Page No.
Date

Module 4

Q.1. What is semantic analysis? Why is it difficult? Explain various approaches to semantic analysis.

Semantic analysis is the process of understanding the meaning of language in a given context. It involves analyzing the syntactic and semantic structures of text to extract the underlying meaning and relationship between words, phrases and sentences.

It is difficult because language is inherently ambiguous and context-dependent. Words can have multiple meanings depending on the context in which they are used.

Page No.
Date

Approaches :-

① Knowledge-based :- These approaches rely on prior knowledge such as formal ontologies, lexicons and semantic networks, to understand the meaning of text. Knowledge-based approaches are typically rule-based.

② Supervised approaches :- These approaches use annotated data, such as labeled examples of text and their corresponding meanings, to train machine learning models that can automatically predict the meaning of new text.

③ Hybrid approaches :- These approaches combine knowledge-based and supervised methods to leverage the strengths of both. Hybrid approaches use knowledge to guide the analysis and statistical models to learn from data. It requires both domain knowledge and annotated data.

Page No.
Date

Q.2 Explain with suitable examples following relationships between word meanings:

Homonymy: It is defined as a relation that holds between words that have the same form with unrelated meanings.
For eg:- Bat (wooden stick-like thing) or (flying mammal thing).

Polysemy :- Multiple related meanings within a single lexeme.
For eg :- The bank was constructed in 1875 out of local red brick.
Sense :- The building belonging to a financial institution.
I withdrew the money from the bank.
Sense : Financial institution

Synonymy :- Words that have same meaning in some or all contexts.
For eg :- couch / sofa

Antonymy :- Senses that are opposite with respect to one feature of their meaning.
For eg :- hold & cold

Page No.
Date

Hyponymy :- One sense is a hyponym of another if the first sense is more specific, denoting a subclass of the other.
For eg :- car is a hyponym of vehicle.

Hypernymy :- It is converse of hyponymy.
For eg :- Vehicle is hypernym of car.

Meronymy :- An asymmetric, transitive relation between senses. X is a meronym of Y if it denotes a part of Y.
Eg :- Leg is a meronym of chair.

Q.3. What is semantic analysis? Discuss different semantic relationships between the words.

Same as Q.1 & Q.2.

Q.4. What is WordNet? How is sense defined in WordNet? Explain with example.

WordNet is a lexical database for the English language that organizes words into groups called synsets based on their meanings. A synset is a set of synonyms that are semantically related and can be used interchangeably in certain contexts.

In WordNet, a sense is defined as a specific meaning of a word. Each sense of a word is represented by a seperate synset.
For example, the word "bank" has multiple senses depending on the context in which it is used. In WordNet, there are different synsets for each sense of "bank".

Here are few different synsets of word "bank" in WordNet:
- (noun) a financial institution
- (noun) relating to river or sea
- (verb) to deposit money
- (verb) to tilt or angle something in a particular direction

Q 6. What do you mean by word sense disambiguation? Discuss knowledge based approach for WSD.

The task of selecting the correct sense for a word is called word sense disambiguation (WSD). WSD algorithms take as input a word in context and a fixed inventory word sense disambiguation of potential word senses and outputs the correct word sense in context.

Knowledge-based approach:-
These rely primarily on dictionaries, thesauri and lexical knowledge bases, without using any corpus evidence.

The Lesk method is the seminal dictionary-based method. It is based on the hypothesis that words use together in text are related to each other and that the relation can be observed in the definitions of the words and their senses.

WordNet is a knowledge-based approach to WSD which identifies different meanings of a word and the associated semantic features for each meaning. These features are then compared to the context of the word to determine the correct meaning. The sense with the highest degree of overlap between its semantic features and the context is selected as the most likely sense.

Q.7. Discuss machine learning based (Naive Bayes) approach for WSD.

The Naive Bayes approach calculates the probability of each sense of a word given its context.

The approach involves following steps:-

1. Identify the target word and its context.
2. Extract features from the context of the target word.
3. Create a training set of annotated examples.
4. Train the Naive Bayes classifier using the annotated training set.
5. Use the trained classifier to predict the sense of the target word in new contexts.

Page No.
Date

Naive Bayes classifier calculates the probability of each sense of the target word given its context. The sense with the highest probability is selected as the most likely sense of the word in the given context.

Applying Naive Bayes to WSD:

$P(c)$ is the prior probability of that sense

$P(w|c)$ - conditional probability of word given particular sense

$$P(w|c) = \frac{count(w, c)}{count(c)}$$

$P(f|c)$ - Conditional probability of feature given a sense

$$\hat{P}(c) = \frac{N_c}{N}$$

$$\hat{P}(w|c) = \frac{count(w, c) + 1}{count(c) + |V|}$$

Page No.
Date

Q. 8. Explain how a supervised learning algorithm can be applied for word sense disambiguation.

Supervised methods are based on the assumption that the context can provide enough evidence on its own to disambiguate words (hence, world knowledge and reasoning are deemed unnecessary). Probably every machine learning algorithm going has been applied to WSD including :- feature selection, parameter optimization and ensemble learning.
SVM and memory based learning have been shown to be most successful approaches because they can cope with high dimensionality of the future space.

