Learning probabilistic finite automata

Tuan-Phong NGO

Reusing slides from Colin de la Higuera
Outline

- Frequency finite automata and probabilistic finite automata
- Types of probabilistic finite automata
- Learning probabilistic finite automata
Recall

- Probability
- Frequency
- Relative frequency

Example: probability to select a specific face of a coin is 1/2. If we do 100 times and get the face 49 times, then frequency is 49, and relative frequency is 49/100
**FFA & PFA**

Frequency finite automata: FFA

![Diagram of FFA]

\[
P(q) = \frac{F(q)}{F_{q' \in Q, a \in \Sigma}(q', a, q)}
\]

Probability of states

Probabilistic finite automata: PFA

![Diagram of PFA]

\[
\partial(q, a, q') = \frac{F(q, a, q')}{F_{q'' \in Q, b \in \Sigma}(q'', b, q)}
\]

Probability of edges
Types of PFA

DPFA: Deterministic Probabilistic Finite Automaton

NPFA: Nondeterministic Probabilistic Finite Automaton

ε-DPFA: Deterministic Probabilistic Finite Automaton with ε

**Probability of strings**

- $P(a) = \frac{1}{2} \times \frac{1}{2} = \frac{1}{4}$
- $P(ab) = \frac{1}{2} \times \frac{1}{2} \times 0 = 0$
- $P(aba) = \frac{1}{2} \times \frac{1}{2} \times \frac{1}{3} \times 0 = 0$
- $P(abab) = \frac{1}{2} \times \frac{1}{2} \times \frac{1}{3} \times \frac{2}{3} \times \frac{3}{4} = \frac{1}{24}$
How useful are these automata

- They can define a distribution over $\Sigma^*$
  - $a, ab, abab...$
  - **Consistent:** $0 \leq P(w) \leq 1, \sum_{w \in \Sigma^*} P(w) = 1$
  - or $\forall q \in Q : P(q) + \sum_{q' \in Q, a \in \Sigma} \partial(q, a, q') = 1$
- They do **not** tell us if a string belongs to a language
\[ P(aba) = 0.7 \times 0.4 \times 0.1 \times 1 + 0.7 \times 0.4 \times 0.35 \times 0.2 = 0.0476 \]
PFA are strictly more powerful than DPFA (Folk theorem)

NPFA over $a^*$ ≠ DPFA over $a^*$
Learning PDA

- Give a sample and a target PDA, how to construct the PDA that compared with target PDA with a threshold $\alpha$

| $\lambda$ | 490 | abb | 4 | abab | 2 | aaaaa | 1 |
| a | 128 | baa | 9 | abba | 2 | aaab | 1 |
| b | 170 | bab | 4 | abbb | 1 | aaaba | 1 |
| aa | 31 | bba | 3 | baaa | 2 | aabaa | 1 |
| ab | 42 | bbb | 6 | baab | 2 | aabab | 1 |
| ba | 38 | aaaa | 2 | baba | 1 | aabba | 1 |
| bb | 14 | aaab | 2 | babb | 1 | abbaa | 1 |
| aaa | 8 | aaba | 3 | bbba | 1 | abbab | 1 |
| aab | 10 | aabb | 2 | bbab | 1 | abab | 1 |
Learning PDA

- Distance between 2 PDA (skip)
- Frequency prefix tree acceptor
- Algeriga algorithm
- Merging and folding
Frequency prefix tree acceptor (FPTA)
Alergia algorithm

• Start with a FPTA

• Try to merge and fold states together until get an acceptable PFA
Merging and folding

- Merge and fold $q_{aa}$ with $q_\lambda$
Merging

• Disconnect and reconnect
Folding

folding $q_{aaa}$

folding $q_{aab}$
Merging and folding?
Merging and folding?