Elementary Language Theory

Roland Backhouse

February 18, 2002

Outline

- Languages
- Regular Expressions
- Context-Free (BNF) Grammars
- Extended BNF
- Parsing (Syntax Analysis)
- Grammar Transformations

Languages

An *alphabet* is a finite set. Eg $\{a,b,c\}$.

A *symbol* is an element of an alphabet. Eg a.

A *word* over alphabet $\sum$ is a finite length string of symbols taken from $\sum$. Eg $abcaa$.

ε denotes the word of length 0, the *empty* word.

A *language* (over alphabet $\sum$) is a set of words. Eg $\{abcaa\,, abc\,, b\,, caa\}$.

$\sum^*$ denotes the set of all words over the alphabet $\sum$.

Concatenation

Concatenation of words is denoted by juxtaposition.

Eg $\mathtt{abca}$ is the concatenation of $\mathtt{ab}$ and $\mathtt{ca}$.

Concatenation is associative and has unit ε.

Associativity

$$u(vw) = (uv)w \quad .$$

Unit

$$u\varepsilon = u = \varepsilon u \quad .$$

(Concatenation is *not* symmetric.)

Concatenation of Languages

Concatenation of words is extended to languages by

$$LM = \{uv\colon u \in L \wedge v \in M\} \quad .$$

Concatenation of languages is associative and has unit $\{\varepsilon\}$.

Associativity

$$L(MN) = (LM)N \quad .$$

Unit

$$L\{\varepsilon\} = L = \{\varepsilon\}L \quad .$$

Concatenation of Languages (Cont)

Concatenation of languages distributes through set union.

Distributivity

$$L(M \cup N) = LM \cup LN \ .$$

$$(M \cup N)L = ML \cup NL \ .$$

The zero of concatenation is the empty set.

Zero

$$L\phi = \phi = \phi L \ .$$

It is common to use arithmetic notation —i.e. 0, 1, $+$ and $\cdot$ to denote the empty set (ϕ) , the set containing just the empty word ($\{\varepsilon\}$), union and concatenation, respectively— particularly for regular

expressions. This is because of the similarity in the algebraic properties. For example, 0 is the unit of addition and the zero of multiplication, and the empty set is the unit of set union and the zero of concatenation.

Regular Expressions (over alphabet $\sum$)

ϕ is a regular expression denoting the empty set.

ε is a regular expression denoting $\{\varepsilon\}$.

For each $a \in \sum$, a is a regular expression denoting $\{a\}$.

If R and S are regular expressions then (RS) is a regular expression denoting the concatenation of the languages denoted by R and S.

If R and S are regular expressions then $(R|S)$ is a regular expression denoting the union of the languages denoted by R and S.

If R is a regular expression then (R^*) is a regular expression denoting the union of the languages R^n where n ranges over all natural numbers, R^0 denotes $\{\varepsilon\}$ and R^{n+1} denotes the language denoted by RR^n .

Parentheses may be omitted using the associativity properties of concatenation and union, and the convention that * has precedence over concatenation, which has precedence over $|$.

Examples

$(a|b)c$ denotes $\{ac\,, bc\}$.

a^* denotes $\{\varepsilon\,, a\,, aa\,, aaa\,, \ldots\}$

$(a|c)^*(a|b)$ denotes the set containing all words that begin with an arbitrary number (including zero) of a's and c's and end with an a or a b.

Properties

Two regular expressions are *equal* if they denote the same language.

For example, for all regular expressions R and S,

$$R(SR)^* = (RS)^*R \ .$$

Concatenation and union of regular expressions have the algebraic properties of concatenation of languages (mentioned above) and of set union.

Let regular expression R denote the language L. The language denoted by R^* is the least solution of the following (in)equations in X

$$X \supseteq LX \cup \{\varepsilon\} \ .$$

$$X \supseteq XL \cup \{\varepsilon\} \ .$$

$$X \supseteq XX \cup L \cup \{\varepsilon\} \ .$$

("Least" means with respect to the subset ordering on sets.)

