1 Expressiveness

1.1 Finite Languages

Finite Languages

Definition 1. A language is finite if it has finitely many strings.

Example 2. $\{0, 1, 00, 10\}$ is a finite language, however, $(00 \cup 11)^*$ is not.

Proposition 3. *If L is finite then L is regular.*

Proof. Let $L = \{w_1, w_2, \ldots w_n\}$. Then $R = w_1 \cup w_2 \cup \cdots \cup w_n$ is a regular expression defining L. □

1.2 Non-Regular Languages

Are all languages regular?

Proposition 4. *The language $L_{\text{eq}} = \{w \in \{0,1\}^* \mid w$ has an equal number of 0s and 1s$\}$ is not regular.*

Proof? No DFA has enough states to keep track of the number of 0s and 1s it might see. □

Above is a weak argument because $E = \{w \in \{0,1\}^* | w$ has an equal number of 01 and 10 substrings$\}$ is regular!

---

2 Proving Non-regularity

2.1 Lower Bound Method

Proving Non-Regularity

Proposition 5. *The language $L_{\text{eq}} = \{w \in \{0,1\}^* \mid w$ has an equal number of 0s and 1s$\}$ is not regular.*

Proof. Suppose (for contradiction) L_{eq} is recognized by DFA $M = (Q, \{0,1\}, \delta, q_0, F)$.

Let $W = \{0\}^*$. For any $w_1, w_2 \in W$ with $w_1 \neq w_2$, $\hat{\delta}_M(q_0, w_1) \neq \hat{\delta}_M(q_0, w_2)$. Let us observe that if the claim holds, then M has infinitely many states, and so is not a finite automaton, giving the desired contradiction.
Claim: For any $w_1, w_2 \in W$ with $w_1 \neq w_2$, $\hat{\delta}_M(q_0, w_1) \neq \hat{\delta}_M(q_0, w_2)$.
Proof of Claim: Suppose (for contradiction) there is w_1 and w_2 such that $\hat{\delta}_M(q_0, w_1) = \hat{\delta}_M(q_0, w_2) = \{q\}$. Without loss of generality we can assume that $w_1 = 0^i$ and $w_2 = 0^j$, with $i < j$. Then, $\hat{\delta}_M(q_0, w_1 1^i = 0^i 1^i) = \hat{\delta}_M(q, 1^i) = \hat{\delta}_M(q_0, w_2 1^i = 0^j 1^i)$. Thus, M either accepts both $0^i 1^i$ and $0^j 1^i$, or neither. But $0^i 1^i \in L_{\text{eq}}$ but $0^j 1^i \notin L_{\text{eq}}$, contradicting the assumption that M recognizes L_{eq}. □

Example I

Proposition 6. $L_{0n1n} = \{0^n 1^n \mid n \geq 0\}$ *is not regular.*

Proof. Suppose L_{0n1n} is regular and is recignized by DFA $M = (Q, \{0,1\}, \delta, q_0, F)$.

- Let $W = \{0\}^*$. For any $w_1, w_2 \in W$ with $w_1 \neq w_2$, $\hat{\delta}_M(q_0, w_1) \neq \hat{\delta}_M(q_0, w_2)$.
  - Suppose (for contradiction) $\hat{\delta}_M(q_0, w_1) = \hat{\delta}_M(q_0, w_2) = \{q\}$, where $w_1 = 0^i$ and $w_2 = 0^j$, with $i < j$.
  - Then, $\hat{\delta}_M(q_0, w_1 1^i = 0^i 1^i) = \hat{\delta}_M(q, 1^i) = \hat{\delta}_M(q_0, w_2 1^i = 0^j 1^i)$.
  - But $0^i 1^i \in L_{0n1n}$ but $0^j 1^i \notin L_{0n1n}$, contradicting the assumption that M recognizes L_{0n1n}.
- Because of the claim, M has infinitely many states, and so is not a finite automaton! □

2.2 Using Closure Properties

Example II
Closure Properties

Proposition 7. $L_{anban} = \{a^n b a^n \mid n \geq 0\}$ *is not regular.*

Proof. We could prove this proposition the way we demonstrated the other languages to be not regular. We could show that for any two (different) strings in $W = \{a\}^* b$, any DFA M recognizing L_{anban} must go to different states, thus showing that M cannot have finitely many states. However, we choose to demonstrate a different technique to prove non-regularity of languages. This relies on closure properties.

