Artificial Intelligence & Expert System

UNIT IV

Natural Language Processing (NLP) & Planning

Overview of NLP

Natural Language Processing (NLP) is the process of computer analysis of input provided in a human language (natural language), and conversion of this input into a useful form of representation.

The field of NLP is primarily concerned with getting computers to perform useful and interesting tasks with human languages.

The field of NLP is secondarily concerned with helping us come to a better understanding of human language.

Overview of NLP

Applications of Natural Language Processing:

- Machine Translation – Translation between two natural languages.
- Information Retrieval – Web search (uni-lingual or multi-lingual).
- Query Answering/Dialogue – Natural language interface with a database system, or a dialogue system.
- Report Generation – such as weather reports.
- Some Small Applications – Grammar Checking, Spell Checking, Spell Corrector.

Overview of NLP

The input/output of a NLP system can be:

- written text
- speech

We will mostly be concerned with written text (not speech).

To process written text, we need:

- lexical, syntactic, semantic knowledge about the language.
- discourse information, real world knowledge

Overview of NLP

There are two components of NLP:

1. Natural Language Understanding
2. Natural Language Generation

1. Natural Language Understanding:

– Mapping the given input in the natural language into a useful representation.

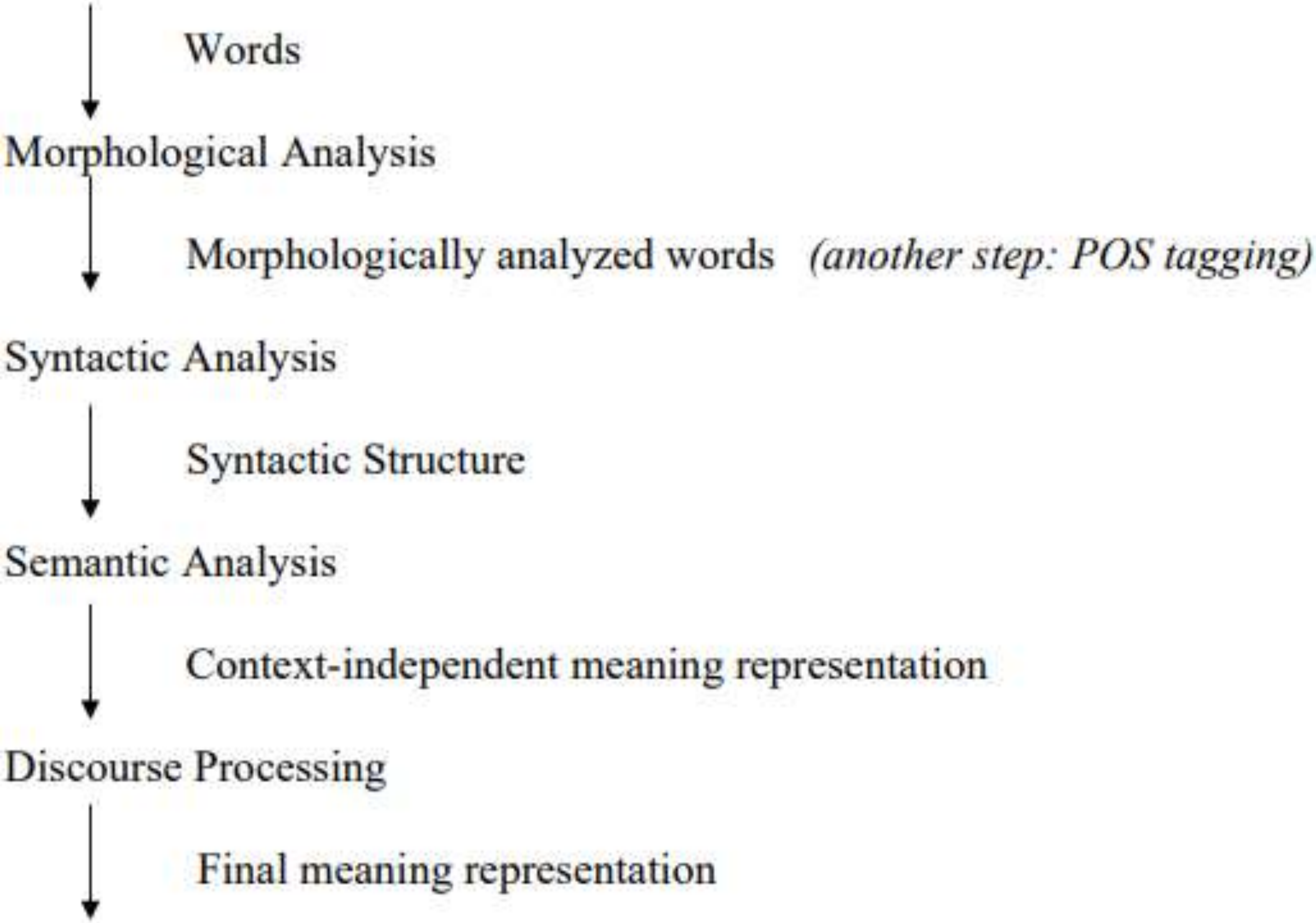
– Different level of analysis required:

