

Syntactic Pattern Recognition

Syntactic Pattern Recognition

- **Statistical pattern recognition** attempts to classify patterns based on a set of extracted features and an underlying statistical model for the generation of these patterns. Ideally, this is achieved with a rather straightforward method.
 - determine the feature vector,
 - train the system,
 - classify the patterns
- Patterns that include structural or relational information are difficult to represent as feature vectors.
- Syntactic pattern recognition uses this structural information for classification and description.

Syntactic Pattern Recognition

- Grammars can be used to create a definition of the structure of each pattern class.
- Then, recognition and classification are done using either:
 - Parsing using formal grammars (called Syntactic Pattern Recognition)
 - or
 - Relational graph matching(called Hierarchical approach)

Classification

- Classification can be done based on a measure of structural similarity in patterns.
- Each pattern class can be represented by a structural representation or description.

Description

- A description of the pattern structure is useful for recognizing entities when a simple classification isn't possible.
- Can also describe aspects that cause a pattern not to be assigned to a particular class.
- In complex cases, recognition can only be achieved through a description for each pattern rather than through classification.

Definitions

- The simplest sub-patterns are called pattern primitives, and should be much easier to recognize than the overall patterns.
- The language used to describe the structure of the patterns in terms of sets of pattern primitives is called the pattern description language.
- The pattern description language will have a grammar that specifies how primitives can be composed into patterns

Syntax Analysis

- When a primitive within the pattern is identified, syntax analysis (parsing) is performed on the sentence describing the pattern to determine if it is correct with respect to the grammar.
- Syntax analysis also gives a structural description of the sentence associated with the pattern.
- One advantage of this approach is that a grammar (rewriting) rule can be applied many times.

Pattern Description Using Grammar

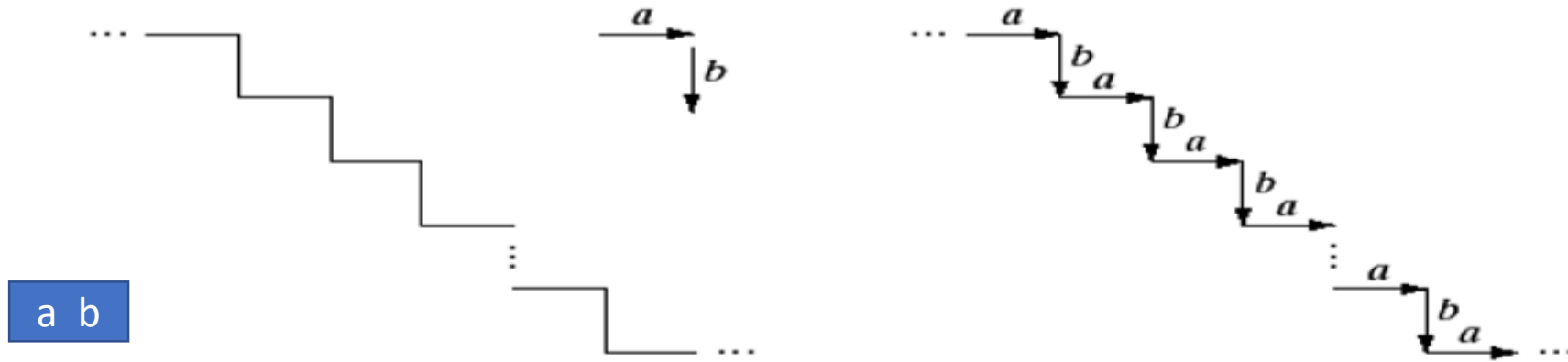


Fig 1: (a) Staircase structure (b) structure encoded in terms of primitive a and b to yield the description ...ababab...