The idea behind the proof is to show that if we had an automaton M accepting L_{anban} then we can construct an automaton accepting $L_{0n1n} = \{0^n 1^n \mid n \geq 0\}$. But since we know L_{0n1n} is not regular, we can conclude L_{anban} cannot be regular. This is the idea of *reductions*, where one shows that one problem (namely, L_{0n1n} in this case) can be solved using a modified version of an algorithm solving another problem (L_{anban} in this case), which plays a central role in showing impossibility results. We will see more examples of this as the course goes on.

How do we show that a DFA recognizing L_{anban} can be modified to obtain a DFA for L_{0n1n}? We will use closure properties for this. More formally, we will show that by applying a sequence of "regularity preserving" operations to L_{anban} we can get L_{0n1n}. Then, since L_{0n1n} is not regular, L_{anban} cannot be regular. The proof is as follows.

- Consider homomorphism $h_1 : \{a, b, c\}^* \to \{a, b\}^*$ defined as $h_1(a) = a$, $h_1(b) = b$, $h_1(c) = a$.
  - $L_1 = h_1^{-1}(L_{anban}) = \{(a \cup c)^n b (a \cup c)^n \mid n \geq 0\}$
- Let $L_2 = L_1 \cap \mathbf{L}(a^* b c^*) = \{a^n b c^n \mid n \geq 0\}$
- Homomorphism $h_2 : \{a, b, c\}^* \to \{0, 1\}^*$ is defined as $h_2(a) = 0$, $h_2(b) = \epsilon$, and $h_2(c) = 1$.

– $L_3 = h_2(L_2) = \{0^n 1^n \mid n \geq 0\} = L_{0n1n}$

- Now if L_{anban} is regular then so are L_1, L_2, L_3, and L_{0n1n}. But L_{0n1n} is not regular, and so L is not regular. □

Example III

Proposition 8. *$L_{\text{neq}} = \{w_1 w_2 \mid w_1, w_2 \in \{0,1\}^*,\ |w_1| = |w_2|,$ but $w_1 \neq w_2\}$ is not regular.*

Proof. As before there are two ways to show this result. First we can show that if M with initial state q_0 is a DFA recognizing L_{ww}, then on any two (different) strings in $W = \{0,1\}^*$, M must be in different states. This is because, suppose on $x, y \in \{0,1\}^*$, $\hat{\delta}_M(q_0, x) = \hat{\delta}_(q_0, y)$ then $\hat{\delta}_M(q_0, xy) = \hat{\delta}_M(q_0, yy)$. But $xy \in L_{\text{neq}}$ and $yy \notin L_{\text{neq}}$, giving us the desired contradiction. Thus, M must have infinitely many states (since $|W|$ is infinite), contradicting the fact that M is a finite automaton.

Another proof uses closure properties. Consider the following sequence of languages.

