Transducers

Session 2: Mealy Machines

Aliaume LOPEZ
TA mail* Course page† Exercises page‡

March 11, 2024

1 Continuity. Again.

Exercise 1 (It is cake?). Let Σ and Γ be two alphabets. Prove or disprove that the following functions are continuous for the regular topologies on Σ^* and Γ^*:

- ☐ The map $w \mapsto ww$
- ☐ The map $w \mapsto w^{|w|}$
- ☐ The map $w \mapsto \odot_{i=1}^{|w|} (w_{<i}\bar{w}_i w_{>i}\#)$
- ☐ A function $f : \Sigma^* \to \{1\}^*$ that is increasing and such that $f(\Sigma^*) \subseteq \{n! \mid n \in \mathbb{N}\}$.
- ☐ The map $w \mapsto \odot_{i=1}^{|w|} w_{<i}$.

Exercise 2 (A Graph Property). Let $f : \Sigma^* \to \Gamma^*$ be a function preserving lengths. Prove that the following are equivalent:

1. The graph of f is a regular language,
2. f can be computed by a Mealy Machine with regular lookahead.

▷ Hint 1
▷ Hint 2
▷ Solution 1 (Solution for the easy implication)
▷ Solution 2 (Solution for the hard implication)

Exercise 3 (Sequential Functions).

Exercise 4 (Sequential Functions and Forward Images). Prove that the image of a rational language through a sequential function is a rational language. Is it true for rational functions?

Is the function f that maps to a binary encoded number n its square n^2 a sequential function?

Exercise 5 (Sequential Functions and Topology). Prove that the function $(\times 3)$ that maps a binary encoded number n to its triple $3n$ is not a sequential function.

▷ Hint 3

*ad.lopez@uw.edu.pl
†https://www.mimuw.edu.pl/~bojan/2023-2024/przeksztalcenia-automatowe-transducers
‡https://aliaumel.github.io/transducer-exercices/

Exercise 6 (Injectivity, fixedpoints)**.** For the following models, is injectivity decidable? Is the property of having a fixed point decidable?

1. Mealy Machines
2. Rational transductions
3. Sequential Functions

Exercise 7 (Is it a code?)**.** Are the following functions sequential?

1. The relation β_1^{-1} where $\beta_1 \colon \{x, y\}^* \to \Sigma^*$ is defined by $\beta_1(x) = a$, $\beta_1(y) = aba$?
2. The relation β_2^{-1} where $\beta_2 \colon \{x, y, z\}^* \to \Sigma^*$ is defined by $\beta_2(x) = ab$, $\beta_2(y) = abb$, and $\beta_2(z) = baab$?

Exercise 8 (Coding and Decoding)**.** Let $\beta \colon \Gamma^* \to \Sigma^*$ be a morphism. Prove that the following are equivalent:

1. The set $X := \beta(\Gamma^*)$ is a code with bounded delay
2. The function β^{-1} is an impure sequential function.

▷ Hint 4

Exercise 9 (Continuous functions ...)**.** Let $\Sigma := \{0, \dots, 9\}$ and $f \colon \Sigma^* \to \Sigma^*$ be a rational function. To a word $u \in \Sigma^*$, we associate the number $\bar{u} := \sum_{i=1}^{|u|} u_i 10^{-i}$. This allows us to lift the usual distance on $\mathbb{R}$ to Σ^* by defining $d(u, v) := |\bar{u} - \bar{v}|$.

Can you provide sufficient conditions for f to be continuous in this new topology?

2 Homework

Exercise 10 (String Manipulation)**.** Let Σ be a fixed finite alphabet, and E be a finite set of rules of the form $u \to v$ where $u, v \in \Sigma^*$. Can you think about a method to obtain a transducer that realizes the search and replace operations defined by E? What about the case where patterns overlap? What happens if we allow for rules defined by regular expressions?

