Write down four desirable properties for a formal language to be used for representing natural language.

1. **Expressiveness:** The formal language should be capable of expressing the full range of meanings and nuances found in natural language, including complex syntactic structures and semantic relationships.
2. **Precision:** It should allow for precise and unambiguous representation of meaning, ensuring that each expression has a clear and specific interpretation.
3. **Scalability:** The language should be scalable to accommodate a wide variety of linguistic phenomena, from simple sentences to complex discourse structures, and to handle large-scale linguistic processing tasks efficiently.
4. **Interpretability:** It should facilitate easy interpretation and understanding by both humans and machines, enabling effective communication and computation based on the represented linguistic information.

Discuss the role and features of a Language Analyzer.

Features:

1. **Tokenization:** Segmenting text into individual units such as words, phrases, or sentences. This process is fundamental for further analysis and processing.
2. **Lemmatization and Stemming:** Reducing words to their base or root forms, enabling normalization and reducing the complexity of the vocabulary. Lemmatization considers the word's meaning, while stemming operates on the word's form.
3. **Part-of-Speech Tagging (POS):** Assigning grammatical categories (e.g., noun, verb, adjective) to each word in a sentence. This helps in understanding the syntactic structure and semantic roles of words in context.
4. **Named Entity Recognition (NER):** Identifying and classifying proper nouns and other named entities such as people, organizations, locations, dates, and numerical expressions.

Roles:

1. **Text Preprocessing:** Language analyzers preprocess raw text data to prepare it for further analysis and processing. This includes tasks like tokenization, lemmatization, and POS tagging.
2. **Information Extraction:** By identifying named entities and extracting key information from text, language analyzers enable applications to extract structured data from unstructured text sources.
3. **Understanding Language Structure:** Analyzing the syntactic and semantic structure of text allows language analyzers to understand the relationships between words and phrases, enabling more accurate interpretation and processing.
4. **Text Classification and Categorization:** Language analyzers play a crucial role in tasks such as text classification, topic modeling, and document categorization by extracting relevant features and information from text data.

Define tree adjoining grammar.

TAG consists of initial trees and auxiliary trees, and two operations substitution and adjoining (or adjunction). **Auxiliary trees** have leaf nodes labelled by terminal symbols and nonterminal symbols; exactly one of the leaf nodes with a non-terminal label same as the root of the auxiliary tree is the foot node and all other leaf nodes with non-terminal labels are substitution nodes. If every auxiliary tree (besides initial trees) has at least one lexical item (i.e., terminal symbol) at a leaf node, the grammar is called lexicalised TAG.

Adjoining Operation: To perform adjoining operation on a tree t at address p using a tree u with a footnode, remove the subtree s of t rooted at p leaving a copy of the root node at p, substitute u with the node at address p in t, and finally substitute s at the footnode of u. Adjoining is disallowed if node at address p in t is a substitution node. Also clearly, this operation can be carried out only if the label of the node at address p in t is the same as the label of the root node of u. An auxiliary tree with no substitution node is a completed tree.

List and explain 5 application of Machine Translation.

1. **Language Translation Apps:** Apps that help translate text or speech from one language to another, making it easier for people who speak different languages to communicate.
2. **Global Business Websites:** Websites of international companies that translate their content into multiple languages to reach customers worldwide.
3. **Travel and Tourism:** Translation services used in travel apps, websites, and tourist guides to help travellers navigate foreign countries and communicate with locals.
4. **Language Learning Platforms:** Online platforms and apps that provide translation services to help learners understand foreign languages better by translating texts or conversations.
5. **Multilingual Customer Support:** Companies offering customer support in multiple languages, using machine translation to communicate with customers who speak different languages.

Differentiate between context free and context sensitive language.

Aspect	Context-Free Language (CFL)	Context-Sensitive Language (CSL)
Definition	Production rules are context-free, meaning rewriting of symbols does not depend on context.	Production rules are context-sensitive, meaning rewriting of symbols depends on context.
Grammar	Defined by Context-Free Grammars (CFG), where rules are context-free.	Defined by Context-Sensitive Grammars (CSG), where rules are context-sensitive.
Examples	Well-formed parentheses expressions, such as (())()(())().	Palindromes, such as $aba, abba, abcbaaba, abba, abcba$.
Parsing Complexity	Efficient parsing algorithms (e.g., CYK algorithm) with polynomial time complexity.	More complex parsing algorithms, sometimes requiring exponential time or being undecidable.

