CS 3813: Introduction to Formal Languages and Automata

Regular grammars (Sec 3.3)

Grammar

- A grammar consists of the following:
  - a set Σ of terminals (same as an alphabet)
  - a set NT of nonterminal symbols, including a starting symbol $S \in NT$
  - a set R of rules
- Example

  $S \rightarrow aS \mid A$

  $A \rightarrow bA \mid e$

Derivation

- Strings are "derived" from a grammar
- Example of a derivation

  $S \Rightarrow aS \Rightarrow aaS \Rightarrow aabA \Rightarrow aab$
- At each step, a nonterminal is replaced by the **sentential form** on the right-hand side of a rule (a sentential form can contain nonterminals and/or terminals)
- Automata recognize languages … grammars generate languages

Context-free grammar

- A grammar is said to be context-free if every rule has a single nonterminal on the left-hand side
- This means you can apply the rule in any context. More complicated languages (such as English) have context-dependent rules. But we only consider context-free grammars in this course.
- A language generated from a context-free grammar is called a context-free language

Regular grammar

- A grammar is said to be **right-linear** if all productions are of the form $A \rightarrow xB|x$, where A and B are nonterminals and x is a string of terminals
- A grammar is said to be **left-linear** if all productions are of the form $A \rightarrow Bx|x$
- A regular grammar is either right-linear or left-linear.

Another formalism for regular languages

- Every regular grammar generates a regular language, and every regular language can be generated by a regular grammar. (We can prove this, but won't in this class …)
- A regular grammar is a simpler, special-case of a context-free grammar
- The regular languages are a proper subset of the context-free languages

Exercises

- Find a regular grammar that generates the language on $\Sigma = \{a,b\}$ consisting of all strings with no more than three a's (page 97 #6)
- Find a regular grammar that generates the language consisting of even-length strings over {a,b}

Exercises

- Find a regular grammar that generates the language L(aa*(ab+a)*). (page 97 #2 in book)
- Find a regular grammar that generates the language $L = \{w \in \{a,b\}^* \mid n_a(w) + n_b(w) \text{ is even}\}$

CS 3813: Introduction to Formal Languages and Automata

Closure properties of regular languages (Sec 4.1)

Languages are just sets of strings. We can use operations on these sets to create other languages. For example, if L_1 and L_2 are regular languages, we can create another language using the union operator as follows:

$$L_3 = L_1 \cup L_2$$

We say that the regular languages are **closed under union** because L_3 is regular whenever L_1 and L_2 are regular. (Why?)

The regular languages are also closed under the following operations:

Concatenation (by construction used in Kleene's theorem)

Kleene star, or star-closure (also by the construction used in Kleene's theorem)

The regular languages are also closed under the following operations:

reversal (given an NFA that accepts language, reverse transitions and switch start and final states)

complement (given DFA, switch final and non-final states)

intersection (because $L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$)

difference (because $L_1 - L_2 = L_1 \cap \overline{L_2}$)

CS 3813: Introduction to Formal Languages and Automata

Questions about regular languages (Sec 4.2)

Consider a language L as defined by a finite acceptor, a regular expression or a regular grammar:

- Given a string w, can we determine whether or not w is a member of L?
- Can we determine whether L is empty, finite or infinite?
- Can we determine whether two regular languages L_1 and L_2 are the same?

CS 3813: Introduction to Formal Languages and Automata

The pumping lemma for regular languages (Sec 4.3)

Non-regular languages

- There are non-regular languages that can be generated by context-free grammars
- The language $\{a^n b^n : n \geq 0\}$ is generated by the grammar $S \rightarrow aSb \mid e$
- The language $L = \{w : n_a(w) = n_b(w)\}$ is generated by the grammar $S \rightarrow SS \mid e \mid aSb \mid bSa$

Pumping Lemma

Let L be a regular language accepted by some DFA with k states. Then for any string $w \in L$ with $|w| \geq m$, w may be written as $w = xyz$, for some x, y, and z satisfying the following:

$$|xy| \leq m,$$
$$|y| \geq 1,$$
$$\text{and} \quad xy^i z \in L \text{ for every } i \geq 1$$

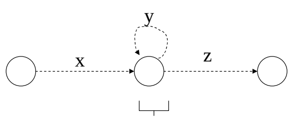
Idea of pumping lemma

If a string in a regular language is sufficiently long, you can always find a substring in it that you can "pump" to get other strings in the language.

So if you find a string in a language (that meets the conditions of the pumping lemma) such that pumping it produces any string that is not in the language, then the language is not regular.

Proof idea

If a DFA has k states, then any path of length k must visit k+1 states, and contains a cycle. (This is an application of the "pigeonhole principle.")



This part of the string can be "pumped" to produce other strings in the language.

Proof idea again

If an infinite language is regular, it is accepted by a DFA.
The DFA has some finite number of states, say, m.
Because the language is infinite, some strings must have length > m.
For a string of length > m accepted by the DFA, a "walk" through the DFA must contain a cycle.
Repeating the cycle an arbitrary number of times must yield another string accepted by the DFA.

How to use the pumping lemma

The Pumping Lemma describes a property that is possessed by every regular language. So if we show that a language does not possess this property, we know that it is not regular.

The strategy is proof by contradiction. We assume a language has the property described by the pumping lemma, and then we show that this leads to a contradiction.

Example

Theorem: The language $L = \{a^n b^n \mid n \geq 0\}$ is not regular.

The proof is by contradiction. If L is regular, it must be accepted by some DFA. Let m be the number of states of the DFA and consider some $w \in L$ such that $|w| \geq m$. By the pumping lemma, we can split w into three pieces, $w = xyz$, such that for any $i \geq 0$, the string xy^iz is in L. So let $w = a^i b^i$. Because $|xy| \leq m$, y must consist of all a's. But then xy^2z will contain more a's than b's, which is a contradiction.

Exercises

Use the pumping lemma to show that the language $L = \{w \in \{a,b\}^* \mid$ w contains equal number of a's and b's$\}$ is not regular.

How can you prove the complement of $L = \{a^n b^n \mid n \geq 0\}$ is not regular? (Remember closure properties.)

Practice with pumping lemma

For each of the following languages, say whether it is regular or not and give a proof.

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

$L = \{w \mid$ w contains 3 more a's than b's$\}$

$L = \{w \in \{a,b\}^* \mid$ w does not have 3 consecutive a's$\}$

$L = \{ww \mid w \in \{a,b\}^*\}$