morphological analysis, syntactic analysis,

semantic analysis, discourse analysis

Overview of NLP

The steps in natural language understanding are as follows:



Words

Morphological Analysis

Morphologically analyzed words *(another step: POS tagging)*

Syntactic Analysis

Syntactic Structure

Semantic Analysis

Context-independent meaning representation

Discourse Processing

Final meaning representation

Overview of NLP

2. Natural Language Generation:

  – Producing output in the natural language from some internal representation.

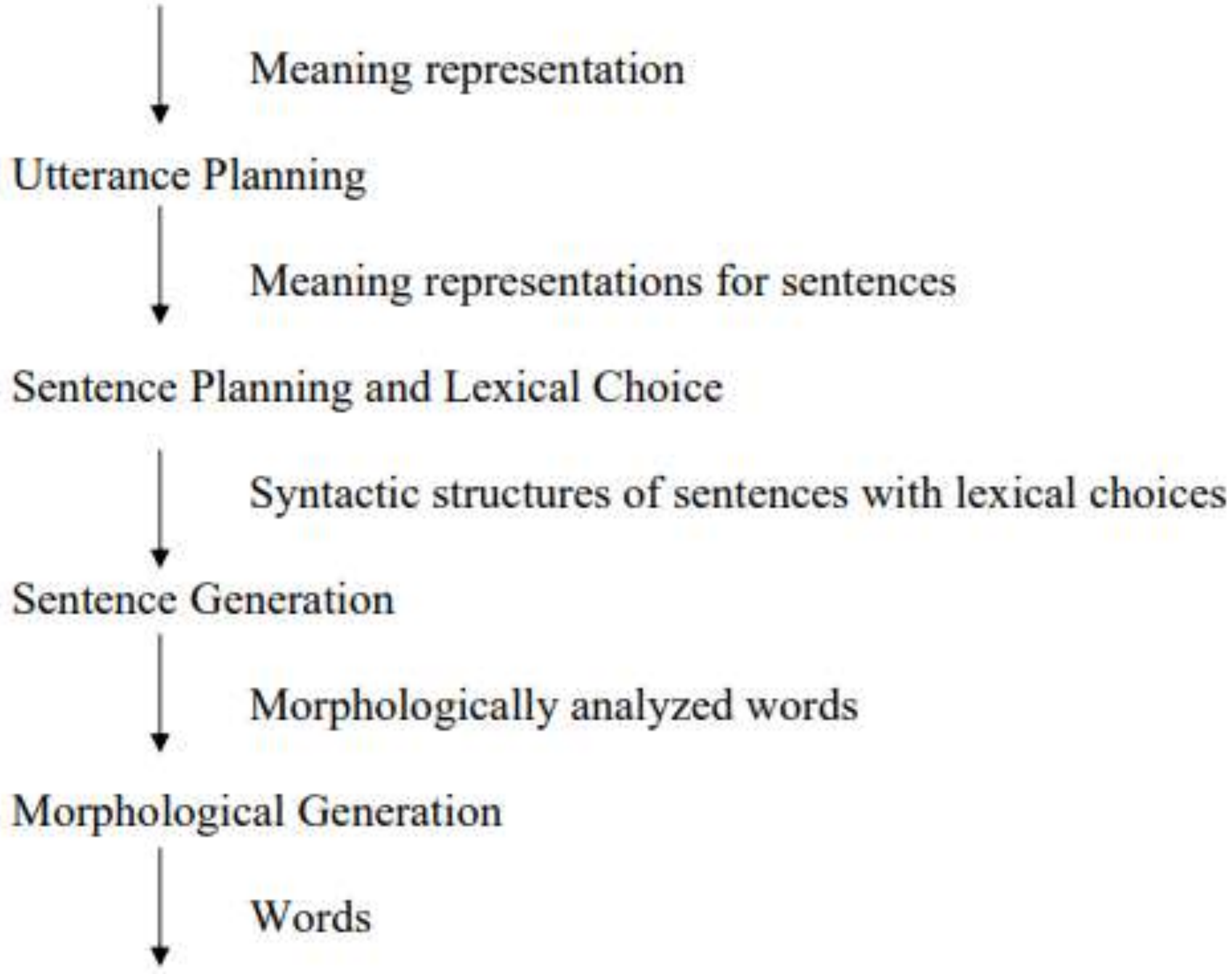
  – Different level of synthesis required:

    - deep planning
    - syntactic generation.

