CMPSCI 250: Introduction to Computation

Lecture #29: Proving Regular Language Identities
David Mix Barrington
6 April 2012

Proving Regular Language Identities

- Regular Language Identities
- The Semiring Axioms Again
- Identities Involving Union and Concatenation
- Proving the Distributive Law
- The Inductive Definition of Kleene Star
- Identities Involving Kleene Star
- $(ST)^*$, S^*T^*, and $(S + T)^*$

Regular Language Identities

- In this lecture and the next we'll use our new formal definition of the regular languages to prove things about them. In particular, in this lecture we'll prove a number of **regular language identities**, which are statements about languages where the types of the free variables are "regular expression" and which are true for all possible values of those free variables.

- For example, if we view the union operator + as "addition" and the concatenation operator $\cdot$ as "multiplication", then the rule $S(T + U) = ST + SU$ is a statement about languages and (as we'll prove today) is a regular language identity. In fact it's a language identity as regularity doesn't matter.

- We can use the inductive definition of regular expressions to prove statements about the whole family of them -- this will be the subject of the next lecture.

The Semiring Axioms Again

- The set of natural numbers, with the ordinary operations + and ×, forms an algebraic structure called a **semiring**. Earlier we proved the semiring axioms for the naturals from the Peano axioms and our inductive definitions of + and ×. It turns out that the languages form a semiring under union and concatenation, and the regular languages are a **subsemiring** because they are **closed** under + and $\cdot$: if R and S are regular, so are $R + S$ and $R \cdot S$.

- Both operations of a semiring must be associative and each must have an identity. For languages, $\emptyset$ is the identity for union and $\{\lambda\} = \emptyset^*$ is the identity for concatenation, as $\emptyset + R = R + \emptyset = R$ and $R\emptyset^* = \emptyset^*R = R$. We also need the distributive law which we'll prove soon.

- Note that + is commutative but $\cdot$ is not as in general XY and YX are different languages. There are other identities like $X + X = X$ that are not true for the natural numbers.

Identities Involving Union and Concatenation

- We've already proved everything we need to know about just + for languages, since they are **set identities** for the union operator. We know that $S + T = T + S$, $S + (T + U) = (S + T) + U$, $S + \emptyset = \emptyset + S = S$, $S + S = S$, and $S + \Sigma^* = \Sigma^*$.

- We looked at concatenation of languages back in Chapter 2. Statements like $S(TU) = (ST)U$, $S\emptyset = \emptyset S = \emptyset$, and $S\emptyset^* = \emptyset^* S = S$ are proved by the equational sequence method -- to prove "X = Y" we let w be an arbitrary string and prove $w \in X \leftrightarrow w \in Y$.

- For example, $w \in (ST)U \leftrightarrow \exists u:\exists z:(w = uz) \wedge (u \in ST) \wedge (z \in U) \leftrightarrow \exists x:\exists y:\exists z:(w = xyz) \wedge (x \in S) \wedge (y \in T) \wedge (z \in U) \leftrightarrow \exists x:\exists v:(w = xv) \wedge (x \in S) \wedge (v \in TU) \leftrightarrow w \in S(TU)$. At each stage we use the definition of concatenation of languages or the associativity of concatenation of *strings* ($x(yz) = (xy)z$), which we've proved.

Proving the Distributive Law

- The equational sequence method also works to prove S(T + U) = ST + SU:

$$
\begin{aligned}
w \in S(T + U) &\leftrightarrow \\
\exists u{:}\exists v{:}(w = uv) \land u \in S \land v \in (T + U) &\leftrightarrow \\
\exists u{:}\exists v{:}\ w = uv \land u \in S \land (v \in T \lor v \in U) &\leftrightarrow \\
\exists u{:}\exists v{:}\ w = uv \land [(u \in S \land v \in T) \lor (u \in S \land v \in U)] &\leftrightarrow \\
(\exists u{:}\exists v{:}\ w = uv \land u \in S \land v \in T) \lor (\exists u{:}\exists v{:}\ w = uv \land u \in S \land v \in U) &\leftrightarrow \\
w \in ST \lor w \in SU &\leftrightarrow \\
w \in ST + SU &
\end{aligned}
$$

- Again we use the definition of concatenation of languages, some boolean rules about $\lor$ and $\land$, and the fact that an $\exists$ statement splits over $\lor$.

The Inductive Definition of Kleene Star

- To prove identities about the Kleene star operation, we use its inductive definition. If A is any language, we define A^* by three rules: (1) $\lambda \in A^*$, (2) if $u \in A^*$ and $v \in A$, then $uv \in A^*$, and (3) a string is only in A^* if it can be proved to be so by rules (1) and (2).

- The definition we gave earlier, "$w \in A^*$ if and only if w is the concatenation of zero or more strings, each of which is in A" is equivalent. By induction on naturals n, we can prove that any concatenation of n strings from A is in A^* according to the second definition. And we can prove by induction on all strings w in A^* (according to the second definition) that there exists an n such that w is the concatenation of n strings from A.

- This is an example of a general phenomenon -- any of our **structural inductions** on the definition of a class could be rephrased as inductions on the naturals.

Identities Involving Kleene Star

- The statement "($u \in A^* \wedge v \in A^*$) $\rightarrow uv \in A^*$", or "A^* is closed under concatenation", is not part of the definition of Kleene star though it is very similar to our rule (2) which says "($u \in A^* \wedge v \in A$) $\rightarrow uv \in A^*$".

- Let's prove the closure rule by induction on all strings v in A^*. Our statement $P(w)$ is "$u \in A^* \rightarrow uv \in A^*$". The base case is $v = \lambda$, and it is clear that if $u \in A^*$ and $v = \lambda$, then $uv \in A^*$ since $uv = u$. For the induction, assume that $v = wx$, that $w \in A^*$, that $x \in A$, and that we already know $P(w)$, that $u \in A^* \rightarrow uw \in A^*$.

- Now to prove $P(v)$, we assume $u \in A^*$, derive $uw \in A^*$ from the IH, and derive that $uv = uwx$ is in A^* from rule (2), because $uw \in A^*$ and $x \in A$.

- This should remind you of the proof that the path relation on graphs is transitive, using the inductive definition of paths.

$(ST)^*$, S^*T^*, and $(S+T)^*$

- It is generally much easier to prove subset relationships that set equalities from the Kleene star definition. The equality identities that are true, like $(S^*)^* = S^*$, are most easily proved by showing both $(S^*)^* \subseteq S^*$ and $S^* \subseteq (S^*)^*$. These in turn follow from the identities $T \subseteq T^*$ and $(S \subseteq T) \rightarrow (S^* \subseteq T^*)$. Both of these in turn follow from $(S \subseteq T^*) \rightarrow (S^* \subseteq T^*)$.

- How to prove this? Assume $S \subseteq T^*$, let $P(w)$ be "$w \in T^*$", and prove $P(w)$ for all w in S^*. For the base case, $w = \lambda$ and we know $\lambda \in T^*$. For the induction, assume $w = xy$ with $P(x)$ true and $y \in S$. So $X \in T^*$ by the IH, $y \in T^*$ because $S \subseteq T^*$, and then $w = xy$ is in T^* by the closure of T^* under concatenation.

- We have seen that parentheses matter, so that $(ST)^*$ and S^*T^* are two different languages for most choices of S and T. (We saw that $(ab)^* \neq a^*b^*$, for example.) But we can prove that both $(ST)^*$ and S^*T^* are contained in $(S + T)^*$, using the identities above.