

NLP MECHANISM

Dr. Krishnendu Guha

Assistant Professor (On Contract)

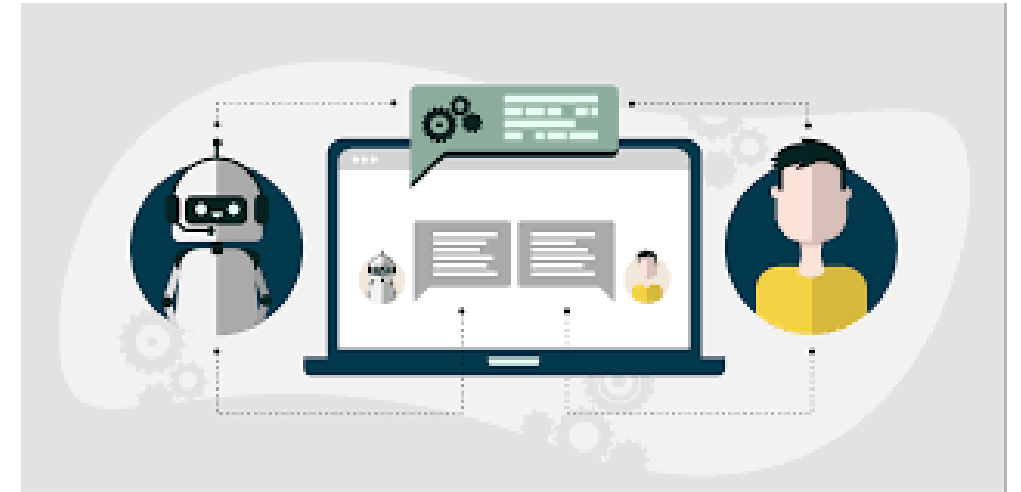
National Institute of Technology (NIT), Jamshedpur

Email: krishnendu.ca@nitjsr.ac.in



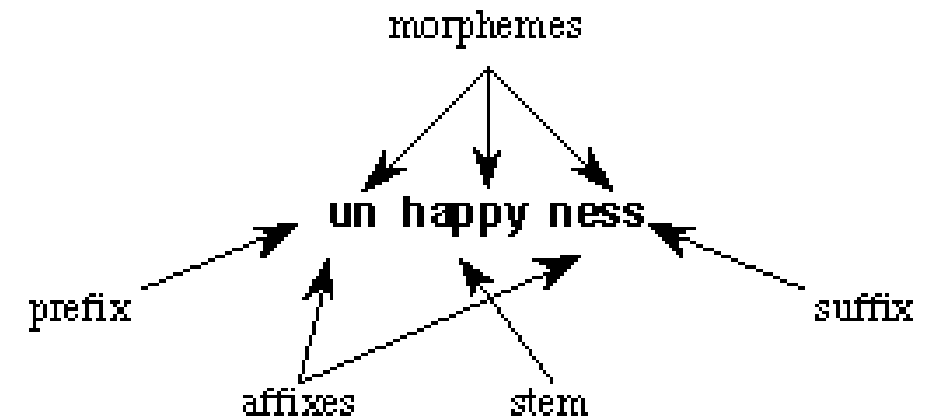
INTRODUCTION

- Natural Language Processing (NLP) problem can divide into two tasks:
 1. Processing written text, using lexical, syntactic and semantic knowledge of the language as well as the required real-world information
 2. Processing spoken language, using all the information needed above plus additional knowledge about phonology as well as enough added information to handle the further ambiguities that arise in speech