3 Cheat Sheet

3.1 Codes

Definition 1 (Codes with Bounded Delay). Let us write $\Gamma := \{x_1, \dots, x_n\} = X$. By definition, X is a code if the map $\beta \colon \Gamma^* \to \Sigma^*$ defined by $\beta(x_i) = x_i$ is injective.

A set $X \subseteq \Sigma^*$ is a prefix code if no word of X is a prefix of another word of X.

A code with bounded delay d is a code X such that for all $u \in \Gamma^{d+1}$, for all $v \in \Gamma^*$ if $\beta(u) \sqsubseteq_{\text{prefix}} \beta(v)$ then $u_1 = v_1$.

3.2 Machines

Definition 2 (Regular Language). A regular language is a language that is recognized by a deterministic finite automaton.

Definition 3 (Mealy Machine). Let Σ and Γ be two alphabets. A Mealy Machine $\mathcal{M}$ is a tuple $(q_0, Q, \delta, \lambda)$ such that

1. Q is a finite set of states.
2. $q_0 \in Q$ is the initial state.
3. $\delta \colon Q \times \Sigma \to Q$ is a transition function.
4. $\lambda \colon Q \times \Sigma \to \Gamma$ is an output function.

The semantics of a Mealy Machine is given by the following inductive equations:

$$\mathcal{M}(w) := \mathcal{M}(q_0, w) \quad \mathcal{M}(q, \varepsilon) := \varepsilon \quad \mathcal{M}(q, au) := \lambda(q, a) \cdot \mathcal{M}(\delta(q, a), u)$$

Definition 4 (Mealy Machine With Lookahead). Let Σ and Γ be two alphabets. A Mealy Machine with Lookahead $\mathcal{M}$ is a tuple $(q_0, Q, \delta, \lambda)$ such that

1. Q is a finite set of states.
2. $q_0 \in Q$ is the initial state.
3. $\delta \subseteq Q \times \Sigma \times Q$ is a transition relation.
4. $\lambda \colon Q \times \Sigma \times Q \to \Gamma$ is an output function.

In addition to this syntactic definition, we furthermore assume that for each $w \in \Sigma^*$, there exists at most one path in the automaton (q_0, Q, δ) starting from q_0 and reading w.

The semantics of the Mealy Machine is given by considering potential runs of the machine. Because of the absence of ambiguity, it defines a partial map $\mathcal{M} \colon \Sigma^* \rightharpoonup \Gamma^*$.

Definition 5 (Sequential Functions). Let Σ and Γ be two alphabets. A sequential transducer A is a tuple $(q_0, Q, \delta, \lambda)$ such that

1. Q is a finite set of states.
2. $q_0 \in Q$ is the initial state.
3. $\delta \colon Q \times \Sigma \rightharpoonup Q$ is a **partial** transition function.
4. $\lambda \colon Q \times \Sigma \to \Gamma^*$ is an output function.

The semantics is defined as for Mealy Machines.

Warning: this is sometimes called pure sequential functions. In this document, we call impure sequential functions those that also have an output function $\rho \colon Q \to \Gamma^*$ that is called at the end of the computation.

Definition 6 (Eilenberg Bimachines)**.** An Eilenberg Bimachine is a tuple (A, B, π, u) where A and B are two deterministic finite automata, and $u \in \Gamma^*$, together with a production function $\pi\colon Q_A \times Q_B \times \Sigma \to \Gamma^*$.with a production function

The semantics of an Eilenberg Bimachine over non-empty words is given as follows:

1. We run A on the input from left to right
2. We run B on the input from right to left
3. We replace every letter a_i of the input by the word $\pi(q_i^A, q_i^B, a_i)$ where q_i^A and q_i^B are respectively the states of A after the letter a_i and B before the letter a_i.

For empty words, the bimachine outputs u.

3.3 Maths

Definition 7 (Graph of a Function)**.** Let $f\colon X \to Y$ be a function. The graph of f is the set $\mathsf{graph}(f) := \{(x, f(x)) \mid x \in X\}$.

