Equivalence of NFA and Regular Expressions

Dr. Neminath Hubballi

Indian Institute of Technology Indore

January 17, 2018

Overview

1. Equivalence Proof
2. Regular Languages and Equivalence

Generalized NFA

Definition

A Generalized Transition Graph is an NFA which has

Generalized NFA

Definition

A Generalized Transition Graph is an NFA which has

1. A single start and final state.

Generalized NFA

Definition

A Generalized Transition Graph is an NFA which has

1. A single start and final state.
2. There is no incoming edge on start state.

Generalized NFA

Definition

A Generalized Transition Graph is an NFA which has

1. A single start and final state.
2. There is no incoming edge on start state.
3. There is no outgoing edge on final state.

Generalized NFA

Definition

A Generalized Transition Graph is an NFA which has

1. A single start and final state.
2. There is no incoming edge on start state.
3. There is no outgoing edge on final state.
4. Any edge between states q_i and q_j is a regular expression r.

Generalized NFA

Definition

A Generalized Transition Graph is an NFA which has

1. A single start and final state.
2. There is no incoming edge on start state.
3. There is no outgoing edge on final state.
4. Any edge between states q_i and q_j is a regular expression r.

Generalized NFA

$\mathcal{M}=(Q, \Sigma, \delta, q_0, F)$

Generalized NFA

$\mathcal{M}=(Q, \Sigma, \delta, q_0, F)$
where

Generalized NFA

$\mathcal{M}=(Q, \Sigma, \delta, q_0, F)$
where
Q - Finite non empty set of states.

Generalized NFA

$\mathcal{M}=(Q, \Sigma, \delta, q_0, F)$
where
Q - Finite non empty set of states.
Σ- Finite non empty set of alphabet

Generalized NFA

$\mathcal{M}$=$(Q, \Sigma, \delta, q_0, F)$
where
Q - Finite non empty set of states.
Σ- Finite non empty set of alphabet
q_0- Start state

Generalized NFA

$\mathcal{M}=(Q, \Sigma, \delta, q_0, F)$
where
Q - Finite non empty set of states.
Σ- Finite non empty set of alphabet
q_0- Start state
$F = \{q_f\}$ (Final state)

Generalized NFA

$\mathcal{M}=(Q, \Sigma, \delta, q_0, F)$
where
Q - Finite non empty set of states.
Σ- Finite non empty set of alphabet
q_0- Start state
$F = \{q_f\}$ (Final state)
δ : {Q-F} X {Q-q_0} $\rightarrow R_\Sigma$ Transition function with mapping of every state to a regular expression

Generalized NFA

1. Transition: NFA reads a block of symbols from input and makes a non-deterministic move from state q_i to state q_j

1. Transition: NFA reads a block of symbols from input and makes a non-deterministic move from state q_i to state q_j
2. Acceptance: NFA reads all input symbols in w through a sequence of moves $q_0, \cdots q_f$ where $q_f \in F$. i.e., $\delta^*(q_0, w) = q_f$

Generalized NFA

Theorem

For any NFA M with $L = L(M)$, $\exists$ a regular expression r such that $L(M) = L(r)$

Generalized NFA

Theorem

For any NFA M with $L = L(M)$, $\exists$ *a regular expression r such that* $L(M) = L(r)$

Proof:

Generalized NFA

Theorem

For any NFA M with $L = L(M)$, $\exists$ a regular expression r such that $L(M) = L(r)$

Proof:

1. Convert given NFA into GNFA.

Generalized NFA

Theorem

For any NFA M with L= L(M), ∃ a regular expression r such that $L(M) = L(r)$

Proof:

1. Convert given NFA into GNFA.
2. If GNFA has only two states q_0 and q_f then regular expression is the label on edge from q_0 to q_f

Generalized NFA

Theorem

For any NFA M with L= L(M), $\exists$ a regular expression r such that $L(M) = L(r)$

Proof:

1. Convert given NFA into GNFA.
2. If GNFA has only two states q_0 and q_f then regular expression is the label on edge from q_0 to q_f
3. IF GNFA has $K \geq 3$ states then reduce it to a GNFA with $K - 1$ states successively to get a GNFA with 2 states.