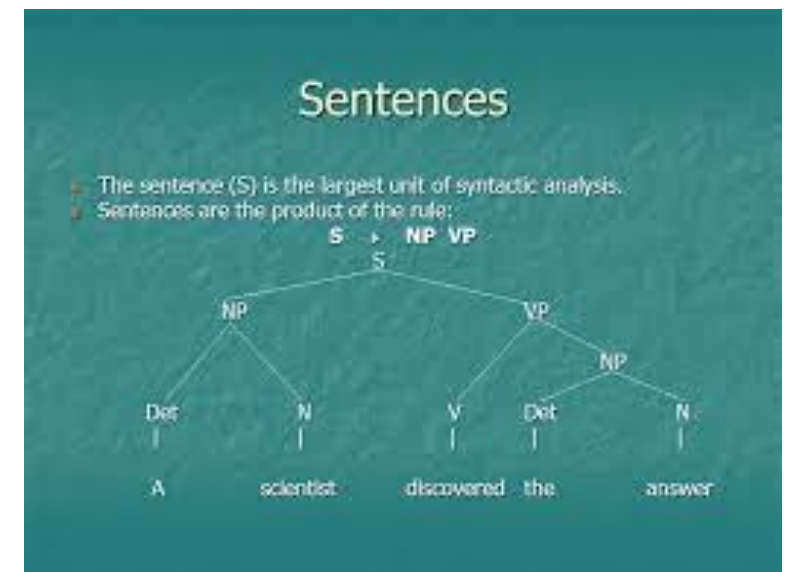
MORPHOLOGICAL ANALYSIS

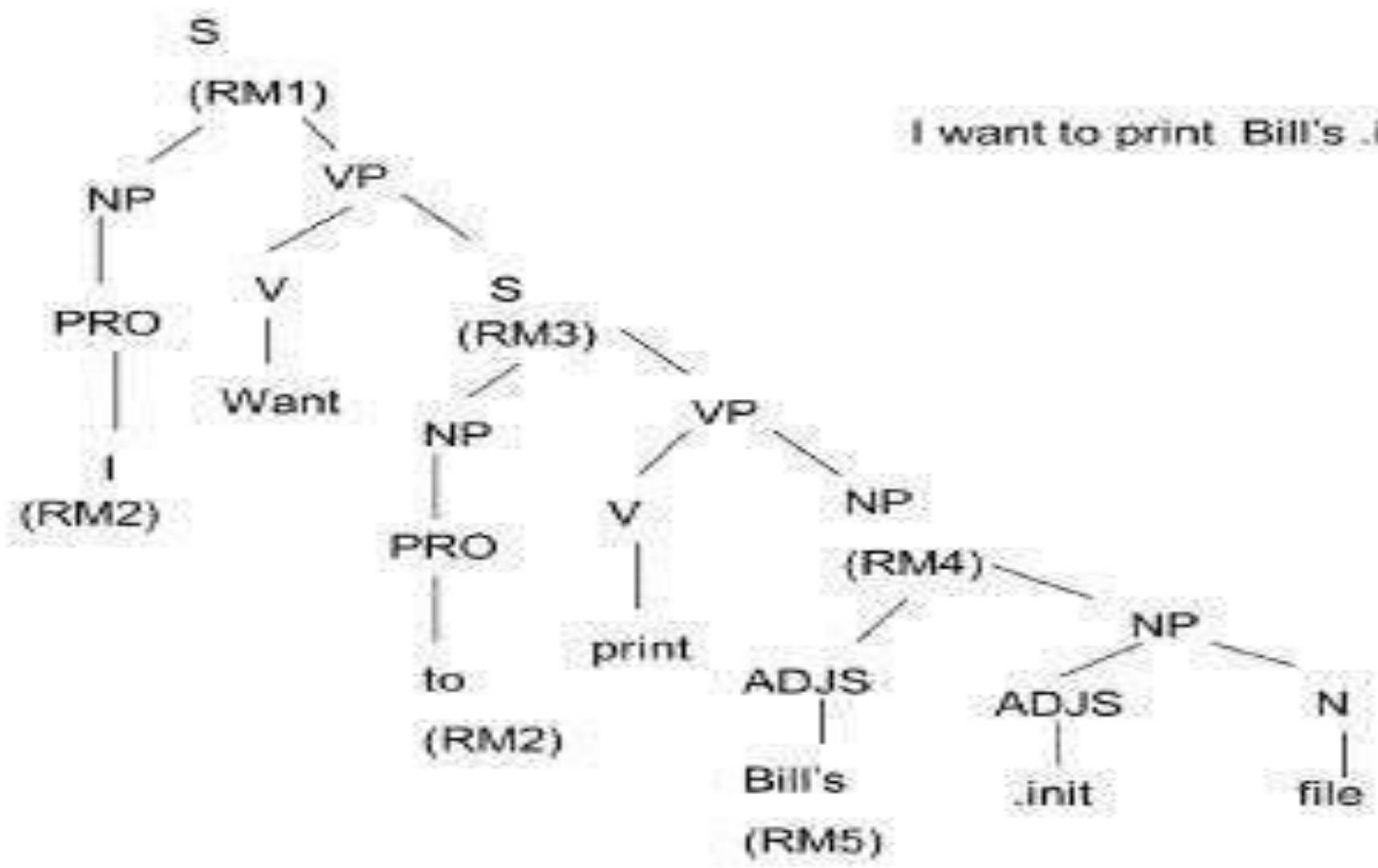
- Individual words analyzed into their components and non-word tokens such as punctuation separated from the words
- Suppose we have an English interface to an operating system and the following sentence typed:
 - I want to print Bill's .init file.
- The morphological analysis must do the following things:
- Pull apart the word "Bill's" into proper noun "Bill" and the possessive suffix "s"
- Recognize the sequence ".init" as a file extension that is functioning as an adjective in the sentence
- This process will usually assign syntactic categories to all the words in the sentence



