LECTURE - 11

In classical automata theory, we have Kleene's Theorem: Rational Languages are ~~Regular~~ Recognizable Languages. We want an analogue for the weighted setting.

Series

Suppose $f : \Sigma^* \longrightarrow \mathbb{S}$ is an weighted language over $\mathbb{S}$, then we represent f by the formal object $\sum_{w \in \Sigma^*} f(w)\, w$, which is what we call <u>series</u>. Note that since Σ^* is non-commutative, the coefficient of, say, 'ab', cannot be identified with the coefficient of 'ba'.

A <u>polynomial</u> is a series in which all but finitely many coefficients ~~were~~ 0. A <u>monomial</u> is a polynomial with exactly one non-zero coefficient.

~~Note that~~ We shall use 1 to mean the polynomial 1, i.e, the function which has weight 1 on ε and 0 on all other words. Note that we may define scalar multiplication on the left or on the right in a natural way.

$$\lambda \sum_{w \in \Sigma^*} f(w)\, w = \sum_{w \in \Sigma^*} (\lambda \cdot f(w))\, w$$

$$\left(\sum_{w \in \Sigma^*} f(w)\, w\right) \lambda = \sum_{w \in \Sigma^*} (f(w)\, \lambda)\, w$$

Series expressions are meant to be an analogue for regular expressions in the general setting.

Metric on weighted languages and Summability

In order to define an analogue for the Kleene Star on weighted languages, we ~~define~~ need a notion of convergence for infinite sums, as we will see. Hence, we equip the space of quantitative languages with a metric.

$$d : (\Sigma^* \to \mathbb{S}) \times (\Sigma^* \to \mathbb{S}) \longrightarrow \mathbb{R}$$

$$d(f,g) = \left.\begin{matrix} 2^{-r} & f \neq g \\ 0 & f = g \end{matrix}\right\} \quad \text{where } r = \min\{ |w| \mid f(w) \neq g(w),\ w \in \Sigma^* \}$$

It is easy to check that d is a matric on $\Sigma^* \to S$. Infact, it is an ultrametric:

$$d(f,h) \leq \max(d(f,g), d(g,h)) \text{ for all } g$$

<u>Defn</u> Suppose $\{f_k\}_{k\in K}$ is a family of series indexed by K. $\{f_k\}_{k\in K}$ is <u>summable</u> if $\exists f$ s.t. $\forall \varepsilon > 0$, $\exists I \subseteq_{finite} K$, $\forall J \underset{finite}{\supseteq} I$

$$d\left(\sum_{j\in J} f_j, f\right) < \varepsilon$$

where $\sum_{j\in J} f_j$ is an usually interpreted finite sum.

If this happens, we say $\sum_{k\in K} f_k = f$

<u>Defn</u> $\{f_k\}_{k\in K}$ is <u>locally finite</u> if for all $w \in \Sigma^*$, there are only finitely many indices $k \in K$ s.t $f_k(w) \neq 0$.

<u>Lemma</u> A locally finite family is summable. ◻

~~Lemma~~

<u>Defn</u> ~~$\{f_k\}_{k\in K}$ is prop~~ f is proper if $f(\varepsilon) = 0$

~~Defn Let f be an~~

Rational Series

The smallest class of weighted languages which includes all polynomials and are closed under the following operations are called rational series:

- Scalar multiplication on left and right; $(sf)(w) = s \cdot f(w)$ $\quad s \in S$
  $(fs)(w) = f(w) \cdot s$
- Binary Sum: $(f+g)(w) = f(w) + g(w)$
- Cauchy Product: $(f \cdot g)(w) = \sum_{\substack{u,v \\ uv=w}} f(u) g(v)$
- Kleene Star: Define $f^i = \begin{cases} 1 & \text{if } i = 0 \\ f \cdot f^i & i > 0 \end{cases}$

$$f^* = \sum_{i \geq 0} f^i$$

We discuss a sufficient condition for f^* to be defined.

Lemma (aka. f is proper) If $f(\varepsilon) = 0$, f^* is well defined.

Proof We show that $\{f^i\}_{i \geq 0}$ is locally finite.

Suppose $w \in \Sigma^*$ and $|w| = l$. Then, for any $j > l$, the coefficient of w in f^j would be $\sum_{\substack{u_1 u_2 \cdots u_j \\ u_1 u_2 \cdots u_j = w}} f(u_1) \cdot f(u_2) \cdots f(u_j)$. But, at least one of the u_i's would be 0. So, at most l indices have a non-zero value at w.

$\boxtimes$

Rational Series are Recognizable

- Recognizing a monomial is easy. We need an automata that matches the exact word and has the exact weight
- Binary sums are realizable via disjoint union.
- Suppose $\mathcal{A}$ and $\mathcal{B}$ are two weighted automata. Below, we construct C which recognizes the Cauchy Product of the corresponding functions. This construction is analogous to the corresponding construction for concatenation in usual finite state automata.

Consider the state space $Q = \mathcal{A} \sqcup \mathcal{B}$

For every transition $q_1 \xrightarrow[w \cdot a]{} q_2$ in $\mathcal{A}$,

add a corresponding transition in the new automaton.

Repeat a similar construction foh transitions of $\mathcal{B}$.

For triples of states $q_f \in \mathcal{A}$ and $q_i, q \in \mathcal{B}$

s.t. $q_f \xrightarrow{w_1}$

$\xrightarrow{w_2} q_i \xrightarrow{w_3} q$

add the transition $q_f \xrightarrow{w_1 w_2 w_3} q$ in the new automaton

For ~~triples~~ pairs of states $q_1 \in \mathcal{A}$ and $q_2 \in \mathcal{B}$

s.t. $\xrightarrow{w_0} q_1 \xrightarrow{w_1}$ $\quad \xrightarrow{w_2} q_2$

add the initial weight $\xrightarrow{w_0 w_1 w_2} q_2$