NL Understanding is much harder than NL Generation. But, still both of them are hard.

Overview of NLP

The steps in natural language generation are as follows:



Meaning representation

Utterance Planning

Meaning representations for sentences

Sentence Planning and Lexical Choice

Syntactic structures of sentences with lexical choices

Sentence Generation

Morphologically analyzed words

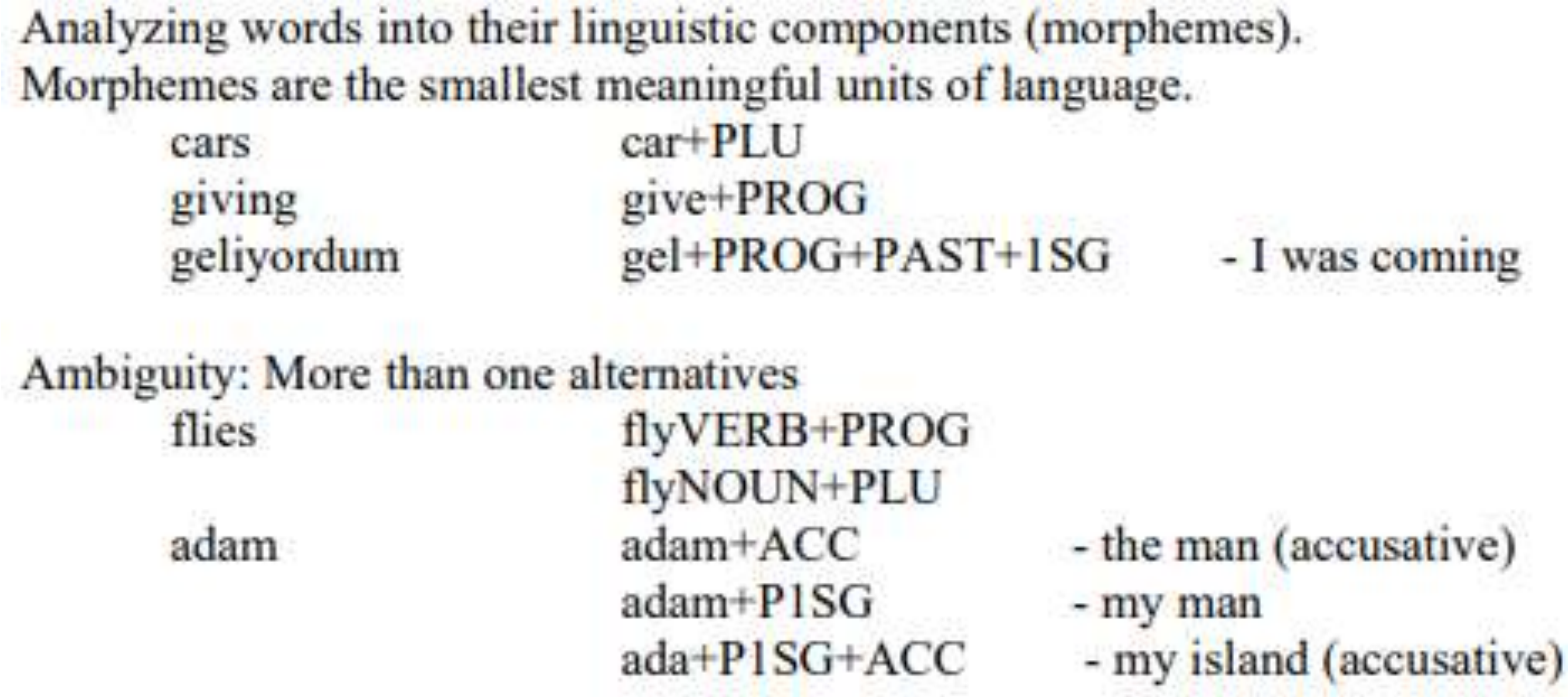
Morphological Generation

Words

Overview of NLP

Steps in Language Understanding and Generation:

1. Morphological Analysis:



Analyzing words into their linguistic components (morphemes).
Morphemes are the smallest meaningful units of language.

cars	car+PLU	
giving	give+PROG	
geliyordum	gel+PROG+PAST+1SG	- I was coming

Ambiguity: More than one alternatives

flies	flyVERB+PROG	
	flyNOUN+PLU	
adam	adam+ACC	- the man (accusative)
	adam+P1SG	- my man
	ada+P1SG+ACC	- my island (accusative)

Overview of NLP

Steps in Language Understanding and Generation:

2. Parts-of-Speech (POS) Tagging:

Part-of-speech tag of a word is one of major word groups (or its subgroups).

