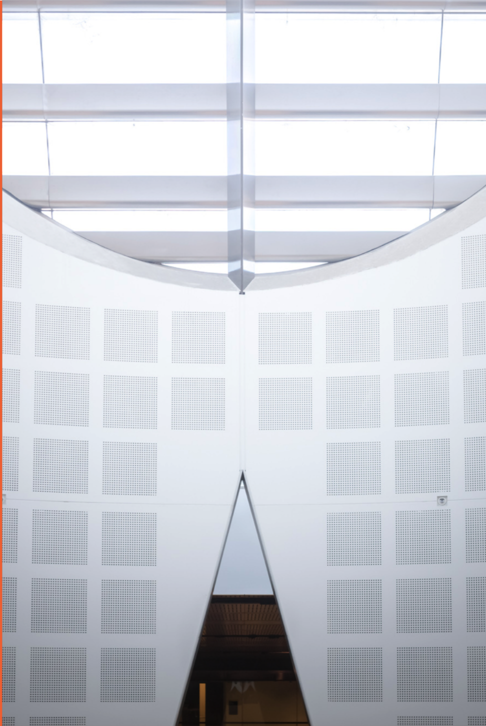
COMP2022|2922
Models of Computation

Non-regularity

Sasha Rubin

August 24, 2022

THE UNIVERSITY OF SYDNEY



To show that a language is regular, it is sufficient to find a DFA, NFA, or Regular Expression for it.

But how do we show a language is not regular?

- One must show that there is no DFA that recognises it.
- We will show that $\{a^n b^n : n \geq 1\} = \{ab, aabb, aaabbb, \ldots\}$ is not regular.
- We will start with an intuition of why this language is not regular, and then prove it mathematically.

Intuition

What happens if you try build a DFA for the language $\{a^n b^n : n \geq 1\}$?

- What information does the DFA have to remember in its state?
- How many states are needed?

Pigeonhole principle (PHP)

If there are n holes and $> n$ objects to put in the holes, then at least one hole must get at least two objects.

- We will apply this with n states and $> n$ strings, and conclude that at least one state must be associated with two strings.
- Then we will use these two strings to "fool" the DFA into accepting (or rejecting) a string that it shouldn't.

Distinguishability

Let L be a language over Σ.

Definition

Two strings x, y are **distinguishable** with respect to L if there is some string z such that exactly one of xz and yz is in L.

So: x, y are **indistinguishable** wrt L means that for every z either both xz and yz are in L or both are not in L.

Example

1. $L = L(a^*b^*)$
   - aa and ab are distinguishable (by $z = a$).
   - aab and ab are indistinguishable.
   - aba and ba are indistinguishable.
2. $L = \{a^n b^n : n \geq 1\}$
   - aa and aaa are distinguishable (by $z = bb$).
   - aaa and $aaaaaaaaa$ are distinguishable (by $z = bbb$).

Distinguishability

What is the largest number of strings that are pairwise distinguishable?

Example

1. $L = L(a^*b^*)$
   - Strings in L that don't have b are pairwise indistinguishable.
   - Strings in L that have b are pairwise indistinguishable.
   - Strings not in L are pairwise indistinguishable.
   - Largest number of strings that are pairwise distinguishable is 3, one from each set, e.g., aa and ab and aba.
2. $L = \{a^n b^n : n \geq 1\}$
   - If $i \neq j$, strings a^i and a^j are distinguishable (by $z = b^i$).
   - So we can find infinitely many strings that are pairwise distinguishable, e.g., $a, aa, aaa, aaaa, \cdots$.

Proving a language is not regular

Theorem

If there are infinitely many strings $x_1, x_2, \cdots$ that are pairwise distinguishable with respect to L then L is not regular.

We prove this by **contradiction**. Here is the structure:

- We are given that there are infinitely many pairwise distinguishable strings wrt L.
- We assume (for a "moment") that our conclusion is false
  - i.e., that there is some DFA recognising L.
- We insert a clever argument here and that ends with a contradictory ("impossible") statement
  - i.e., that some pair of the given strings are indistinguishable!
- Immediately conclude that our assumption is false
  - i.e., L cannot be regular, which is what we want to show!

Proving a language is not regular

Theorem
If there are infinitely many strings $x_1, x_2, \cdots$ that are pairwise distinguishable with respect to L then L is not regular.

1. Given $x_1, x_2, \cdots$ pairwise distinguishable.
2. Assume L is recognised by some DFA, say with state set Q.
3. Write $f(x)$ for the state that M reaches after reading input x.
4. There must exist $i \neq j$ such that $f(x_i) = f(x_j)$.
   - Why? By the pigeonhole principle. There are infinitely many pigeons $x_1, x_2, x_3, \cdots$, but only $|Q|$ many pigeonholes.
5. But if $f(x_i) = f(x_j)$ then x_i and x_j are indistinguishable.
   - Why? if x_i, x_j go to the same state then for every string z both $x_i z$ and $x_j z$ go to the same state, and thus either both $x_i z$ and $x_j z$ are accepted, or both are rejected.