SYNTACTIC ANALYSIS

- Linear sequences of words transformed into structures that show how the words relate to each other.
- Moreover, Some word sequences may reject if they violate the language's rule for how words may combine
- A syntactic analysis must exploit the results of the morphological analysis to build a structural description of the sentence.
- The goal of this process, called parsing, is to convert the flat list of words that form the sentence into a structure that defines the units that represented by that flat list.
- The important thing here is that a flat sentence has been converted into a hierarchical structure
- And that the structure corresponds to meaning units when a semantic analysis performed.
- Reference markers (set of entities) shown in the parenthesis in the parse tree.
- Each one corresponds to some entity that has mentioned in the sentence.
- These reference markers are useful later since they provide a place in which to accumulate information about the entities as we get it.





I want to print Bill's .init file.

SEMANTIC ANALYSIS

- The structures created by the syntactic analyzer assigned meanings.
- Also, a mapping made between the syntactic structures and objects in the task domain.
- Moreover, Structures for which no such mapping possible may reject
- The semantic analysis must do two important things:
 1. It must map individual words into appropriate objects in the knowledge base or database.
 2. It must create the correct structures to correspond to the way the meanings of the individual words combine with each other.



A Knowledge Base Fragment

```
User
  isa :
    * login-name :      Person
                        must be <string>

User068
  instance :           User
  login-name :        Susan-Black

User073
  instance :           User
  login-name :        Bill-Smith

F1
  instance :           File-Struct
  name :              stuff
  extension :         .init
  owner :             User073
  in-directory :     /wsmith/

File-Struct
  isa :               Information-Object

Printing
  isa :               Physical-Event
  * agent :           must be <animate or program>
  * object :          must be <information-object>

Wanting
  isa :               Mental-Event
  * agent :           must be <animate>
  * object :          must be <state or event>

Commanding
  isa :               Mental-Event
  * agent :           must be <animate>
  * performer :       must be <animate or program>
  * object :          must be <event>

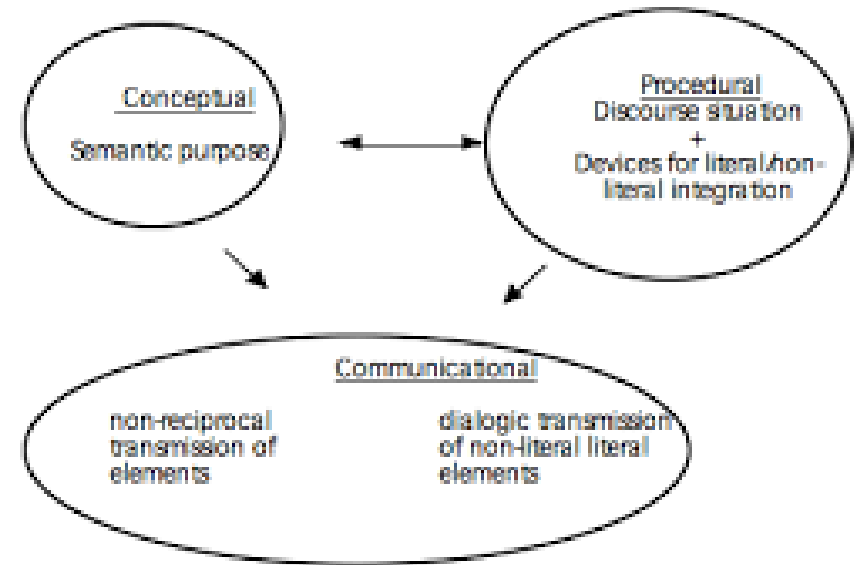
This-System
  instance :          Program
```

