Q4. Statistical Methods for ambiguity resolution

- Statistical methods play a significant role in solving ambiguity in Natural Language processing (NLP)

- Some commonly used statistical approaches include →

1. Word Sense Disambiguation (WSD) →

- Statistical methods utilize corpus-based frequencies to determine the most probable sense of a word in a given context.

- Techniques like Lesk algorithm, supervised learning models (eg. Naive Bayes) and more recently, neural network models leverage contextual information to disambiguate word senses.

eg. "bank" can mean a financial institution or the side of a river

"I went to the bank to deposit my paycheck"

- In this sentence, without context, it is unclear if "bank" refers to a financial institution or a river bank.

- Statistical methods for WSD analyze the surrounding words and their frequencies in a large corpus to determine the most probable sense

- Eg. if words like "money", "ATM" etc. frequently occur with "bank" in a specific context in training data, the statistical model might lean towards the financial institution sense.

Date
Page

Lesk algorithm

- Lesk algorithm is a computational approach used in NLP to disambiguate the meanings of words in context. It was developed by Michael E. Lesk

- Overwiew of how it works →

i. Gather context - Collect the surrounding words (context) of the target word that we need to disambiguate. (usually consists of words within a certain window around the target word)

ii. Retrieve definitions - Retrieve the definitions of the ambiguous word from a lexical database like WordNet (organizes words into sets of synonyms called synsets, with definitions for each synset)

iii. Overlap calculation: Calculate overlap between the words in the context and the words in each definition of the ambiguous word. The sense of the word with the highest overlap with the context is likely to be the correct one.

iv. Sense calculation: Select the sense with the highest overlap as the disambiguated sense for the ambiguous word.

2. Probablistic parsing →

- Statistical parsing techniques assign probabilities to different parse trees based on training data

- This helps in resolving syntactic ambiguity by selecting the most probable parse for a given sentence.

- eg. Probablistic context-free grammars (PCFG's) and statistical dependency ~~parsing~~ parsers etc.

eg. "Time flies like an arrow"

Probablistic parsing assigns probabilities to various ways of parsing this sentence based on training data.
It might assign a higher probability to the tree where "like an arrow" is a modifier for "flies" rather than for "time", based on the statistical frequency of such structures in the training corpus.

3. N-gram models →

- These words calculate the probability of a word given it's preceeding context of 'n' words.

- It helps in disambiguating based on the likelihood of certain word sequences occuring together.

eg. "Mary had a little lamb, It's fleece was white as snow"

* Creating N grams →

Unigrams (1-gram) : single words
[Mary, had, a, little, lamb, It's, fleece, was, white, as, snow]

Bigrams (2-gram) : Pairs of consecutive ~~consequet~~ words
[(Mary, had), (had, a), (a, little), (little, lamb), (lamb, It's), (It's, fleece), (fleece, was), (was, white), (white, as), (as, snow)]

Trigrams (3-gram) : Sequence of three consecutive words
[(Mary, had, a), (had, a, little), (a, little, lamb), (little, lamb, It's), (lamb, It's, fleece) ....]

★ Using N-gram models →

- Suppose we are using a bigram model to predict the next word in the sequence "It's fleece ____"

- From the bigrams in the text, we see that "It's fleece" is followed by "was" and "was" is followed by "white".
So, using the bigram probabilities, we predict that most likely the next word after "It's fleece" is "was".

★ Calculating Probabilities →

N-gram models calculate probabilities based on the frequency of the occurence of ~~such~~ these sequences in a given text

$$\boxed{P(w \mid w_{-1}) = \frac{\text{Count}(w_{-1}, w)}{\text{Count}(w_{-1})}}$$

Where $P(w \mid w_{-1})$ → Probability of the word w, given the preceding word w_{-1}
Count (w_{-1}, w) → Count of the bigram (w_{-1}, w)
Count (w_{-1}) → The count of occurences of the word w_{-1}.

4. Statistical Machine Translation → (SMT)

- Ambiguity often arises in translation tasks.

- Statistical approaches in machine translation use large bilingual corpora to determine the most likely translation.

- In translation, words and phrases can have multiple possible translations depending on the context.

eg. Translating "I saw her duck" to spanish from Eng

the word "duck" can be a verb or a noun.
Statistical machine translation models trained on bilingual corpora use context and probabilities to determine the most likely translation.
If it sees that "duck" as a noun is often translated to a spanish word for a bird, and "duck" as a verb is ~~often~~ translated to a word for avoiding something, it will choose the translation based on the statistical likelihood within the context of the sentence.

5. Statistical Semantic Analysis →

- Methods like distributional semantics use statistical patterns from large text corpora to represent word meanings based on their usage pattern in the corpus

eg. Consider a corpus with the sentences
"The cat sat on the mat"
"The dog lay on the rug"

- Statistical semantic analysis would analyze the co-occurence pattern of words.

eg. "cat" and "dog" occur in similar contexts (with words like "on" and "the"), suggesting a degree of semantic similarity.