- – open classes -- noun, verb, adjective, adverb
- – closed classes -- prepositions, determiners, conjunctions, pronouns, participles

duck is a verb or noun? (morphological analyzer cannot make decision).

A POS tagger may make that decision by looking the surrounding words.

Overview of NLP

Steps in Language Understanding and Generation:

3. Lexical Processing:

- The purpose of lexical processing is to determine meanings of individual words.
- Basic methods is to lookup in a database of meanings – lexicon
- We should also identify non-words such as punctuation marks.
- Word-level ambiguity -- words may have several meanings, and the correct one cannot be chosen based solely on the word itself. – bank in English

Overview of NLP

Steps in Language Understanding and Generation:

3. Lexical Processing (continued):

Solution -- resolve the ambiguity on the spot by POS tagging (if possible) or pass on the ambiguity to the other levels.

4. Syntactic Processing:

Parsing -- converting a flat input sentence into a hierarchical structure that corresponds to the units of meaning in the sentence.

Overview of NLP

Steps in Language Understanding and Generation:

4. Syntactic Processing(continued):

There are different parsing formalisms and algorithms.

Most formalisms have two main components:

– grammar -- a declarative representation describing the syntactic structure of sentences in the language.

– parser -- an algorithm that analyzes the input and outputs its structural representation (its parse) consistent with the grammar specification.

Overview of NLP

Steps in Language Understanding and Generation:

5. Semantic Analysis:

- Assigning meanings to the structures created by syntactic analysis.

- Mapping words and structures to particular domain objects in way consistent with our knowledge of the world.

- Semantic can play an import role in selecting among competing syntactic analyses and discarding illogical analyses.

- Example: I robbed the bank -- bank is a river bank or a financial institution needs to be determined.

Overview of NLP

Models to represent Linguistic Knowledge:

- State Machines -- FSAs, FSTs, HMMs, ATNs, RTNs
- Formal Rule Systems -- Context Free Grammars, Unification Grammars, Probabilistic CFGs.
- Logic-based Formalisms -- first order predicate logic, some higher order logic.
- Models of Uncertainty -- Bayesian probability theory.

Overview of NLP

The following language related information are useful in NLP:

Phonology – concerns how words are related to the sounds that realize them.

Morphology – concerns how words are constructed from more basic meaning units called morphemes. A morpheme is the primitive unit of meaning in a language.

Syntax – concerns how can be put together to form correct sentences and determines what structural role each word plays in the sentence and what phrases are subparts of other phrases.

Overview of NLP

Semantics – concerns what words mean and how these meaning combine in sentences to form sentence meaning. The study of context-independent meaning.

Pragmatics – concerns how sentences are used in different situations and how use affects the interpretation of the sentence.

Discourse – concerns how the immediately preceding sentences affect the interpretation of the next sentence. For example, interpreting pronouns and interpreting the temporal aspects of the information.

Overview of NLP

World Knowledge – includes general knowledge about the world. What each language user must know about the other’s beliefs and goals.

Ambiguity in NLP:

Ambiguity is an intrinsic characteristic of human conversations and one that is particularly challenging in natural language understanding(NLU).

Scenarios by ambiguity, are essentially referring to sentences that have multiple alternative interpretations.

Overview of NLP

Ambiguity in NLP:

Example: I made her duck

For this sentence, different interpretations possible are:

1. I cooked duck for her.
2. I cooked duck belonging to her.
3. I created a toy duck which she owns.
4. I caused her to quickly lower her head or body.
5. I used magic and turned her into a duck.

Overview of NLP

Ambiguities are resolved using the following methods.

- *part-of-speech tagging* -- Deciding whether duck is verb or noun.
- *word-sense disambiguation* -- Deciding whether make is create or cook.
- *lexical disambiguation* -- Resolution of part-of-speech and word-sense ambiguities are two important kinds of lexical disambiguation.
- *syntactic ambiguity* -- her duck is an example of syntactic ambiguity, and can be addressed by probabilistic parsing.

