Automata & Formal Languages

Homework 3 – Closure Properties of Regular Languages

Abraham Murciano

March 31, 2020

1. (a) The language $\mathcal{L} = \mathcal{L}_1 \cap \mathcal{L}_2 \cap \cdots \cap \mathcal{L}_n$, where every $\mathcal{L}_i$ is a regular language, is regular, since intersection is closed under finite intersection.

   (b) The language $\mathcal{L} = \mathcal{L}_1 \cap \mathcal{L}_2 \cap \ldots$, where every $\mathcal{L}_i$ is a regular language, is not regular, since intersection is not closed under infinite intersection.

   (c) If a language $\mathcal{R}$ is regular, and for some language $\mathcal{L}$, $\mathcal{L} - \mathcal{R}$ is regular, then $\mathcal{L}$ is regular. This is because $\mathcal{L} = (\mathcal{L} - \mathcal{R}) \cup \mathcal{R}$, and regular languages are closed under finite union.

   (d) If a language $\mathcal{F}$ is finite, and for some language $\mathcal{L}$, $\mathcal{L} - \mathcal{F}$ is regular, then $\mathcal{L}$ must be regular. This follows from part (c) since all finite languages are regular.

2. If two languages $\mathcal{L}_1$ and $\mathcal{L}_2$ are regular, then the following language $\mathcal{L}$ must be regular.

$$\mathcal{L} = \{a_1 b_1 a_2 b_2 \ldots a_n b_n \in \Sigma^* : a_1 a_2 \ldots a_n \in \mathcal{L}_1 \land b_1 b_2 \ldots b_n \in \mathcal{L}_2\}$$

$\mathcal{L}$ can be shown to be regular since it is the intersection of two regular languages. Let these two languages be called $\mathcal{L}_1'$ and $\mathcal{L}_2'$. We will define these as follows.

$$\mathcal{L}_1' = \{a_1 b_1 a_2 b_2 \ldots a_n b_n \in \Sigma^* : a_1 a_2 \ldots a_n \in \mathcal{L}_1\}$$
$$\mathcal{L}_2' = \{a_1 b_1 a_2 b_2 \ldots a_n b_n \in \Sigma^* : b_1 b_2 \ldots b_n \in \mathcal{L}_2\}$$

Each of these is the language that can be formed by taking $\mathcal{L}_1$ and $\mathcal{L}_2$ respectively and inserting any symbol in the language after or before each symbol in every word, respectively. To show that both $\mathcal{L}_1'$ and $\mathcal{L}_2'$ are regular given that $\mathcal{L}_1$ and $\mathcal{L}_2$ are regular, we will show that it is possible to 'extend' the DFA of each of $\mathcal{L}_1$ and $\mathcal{L}_2$ to create DFAs which accept $\mathcal{L}_1'$ and $\mathcal{L}_2'$.

To construct a DFA for $\mathcal{L}_1'$ from the DFA which accepts $\mathcal{L}_1$, we can replace each transition on a symbol σ out of a state q_r to q_s in $\mathcal{L}_1$ into a transition on the same symbol σ from q_r to a new intermediate state q_t, and from q_t it will transition to q_s on any symbol. If q_r was a final state in the DFA for $\mathcal{L}_1$ it will not be for $\mathcal{L}_1'$, and q_t will be instead. See Figure 1.

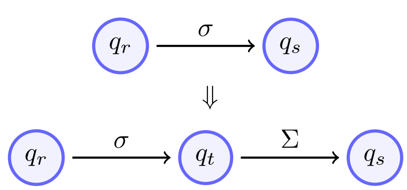

Figure 1: Conversion from $\mathcal{L}_1$ to $\mathcal{L}_1'$

Similarly, a DFA for $\mathcal{L}_2'$ can be constructed from the DFA which accepts $\mathcal{L}_2$. We can replace each transition on a symbol σ into a state q_s from q_r in $\mathcal{L}_2$ into a transition on any symbol from

q_r to a new intermediate state q_t, and from q_t it will transition to q_s on the symbol σ. The final states are unchanged, and a new start state is added to the DFA for $\mathcal{L}_2'$ which transitions to the original start state on any symbol. See Figure 2.

Figure 2: Conversion from $\mathcal{L}_2$ to $\mathcal{L}_2'$

Since $\mathcal{L}_1'$ and $\mathcal{L}_2'$ are regular, their intersection, $\mathcal{L}$, is also regular.

3. Let $\mathcal{L}$ be a regular language over some alphabet Σ.

   (a) The language containing all words which are double the length of any word in $\mathcal{L}$,

   $$\mathcal{L}' = \{a^{2|w|} : w \in \mathcal{L}\}$$

   is regular. If the DFA for language $\mathcal{L}$ is modified such that every edge from q_a to q_b is exchanged for an edge leading to a new state on any symbol in Σ, and the new state also leads to q_b on any symbol in Σ, and the start state is replaced with a new state which transitions to the original start state, then an NFA which accepts $\mathcal{L}'$ can be constructed. Since this is possible, then $\mathcal{L}'$ must be regular.

   (b) The language consisting of all prefixes of any word in $\mathcal{L}$,

   $$\text{prefix}(\mathcal{L}) = \{w \in \Sigma^* : \exists u \in \Sigma^*, wu \in \mathcal{L}\}$$

   is regular, since a DFA can be built for it by making any state which leads to an accepted state in $\mathcal{L}$'s DFA into an accepted state in prefix($\mathcal{L}$)'s DFA.

   (c) The language containing all 'superstrings' of a word in $\mathcal{L}$, i.e. strings that contain a word in $\mathcal{L}$ as a substring,

   $$\sup(\mathcal{L}) = \{w \in \Sigma^* : \exists x, z \in \Sigma^*, \exists y \in \mathcal{L}, w = xyz\}$$

   is a regular language. An NFA with ε transitions can easily be constructed for this language. Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be two automatons that accept Σ^*, and DFA$_\mathcal{L}$ be an automaton that accepts $\mathcal{L}$. We can now 'concatenate' $\mathcal{A}_1$ to DFA$_\mathcal{L}$ and then 'concatenate' DFA$_\mathcal{L}$ to $\mathcal{A}_2$. The way in which we will concatenate two automata is by adding an ε-transition from every accepted state in the first one to the start state of the second.

   This operation will give us an NFA that accepts sup($\mathcal{L}$), so it must be regular.