- Let $h_1 : \{0,1,\#\}^* \to \{0,1\}^*$ be a homomorphism such that $h_1(0) = 1$, $h_1(1) = 1$ and $h_1(\#) = \epsilon$. Consider

  $L_1 = h_1^{-1}(L_{\text{neq}}) \cap \mathbf{L}((0 \cup 1)^* \# (0 \cup 1)^*) = \{w_1 \# w_2 | w_1, w_2 \in \{0,1\}^*, |w_1| + |w_2| \text{ is even, and } w_1 \neq w_2\}$

- $L_2 = \{0,1,\#\}^* \setminus L_1$

- $L_3 = L_1 \cap \mathbf{L}((0 \cup 1)^* \# (0 \cup 1)^*) \cap ((\{0,1,\#\}\{0,1,\#\})^* \{0,1,\#\}) = \{w_1 \# w_2 \mid w_1, w_2 \in \{0,1\}^*, \text{ and } w_1 = w_2\}$

- Let $h_2 : \{0,1,\bar{0},\bar{1},\#\}^* \to \{0,1,\#\}^*$ be a homomorphism where $h_2(0) = h_2(\bar{0}) = 0$, $h_2(1) = h_2(\bar{1}) = 1$ and $h_2(\#) = \#$. Let $L_4 = h_2^{-1}(L_3) \cap \mathbf{L}((\bar{0} \cup \bar{1})^* \# (0 \cup 1)^*)$. Observe that

  $$L_4 = \{w_1 \# w_2 \mid w_1 \in \{\bar{0},\bar{1}\}^*,\ w_2 \in \{0,1\}^* \text{ and } w_1 \text{ is same as } w_2 \text{ except for the bars}\}$$

- Let $h_3 : \{0,1,\bar{0},\bar{1},\#\}^* \to \{0,1\}^*$ be the homomorphism where $h_3(\bar{0}) = 0$, $h_3(\bar{1}) = h_3(\#) = h_3(1) = \epsilon$, and $h_3(0) = 1$. Observe that $h_3(L_4) = L_{0n1n}$.

Due the closure properties of the regular languages, if L_{neq} is regular, then so are $L_1, L_2, L_3, L_4, h_3(L_4 = L_{0n1n}$. But since L_{0n1n} is not regular, L_{neq} is not regular. □

2.3 Pumping Lemma

Pumping Lemma: Overview

Pumping Lemma
Gives the template of an argument that can be used to easily prove that many languages are non-regular.

Pumping Lemma

Lemma 9. *If L is regular then there is a number p (the pumping length) such that $\forall w \in L$ with $|w| \geq p$, $\exists x, y, z \in \Sigma^*$ such that $w = xyz$ and*

1. $|y| > 0$
2. $|xy| \leq p$
3. $\forall i \geq 0.\ xy^i z \in L$

Proof. Let $M = (Q, \Sigma, \delta, q_0, F)$ be a DFA such that $L(M) = L$ and let $p = |Q|$. Let $w = w_1 w_2 \cdots w_n \in L$ be such that $n \geq p$. For $1 \leq i \leq n$, let $\{s_i\} = \hat{\delta}_M(q_0, w_1 \cdots w_i)$; define $s_0 = q_0$.

- Since $s_0, s_1, \ldots, s_i, \ldots s_p$ are $p+1$ states, there must be j, k, $0 \leq j < k \leq p$ such that $s_j = s_k$ ($= q$ say).
- Take $x = w_1 \cdots w_j$, $y = w_{j+1} \cdots w_k$, and $z = w_{k+1} \cdots w_n$
- Observe that since $j < k \leq p$, we have $|xy| \leq p$ and $|y| > 0$.

Claim
For all $i \geq 1$, $\hat{\delta}_M(q_0, xy^i) = \hat{\delta}_M(q_0, x)$.

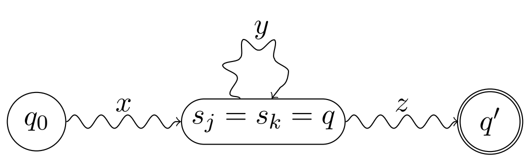
Proof. We will prove it by induction on i.

