Aho- Corasick Algorithm: The aho- Corasick algorithm is a string searching algorithm. It is a kind of dictionary matching algorithm that locates elements of a finite set of strings within an input text. It matches all patterns simultaneously.

The data structure has one node for every prefix of every string in the dictionary. So if (bca) is in the dictionary, then there will be nodes for (bca), (bc), (b), and (). There is a black directed "child" arc from each node to a node whose name is found by appending one character. So there is a black arc from (bc) to (bca). There is a blue directed "suffix" arc from each node to the node that is the longest possible strict suffix of it in the graph.For example, there is a green arc from (bca) to (a) because (a) is the first node in the dictionary (i.e. a white node) that is reached when following the blue arcs to (ca) and then on to (a).

Dictionary {a, ab, bc, bca, c}

Path	In dictionary	Suffix link	Dict suffix link	
()	-			
(a)	+	()		
(ab)	+	(b)		
(b)	-	()		
(bc)	+	(c)	(c)	
(bca)	+	(ca)	(a)	
(c)	+	()		

At each step, the current node is extended by finding its child, and if that doesn't exist, finding its suffix's child, and if that doesn't work, finding its suffix's, suffix's child, and so on, finally ending in the root node if nothing's seen before.When the algorithm reaches a node, it outputs all the dictionary entries that end at the current character position in the input text. This is done by printing every node reached by following the dictionary suffix links, starting from that node, and continuing until it reaches a node with no dictionary suffix link. In addition, the node itself is printed, if it is a dictionary entry.