Generalized NFA

Theorem

For any NFA M with L= L(M), ∃ a regular expression r such that $L(M) = L(r)$

Proof:

1. Convert given NFA into GNFA.
2. If GNFA has only two states q_0 and q_f then regular expression is the label on edge from q_0 to q_f
3. IF GNFA has $K \geq 3$ states then reduce it to a GNFA with $K - 1$ states successively to get a GNFA with 2 states.

Generalized NFA

GNFA with K states to GNFA with $K-1$ states

[1] q_a and q_b may be same

Generalized NFA

GNFA with K states to GNFA with $K-1$ states
$\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$

[1] q_a and q_b may be same

Generalized NFA

GNFA with K states to GNFA with $K-1$ states

$\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$

$\mathcal{M}' = (Q', \Sigma, \delta', q_0, F)$

[1] q_a and q_b may be same

Generalized NFA

GNFA with K states to GNFA with $K-1$ states
$\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$

$\mathcal{M}' = (Q', \Sigma, \delta', q_0, F)$
where

[1] q_a and q_b may be same

Generalized NFA

GNFA with K states to GNFA with $K-1$ states
$\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$

$\mathcal{M}^{'} = (Q^{'}, \Sigma, \delta^{'}, q_0, F)$
where
$Q^{'} = \{$Q-$q_i\}$ $q_i \in Q$ and $q_i \neq q_0$ and $q_i \neq q_f$

[1] q_a and q_b may be same

Generalized NFA

GNFA with K states to GNFA with $K-1$ states
$\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$

$\mathcal{M}^{'} = (Q^{'}, \Sigma, \delta^{'}, q_0, F)$
where
$Q^{'} = \{$Q-$q_i\}$ $q_i \in Q$ and $q_i \neq q_0$ and $q_i \neq q_f$
Transitions:

[1] q_a and q_b may be same

GNFA with K states to GNFA with $K-1$ states
$\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$

$\mathcal{M}' = (Q', \Sigma, \delta', q_0, F)$
where
$Q' = \{Q\text{-}q_i\}$ $q_i \in Q$ and $q_i \neq q_0$ and $q_i \neq q_f$
Transitions:

1. For any two states q_a and $q_b \in Q'$[1] with transitions
$\delta(q_a, r_1) = q_i$, $\delta(q_i, r_2) = q_i$ and $\delta(q_i, r_3) = q_b$ and $\delta(q_a, r_4) = q_b$
Add an edge from q_a to q_b with the regular expression $r_1 r_2^* r_3 + r_4$

[1] q_a and q_b may be same

Generalized NFA

GNFA with K states to GNFA with $K-1$ states
$\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$

$\mathcal{M}' = (Q', \Sigma, \delta', q_0, F)$
where
$Q' = \{Q\text{-}q_i\}$ $q_i \in Q$ and $q_i \neq q_0$ and $q_i \neq q_f$
Transitions:

1. For any two states q_a and $q_b \in Q'$[1] with transitions $\delta(q_a, r_1) = q_i$, $\delta(q_i, r_2) = q_i$ and $\delta(q_i, r_3) = q_b$ and $\delta(q_a, r_4) = q_b$
   Add an edge from q_a to q_b with the regular expression $r_1 r_2^* r_3 + r_4$
2. For cases q_a and $q_b \in Q'$ where $\nexists$ a path from q_a to q_b through q_i but there is a direct edge with label r, retain that label.

