فصل اول

مقدمه‌ای بر نظریه‌ی محاسبات (۲)

An Introduction to the Theory of Computation (2)

کاظم فولادی
kazim@fouladi.ir
دانشکده‌ی مهندسی برق و کامپیوتر
دانشگاه تهران

Languages

Languages

A language is a set of strings

String: A sequence of letters

Examples: "cat", "dog", "house", ...

Defined over an alphabet:

$$\Sigma = \{a, b, c, \ldots, z\}$$

Alphabets and Strings

We will use small alphabets: $\Sigma = \{a, b\}$

Strings

a

ab $\qquad u = ab$

$abba$ $\qquad v = bbbaaa$

$baba$ $\qquad w = abba$

$aaabbbaabab$

String Operations

$$w = a_1 a_2 \cdots a_n \qquad abba$$

$$v = b_1 b_2 \cdots b_m \qquad bbbaaa$$

Concatenation

$$wv = a_1 a_2 \cdots a_n b_1 b_2 \cdots b_m \qquad abbabbbaaa$$

$$w = a_1 a_2 \cdots a_n$$

$$ababaaabbb$$

Reverse

$$w^R = a_n \cdots a_2 a_1$$

$$bbbaaababa$$

String Length

$$w = a_1 a_2 \cdots a_n$$

Length: $|w| = n$

Examples:

$$|abba| = 4$$

$$|aa| = 2$$

$$|a| = 1$$

Recursive Definition of Length

For any letter: $|a| = 1$

For any string wa : $|wa| = |w| + 1$

Example:
$$\begin{aligned} |abba| &= |abb| + 1 \\ &= |ab| + 1 + 1 \\ &= |a| + 1 + 1 + 1 \\ &= 1 + 1 + 1 + 1 \\ &= 4 \end{aligned}$$

Length of Concatenation

$$|uv| = |u| + |v|$$

Example: $u = aab, \quad |u| = 3$

$v = abaab, \quad |v| = 5$

$$|uv| = |aababaab| = 8$$

$$|uv| = |u| + |v| = 3 + 5 = 8$$

Proof of Concatenation Length

Claim: $|uv| = |u| + |v|$

Proof: By induction on the length $|v|$

Induction basis: $|v| = 1$

From definition of length:

$$|uv| = |u| + 1 = |u| + |v|$$

Inductive hypothesis: $|uv| = |u| + |v|$

for $|v| = 1, 2, \ldots, n$

Inductive step: we will prove $|uv| = |u| + |v|$

for $|v| = n + 1$

Inductive Step

Write $v = wa$, where $|w| = n, \quad |a| = 1$

From definition of length: $|uv| = |uwa| = |uw| + 1$

$$|wa| = |w| + 1$$

From inductive hypothesis: $|uw| = |u| + |w|$

Thus: $|uv| = |u| + |w| + 1 = |u| + |wa| = |u| + |v|$

Empty String

A string with no letters: λ

Observations: $|\lambda| = 0$

$$\lambda w = w\lambda = w$$

$$\lambda abba = abba\lambda = abba$$

Substring

Substring of string:

a subsequence of consecutive characters

String	Substring
$\underline{ab}bab$	ab
$\underline{abba}b$	$abba$
$ab\underline{b}ab$	b
$a\underline{bbab}$	$bbab$

Prefix and Suffix

$abbab$

Prefixes	Suffixes
λ	$abbab$
a	$bbab$
ab	bab
abb	ab
$abba$	b
$abbab$	λ

$$w = uv$$

prefix

suffix

Another Operation

$$w^n = \underbrace{ww\cdots w}_{n}$$

Example: $(abba)^2 = abbaabba$

Definition: $w^0 = \lambda$

$$(abba)^0 = \lambda$$

The * Operation

Σ^* : the set of all possible strings from alphabet Σ

$$\Sigma = \{a, b\}$$

$$\Sigma^* = \{\lambda, a, b, aa, ab, ba, bb, aaa, aab, \ldots\}$$

The + Operation

Σ^+ : the set of all possible strings from alphabet Σ except λ

$$\Sigma = \{a, b\}$$

$$\Sigma^* = \{\lambda, a, b, aa, ab, ba, bb, aaa, aab, \ldots\}$$

$$\Sigma^+ = \Sigma^* - \lambda$$

$$\Sigma^+ = \{a, b, aa, ab, ba, bb, aaa, aab, \ldots\}$$

Language

A language is any subset of Σ^*

Example: $\Sigma = \{a, b\}$

$$\Sigma^* = \{\lambda, a, b, aa, ab, ba, bb, aaa, \ldots\}$$

Languages: $\{\lambda\}$

$$\{a, aa, aab\}$$

$$\{\lambda, abba, baba, aa, ab, aaaaaa\}$$

Another Example

An infinite language $L = \{a^n b^n : n \geq 0\}$

$$\left.\begin{array}{l} \lambda \\ ab \\ aabb \\ aaaaabbbbb \end{array}\right\} \in L \qquad abb \notin L$$