- *Base Case:* By our assumption that $s_j = s_k$ and the definition of x and y, we have $\hat{\delta}_M(q_0, xy) = \{s_k\} = \{s_j\} = \hat{\delta}_M(q_0, x)$.
- *Induction Step:* We have

$$\begin{aligned}
\hat{\delta}_M(q_0, xy^{\ell+1}) &= \hat{\delta}_M(q, y) \text{ where } \{q\} = \hat{\delta}_M(q_0, xy^\ell) \\
&= \hat{\delta}_M(q, y) \text{ where } \{q\} = \hat{\delta}_M(q_0, x) \\
&= \hat{\delta}_M(q_0, xy) = \hat{\delta}_M(q_0, x)
\end{aligned}$$

□

We now complete the proof of the pumping lemma.



- We have $\hat{\delta}_M(q_0, xy^i) = \hat{\delta}_M(q_0, x)$ for all $i \geq 1$
- Since $w \in L$, we have $\hat{\delta}_M(q_0, w) = \hat{\delta}_M(q_0, xyz) \subseteq F$
- Observe, $\hat{\delta}_M(q_0, xz) = \hat{\delta}_M(q, z) = \hat{\delta}_M(q_0, w)$, where $\{q\} = \hat{\delta}_M(q_0, x) = \hat{\delta}_M(q_0, xy)$. So $xz \in L$
- Similarly, $\hat{\delta}_M(q_0, xy^i z) = \hat{\delta}_M(q_0, xyz) \subseteq F$ and so $xy^i z \in L$

□

Finite Languages and Pumping Lemma

Question
Do finite languages really satisfy the condition in the pumping lemma?

Recall Pumping Lemma: If L is regular then *there is a number* p (the pumping length) such that $\forall w \in L$ with $|w| \geq p$, $\exists x, y, z \in \Sigma^*$ such that $w = xyz$ and

1. $|y| > 0$
2. $|xy| \leq p$
3. $\forall i \geq 0.\ xy^i z \in L$

Answer
Yes, they do. Let p be larger than the longest string in the language. Then the condition "$\forall w \in L$ with $|w| \geq p, \ldots$" is *vaccuously* satisfied as there are no strings in the language longer than p!

Using the Pumping Lemma
L regular implies that L satisfies the condition in the pumping lemma. If L is not regular *pumping lemma says nothing about L!*

Pumping Lemma, in contrapositive
If L does not satisfy the pumping condition, then L not regular.

Negation of the Pumping Condition

$$\forall p. \quad \exists w \in L.\text{ with } |w| \geq p \quad \forall x, y, z \in \Sigma^*.\ w = xyz$$
$$\left.\begin{array}{ll} (1) & |y| > 0 \\ (2) & |xy| \leq p \\ (3) & \forall i \geq 0.\ xy^i z \in L \end{array}\right\} \text{not all of them hold}$$

Equivalent to showing that if $(1), (2)$ then (3) does not. In other words, we can find i such that $xy^i z \notin L$

Game View

Think of using the Pumping Lemma as a game between you and an *opponent*.

L	Task: To show that L is not regular
$\forall p.$	*Opponent picks p*
$\exists w.$	Pick w that is of length at least p
$\forall x, y, z$	*Opponent divides w into x, y, and z such that $|y| > 0$, and $|xy| \leq p$*
$\exists k.$	You pick k and win if $xy^k z \notin L$

Pumping Lemma: If L is regular, *opponent* has a winning strategy (no matter what you do).
Contrapositive: If you can beat the opponent, L not regular.
Your strategy should work for any p and any subdivision that the opponent may come up with.

Example I

Proposition 10. $L_{0n1n} = \{0^n 1^n \mid n \geq 0\}$ *is not regular.*

Proof. Suppose L_{0n1n} is regular. *Let p be the pumping length for L_{0n1n}.*

- Consider $w = 0^p 1^p$
- *Since $|w| > p$, there are x, y, z such that $w = xyz$, $|xy| \leq p$, $|y| > 0$, and $xy^i z \in L_{0n1n}$, for all i.*
- Since $|xy| \leq p$, $x = 0^r$, $y = 0^s$ and $z = 0^t 1^p$. Further, as $|y| > 0$, we have $s > 0$.