(To think about: where do we use that we are working with a DFA and not an NFA?)

Proving a language is not regular

Theorem

If there are infinitely many strings $x_1, x_2, \cdots$ that are pairwise distinguishable with respect to L then L is not regular.

That proves the theorem.

To apply this theorem to show that a specific L is not regular, we must find infinitely many strings that are pairwise distinguishable wrt L.

Proving a language is not regular

Theorem

If there are infinitely many strings $x_1, x_2, \cdots$ that are pairwise distinguishable with respect to L then L is not regular.

Example 1

$L = \{a^n b^n : n \geq 1\}$ is not regular. We will show that $a, aa, aaa, \cdots$ are pairwise distinguishable.

1. Let $x_n = a^n$ for $n \geq 1$.
2. Then x_n, x_m (for $n \neq m$) are distinguished by $z = b^n$. Why?
3. String $x_n z = a^n b^n \in L$ (obvious).
4. String $x_m z = a^m b^n \notin L$ since $n \neq m$.

Proving a language is not regular

Theorem

If there are infinitely many strings $x_1, x_2, \cdots$ that are pairwise distinguishable with respect to L then L is not regular.

Example 2

$L = \{ww : w \in \{0,1\}^*\}$ is not regular. We will show that $ab, aab, aaab, \cdots$ are pairwise distinguishable wrt L.

1. Let $x_n = a^n b$ for $n \geq 1$.
2. Then x_n, x_m (for $n < m$) are distinguished by $z = a^n b$. Why?
3. $x_n z = a^n b a^n b \in L$ since it is of the form ww for $w = a^n b$.
4. $x_m z = a^m b a^n b \notin L$ since the left half of this string only contains as while the right contains two b's.

Proving a language is not regular

Theorem

If there are infinitely many strings $x_1, x_2, \cdots$ that are pairwise distinguishable with respect to L then L is not regular.

Example 3

$L = \{w \in \{a\}^* : |w| \text{ is a power of } 2\}$ is not regular. We will show that $a, aa, aaaa, aaaaaaaa, \cdots$ are pairwise distinguishable wrt L.

1. Let $x_n = a^{2^n}$ for $n \geq 1$.
2. Then x_n, x_m (for $n < m$) are distinguished by $z = a^{2^n}$. Why?
3. String $x_n z \in L$ since $|x_n z| = 2^n + 2^n = 2^{n+1}$.
4. String $x_m z \notin L$ since

   $$2^m < |x_m z| = 2^m + 2^n < 2^m + 2^m = 2^{m+1}$$

   so $|x_m z|$ is not a power of 2 (because there is no power of 2 between 2^m and 2^{m+1}).

Other techniques

Once we know that a language is not regular, we can deduce that other languages are not regular using the closure properties of regular languages.

Showing a language is not regular

Example

Prove that the set L_1 of strings with the same number of as as bs is not regular.

We will use that $L_0 = \{a^n b^n : n \geq 1\}$ is not regular.

1. We know that L_0 is not regular (previous slides).
2. Assume L_1 is regular.
3. Then $L_1 \cap L(a^* b^*)$ is regular. Why?
4. But $L_0 = L_1 \cap L(a^* b^*)$. Why?
5. So L_0 is regular. But this is a contradiction to what we know.
6. So, our assumption is wrong, i.e., L_1 cannot be regular.

Showing a language is not regular

Example

Prove that the language L_2 consisting of strings over $\{a, b\}$ with a different number of as as bs is not regular.

We will use that L_1 (consisting of strings over $\{a, b\}$ with the same number of as as bs) is not regular.

1. We know that L_1 is not regular (previous slide).
2. Assume L_2 is regular.
3. Then $\{a, b\}^* \setminus L_2$ is regular. Why?
4. So $L_1 = \{a, b\}^* \setminus L_2$ is regular. But this is a contradiction to what we know.
5. So, deduce that L_2 is not regular.

Summary

- We use a fooling argument to show that certain languages are not regular.
  - A variation is called the *pumping lemma*, see Sipser.
- But these particular non-regular languages are still quite simple!
  - e.g., there is an algorithm for deciding if a given string is of the form $a^n b^n$ for some n. How would it work?
- So, we need a more powerful model of computation to recognise more complex languages.
- Next we will see a very powerful model of computation that captures our intuitive idea of what an arbitrary algorithm can do.