3. Finite automata and regular languages: theory

Probabilistic FA
- a NFA whose transitions
  are assigned probabilities

State Minimization
- dynamic programming algorithm
- pairs of non-equivalent states are gradually being identified

variant

(series-parallel-loop construction)

$\leq$

related

**Non-deterministic FA
(NFA)**

contained in

ε-free NFA

contained in

**Deterministic FA
(DFA)**

$\leq$
(ε-Hull)

$\leq$
(power set)

$\leq$
(dynamic
programming)

**Regular Expressions
(RE)**
- primitive and compound
  expressions

is a

is a

is a

Finite Automata (FA)
- $M = (Q,A,f,s0,F)$

variant

described by

described by

**Finite State Machine
(fsm)**
- with output alphabet
- output either attached to transition
  (h: Q x A $\rightarrow$ B)
  or to current state
  (g: Q $\rightarrow$ B)

Regular Languages

related

related

Pumping Lemma
- used to prove non-regularity
- Def.: L regular, $\exists n(L)$, $\forall\, w \in L$, $|w| \geq n$, $w = xyz$:
  1) $|xy| \leq n$, 2) $|y| > 0$, 3) $\forall i \geq 0$, $x\, y^i\, z \in L$.

Closure Properties
- closed under
  union, concatenation, Kleene star,
  intersection and complement
- proof by construction of suitable
  FAs