Operations on Languages

The usual set operations

$$\{a, ab, aaaa\} \cup \{bb, ab\} = \{a, ab, bb, aaaa\}$$

$$\{a, ab, aaaa\} \cap \{bb, ab\} = \{ab\}$$

$$\{a, ab, aaaa\} - \{bb, ab\} = \{a, aaaa\}$$

Complement: $\overline{L} = \Sigma^* - L$

$$\overline{\{a, ba\}} = \{\lambda, b, aa, ab, bb, aaa, \ldots\}$$

Reverse

Definition: $L^R = \{w^R : w \in L\}$

Examples: $\{ab, aab, baba\}^R = \{ba, baa, abab\}$

$$L = \{a^n b^n : n \geq 0\}$$

$$L^R = \{b^n a^n : n \geq 0\}$$

Concatenation

Definition: $L_1 L_2 = \{xy : x \in L_1, y \in L_2\}$

Example: $\{a, ab, ba\}\{b, aa\}$

$= \{ab, aaa, abb, abaa, bab, baaa\}$

Another Operation

Definition: $$L^n = \underbrace{LL \cdots L}_{n}$$

$$\{a,b\}^3 = \{a,b\}\{a,b\}\{a,b\} =$$
$$\{aaa,aab,aba,abb,baa,bab,bba,bbb\}$$

Special case: $L^0 = \{\lambda\}$

$$\{a,bba,aaa\}^0 = \{\lambda\}$$

More Examples

$$L = \{a^n b^n : n \geq 0\}$$

$$L^2 = \{a^n b^n a^m b^m : n, m \geq 0\}$$

$$aabbaaabbb \in L^2$$

Star-Closure (Kleene *)

Definition: $L^* = L^0 \bigcup L^1 \bigcup L^2 \cdots$

Example:

$$\{a,bb\}^* = \left\{ \begin{array}{l} \lambda, \\ a,bb, \\ aa,abb,bba,bbbb, \\ aaa,aabb,abba,abbbb,\ldots \end{array} \right\}$$

Positive Closure

Definition: $$L^+ = L^1 \cup L^2 \cup \cdots$$
$$= L^* - \{\lambda\}$$

$$\{a, bb\}^+ = \left\{\begin{matrix} a, bb, \\ aa, abb, bba, bbbb, \\ aaa, aabb, abba, abbbb, \ldots \end{matrix}\right\}$$

Grammars

Grammars

Grammars express languages

Example: the English language

$$\langle sentence \rangle \rightarrow \langle noun\_phrase \rangle \; \langle predicate \rangle$$

$$\langle noun\_phrase \rangle \rightarrow \langle article \rangle \; \langle noun \rangle$$

$$\langle predicate \rangle \rightarrow \langle verb \rangle$$

$$\langle article \rangle \rightarrow a$$

$$\langle article \rangle \rightarrow the$$

$$\langle noun \rangle \rightarrow boy$$

$$\langle noun \rangle \rightarrow dog$$

$$\langle verb \rangle \rightarrow runs$$

$$\langle verb \rangle \rightarrow walks$$

A derivation of "the boy walks":

$$\langle sentence \rangle \Rightarrow \langle noun\_phrase \rangle \ \langle predicate \rangle$$

$$\Rightarrow \langle noun\_phrase \rangle \ \langle verb \rangle$$

$$\Rightarrow \langle article \rangle \ \langle noun \rangle \ \langle verb \rangle$$

$$\Rightarrow the \ \ \langle noun \rangle \ \langle verb \rangle$$

$$\Rightarrow the \ \ boy \ \ \langle verb \rangle$$

$$\Rightarrow the \ \ boy \ \ walks$$

A derivation of "a dog runs":

$$\begin{aligned}\langle sentence \rangle &\Rightarrow \langle noun\_phrase \rangle \ \langle predicate \rangle \\ &\Rightarrow \langle noun\_phrase \rangle \ \langle verb \rangle \\ &\Rightarrow \langle article \rangle \ \langle noun \rangle \ \langle verb \rangle \\ &\Rightarrow a \ \ \langle noun \rangle \ \langle verb \rangle \\ &\Rightarrow a \ \ dog \ \ \langle verb \rangle \\ &\Rightarrow a \ \ dog \ \ runs\end{aligned}$$