Overview of NLP

Knowledge Representation for NLP:

Which knowledge representation will be used depends on the application -- Machine Translation, Database Query System.

Common representational formalisms:

– first order predicate logic

– conceptual dependency graphs

– semantic networks

– Frame-based representations

Parsing

The word ‘Parsing’ whose origin is from Latin word ‘pars’ (which means ‘part’), is used to draw exact meaning or dictionary meaning from the text.

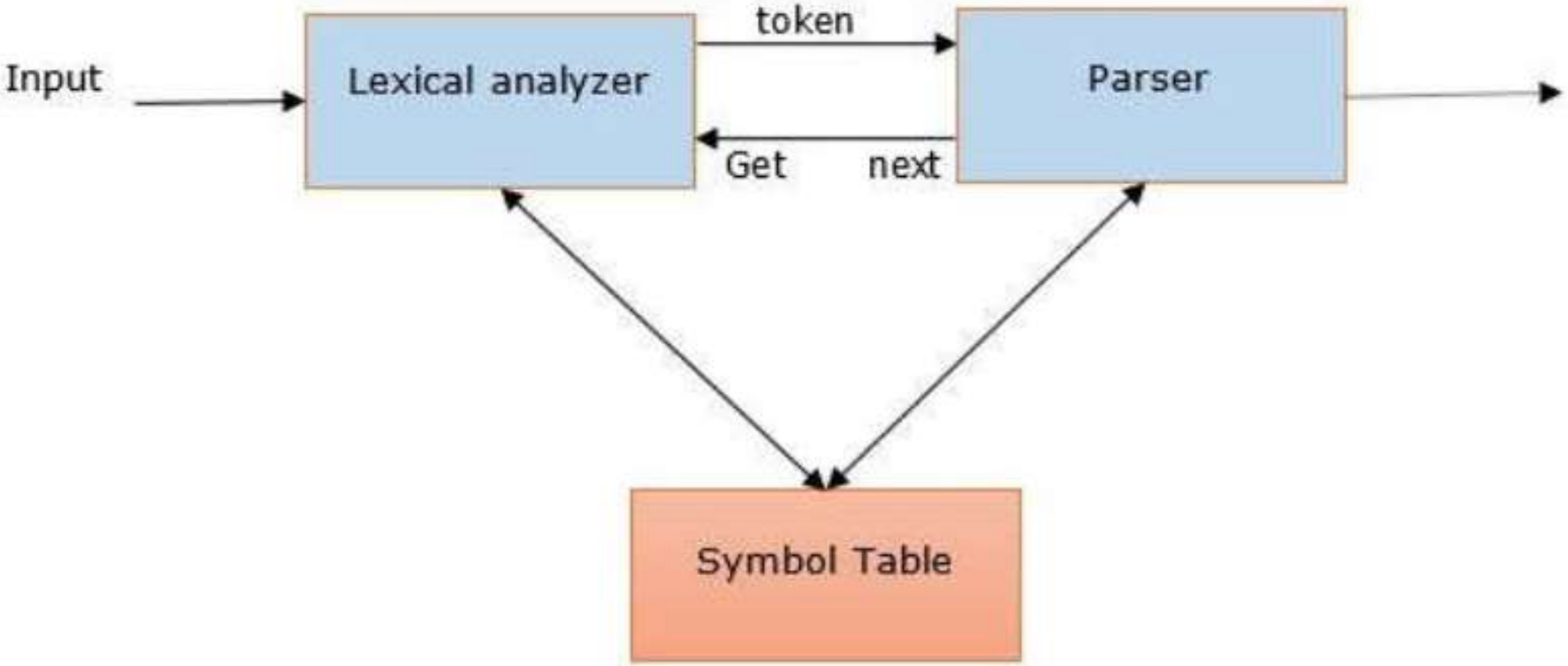
It is also called Syntactic analysis or syntax analysis.

Comparing the rules of formal grammar, syntax analysis checks the text for meaningfulness.

The sentence like “Give me hot ice-cream”, for example, would be rejected by parser or syntactic analyzer.

Parsing

It may be defined as the process of analyzing the strings of symbols in natural language conforming to the rules of formal grammar.



Parsing

We can understand the relevance of parsing in NLP with the help of following points:

1. Parser is used to report any syntax error.
2. It helps to recover from commonly occurring error so that the processing of the remainder of program can be continued.
3. Parse tree is created with the help of a parser.
4. Parser is used to create symbol table, which plays an important role in NLP.
5. Parser is also used to produce intermediate representations (IR).

Parsing: Deep Parsing

In deep parsing, the search strategy will give a complete syntactic structure to a sentence.

