}(\alpha)$, $(\alpha) \oplus (\beta)$, $(\alpha) \ominus (\beta)$, $(\alpha)^{*\oplus}$, $(\alpha)^{*\ominus}$ are regular expressions.

Definition

A two-dimensional language $L \subseteq \Sigma^{**}$ is regular if it is denoted by a regular expression over Σ.

Example

Let $\Sigma = \{a, b\}$. The regular expression

$$(((a \ominus b)^{*\ominus}) \oplus ((b \ominus a)^{*\ominus}))^{*\oplus}$$

denotes language consisting of all "chesboards" with even side-length.

Denotation

- *The regular expressions that not contain complement operation are called complementation-free regular expressions (CFRE).*
- *Similarly, the regular expressions that not contain closure operations are called star-free regular expressions (SFRE).*

Four-way automata

M. Blum, C. Hewitt

Definition

A non-deterministic (deterministic) four-way automata, 4NFA (4DFA), is a 7-tuple $\mathcal{A} = (\Sigma, Q, \Delta, q_0, q_a, q_r, \delta)$ where:

- Σ is the input alphabet
- Q is finite set of states
- $\Delta = R, L, U, D$ is the set of directions.
- $q_0 \in Q$ is the initial state
- $q_a, q_r \in Q$ are the accepting and the rejecting states
- $\delta : Q \backslash \{q_a, q_r\} \times \Sigma \rightarrow 2^{Q \times \Delta}$ $(\delta : Q \backslash \{q_a, q_r\} \times \Sigma \rightarrow Q \times \Delta)$ is the transition function

Two-dimensional on-line tesselation automata

K. Inoue, A. Nakamura

Definition

A non-deterministic (deterministic) two-dimensional online tesselation automata, referred as 2OTA (2-DOTA), is defined as $\mathcal{A} = (\Sigma, Q, q_0, F, \delta)$ where:

- Σ is the input alphabet
- Q is the finite set of states
- $I \subseteq Q$ ($I = \{i\} \subseteq Q$) is the set of initial states
- $F \subseteq Q$ is the set of final states
- $\delta : Q \times Q \times \Sigma \to 2^Q$ ($\delta : Q \times Q \times \Sigma \to Q$) is the transition function.

- Run of $\mathcal{A}$ on p associate a state (from Q) to each position of picture p.
- All Positions of the first row and first column of $\widehat{p}$ are initialized to state Q_0.
- Each state at position (i, j) is given by a transition function δ and depends on the states at $(i-1, j)$ and $(i, j-1)$ and input symbol $p(i, j)$.
- A 2OTA $\mathcal{A}$ recognizes a picture p if there exist a run of $\mathcal{A}$ on p such that a state at position $(\ell_1(p), \ell_2(p))$ is a final state.

Two-dimensional right-linear grammar

Definition

A two-dimensional right-linear grammar (2RLG) is defined by a 7-tuple $G = (V_h, V_v, \Sigma_I, \Sigma, S, R_h, R_v)$, where:

- V_h is a finite set of horizontal variables
- V_v is a finite se of vertical variables
- $\Sigma_I \subseteq V_v$ is a finite set of intermediates
- Σ is a finite set of terminals
- $S \in V_h$ is a starting symbol
- R_h is a finite set of horizontal rules of the form $V \rightarrow AV'$ or $V \rightarrow A$, where $V, V' \in V_h$ and $A \in \Sigma_I$
- R_v is a finite set of vertical rules of the form $W \rightarrow aW'$ or $W \rightarrow a$, where $W, W' \in V_v$ and $a \in \Sigma$.

- The string grammar $G_h = (V_h, \Sigma_I, S, R_h)$ generates a string language $H(G)$ over the intermediate alphabet Σ_I.
- The string in $H(G)$ defines first row of generated picture.
- Each intermediate symbol is threated as a start symbol of vertical grammar $G_v = (V_v, \Sigma, \Sigma_I, R_v)$.
- The vertical generation of the columns is done in parallel by applying the rules in R_v.

Tiling systems

Denotation

Given a picture p of size (m, n)*, let* $h \leq m, k \leq n$*: we denote by* $B_{h,k}(p)$ *the set of all sub-pictures of p of size* (h, k)*.*

Definition (Local two-dimensional language)

Let Γ be a finite alphabet. A two-dimensional language $L \subseteq \Gamma^{**}$ is local if there exist finite set Θ of tiles over the alphabet $\Gamma \cup \{\#\}$ such that $L = \{p \in \Gamma^{**} | B_{2,2}(\widehat{p}) \subseteq \Theta\}$.

Definition (Tiling system)

A tiling system (TS) is a 4-tuple $\mathcal{T} = (\Sigma, \Gamma, \Theta, \pi)$, where Σ and Γ are two finite alphabets, Θ is finite set of tiles over the alphabet $\Gamma \cup \{\#\}$ and $\pi : \Gamma \to \Sigma$ is a projection.

- The TS $\mathcal{T}$ defines language L over alphabet Σ as follows: $L = \pi(L')$ where $L' = L(\Theta)$ is the local two-dimensional language over Γ.
- We write $L = L(\mathcal{T})$, and we say that L is recognized by $\mathcal{T}$.
- We will refer to $L' \subseteq \Gamma^{**}$ as an underlaying local language for L, while we will call Γ local alphabet.

Equivalences

- $\mathcal{L}(SFRE) \subseteq \mathcal{L}(RE)$
- $\mathcal{L}(CFRE) \subseteq \mathcal{L}(RE)$
- $\mathcal{L}(4DFA) \subset \mathcal{L}(4NFA)$
- $\mathcal{L}(2DOTA) \subset \mathcal{L}(2OTA)$
- $\mathcal{L}(4NFA) \subset \mathcal{L}(2OTA)$
- $\mathcal{L}(2RLG) \subset \mathcal{L}(4DFA)$
- $\mathcal{L}(TS) = \mathcal{L}(DS)$
- $\mathcal{L}(2OTA) = \mathcal{L}(TS)$
- $\mathcal{L}(TS) = \mathcal{L}(EMSO)$
- $\mathcal{L}(TS) = \mathcal{L}(PCFRE)$

Bibliography

G. Rozenberg, A. Salomaa (Eds.):
Chapter 4. in Volume 3 of Handbook of formal languages.
Berlin; New York: Springer, c1997.