Supervised Machine Learning :-

Input :- a word w in a text window d (document)
a fixed set of classes $C = \{c1, c2, \ldots, cJ\}$
a training set of m hand-labeled text windows again called "documents". $(d1, c1), \ldots, (dm, cm)$

Output :- a learned classifier $\gamma : d \rightarrow c$

Page No.
Date

Q 9. Explain Bag of Words with the help of suitable example.

Bag of Words is a method for feature extraction with text data. It keeps track of word counts and disregards the grammatical details and word order.

By using Bag of words, we can convert variable length text into a fixed length vector.

For eg:- S1 : "Welcome to Great Learning, Now start learning"

S2 : "Learning is a good practice"

Step 1: Create the vocabulary.

- Welcome
- To
- Great
- Learning
- ,
- Now
- start
- learning
- is
- a
- good
- practice

Page No.
Date

Because, we know the vocabulary has 12 words, we can use a fixed-length document-representation of 12, with one position in the vector to score each word. The scoring method we use here is to count the presence of each word and mark 0 for absence.

	Welcome	To	Great	Learning	,	Now	start	learning	is	a
S1	1	1	1	1	1	1	1	1	0	0
S2	0	0	0	0	0	0	0	1	1	1

	good	practice
S1	0	0
S2	1	1

Writing above frequencies in vector form :-

S1 → [1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0]

S2 → [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1]

Q·10. Explain vector semantics analysis.

The idea of vector semantics is to represent a word as a point in some multidimensional semantic space. Vectors for representing words are generally called embeddings, because the word is embedded in a particular vector space. Vector semantic models are also extremely practical because they can be learned automatically from text without any complex labeling or supervision.

Page No.
Date

Q.11. Explain what is Distributional Hypothesis.

The Distributional Hypothesis states that words that have similar meanings tend to occur in similar contexts. This hypothesis suggests that the meaning of a word can be inferred from the pattern of words that typically occur in its context.

It is often used as the basis for various NLP techniques including vector semantic analysis, which seeks to capture the meaning of words based on their distribution in a large corpus of text.

Q.12. Explain word space and vector space in distributional semantics.

A word space is a high-dimensional space where each word in a vocabulary is represented as a vector. The word space is created by analyzing the distribution of words in a large corpus of text and representing each word as a vector based on its co-occurrence with other words in the corpus.

A vector space is a lower-dimensional subspace of the word space, created by applying dimensionality reduction techniques to the co-occurence matrix that captures the distributional properties of the word. The vectors in the vector space are created by projecting the high-dimensional word vectors onto a lower dimensional subspace which captures most important dimensions of word space.

Page No.
Date

Q.13. What is One-hot representation? Explain with example.

One-hot representation is a method for representing categorical data, as binary vectors where only one element is 1 and all other elements are zero. Each dimension of the vector corresponds to a category, and the value 1 is placed in the dimension that corresponds to the category of interest.

For eg:- Consider vocabulary of 4 words, "cat", "dog", "bird" and "fish". To represent these words using one-hot encoding, a binary vector of length 4 is created for each word.
The vector has 1 in the position that corresponds to the word else others are zero.

cat : [1, 0, 0, 0]
dog : [0, 1, 0, 0]
bird : [0, 0, 1, 0]
fish : [0, 0, 0, 1]

Page No.
Date

Q.15. Explain steps in building distributional semantics model.

Linguistic Steps:

① Pre-process a corpus (to define targets and contexts)

A large corpus of text data is collected and preprocessed to prepare it for analysis.

② Select the targets and the contexts

Mathematical Steps:-

① Count the target-context co-occurences:
We count how often a target word occurs together with other words in the corpus.

② Weight the contexts (optional): In some cases, we may want to weight the context words based on their importance in determining the meaning of the target word.

③ Build the distributional matrix: that represents the relationship between words in the corpus. Each row represents target word and each column represents context word.

④ Reduce the matrix dimensions (optional): If the distributional matrix is large and sparse, we can reduce the dimensionality of matrix using techniques such as PCA.

⑤ Compute the vector distances on the (reduced) matrix:

Once we have the final distributional matrix, we can compute the vector distances between words to determine their semantic similarity.

Q.16. Consider following sentences and represent it into ~~string~~ term document matrix format

D1: Text mining is to find useful information from text.
D2: Useful information is mined from the text.
D3: Dark came.