$$xy^0 z = 0^r \epsilon 0^t 1^p = 0^{r+t} 1^p$$

Since $r + t < p$, $xy^0 z \notin L_{0n1n}$. Contradiction! □

Example II

Proposition 11. $L_{\text{eq}} = \{w \in \{0,1\}^* \mid w$ *has an equal number of 0s and 1s*$\}$ *is not regular.*

Proof. Suppose L_{eq} is regular. *Let p be the pumping length for L_{eq}.*

- Consider $w = 0^p 1^p$
- *Since $|w| > p$, there are x, y, z such that $w = xyz$, $|xy| \leq p$, $|y| > 0$, and $xy^i z \in L_{\text{eq}}$, for all i.*
- Since $|xy| \leq p$, $x = 0^r$, $y = 0^s$ and $z = 0^t 1^p$. Further, as $|y| > 0$, we have $s > 0$.

$$xy^0 z = 0^r \epsilon 0^t 1^p = 0^{r+t} 1^p$$

Since $r + t < p$, $xy^0 z \notin L_{\text{eq}}$. Contradiction! □

Example III

Proposition 12. $L_p = \{0^i \mid i$ *prime*$\}$ *is not regular*

Proof. Suppose L_p is regular. *Let p be the pumping length for L_p.*

- Consider $w = 0^m$, where $m \geq p + 2$ and m is prime.
- *Since $|w| > p$, there are x, y, z such that $w = xyz$, $|xy| \leq p$, $|y| > 0$, and $xy^i z \in L_p$, for all i.*

- Thus, $x = 0^r$, $y = 0^s$ and $z = 0^t$. Further, as $|y| > 0$, we have $s > 0$. $xy^{r+t}z = 0^r(0^s)^{(r+t)}0^t = 0^{r+s(r+t)+t}$. Now $r + s(r+t) + t = (r+t)(s+1)$. Further $m = r + s + t \geq p + 2$ and $s > 0$ mean that $t \geq 2$ and $s + 1 \geq 2$. Thus, $xy^{r+t}z \notin L_p$. Contradiction! □

Example IV

Question
Is $L_{\text{eq}} = \{xx \mid x \in \{0,1\}^*\}$ is regular?

Suppose L_{eq} is regular, and let p be the pumping length of L_{eq}.

- Consider $w = 0^p0^p \in L$.
- Can we find substrings x, y, z satisfying the conditions in the pumping lemma? Yes! Consider $x = \epsilon, y = 00, z = 0^{2p-2}$.
- Does this mean L_{eq} satisfies the pumping lemma? Does it mean it is regular?
  - No! We have chosen a bad w. To prove that the pumping lemma is violated, we only need to exhibit *some* w that cannot be pumped.
- Another bad choice $(01)^p(01)^p$.

Example IV
Reloaded

Proposition 13. $L_{\text{eq}} = \{xx \mid x \in \{0,1\}^*\}$ *is not regular.*

Proof. Suppose L_{eq} is regular. *Let p be the pumping length for L_{xx}.*

- Consider $w = 0^p10^p1$.
- *Since $|w| > p$, there are x, y, z such that $w = xyz$, $|xy| \leq p$, $|y| > 0$, and $xy^iz \in L_p$, for all i.*
- Since $|xy| \leq p$, $x = 0^r$, $y = 0^s$ and $z = 0^t10^p1$. Further, as $|y| > 0$, we have $s > 0$.

$$xy^0z = 0^r\epsilon 0^t10^p1 = 0^{r+t}10^p1$$

  Since $r + t < p$, $xy^0z \notin L_{\text{eq}}$. Contradiction! □

Lessons on Expressivity

Limits of Finite Memory
Finite automata cannot

- "keep track of counts": e.g., L_{0n1n} not regular.
- "compare far apart pieces" of the input: e.g. L_{xx} not regular.
- do "computations that require it to look at global properties" of the input.