Pattern Description Using Grammar

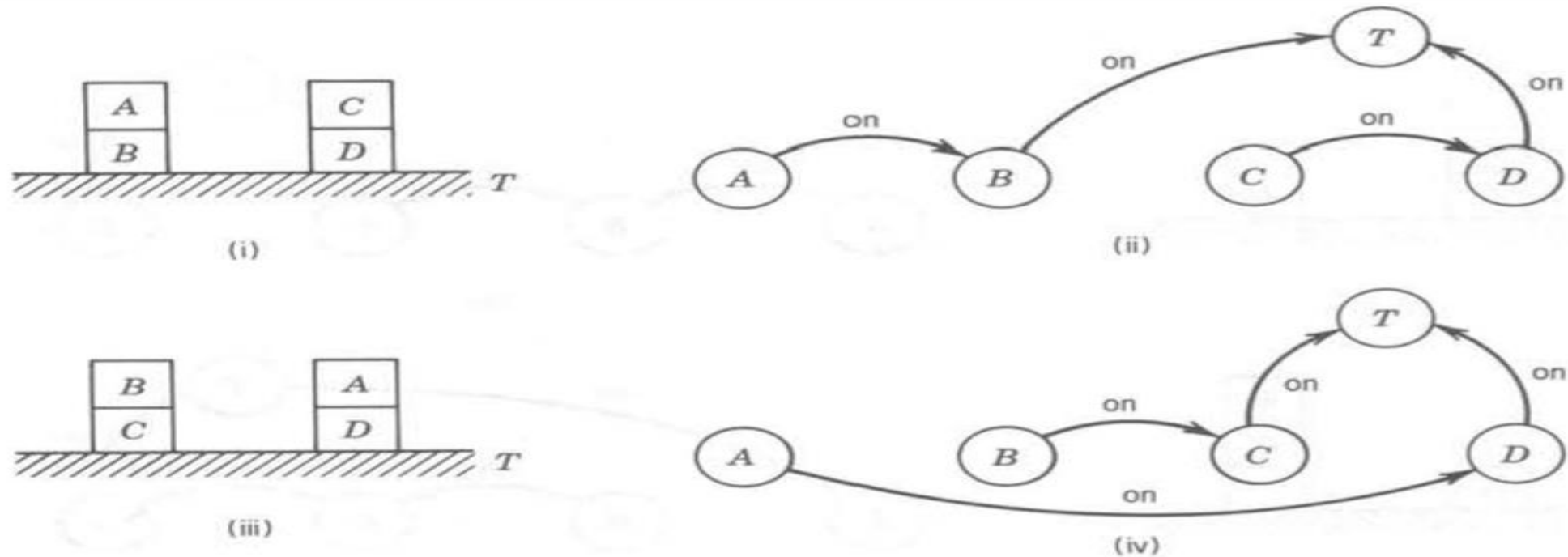


Fig 2: A grammar describing four blocks arranged in 2-block stacks. (i) An example. (ii) Graphical description corresponding to (i). (iii) Another example. (iv) Graphical description corresponding to (iii).

Pattern Description Using Grammar

➤ A grammar describing four blocks arranged in 2-block stacks:

➤ $V_T = \{ \text{table}, \text{block}, +, \uparrow \}$ (terminal symbols)

$V_N = \{ \text{DESK}, \text{LEFT STACK}, \text{RIGHT STACK} \}$

(non-terminal symbols)

$S = \text{DESK} \in V_N$ (root symbol)

$P = \{ \text{DESK} \rightarrow \text{LEFT STACK} + \text{RIGHT STACK}$

$\text{DESK} \rightarrow \text{RIGHT STACK} + \text{LEFT STACK}$

$\text{LEFT STACK} \rightarrow \text{block} \uparrow \text{block} \uparrow \text{table}$

$\text{RIGHT STACK} \rightarrow \text{block} \uparrow \text{block} \uparrow \text{table} \}$

(production rules)

