Languages

Geoffrey Matthews

Department of Computer Science
Western Washington University

November 2, 2015

Readings

- http://web.science.mq.edu.au/~chris/notes/second_langmach.html
- http://en.wikipedia.org/wiki/Formal_language

Strings

- A finite set A of symbols is given as the **alphabet**
- A **string** or **word** or **sentence** is a finite sequence of symbols from the alphabet.
- The **length** of a string is denoted $|s|$
- ϵ denotes the empty string. $|\epsilon| = 0$

- ϵ, a, $abbca$, and $bccb$ are strings over the alphabet $\{a, b, c\}$
- ϵ, 110101 and 0011 are strings over the alphabet $\{0, 1\}$
- ϵ, "the black cat" and "cat cat the the" are strings over the alphabet {black, cat, the}

Concatenation

- The concatenation of two strings is the string obtained by placing them next to each other.
- The concatenation of aaa and $bccb$ is $aaabccb$
- The concatenation of a string s with itself n times is denoted s^n
- $(ab)^2 = abab$, $(aba)^3 = abaabaaba$, $(ab)^0 = \epsilon$

Languages

- A set of strings over an alphabet is a **language**.
- Languages over $\{a, b\}$:
  $\emptyset$, $\{\epsilon\}$, $\{b\}$, $\{\epsilon, abb, aaaa\}$,
  $\{a^n | n \in \mathbb{N}\} = \{\epsilon, a, aa, aaa, aaaa, \ldots\}$,
  $\{ab^n | n \in \mathbb{N}\} = \{a, ab, abb, abbb, \ldots\}$,
  $\{(ab)^n | n \in \mathbb{N}\} = \{\epsilon, ab, abab, ababab, \ldots\}$
- Since they are sets, we can make new languages from old with:
  $L \cup M \qquad L \cap M \qquad L - M \qquad \overline{L}$

Product of Languages

- The product of languages L and M is

$$LM = \{st | s \in L \wedge t \in M\}$$

- If $L = \{ab, bb\}$ and $M = \{a, b, c\}$ then $LM = \{aba, abb, abc, bba, bbb, bbc\}$
- Is it always the case that $|LM| = |L||M|$?

Product of Languages

- The product of languages L and M is

$$LM = \{st | s \in L \wedge t \in M\}$$

- If $L = \{ab, bb\}$ and $M = \{a, b, c\}$ then $LM = \{aba, abb, abc, bba, bbb, bbc\}$
- Is it always the case that $|LM| = |L||M|$?
- If $L = \{a, ab\}$ and $M = \{\epsilon, b\}$, then $LM = \{a, ab, abb\}$

Product of Languages

- The product of languages L and M is

$$LM = \{st | s \in L \wedge t \in M\}$$

- If $L = \{ab, bb\}$ and $M = \{a, b, c\}$ then $LM = \{aba, abb, abc, bba, bbb, bbc\}$
- Is it always the case that $|LM| = |L||M|$?
- If $L = \{a, ab\}$ and $M = \{\epsilon, b\}$, then $LM = \{a, ab, abb\}$
- If $L = \{a, ab\}$ and $M = \{a, ba\}$, then $LM = \{aa, aba, abba\}$

Product of Languages

- The product of languages L and M is

$$LM = \{st | s \in L \land t \in M\}$$

- If $L = \{ab, bb\}$ and $M = \{a, b, c\}$ then $LM = \{aba, abb, abc, bba, bbb, bbc\}$
- Is it always the case that $|LM| = |L||M|$?
- If $L = \{a, ab\}$ and $M = \{\epsilon, b\}$, then $LM = \{a, ab, abb\}$
- If $L = \{a, ab\}$ and $M = \{a, ba\}$, then $LM = \{aa, aba, abba\}$
- Why is it always the case that $|L \times M| = |L| \times |M|$?

Properties of Language Products

- $L\{\epsilon\} = \{\epsilon\}L = L$
- $L\emptyset = \emptyset L = \emptyset$

Product of a language with itself

- $L^n = \{s_1 s_2 s_3 \dots s_k | k \in \mathbb{N} \land \forall i, s_i \in L\}$
- If $L = \{a, bb\}$, then
- $L^0 = \{\epsilon\}$
- $L^1 = L = \{a, bb\}$
- $L^2 = LL = \{aa, abb, bba, bbbb\}$

Closure of a Language (Kleene Star)

- The **closure** L^* of a language L is

$$\begin{aligned} L^* &= \bigcup_{i=0}^{\infty} L^i \\ &= L^0 \cup L^1 \cup L^2 \cup \ldots \end{aligned}$$

- The **positive closure** L^+ of a language L is

$$\begin{aligned} L^+ &= \bigcup_{i=1}^{\infty} L^i \\ &= L^1 \cup L^2 \cup L^3 \cup \ldots \end{aligned}$$

Properties of Closure

- $\emptyset^* = \{\epsilon\}^* = \{\epsilon\}$
- $\epsilon \in L$ if and only if $L^+ = L^*$
- $L^* = L^*L^* = (L^*)^*$
- $(L^*M^*)^* = (L \cup M)^*$
- $L(ML)^* = (LM)^*L$

String Substitution

- Start with the string $ABBA$
- If we make the substitutions $A \to a$ and $B \to b$
- $ABBA \Rightarrow abba$
- If we make the substitutions $A \to ab$ and $B \to ba$
- $ABBA \Rightarrow abbababaab$
- If we make the substitutions $A \to bab$ and $B \to bbb$
- $ABBA \Rightarrow babbbbbbbab$

Formal Grammars

- A set of **terminals**, e.g. {the,cat,sat,on,mat}
- A set of **nonterminals**, or **variables**, e.g. $\{S, N\}$
- A special nonterminal, the **start symbol**, e.g. S
- A set of **production rules**:

$$\begin{array}{rcl} S & \rightarrow & \text{the } N \text{ sat on the } N \\ N & \rightarrow & \text{cat} \\ N & \rightarrow & \text{mat} \end{array}$$

- A **derivation** is any string we get by starting with the start symbol and repeatedly making a single substitution until we only have terminals.
- $S \Rightarrow$ the N sat on the $N \Rightarrow$ the cat sat on the $N \Rightarrow$ the cat sat on the mat
- $S \Rightarrow$ the N sat on the $N \Rightarrow$ the mat sat on the $N \Rightarrow$ the mat sat on the mat

Vertical bar means "or"

This grammar:

$$\begin{aligned} S &\rightarrow \text{the } N \text{ sat on the } N \\ N &\rightarrow \text{cat} \\ N &\rightarrow \text{mat} \end{aligned}$$

is equivalent to this grammar:

$$\begin{aligned} S &\rightarrow \text{the } N \text{ sat on the } N \\ N &\rightarrow \text{cat} \mid \text{mat} \end{aligned}$$

Rules can be recursive

$$
\begin{aligned}
S &\rightarrow S \text{ and } S \\
S &\rightarrow \text{the } N \text{ sat on the } N \\
N &\rightarrow \text{cat} \mid \text{mat}
\end{aligned}
$$