Example

The language denoted by a^* is the least solution of the equation in X

$$X \supseteq \{a\}X \cup \{\varepsilon\} \ .$$

It is also the least solution of the equation in X

$$X \supseteq X\{a\} \cup \{\varepsilon\}$$

and the equation in X

$$X \supseteq XX \cup \{a\} \cup \{\varepsilon\} \ .$$

(Extreme) Examples

The language denoted by ϕ^* is the least solution of the equation in X

$$X \supseteq \phi X \cup \{\varepsilon\} \ .$$

Simplifying, the language denoted by ϕ^* is the least solution of the equation in X

$$X \supseteq \{\varepsilon\} \ .$$

That is, ϕ^* denotes $\{\varepsilon\}$. Thus,

$$\phi^* = \varepsilon \ .$$

The language denoted by ε^* is the least solution of the equation in X

$$X \supseteq \{\varepsilon\} X \cup \{\varepsilon\} \ .$$

Simplifying, the language denoted by ε^* is the least solution of the equation in X

$$X \supseteq X \cup \{\varepsilon\} \ .$$

But $X \supseteq X \cup \{\varepsilon\} \equiv X \supseteq \{\varepsilon\}$. Thus,

$$\varepsilon^* = \varepsilon \ .$$

Properties (Continued)

More generally, the language denoted by L^*M is the least solution of the equation in X

$$X \supseteq LX \cup M \ .$$

Also, the language denoted by ML^* is the least solution of the equation in X

$$X \supseteq XL \cup M \ .$$

Examples

The language denoted by a^*b is the least solution of the equation in X

$$X \supseteq \{a\}X \cup \{b\} \ .$$

The language denoted by ba^* is the least solution of the equation in X

$$X \supseteq X\{a\} \cup \{b\} \ .$$

Grammars

A *context-free grammar* is a 4-tuple (N, T, P, S) where

- N is a finite set (the *nonterminals*),
- T is an alphabet (the *terminals*),
- S is a distinguished element of N (the *start* or *sentence* symbol)
- P is a set of pairs, usually written

$$A ::= \alpha$$

  where $A \in N$ and $\alpha \in (N \cup T)^*$ (the *productions*)

Context-free grammars are also called *BNF* grammars. (BNF is short for Backus Normal Form.)

Example

$$G = (\{S,T\},\{a,b\},P,S)$$

where P consists of the productions

$$S ::= \varepsilon$$

$$S ::= aT$$

$$T ::= bS$$

Productions with the same lefthand side are usually grouped together as in

$$S ::= \varepsilon \mid aT \quad .$$

Extended BNF

An *extended* BNF grammar is a context-free grammar in which the righthand sides of productions are allowed to be regular expressions over the alphabet $\mathsf{N} \cup \mathsf{T}$.

Example:

$$\mathsf{digit} ::= 0 \mid 1 \mid \ldots \mid 9$$

$$\mathsf{digits} ::= \mathsf{digit}\ \mathsf{digit}^*$$

$$\mathsf{OptionalFraction} ::= \ . \ \mathsf{digits} \mid \varepsilon$$

$$\mathsf{OptionalExponent} ::= (\mathsf{E}\ (+ \mid - \mid \varepsilon)\ \mathsf{digits}) \mid \varepsilon$$

$$\mathsf{Number} ::= \mathsf{digits}\ \mathsf{OptionalFraction}\ \mathsf{OptionalExponent}$$

Note the use of colours to distinguish metalanguage from object language. This is necessary, for example, to distinguish grouping

$$(+ \mid -)$$

from parentheses in the object language

$$(\mathsf{Expression}) \quad .$$

Grammar of Regular Expressions

$$G = (\{R\}, \sum, P, R)$$

where P consists of the productions

$$R ::= (RR) \mid (R|R) \mid (R^*) \mid T \mid \varepsilon \mid \phi$$

$$T ::= a|b|\ldots|z$$

(where a, b, etc. are the elements of $\sum$).

Take care to distinguish between meta and object language. Here the object language is the language of regular expressions and the meta language is BNF.

Exercise

The above grammar requires that all non-trivial regular expressions are parenthesised. Rewrite the grammar so that this is not necessary. Instead iteration should be given precedence over concatenation which in turn has precedence over alternation.

Languages and Grammars

A grammar defines a system of simultaneous (in)equations.

For example, the above grammar defines the simultaneous equations in the unknowns digit, digits, OptionalFraction, OptionalExponent and Number,

$$\mathsf{digit} \supseteq \{0\} \cup \{1\} \cup \ldots \cup \{9\}$$

$$\mathsf{digits} \supseteq \mathsf{digit}\,\mathsf{digit}^*$$

$$\mathsf{OptionalFraction} \supseteq \{.\}\,\mathsf{digits} \cup \{\varepsilon\}$$

$$\mathsf{OptionalExponent} \supseteq \{\mathsf{E}\}\,\{+, -, \varepsilon\}\,\mathsf{digits} \cup \{\varepsilon\}$$

$$\mathsf{Number} \supseteq \mathsf{digits}\ \mathsf{OptionalFraction}\ \mathsf{OptionalExponent}$$

The *language generated* by a nonterminal in the grammar in the grammar is the least solution of its defining equation. The *language generated* by the *grammar* is the language generated by the sentence symbol of the grammar.