It is suitable for complex NLP applications.

Dialogue systems and summarization are the examples of NLP applications where deep parsing is used.

It is also called full parsing.

Parsing: Shallow Parsing

It is the task of parsing a limited part of the syntactic information from the given task.

It can be used for less complex NLP applications.

Information extraction and text mining are the examples of NLP applications where deep parsing is used.

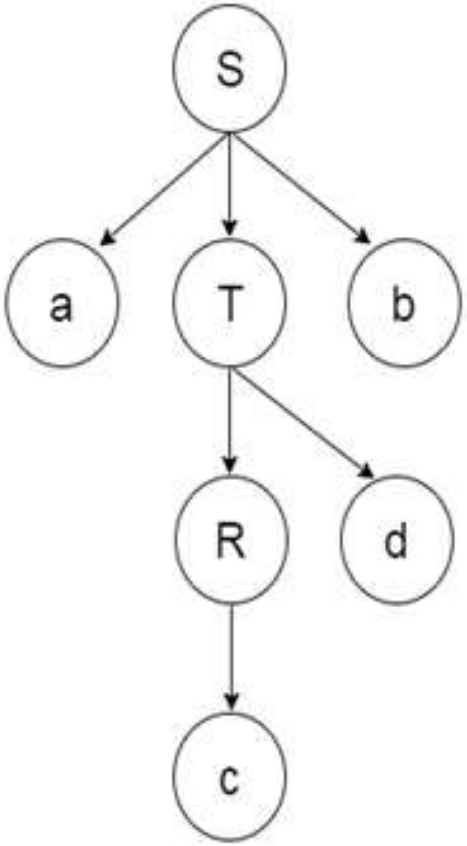
It is also called chunking.

Top down parsing

The top down parsing is known as recursive parsing or predictive parsing.

In the top down parsing, the parsing starts from the start symbol and transform it into the input symbol.

Parse Tree representation of input string "acdb" is as shown in diagram:



Bottom up parsing

Bottom up parsing is also known as shift-reduce parsing.

Bottom up parsing is used to construct a parse tree for an input string.

In the bottom up parsing, the parsing starts with the input symbol and construct the parse tree up to the start symbol by tracing out the rightmost derivations of string in reverse.

Bottom up parsing

Example:

Productions:

E → T

T → T * F

T → id

F → T

F → id

Parse Tree representation of input string "id * id" is as follows:

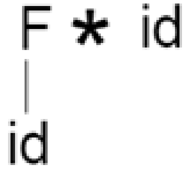
Step 1:

id * id

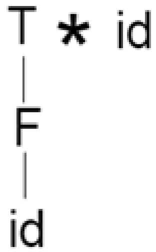
Bottom up parsing

Example (continued):

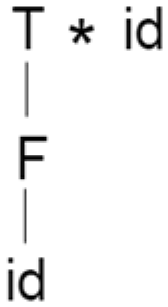
Step 2:



Step 3:



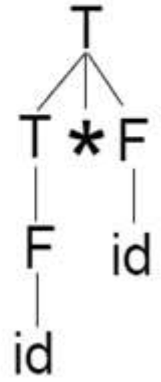
Step 4:



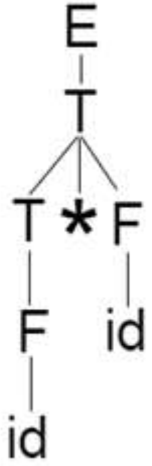
Bottom up parsing

Example (continued):

Step 5:



Step 6:



Types of parsers

Recursive descent parser

- Recursive descent parsing is one of the most straightforward forms of parsing.
- It follows a top down process.
- It attempts to verify that the syntax of the input stream is correct or not.
- It reads the input sentence from left to right.
- One necessary operation for recursive descent parser is to read characters from the input stream and matching them with the terminals from the grammar.

Types of parsers

Shift-reduce parser:

- It follows a simple bottom-up process.
- It tries to find a sequence of words and phrases that correspond to the right-hand side of a grammar production and replaces them with the left-hand side of the production.
- The above attempt to find a sequence of word continues until the whole sentence is reduced.
- In other simple words, shift-reduce parser starts with the input symbol and tries to construct the parser tree up to the start symbol.

Types of parsers

Chart parser:

- It is mainly useful or suitable for ambiguous grammars, including grammars of natural languages.
- It applies dynamic programing to the parsing problems.
- Because of dynamic programing, partial hypothesized results are stored in a structure called a ‘chart’.
- The ‘chart’ can also be re-used.