A Partial Meaning for a Sentence

| | | |
|---------------------|--------------------|----------------------|
| | | {the whole sentence} |
| <i>RM1</i> | | |
| <i>instance :</i> | <i>Wanting</i> | {I} |
| <i>agent :</i> | <i>RM2</i> | {a printing event} |
| <i>object :</i> | <i>RM3</i> | {I} |
| <i>RM2</i> | | {a printing event} |
| <i>RM3</i> | | |
| <i>instance :</i> | <i>Printing</i> | {I} |
| <i>agent :</i> | <i>RM2</i> | {Bill's .init file} |
| <i>object :</i> | <i>RM4</i> | {Bill's .init file} |
| <i>RM4</i> | | |
| <i>instance :</i> | <i>File-Struct</i> | |
| <i>extension :</i> | <i>.init</i> | |
| <i>owner :</i> | <i>RM5</i> | {Bill} |
| <i>RM5</i> | | {Bill} |
| <i>instance :</i> | <i>Person</i> | |
| <i>first-name :</i> | <i>Bill</i> | |

DISCOURSE INTEGRATION

- The meaning of an individual sentence may depend on the sentences that precede it
- And also, may influence the meanings of the sentences that follow it
- Specifically, we do not know whom the pronoun “I” or the proper noun “Bill” refers to.
- To pin down these references requires an appeal to a model of the current discourse context, from which we can learn that the current user named “Bill” about whom we could be talking
- Once the correct referent for Bill known, we can also determine exactly which file referred to.



PRAGMATIC ANALYSIS

- Moreover, The structure representing what said reinterpreted to determine what was actually meant
- The final step toward effective understanding is to decide what to do as a result
- One possible thing to do to record what was said as a fact and done with it.
- For some sentences, a whose intended effect is clearly declarative, that is the precisely correct thing to do.
- But for other sentences, including this one, the intended effect is different.
- We can discover this intended effect by applying a set of rules that characterize cooperative dialogues.
- The final step in pragmatic processing to translate, from the knowledge-based representation to a command to be executed by the system



Representing the Intended Meaning

| | |
|--------------------|--------------------|
| <i>Meaning</i> | |
| <i>instance :</i> | <i>Commanding</i> |
| <i>agent :</i> | <i>User068</i> |
| <i>performer :</i> | <i>This-System</i> |
| <i>object :</i> | <i>P27</i> |
| <i>P27</i> | |
| <i>instance :</i> | <i>Printing</i> |
| <i>agent :</i> | <i>This-System</i> |
| <i>object :</i> | <i>F1</i> |



SUMMARY

- Results of each of the main processes combine to form a natural language system.
- All of the processes are important in a complete natural language understanding system.
- Not all programs are written with exactly these components.
- Sometimes two or more of them collapsed.
- Doing that usually results in a system that is easier to build for restricted subsets of English but one that is harder to extend to wider coverage