Discuss the advantages and disadvantages for NLP applications of grammar formalisms that use features structures compared with context free grammars.

Advantages:

1. **Expressiveness:** Allows for rich linguistic representations with features capturing syntactic, semantic, and pragmatic information.
2. **Linguistic Coverage:** Can model a wide range of linguistic phenomena, leading to more accurate analyses.
3. **Modularity and Reusability:** Promotes modular and reusable grammar development, enhancing efficiency.
4. **Robustness:** Handles variations in linguistic input well, making it suitable for parsing noisy or ambiguous text.

Disadvantages:

1. **Complexity:** Feature-based grammars are more complex, potentially leading to increased computational overhead.
2. **Parsing Efficiency**: Parsing may be slower due to additional processing required for feature matching and unification.
3. **Learning and Training:** Requires more linguistic expertise and effort for training and development.
4. **Interpretability:** More difficult to interpret and debug, especially for non-linguists or developers without expertise in feature-based formalisms.

Briefly define what is meant by the semantics of a natural language utterance, and how this differs from the pragmatics.

Aspect	Semantics	Pragmatics
Focus	Literal meaning of linguistic expressions based on grammar and vocabulary.	Context-dependent interpretation of language beyond literal meaning.
Scope	Deals with compositional meanings of sentences.	Considers speaker's intentions and situational context.
Concerned with	What is being said and what it refers to in a linguistic expression.	How language is used in real-life situations to convey meaning effectively.
Examples	Understanding the meaning of words and sentences based on their dictionary definitions and grammatical structures.	Interpreting sarcasm, understanding indirect speech acts, and recognizing conversational implicatures.

(a) Language as rule based system.

Language can be viewed as a rule-based system where words and sentences follow specific rules or patterns to convey meaning. These rules govern the formation of sentences, the arrangement of words, and the relationships between them. For example, in English, the rule for forming a simple declarative sentence is subject-verb-object (SVO) order.

Example:

Consider the sentence: "The cat chased the mouse."

Here, the rule dictates that the subject "cat" comes before the verb "chased," and the object "mouse" comes after the verb.

Similarly, grammatical rules govern other aspects of language, such as verb conjugation, pluralization of nouns, and formation of questions. By following these rules, speakers and writers can communicate effectively and ensure that their messages are understood.

(b) Part of speech (POS) tagging.

POS tagging is a process in natural language processing (NLP) that involves assigning a grammatical category or part of speech to each word in a sentence. This categorization helps in understanding the syntactic structure of a sentence and aids in various NLP tasks such as parsing, information extraction, and machine translation.

Example:

Consider the sentence: "The quick brown fox jumps over the lazy dog."

POS tagging assigns a part of speech tag to each word:

- "The" -> Determiner (DET)
- "quick" -> Adjective (ADJ)
- "brown" -> Adjective (ADJ)
- "fox" -> Noun (NOUN)
- "jumps" -> Verb (VERB)
- "over" -> Preposition (PREP)
- "the" -> Determiner (DET)
- "lazy" -> Adjective (ADJ)
- "dog" -> Noun (NOUN)

POS tagging helps in analyzing the grammatical structure of sentences, enabling computers to understand and process natural language text more effectively.

What are the differences between pragmatics and discourse analysis?

Aspect	Pragmatics	Discourse Analysis
Focus	Interpretation of language beyond literal meaning.	Examination of larger units of language in context.
Scope	Study of language use and meaning in context.	Analysis of structure, coherence, and functions of discourse.
Units of Analysis	Individual utterances or speech acts.	Conversations, narratives, speeches, or written texts.
Methods and Approaches	Speech act theory, politeness theory, relevance theory.	Conversation analysis, narrative analysis, critical discourse analysis.

Explain the various issues in Indian languages with respect to LFG.

Indian languages are basically free word-order languages whereas CFG formalism is designed to handle order or position elegantly. As a result, it is a misfit.

One can see that CFG is not designed to handle free word-order. Two problems occur: First, to capture free word-order we have to increase the number of rules. Second, free word-order languages have a rich system of case endings. Far from capturing that richness, it leads to yet greater increase in the number of rules. Finally, what it does capture compactly, can be captured by regular grammar

Write an algorithm for converting an arbitrary context-free Chomsky normal from. Explain it with a suitable example.