Pattern Description Using Grammar

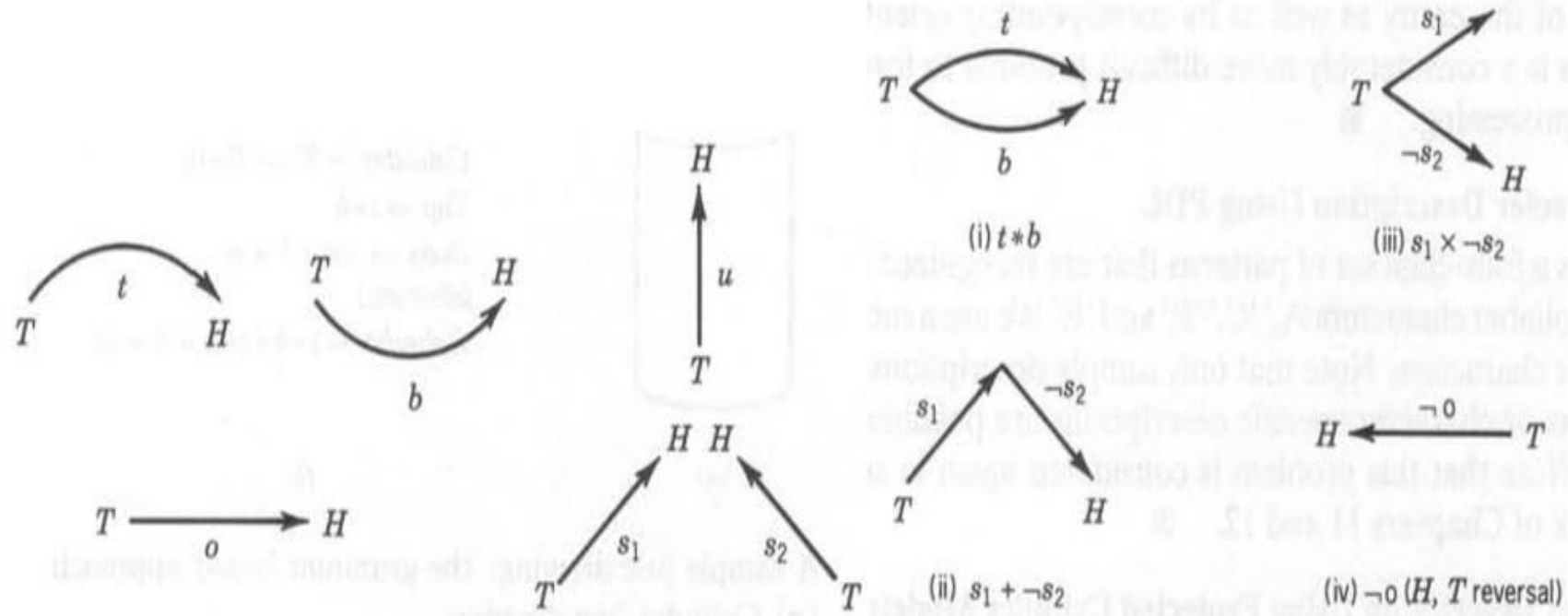


Fig 3: A 2-D line drawing picture description grammar

A set of terminal symbols $\{ t, b, u, o, s, *, -, + \}$ where

- $+$ represents head to tail concatenation,
- $*$ represents head-head and tail-tail attachment,
- $-$ represents the head-tail reversal. (x should be $*$ in (iii))

Pattern Description Using Grammar

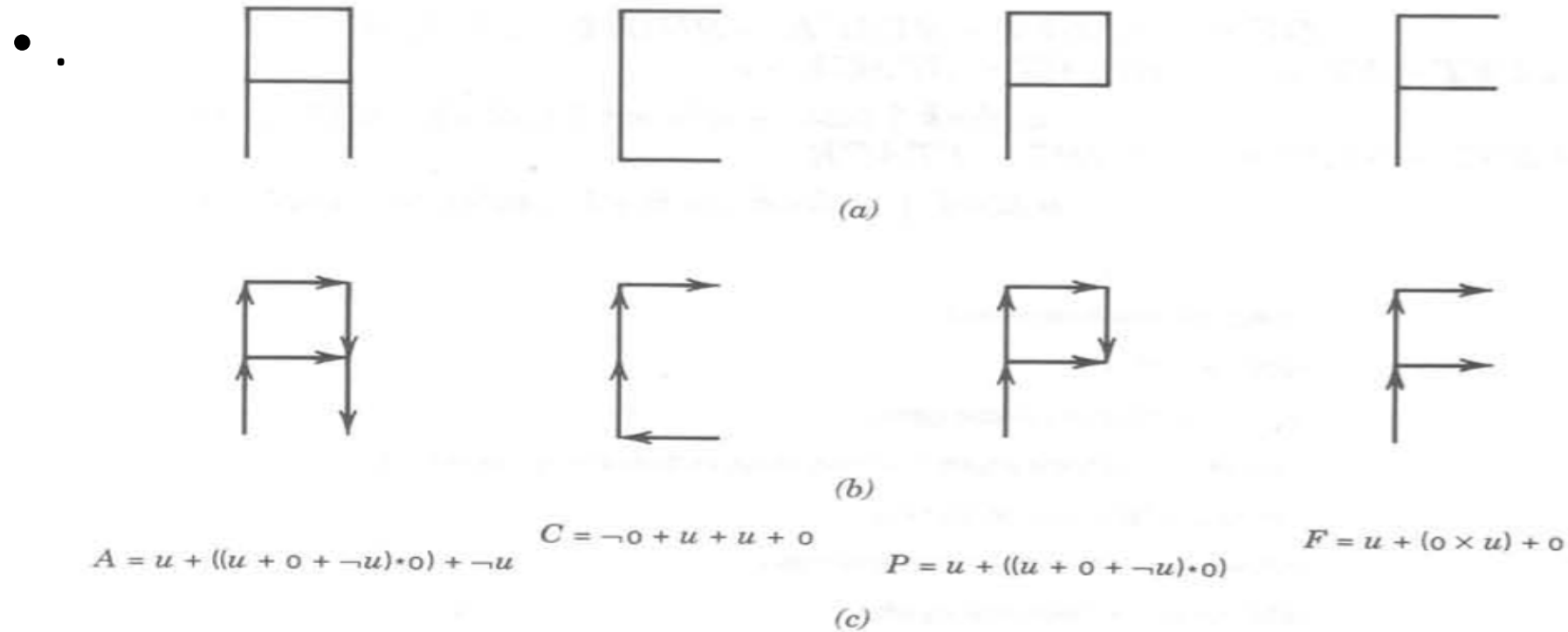
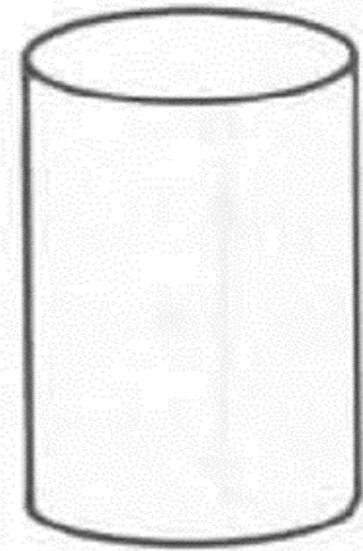