Types of parsers

Regexp parser:

- Regexp parsing is one of the mostly used parsing technique.
- As the name implies, it uses a regular expression defined in the form of grammar on top of a POS-tagged string.
- It basically uses these regular expressions to parse the input sentences and generate a parse tree out of this.

Machine Translation

Machine Translation (MT) is the task of automatically converting one natural language into another, preserving the meaning of the input text, and producing fluent text in the output language.

While machine translation is one of the oldest subfields of artificial intelligence research, the recent shift towards large-scale empirical techniques has led to very significant improvements in translation quality.

Machine Translation

Types of Machine Translation Systems:

There are different types of machine translation systems. Let us see what the different types are.

Bilingual MT System:

Bilingual MT systems produce translations between two particular languages.

Multilingual MT System:

Multilingual MT systems produce translations between any pair of languages. They may be either uni-directional or bi-directional in nature.

Rule-based Machine Translation

A rule-based system requires experts’ knowledge about the source and the target language to develop syntactic, semantic and morphological rules to achieve the translation.

Example:

SYSTRAN is one of the oldest Machine Translation company. It translates from and to around 20 languages.

SYSTRAN was used for the Apollo-Soyuz project (1973) and by the Europian Commission (1975).

SYSTRAN was used by Google’s language tools until 2007.

Rule-based Machine Translation

Advantages:

- No bilingual text required
- Domain-independent
- Total control (a possible new rule for every situation)
- Reusability (existing rules of languages can be transferred when paired with new languages)

Disadvantages:

- Requires good dictionaries
- Manually set rules (requires expertise)
- The more the rules the harder to deal with the system

Statistical Machine Translation (SMT)

This approach uses statistical models based on the analysis of bilingual text corpora. It was first introduced in 1955, but it gained interest only after 1988 when the IBM Watson Research Center started using it.

The idea behind statistical MT is the following:

Given a sentence T in the target language, we seek the sentence S from which the translator produced T. We know that our chance of error is minimized by choosing that sentence S that is most probable given T. Thus, we wish to choose S so as to maximize Pr(S|T).

Statistical Machine Translation (SMT)

Using Bayes’ theorem, we can transform this maximisation problem to the product of Pr(S) and Pr(T|S),

where Pr(S) is the language model probability of S (S is the right sentence in that place) and Pr(T|S) is the translation probability of T given S.

In other words, we are seeking the most likely translation given how correct a candidate translation is and how well it fits in the context.

$$\Pr(S|T) = \frac{\Pr(S)\Pr(T|S)}{\Pr(T)}$$

Statistical Machine Translation (SMT)

Therefore, an SMT requires three steps:

1) A Language Model (what is the correct word given its context?);
2) A Translation Model (what is the best translation of a given word?);
3) A method to find the right order of words.

Real time examples where SMT is used:

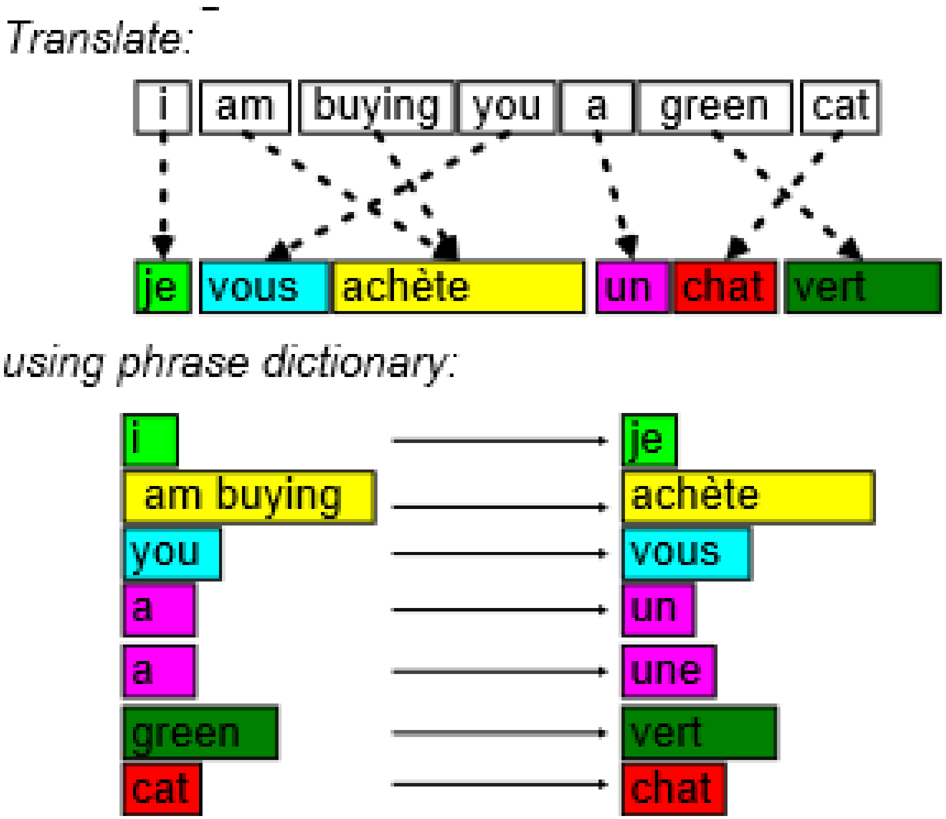
Google Translate (between 2006 and 2016)