[1] q_a and q_b may be same

Generalized NFA

Correctness Claim: Has two parts

Correctness Claim: Has two parts

Part 1) If $w \in L(M) \Rightarrow w \in L(M^{'})$

Correctness Claim: Has two parts

Part 1) If $w \in L(M) \Rightarrow w \in L(M^{'})$

Let $w = a_1 a_2, \cdots, a_n \in L(M) \Rightarrow$

Generalized NFA

Correctness Claim: Has two parts
Part 1) If $w \in L(M) \Rightarrow w \in L(M^{'})$
Let $w = a_1 a_2, \cdots, a_n \in L(M) \Rightarrow$
$\delta^*(q_0, w) = q_f$

Generalized NFA

Correctness Claim: Has two parts

Part 1) If $w \in L(M) \Rightarrow w \in L(M^{'})$

Let $w = a_1 a_2, \cdots, a_n \in L(M) \Rightarrow$

$\delta^*(q_0, w) = q_f$

$=\delta^*(q_0, a_1 a_2, \cdots, a_n) = q_f$

Generalized NFA

Correctness Claim: Has two parts

Part 1) If $w \in L(M) \Rightarrow w \in L(M')$

Let $w = a_1 a_2, \cdots, a_n \in L(M) \Rightarrow$

$\delta^*(q_0, w) = q_f$

$=\delta^*(q_0, a_1 a_2, \cdots, a_n) = q_f$

$=\delta^*(\delta(q_0, a_1), a_2, \cdots, a_n) = q_f$

Generalized NFA

Correctness Claim: Has two parts

Part 1) If $w \in L(M) \Rightarrow w \in L(M^{'})$

Let $w = a_1 a_2, \cdots, a_n \in L(M) \Rightarrow$

$\delta^*(q_0, w) = q_f$

$=\delta^*(q_0, a_1 a_2, \cdots, a_n) = q_f$

$=\delta^*(\delta(q_0, a_1), a_2, \cdots, a_n) = q_f$

$=\delta^*(q_l, a_2, \cdots, a_n) = q_f$

Generalized NFA

Correctness Claim: Has two parts
Part 1) If $w \in L(M) \Rightarrow w \in L(M^{'})$
Let $w = a_1 a_2, \cdots, a_n \in L(M) \Rightarrow$
$\delta^*(q_0, w) = q_f$
$=\delta^*(q_0, a_1 a_2, \cdots, a_n) = q_f$
$=\delta^*(\delta(q_0, a_1), a_2, \cdots, a_n) = q_f$
$=\delta^*(q_l, a_2, \cdots, a_n) = q_f$
$=\delta^*(\delta(q_l, a_2), a_3, \cdots, a_n) = q_f$

Correctness Claim: Has two parts

Part 1) If $w \in L(M) \Rightarrow w \in L(M^{'})$

Let $w = a_1 a_2, \cdots, a_n \in L(M) \Rightarrow$

$\delta^*(q_0, w) = q_f$

$=\delta^*(q_0, a_1 a_2, \cdots, a_n) = q_f$

$=\delta^*(\delta(q_0, a_1), a_2, \cdots, a_n) = q_f$

$=\delta^*(q_l, a_2, \cdots, a_n) = q_f$

$=\delta^*(\delta(q_l, a_2), a_3, \cdots, a_n) = q_f$

.

.

.

$= \delta(q_k, a_n) = q_f$

Generalized NFA

The sequence of states visited are $q_0, q_1, \cdots, q_f$

Generalized NFA

The sequence of states visited are $q_0, q_1, \cdots, q_f$
Two cases

Generalized NFA

The sequence of states visited are $q_0, q_1, \cdots, q_f$
Two cases

1. $\delta^*(q_0, w) = q_0, q_1, \cdots q_f$ where $q_y \neq q_i$. $\Rightarrow$

Generalized NFA

The sequence of states visited are $q_0, q_1, \cdots, q_f$
Two cases

1. $\delta^*(q_0, w) = q_0, q_1, \cdots q_f$ where $q_y \neq q_i$. $\Rightarrow$
$\delta^{'*}(q_0, w) = q_0, q_1, \cdots q_f$ where $q_y \neq q_i$

Generalized NFA

The sequence of states visited are $q_0, q_1, \cdots, q_f$

Two cases

1. $\delta^*(q_0, w) = q_0, q_1, \cdots q_f$ where $q_y \neq q_i$. $\Rightarrow$
   $\delta'^*(q_0, w) = q_0, q_1, \cdots q_f$ where $q_y \neq q_i$
2. $\delta^*(q_0, w) = q_0, \cdots, q_j, q_i, \cdots, q_i, q_m, \cdots, q_f$ After reading $a_1, \cdots, a_{i-1}, a_i, \cdots, a_{i+1}, \cdots a_n$