Fig 4: Representation of four characters using the line drawing picture description grammar
 (a) Pattern data. (b) Primitive representation and interconnection. (c) Corresponding descriptions

Pattern Description Using Grammar



Cylinder \rightarrow *Top* * *Body*

Top \rightarrow $t * b$

Body $\rightarrow \neg u + b + u$

(alternate:)

Cylinder $\rightarrow t * b * (\neg u + b + u)$

Fig 5: Representation of a cylinder using the line drawing picture description grammar

When to Use It

- Picture recognition and scene analysis are problems in which there are a large number of features and the patterns are complex.
 - For example, recognizing areas such as highways, rivers, and bridges in satellite pictures.
- In this case, a complex pattern can be described in terms of a hierarchical composition of simpler sub patterns.

Syntactic System

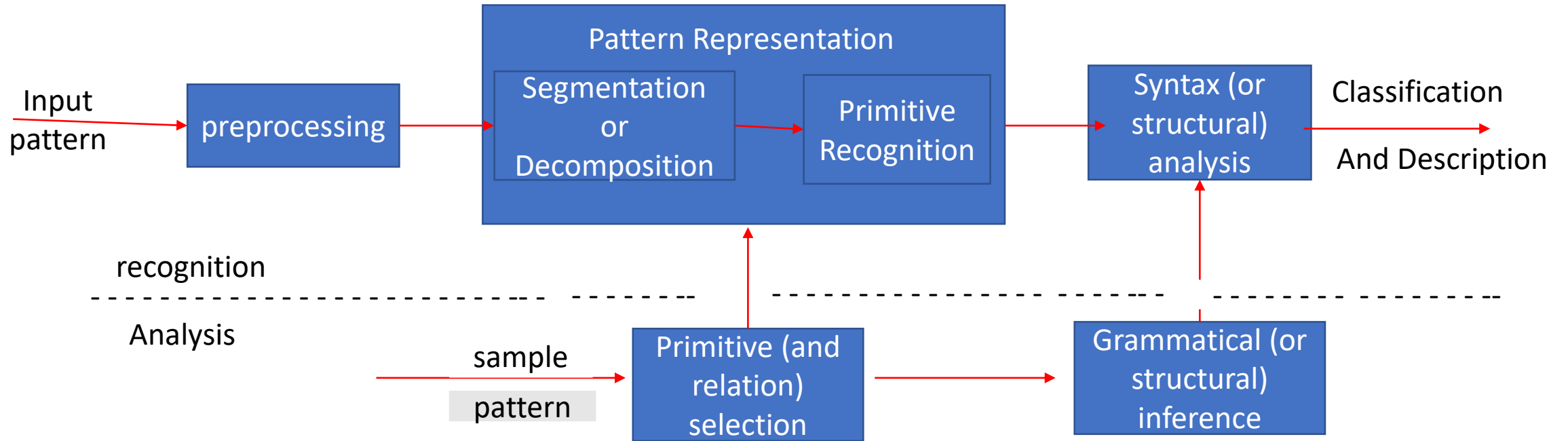


Fig 6 :Block Diagram of a syntactic pattern recognition system

Syntactic System

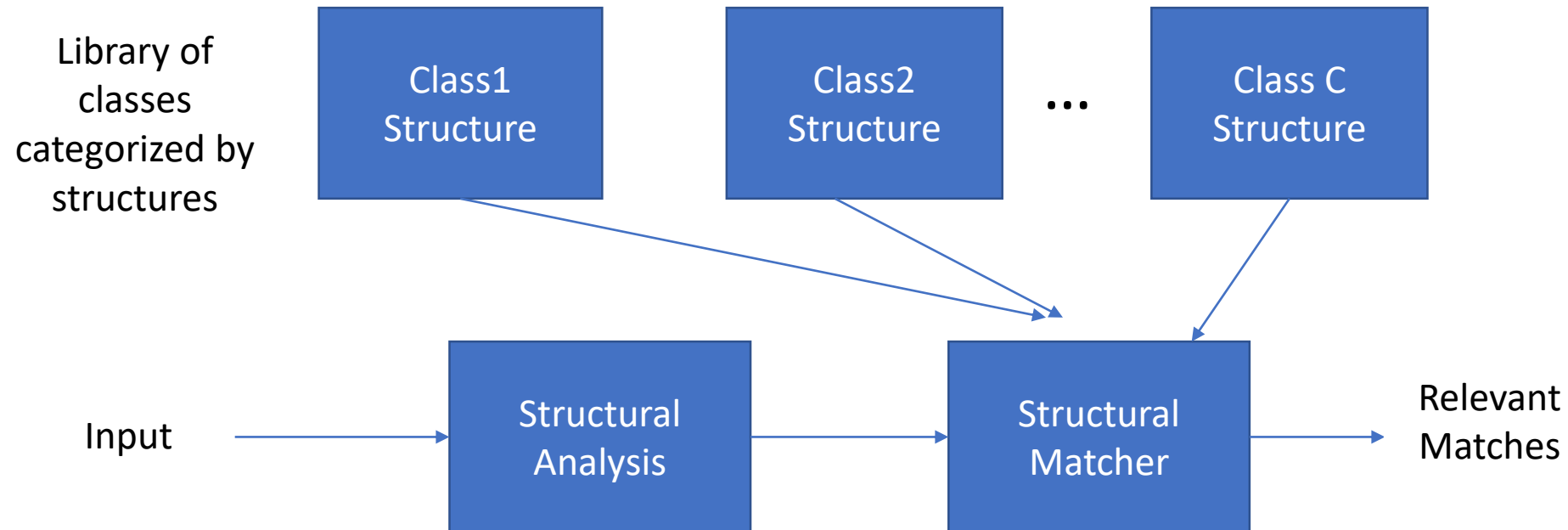


Fig 7: Using Syntactic Patter Recognition for Classification

Syntactic System

- Consists of two main parts:
 - Analysis -primitive selection and grammatical or structural inference
 - Recognition -preprocessing, segmentation or decomposition, primitive and relation recognition, and syntax analysis.
- Preprocessing includes the tasks of pattern encoding and approximation, filtering, restoration, and enhancement.
- After preprocessing, the pattern is segmented into sub-patterns and primitives using predefined operations.
- Sub-patterns are identified with a given set of primitives, so each pattern is represented by a set of primitives with the specified syntactic operations.

Syntax Parsing