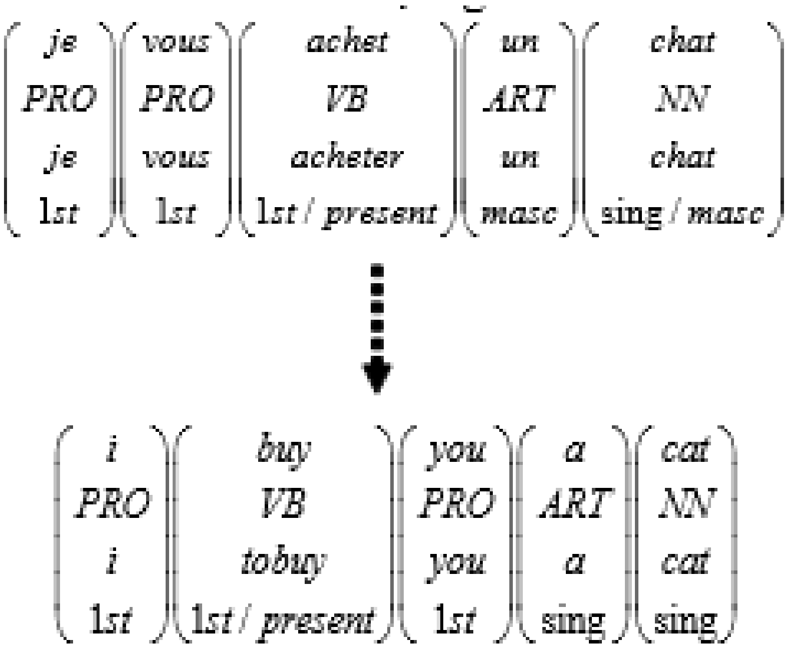
Microsoft Translator (upto 2016)

Moses: Open source toolkit for statistical machine translation.

Statistical Machine Translation (SMT)

Sentence in English translated to French as shown below:





Statistical Machine Translation (SMT)

Advantages:

- Less manual work from linguistic experts
- One SMT suitable for more language pairs
- Less out-of-dictionary translation: with the right language model, the translation is more fluent

Disadvantages:

- Requires bilingual corpus
- Specific errors are hard to fix
- Less suitable for language pairs with big differences in word order

Neural Machine Translation (NMT)

The neural approach uses neural networks to achieve machine translation.

Compared to the previous models, NMTs can be built with one network instead of a pipeline of separate tasks.

In 2014, sequence-to-sequence models were introduced opening new possibilities for neural networks in NLP.

Before the seq2seq models, the neural networks needed a way to transform the sequence input into computer-ready numbers (one-hot encoding, embeddings).

Neural Machine Translation (NMT)

With seq2seq, the possibility of training a network with input and output sequences became possible.

The NMT emerged quickly. After a few years of research, these models outperformed the SMTs.

Real time examples where NMT is used:

Google Translate (from 2016)

Microsoft Translate (from 2016)

Translation on Facebook

OpenNMT: An open-source neural machine translation system.

Neural Machine Translation (NMT)

With seq2seq, the possibility of training a network with input and output sequences became possible.

The NMT emerged quickly. After a few years of research, these models outperformed the SMTs.

Real time examples where NMT is used:

Google Translate (from 2016)

Microsoft Translate (from 2016)

Translation on Facebook

OpenNMT: An open-source neural machine translation system.

Neural Machine Translation (NMT)

Advantages:

End-to-end models (no pipeline of specific tasks)

Disadvantages:

Rare word problem:

A problem with neural networks occurs if the training data is unbalanced, the model cannot learn from the rare samples as well as frequent ones.

In the case of languages, it is a common problem as there are many rare words used only a few times while training.