Step 1. Eliminate start symbol from RHS.
If start symbol S is at the RHS of any production in the grammar, create a new production as:
S0->S
where S0 is the new start symbol.
Step 2. Eliminate null, unit and useless productions.
If CFG contains null, unit or useless production rules, eliminate them.
Step 3. Eliminate terminals from RHS if they exist with other terminals or non-terminals.
e.g,; production rule X->xY can be decomposed as:
X->ZY
Z->x
Step 4. Eliminate RHS with more than two non-terminals.
e.g,; production rule X->XYZ can be decomposed as:
X->PZ
P->XY
https://www.youtube.com/watch?v=6IolsIONBT0

What is Chomsky normal form? What is the use of Chomsky normal form? Explain with example.

Chomsky Normal Form (CNF) is a specific form of context-free grammars (CFGs) in formal language theory. In CNF, each production rule has one of two forms:

$A \rightarrow BC$

$A \rightarrow a$

where A, B, and C are non-terminal symbols, and a is a terminal symbol. In other words, each production rule either has a single terminal symbol on the right-hand side, or two non-terminal symbols.

Uses of CNF:

1. **Parsing Algorithms:** CNF simplifies parsing sentences, enabling efficient parsing algorithms like CYK.
2. **Parsing Efficiency:** CNF leads to faster parsing algorithms for large texts and real-time applications.
3. **Language Modeling:** CNF provides a standard format for representing language structure, aiding in language modeling tasks.
4. **Grammar Simplification:** Converting to CNF simplifies grammars for analysis and manipulation.
5. **Formal Language Theory:** CNF aids theoretical analysis of formal languages and automata.
6. **Machine Translation:** CNF aids in representing syntactic structures for consistent and efficient translation.
7. **Natural Language Understanding:** CNF helps extract syntactic information for tasks like semantic analysis and information extraction.

Language accesor

Language accessor is a software component responsible for providing access to linguistic resources and tools for a specific language. It serves as an interface between NLP systems and linguistic data, including lexicons, grammars, parsers, and other language processing modules. The language accessor facilitates tasks such as morphological analysis, part-of-speech tagging, syntactic parsing, and semantic analysis by providing linguistic knowledge and tools tailored to a particular language. Language accessors play a crucial role in enabling NLP systems to effectively process and understand natural language input in various applications, including machine translation, information retrieval, sentiment analysis, and chatbots.

LFG formalism

LFG formalism has two major components, a ggcontext free grammar and a functional specification. The former gives the c-structure for a sentence, and the latter gives the f-structure. The two components are interrelated, however, and the f-structure is produced by using functional specification together with the c-structure.

The functional specifications usually consist of equalities associated with each non-terminal on the right-hand side of the context free (CF) rule. In the example grammar from Kaplan and Bresnan (1982) given below, there are two special symbols: up-arrow and down-arrow (called meta-variables). The down-arrow in a functional specification associated with a non-terminal refers to the f-structure with the non-terminal, while the up-arrow refers to the f-structure associated with the symbol on the left-hand side of the CF rule.

(R1) S $\rightarrow$	NP	VP		
	$\uparrow$ subj = $\downarrow$	$\uparrow = \downarrow$		
(R2) VP $\rightarrow$ V	{ NP }	{ NP }	PP*	
	$\uparrow$ obj = $\downarrow$	$\uparrow$ obj2 = $\downarrow$	$\uparrow$ ($\downarrow$ pcase) = $\downarrow$ obj	
(R3) PP $\rightarrow$	prep	NP		
		$\uparrow$ obj = $\downarrow$		
(R4) NP $\rightarrow$	det	noun |	pronoun	

Rule (R1) says that a sentence (S) consists of a noun phrase (NP) followed by a verb phrase (VP). The functional specification associated with the NP says that the f-structure for S has an attribute subj whose value is the f-structure for NP. Thus, the f-structure for NP is the subject in the f-structure for S. The second specification says that the f-structures for VP and S are equal.

In rule (R2), both NPs are optional, which are followed by prepositional phrase (PP) repeating zero or more times. The NPs contribute to object or object2, in the f-structure of the sentence, PPs are stored as adjunct.

To obtain the f-structure, we must use the functional specification along with the c-structure. Consider as an example, sentence (5) whose c-structure is given in Figure 8.1 (a). Let f1 be the f-structure of the sentence and f2 that of the NP. Therefore, on using the subject specification of the c-structure we get:

```
f1  subj = f2
```

Similarly, other equations can be written down. The terminals also yield equations, using the lexicon. Solution to the equations is an f-structure shown in Figure 8.1 (b). It is associated with the root node S in the c-structure.

Some example lexicon entries are:

the, det
↑ spec = the

boy, noun
↑ pred = 'boy'
↑ num = singular