CSC444

October 26, 2014

Contents

1 Languages that are not regular

We have seen many ways to show that a language is regular.

But we know that there are many languages that cannot be expressed finitely, not to mention by something as simple as finite automata or regular expressions.

What we need is a way to show that a language is not regular.

(And clearly "it's not regular because I can't come up with a finite automaton/regular expression for it" is not very compelling. Maybe you're not clever enough!)

1.1 Intuition

The following are two intuitive descriptions of regular languages:

- The amount of memory used by a finite automaton cannot depend on the length of the string.

Candidate non-regular language: $\{ a^n b^n \mid n \geq 0 \}$.

A finite automaton couldn't "remember" when it reached the b's how many a's it had seen, since that depends on the length of the string.

- An infinite regular language can be represented by an automaton with cycles and by regular expressions using the Kleene star. For this reason, the structure of these languages must have a simple repetitive structure.

Candidate non-regular language: $\{ a^p \mid p \geq 1 \text{ and } p \text{ is prime} \}$.

The primes do not have a simple periodic structure.

These observations give us a way to exploit the simple structure of regular languages to show that a given language is not regular. The following theorem formalizes this.

1.2 The pumping lemma for regular sets

Theorem 1.1. *Let L be a regular language. There is an integer $n \geq 1$ such that any string $w \in L$ with $|w| \geq n$ can be rewritten as $w = xyz$ such that $y \neq \varepsilon$, $|xy| \leq n$ and $xy^iz \in L$ for each $i \geq 0$.*

Proof. Since L is regular, L is accepted by a DFA M. Let n be the number of states of M, and let w be a string of length n or greater.

Consider the first n *steps of the computation* of M on w:

$(q_0, w_1 w_2 ... w_n) \mapsto_M (q_1, w_2 ... w_n) \mapsto_M ... \mapsto_M (q_n, \varepsilon)$,

where q_o is the initial state of M and $w_1 ... w_n$ are the first n symbols of w.

M has only n states, but there are $n+1$ configurations listed above.

Thus by the *pigeonhole principle* $\exists i, j.\ 0 \leq i < j \leq n$ and $q_i = q_j$. $\square$

The Pigeonhole Principle: If A and B are finite sets and $|A| > |B|$, then there is no 1-1 function from A to B.

This means that $y = w_{i+1}...w_j$ takes M from state q_i back to state q_i.

See the diagram on page 56 of Hopcroft and Ullman.

y is also non-empty since $i < j$.

So we could delete y or repeat y as many times as we like and still end up with a string that will be accepted by M.

So M accepts xy^kz for each $k \geq 0$ where $x = w_1...w_i$ and $z = w_{j+1}...w_n$.

Note that $j \leq n$ since $|xy| \leq n$ by definition.

This is what is required by the statement of the theorem. ?

1.3 Using the pumping lemma

The theorem given above is called a pumping lemma because it asserts the existence of certain points in certain strings where a substring can be repeatedly inserted without affecting the acceptability of the string.

Although it is easy to state and to prove, we need to be careful when using it.

The best way to use it is to consider the five alternating quantifiers of the theorem as a *game* between you, the *prover* who is trying to establish that a given language is *not regular*, and an *adversary*, who insists that the language is *regular*.

What quantifiers?

- $\forall$ regular languages L
- $\exists n \geq 1$
- $\forall w \in L$ where $|w| \geq n$
- $\exists x, y, z.\ w = xyz$ and $y \neq \varepsilon$ and $|xy| \leq n$
- $\forall i \geq 0, xy^iz \in L$

The game then goes as follows:

- You and the adversary agree on a language L.
- The adversary picks a number n.
- You come up with a string w in L where $|w| \geq n$.
- The adversary must now divide up w into its pieces x, y, z so that the requirements of the theorem are satisified.
- You win if you can give an i for which $xy^iz \notin L$. The adversary wins if you can't.

If you have a *strategy that always wins*, then the language is not regular.

1.3.1 Examples

$L = \{ a^ib^i \mid i \geq 0 \}$ is not regular.

Suppose it were. Let n be the integer picked by the adversary. Consider the string $w = a^nb^n \in L$.

By the theorem, it can be written as $w = xyz$ s.t. $|xy| \leq n$ and $y \neq \varepsilon$. This means that $y = a^i$ for some $i > 0$.

But then $a^{n-i}b^n$ should be in L. It's not and this contradicts the theorem.

Example 1.2. $L = \{ a^n \mid n \text{ is prime} \}$ is not regular.

Suppose it were. Let x, y, z be as specified in the pumping lemma. Then $x = a^p$, $y = a^q$ and $z = a^r$ where $p, r \geq 0$ and $q > 0$.

By the theorem, $xy^iz \in L$ for each $i \geq 0$.

That means that $p + iq + r$ is prime for each $i \geq 0$.

This is not true!

For example, let $i = p + 2q + r + 2$. Then $p + iq + r = (q + 1)(p + 2q + r)$.

Both $q + 1$ and $p + 2q + r$ are greater than 1.

So $p + iq + r$ is not prime for that value of i.

This violates the theorem, so L must not be regular.

Example 1.3. $L = \{ 0^{i^2} \mid i \geq 1 \}$ is not regular. This is the language of all strings of 0's whose length is a perfect square.

Suppose it is. Let $n \geq 1$ be the integer given by the adversary. Consider $w = 0^{n^2}$.

By the theorem $w = xyz$ where $y \neq \varepsilon$ and xy^iz is in L for all $i \geq 0$.

Consider $i = 2$. $|y| \leq n$ so that $n^2 < |xy^2z| \leq n^2 + n < (n+1)^2$, since $n \geq 1$.

Thus $|xy^2z|$ is not a perfect square and $xy^2z \notin L$.

This contradicts the theorem and shows that L is not regular.

1.4 Other ways to show non-regularity

Let L be a language that we know is not regular (say one that violates the pumping lemma).

Suppose that we have a language L' s.t. $L' = R\text{op}L$ where R is some regular expression and op $\in \{union, concatenation, Kleenestar, intersection\}$. What does this tell us about L' ?

Then by Theorem 2.4.1 we know that L' is not regular since the regular languages are closed under those operations.

Note too that the complement of any non-regular language cannot be regular.

1.4.1 Example

Let $L = \{ w \in \{a, b\}^* \mid w$ has an equal number of a's and b's $\}$

Then L is not regular since $L \cap a^*b^* = \{ a^nb^n \mid n \geq 0 \}$. If L were regular then $\{ a^nb^n \mid n \geq 0 \}$ would be regular. We know this is not true.