6. Named Entity Recognition (NER) →

- Statistical models are employed to identify and classify named entities (such as names of people, organizations, locations) in text.

eg. "Apple Inc. was founded by Steve Jobs, Steve Wozniak and Ronald wayne in Cupertino, California"

In this sentence NER would identify and classify different types of entities →

Organization : "Apple Inc."
Persons : "Steve Jobs", "Steve Wozniak", "Ronald Wayne"
Location : "Cupertino, California"

7. Co-reference Resolution →

- Statistical methods analyse patterns in text to determine relationship between pronouns and their antecedents., resolving referential ambiguity.

eg. "Sarah went to the park, she enjoyed the sunshine. The girl had a picnic with her friends."

Co-reference aims to link the pronouns ("she", "the girl", "her") to the antecedent "Sarah" to understand that they all refer to the same person.

CFG Techniques and methods:- Efficient Parsing:-

Syntax Modeling:-

Syntax modeling is the arrangement in words in a sentence to make grammatical sense.

NLP uses syntax to assess meaning from a language based on grammatical rules.

Syntax Modelling vaild sentences can be formed from constituents like nouns, verbs, adjectives, etc..,

Phrase structure:-

CFG's define the Phrase structure of sentences, breaking them down into constituents such as noun Phrase (NP), verb phrase (VP), Prepositional Phrases (PP), etc..,

Parsing:-

The Process of analyzing the syntactic structure of sentences according to the rules specified in the grammar.

Parsing Consists of two types of Techniques.

1. Top-down Parsing
2. Bottom-up Parsing.

Top-down Parsing:-

Top-down Parsing is a Parsing technique that looks at the highest level of the tree at start and then moves down to the Parse tree.

The top-down Parsing technique tries to identify the leftmost derivation for an input. It evaluates the rules of grammar while Parsing.

In each terminal symbol in the top-down Parsing is Produced by multiple Production of grammar rules

~~Ex:- She painted a beautiful~~

Ex:- The Cat chased the mouse.

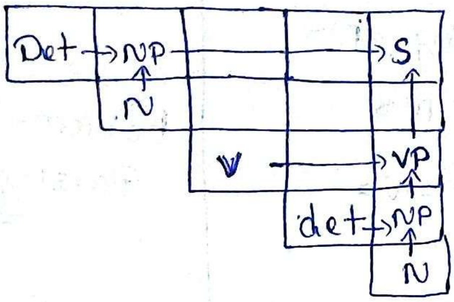
The → "Determinar"

cat → "Noun"

chased - "verb"

the → "determiner"

mouse → "Noun"



S → NP VP

NP → DET N

VP → V NP

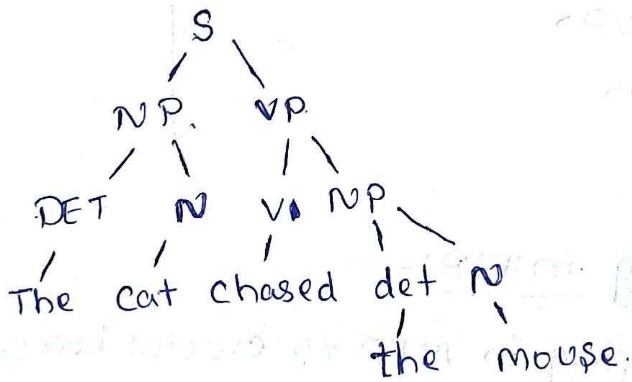
N → Cat

→ Mouse

V → chased

Det → The.

Phrase structure Grammar



Top-down Parsing

Bottom-up Parsing:-

Bottom-up-Parsing technique is which starts from the lowest level of the Parse tree, move upwards and evalutes the rules of grammar.

Bottom-up Parsing technique makes an attempt to decrease the input string to the start symbol of a grammar.

→ The bottom-up Parsing tree technique makes use of rightmost derviation. The main right most ~~derviation~~ decision is to select when a Production rule is used to reduce the string to get the starting symbol of the Parsing tree.

Ex:-

He read an interesting book.

He read an interesting <N>

He read an <Adj> <N>

He read <Det> <Adj> <N>

He read <NP>

He <V> <NP>

He <VP>

<Pronoun> <VP>

<NP> <VP>

<S>

bottom-up Parsing

Efficient Parsing in NLP:-

Efficient Parsing in NLP is crucial because to analyze and understanding the syntactic structure of sentences.

There are various Parsing techniques and algorithms designed to achieve efficient and accurate Parsing in NLP.

1. Shift reduce Parsing:-

Shift reduce Parsing works by shifting inputs words onto a stack and then reducing them according to grammar rules.

The Parsers can be implemented efficiently, and variations like the LR [Left to right, rightmost derivation] and SLR Parsers are widely they used.

Parallel ~~Processing~~ Parsing :-

Parallel ~~Processing~~ Parsing Computing techniques to Process multiple sentences concurrently, speeding up Parsing, especially in large-scale NLP tasks.

Dependency Parsing :-

Dependency Parsing fouses on the relationships b/w words in a sentence rather than Constituency Parsing [hierarchical structures].

Dependency Parsing algorithms, such as transition-based or graph-based methods, can be more efficient for certain applications.

Optimized Algorithms :-

Parsing algorithms tailored for NLP tasks. Techniques like transition-based Parsing (shift-Reduce Parsers) or chart-based Parsing can efficiently Parse sentences.