

Regular Languages : DFA, NFA and epsilon-NFA.

These machines can be imagined to have a read-only tape holding the input. A tape-head can only move forward reading the current symbol from the tape. The current state can be remembered in the tape-head. For epsilon move the head does not move – only the state changes.

Context-free languages : PDA

PDA has two tapes. One tape for input just like epsilon-NFA and another tape as a stack – read/write can be done at one end only.

Context-sensitive Languages : LBA (Linear Bounded Automaton) which is a special case of a Turing Machine. A Turing Machine has a single tape with a fixed left end. The tape head can move in either direction and in one step reads a symbol from the tape and depending on this and the current state, changes state, writes a symbol back at the

same position and moves either to the left or right. Only at the left end it is not allowed to move left. The tape can be unbounded in the right. In a LBA only  $O(n)$  positions from the tape is used. TM's and LBA's can be deterministic or non-deterministic. A language is context-sensitive iff it is accepted by a non-deterministic LBA.

The notion of Halting in a TM : A TM has only two final states accept and reject. A TM is said to halt on an input if it accepts or rejects in a finite number of steps. A general TM has the possibility of not halting on some inputs. This leads to the notion of undecidable problems or languages and we get a precise definition of algorithms as TM's which halt on every input.