Definition 8 (Topology and Continuous functions)**.** Let X be a set. A topology over X is a subset τ of $\mathcal{P}(X)$ closed under finite intersections and arbitrary unions. In a topological space (X, τ), the subsets in τ are called open subsets, and their complement are called closed subsets.

A function $f\colon (X, \tau) \to (Y, \theta)$ is continuous whenever for all open subset $U \in \theta$, its pre-image $f^{-1}(U)$ is an open subset of τ. Equivalently, it is continuous if the pre-image of closed subsets are closed subsets.

Definition 9 (Lipschitz functions)**.** A function $f\colon (X, d_X) \to (Y, d_Y)$ is Lipschitz if there exists a constant $K \geq 0$ such that for all $x_1, x_2 \in X^2$, $d_Y(f(x_1), f(x_2)) \leq K d_X(x_1, x_2)$.

Definition 10 (Prefix Distance)**.** Let Σ^* be a finite alphabet. The prefix distance between two words u, v is $|u| + |v| - 2\,|w|$ where w is the longest common prefix of u and v.

Definition 11 (Regular Topology)**.** Let Σ be a finite alphabet. We equip Σ^* with a metric distance as follows: to a pair of words u, w, we associate the minimal size $s(u, w)$ of a deterministic automaton that separates u from w. The distance between two words u and w, is defined as $d(u, w) := 2^{-s(u,w)}$. The regular topology is the topology defined by this metric on Σ^*.

Equivalently, the regular topology is the coarsest topology containing the regular languages as closed subsets.

A Hints

Hint 1 (Exercise 2 Simple conversions). Transform transitions of the form $p \to^{a/b} q$ into transitions of the form $p \to^{(a,b)} q$. I.e., use the fact that $Q \times (\Sigma \times \Gamma) \to Q \subseteq (Q \times \Sigma) \times (Q \times \Gamma)$.

Hint 2 (Exercise 2 Semantic and Syntaxic Unabmiguity). To convert a graph into a Mealy Machine with regular lookahead, start from a deterministic finite automaton that recognizes the graph.

Hint 3 (Exercise 5 Use the topological characterization). Recall that a sequential function is continuous, and Lipschitz for the prefix distance.

Hint 4 (Exercise 8 Start with sequential functions). Show that if X is a prefix code (no two words of X are related by the prefix relation), then β^{-1} is a pure sequential function.

B Solutions

Solution 1 (Solution to Exercise 2). Consider a Mealy Machine with regular lookahead. It is defined as $T := (Q, q_0, \delta, \lambda)$. Let us write $A := (Q, q_0, \delta', Q)$ where $\delta'(q, (a, b)) = \delta(q, a)$ if $\lambda(q, a) = b$, and is undefined otherwise. An easy induction on the size of the input shows that A recognizes exactly the graph of f.

Solution 2 (Solution to Exercise 2). For simplicity, let us start with a deterministic and co-deterministic finite automaton $A := (Q, q_0, \delta, F)$ that recognizes the graph of f. Define the following Mealy Machine $T := (Q, q_0, \delta', \lambda')$, where $(q, a, q') \in \delta'$ if and only if there exists $b \in \Gamma$ such that $\delta(q, (a, b)) = q'$ in A, and $\lambda(q, a, q') = b$ where b is the unique letter in Γ such that $\delta(q, (a, b)) = q'$ (because the automaton is co-deterministic).

It is an easy induction on the size of the input to prove that T computes f, i.e., that accepting runs produce $f(w)$. Let us now prove that T is unambiguous. Assume that two runs $\rho, \theta \in Q^*$ of T are accepting on a given word w. To each of these runs, one can associate a word u and v, both in Γ^*, and such that $\delta(\rho_i, w_{i+1}, u_{i+1}) = \rho_{i+1})$ for $0 \leq i < |w|$ (and similarly for θ). Because both runs are accepting, and since T produces $f(w)$, we conclude that $u = v = f(w)$. In particular, we can now use the fact that A is deterministic to conclude that $\rho = \theta$.