Language of the grammar:

L = { "a boy runs",
"a boy walks",
"the boy runs",
"the boy walks",
"a dog runs",
"a dog walks",
"the dog runs",
"the dog walks" }

Notation

$$\langle noun \rangle \rightarrow boy$$

$$\langle noun \rangle \rightarrow dog$$

Variable
or
Non-terminal

Production
rule

Terminal

Another Example

Grammar: $S \to aSb$

$S \to \lambda$

Derivation of sentence ab:

$$S \Rightarrow aSb \Rightarrow ab$$

$S \to aSb$ $\qquad$ $S \to \lambda$

Grammar: $S \rightarrow aSb$

$S \rightarrow \lambda$

Derivation of sentence $aabb$:

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aabb$$

$S \rightarrow aSb$ $\qquad$ $S \rightarrow \lambda$

Other derivations:

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb \Rightarrow aaabbb$$

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb$$
$$\Rightarrow aaaaSbbbb \Rightarrow aaaabbbb$$

Language of the grammar

$$S \to aSb$$

$$S \to \lambda$$

$$L = \{a^n b^n : n \geq 0\}$$

More Notation

Grammar $\quad G = (V, T, S, P)$

$V:$ Set of variables

$T:$ Set of terminal symbols

$S:$ Start variable

$P:$ Set of Production rules

Example

Grammar G :
$$S \to aSb$$
$$S \to \lambda$$

$$G = (V, T, S, P)$$

$$V = \{S\} \qquad T = \{a, b\}$$

$$P = \{S \to aSb,\ S \to \lambda\}$$

More Notation

Sentential Form:

A sentence that contains variables and terminals

Example:

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb \Rightarrow aaabbb$$

Sentential Forms

sentence

We write: $S \overset{*}{\Rightarrow} aaabbb$

Instead of:

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb \Rightarrow aaabbb$$

In general we write: $w_1 \overset{*}{\Rightarrow} w_n$

If: $w_1 \Rightarrow w_2 \Rightarrow w_3 \Rightarrow \cdots \Rightarrow w_n$

By default: $w \overset{*}{\Rightarrow} w$

Example

Grammar

$$S \rightarrow aSb$$

$$S \rightarrow \lambda$$

Derivations

$$S \overset{*}{\Rightarrow} \lambda$$

$$S \overset{*}{\Rightarrow} ab$$

$$S \overset{*}{\Rightarrow} aabb$$

$$S \overset{*}{\Rightarrow} aaabbb$$

Example

Grammar

$$S \to aSb$$

$$S \to \lambda$$

Derivations

$$S \overset{*}{\Rightarrow} aaSbb$$

$$aaSbb \overset{*}{\Rightarrow} aaaaaSbbbbb$$

Another Grammar Example

Grammar G :
$$S \to Ab$$
$$A \to aAb$$
$$A \to \lambda$$

Derivations:

$$S \Rightarrow Ab \Rightarrow b$$

$$S \Rightarrow Ab \Rightarrow aAbb \Rightarrow abb$$

$$S \Rightarrow aAbb \Rightarrow aaAbbb \Rightarrow aabbb$$

More Derivations

$$S \Rightarrow Ab \Rightarrow aAbb \Rightarrow aaAbbb \Rightarrow aaaAbbbb$$

$$\Rightarrow aaaaAbbbbb \Rightarrow aaaabbbbb$$

$$S \overset{*}{\Rightarrow} aaaabbbbb$$

$$S \overset{*}{\Rightarrow} aaaaaabbbbbbb$$

$$S \overset{*}{\Rightarrow} a^n b^n b$$

Language of a Grammar

For a grammar G
with start variable S :

$$L(G) = \{w : \quad S \overset{*}{\Rightarrow} w\}$$

String of terminals

Example

For grammar G :
$$S \to Ab$$
$$A \to aAb$$
$$A \to \lambda$$

$$L(G) = \{a^n b^n b : \quad n \geq 0\}$$

Since: $S \overset{*}{\Rightarrow} a^n b^n b$

A Convenient Notation

$$A \rightarrow aAb$$
$$A \rightarrow \lambda$$

$$\Longrightarrow \quad A \rightarrow aAb \mid \lambda$$

$$\langle article \rangle \rightarrow a$$
$$\langle article \rangle \rightarrow the$$

$$\Longrightarrow \quad \langle article \rangle \rightarrow a \mid the$$