- For example, using the concatenation operation, each pattern is recognized by a string of concatenated primitives.
- At this point, the parser will determine if the pattern is syntactically correct.
 - It belongs to the class of patterns described by the grammar if it is correct.
- During parsing/syntax analysis, a description is produced in terms of a parse tree, assuming the pattern is syntactically correct.
- If it isn't correct, it will either be rejected or analyzed based on a different grammar, which could represent other possible pattern classes.

Matching

- The simplest form of recognition is template matching, in which a string of primitives representing an input pattern is compared to strings of primitives representing reference patterns.
- The input pattern is classified in the same class as the prototype that is the best match, which is determined by a similarity criterion.

Matching vs. Complete Parsing

- In this case, the structural description is ignored.
- The opposite approach is a complete parsing that uses the entire structural description.
- There are many intermediate approaches; for example, a series of tests designed to test the occurrence of certain primitives, sub-patterns, or combinations of these. The result of these tests will determine a classification

- Parsing is required if the problem necessitates using a complete pattern description for recognition.
- Efficiency of the recognition process is improved by simpler approaches that do not require a complete parsing.
- Basically, parsing can be expensive, so don't use it unnecessarily.

Inferring Grammars

- Grammatical inference machine similar to “learning” in the discriminant approach; it infers a grammar from a set of training patterns.
- The inferred grammar can then be used for pattern description and syntax analysis.