Generalized NFA

The sequence of states visited are $q_0, q_1, \cdots, q_f$
Two cases

1. $\delta^*(q_0, w) = q_0, q_1, \cdots q_f$ where $q_y \neq q_i$. $\Rightarrow$
   $\delta'^*(q_0, w) = q_0, q_1, \cdots q_f$ where $q_y \neq q_i$
2. $\delta^*(q_0, w) = q_0, \cdots, q_j, q_i, \cdots, q_i, q_m, \cdots, q_f$ After reading
   $a_1, \cdots, a_{i-1}, a_i, \cdots, a_{i+1}, \cdots a_n$
   $\Rightarrow a_i \in r_{i-1}$ and $a_{i+1} \in r_{i+1}$

Generalized NFA

The sequence of states visited are $q_0, q_1, \cdots, q_f$
Two cases

1. $\delta^*(q_0, w) = q_0, q_1, \cdots q_f$ where $q_y \neq q_i$. $\Rightarrow$
   $\delta^{'*}(q_0, w) = q_0, q_1, \cdots q_f$ where $q_y \neq q_i$
2. $\delta^*(q_0, w) = q_0, \cdots, q_j, q_i, \cdots, q_i, q_m, \cdots, q_f$ After reading $a_1, \cdots, a_{i-1}, a_i, \cdots, a_{i+1}, \cdots a_n$
   $\Rightarrow a_i \in r_{i-1}$ and $a_{i+1} \in r_{i+1}$
   which is nothing but $r_{i-1} r_i^* r_{i+1}$

Generalized NFA

The sequence of states visited are $q_0, q_1, \cdots, q_f$

Two cases

1. $\delta^*(q_0, w) = q_0, q_1, \cdots q_f$ where $q_y \neq q_i$. $\Rightarrow$
   $\delta^{'*}(q_0, w) = q_0, q_1, \cdots q_f$ where $q_y \neq q_i$
2. $\delta^*(q_0, w) = q_0, \cdots, q_j, q_i, \cdots, q_i, q_m, \cdots, q_f$ After reading
   $a_1, \cdots, a_{i-1}, a_i, \cdots, a_{i+1}, \cdots a_n$
   $\Rightarrow a_i \in r_{i-1}$ and $a_{i+1} \in r_{i+1}$
   which is nothing but $r_{i-1} r_i^* r_{i+1}$
   $\Rightarrow \delta^{'*}(q_0, w) = q_f$

Generalized NFA

The sequence of states visited are $q_0, q_1, \cdots, q_f$
Two cases

1. $\delta^*(q_0, w) = q_0, q_1, \cdots q_f$ where $q_y \neq q_i$. $\Rightarrow$
   $\delta^{'*}(q_0, w) = q_0, q_1, \cdots q_f$ where $q_y \neq q_i$
2. $\delta^*(q_0, w) = q_0, \cdots, q_j, q_i, \cdots, q_i, q_m, \cdots, q_f$ After reading
   $a_1, \cdots, a_{i-1}, a_i, \cdots, a_{i+1}, \cdots a_n$
   $\Rightarrow a_i \in r_{i-1}$ and $a_{i+1} \in r_{i+1}$
   which is nothing but $r_{i-1} r_i^* r_{i+1}$
   $\Rightarrow \delta^{'*}(q_0, w) = q_f$

Part 2: If $w \in L(M') \Rightarrow w \in L(M)$

Part 2: If $w \in L(M^{'}) \Rightarrow w \in L(M)$
Take it as exercise !

Reular Languages

Theorem

For any regular expression r with $L(r)$, $\exists$ *an NFA M such that* $L(M) = L(r)$.

Theorem

For any NFA M with $L = L(M)$, $\exists$ *a regular expression r such that* $L(r) = L(M)$

Theorem

For any NFA M with $L = L(M)$, $\exists$ *a DFA* M' *such that* $L(M) = L(M')$