1 Overview of NLP content

1.1 Finite State Automata and Regular Expressions

- A finite state automaton (FSA) is a directed graph with a finite number of nodes.
- A FSA is described by a 5 tuple: (states, alphabet, initial state, final state, transition)

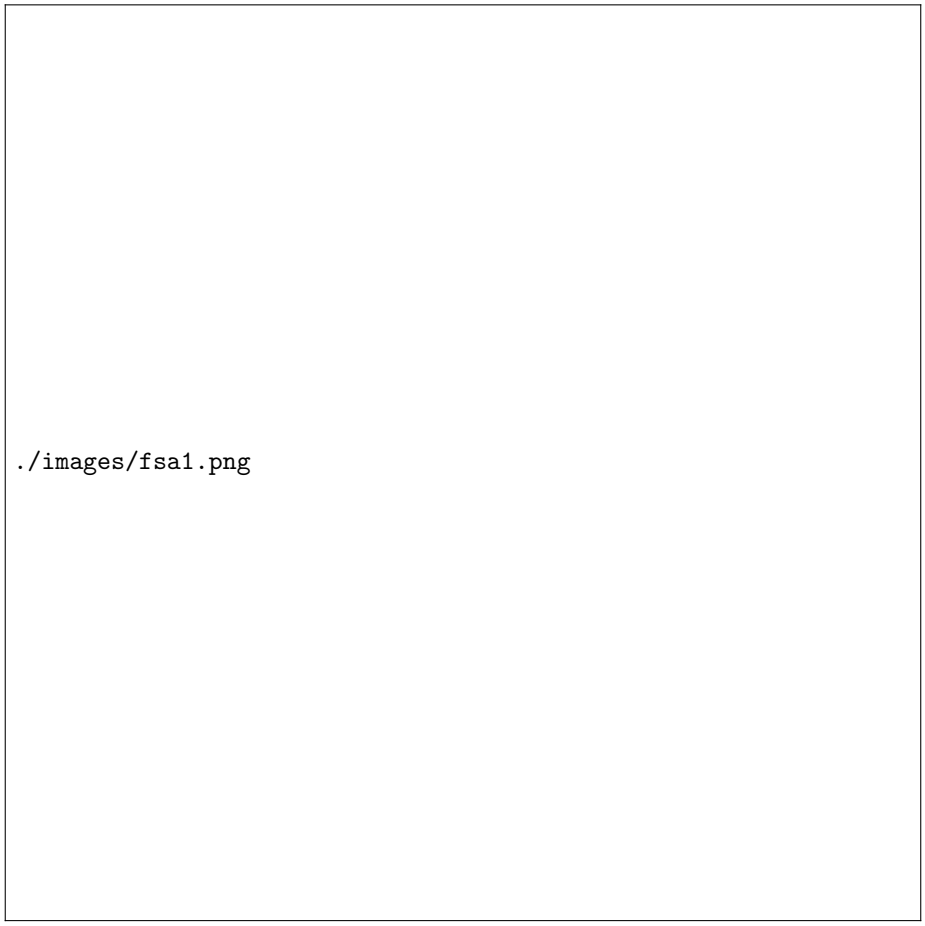

Figure **??**: Example FSA representation

- A deterministic FSA is one whose behaviour is fully determined by the state it is in and the input

- a non-deterministic FSA has an element of stochasticity; perhaps two paths for the same input, or a ϵ (i.e. spontaneous) transition
- NDFSA's present challenges when determinig whether a string should be accepted (by the language) as 'the wrong path' may be taken. A solution to this is to use a backtrack algorithm.
- Any NDFSA can be converted to a DFSA (See: parallel algorithm)

Regular expressions are a powerful tool for pattern matching. They are one way to define an FSA, and also to define a formal (specifically regular) language. Any regular expression can be defined as an NDFSA and hence a DFSA

A formal language is a set of strings that are composed entirely from a finite symbol set Σ

1.2 Finite State Transducers

a FST is a more general function than an FSA. It 'reads' one string and generates another