	D1	D2	D3
Text	1	0	0
mining	1	0	0
is	1	1	0
to	1	0	0
find	1	0	0
useful	1	0	0
information	1	1	0
from	1	1	0
text	1	1	0
Useful	0	1	0
mined	0	1	0
the	0	1	0
Dark	0	0	1
came	0	0	1

Page No.
Date

Q.17.

Step 1 :- Calculating TF

Calculating TF(t, d)

$$TF(t,d) = \left(\frac{\text{Total Number of ferme t in doc A}}{\text{Total no. of tokens in doc A}} \right)$$

	D1	D2	D3
Text	1/9	0	0
mining	1/9	0	0
is	1/9	1/7	0
to	1/9	0	0
find	1/9	0	0
useful	1/9	0	0
information	1/9	1/7	0
from	1/9	1/7	0
text	1/9	1/7	0
Useful	0	1/7	0
mined	0	1/7	0
the	0	1/7	0
Dark	0	0	1/2
came	0	0	1/2

Page No.
Date

step 2:-
Calculate DF & IDF

DF(t) = No. of times term t is present in all docs

$$IDF(t) = \log_{10}\left(\frac{\text{Total Documents}}{DF(t)}\right)$$

	DF	IDF
Text	1	$\log(3/1) = 0.4771$
mining	1	$\log(3/1) = 0.4771$
is	2	$\log(3/2) = 0.1761$
to	1	0.4771
find	1	0.4771
useful	1	0.4771
information	2	0.1761
from	2	0.1761
text	2	0.1761
Useful	1	0.4771
mined	1	0.4771
the	1	0.4771
Dark	1	0.4771
came	1	0.4771

Page No.
Date

Step 3: ~~to~~ Calculate TF-IDF

TF-IDF = TF * IDF

	D1	D2	D3
Text	1/9 * 0.4771 = 0.053	0	0
mining	0.053	0	0
is	0.0195	0.0252	0
to	0.053	0	0
find	0.053	0	0
useful	0.053	0	0
information	0.0195	0.0252	0
from	0.0195	0.0252	0
text	0.0195	0.0252	0
Useful	0	0.0682	0
mined	0	0.0682	0
the	0	0.0682	0
Dark	0	0	0.2385
came	0	0	0.2385

~~Step 3. Calculate IDF~~

Q.18. Calculate point wise mutual information for word (Digital, Computer)

	Count (W, context)					c(w)
	Computer	Data	Pinch	Result	Sugar	
Apricot	0	0	1	0	1	2
Pineapple	0	0	1	0	1	2
Digital	2	1	0	1	0	4
Information	1	6	0	4	0	11
c(context)	3	7	2	5	2	19

$$P(w = \text{Digital}, c = \text{Computer}) = \frac{2}{19}$$

$$P(w = \text{Digital}) = \frac{4}{19}$$

$$P(c = \text{Computer}) = \frac{3}{19}$$

$$PMI(\text{Digital}, \text{Computer}) = \log_2 \left[\frac{P(w = \text{Digital}, c = \text{Computer})}{P(w = \text{Digital}) \cdot P(c = \text{Computer})} \right]$$

$$= \log_2 \left[\frac{2/19}{4/19 \times 3/19} \right]$$

$$PMI(\text{Digital}, \text{Computer}) = 1.663$$

Page No.
Date

Q 20. Discuss the problem with raw dot-product for finding the similarity between vectors.

Dot product favors long vectors.
Dot product is higher if a vector is longer (has higher values in many dimension)

Vector length: $|v| = \sqrt{\sum_{i=1}^{N} v_i^2}$

Frequent words (of, the, you) have long vectors (since they occur many times with other words)
So dot product overly favours frequent words.

Q.21. Find cosine similarity between (cherry, data) and (digital, data)

	pie	data	computer
cherry	442	8	2
digital	5	1683	1670
information	5	3982	3325

$$\cos(\vec{v}, \vec{w}) = \frac{\vec{v} \cdot \vec{w}}{|\vec{v}||\vec{w}|} = \frac{\sum_{i=1}^{N} v_i w_i}{\sqrt{\sum_{i=1}^{N} v_i^2} \sqrt{\sum_{i=1}^{N} w_i^2}}$$

$$\cos(\text{digital}, \text{data}) = \frac{5*8 + 1683*1683 + 1670*3982}{\sqrt{5^2 + 1683^2 + 1670^2}\sqrt{8^2 + 1683^2 + 3982^2}}$$

$$= \frac{9482479}{10249776.33}$$

$$\cos(\text{digital}, \text{data}) = 0.9251$$

~~cos(digital, data) =~~

$$\cos(\text{cherry}, \text{data}) = \frac{442*8 + 8*1683 + 2*3982}{\sqrt{442^2 + 8^2 + 2^2}\sqrt{8^2 + 1683^2 + 3982^2}}$$

$$= \frac{24964}{1911126.53}$$

$$\cos(\text{cherry}, \text{data}) = 0.0131$$

Q.22. What are the disadvantages of TFIDF method?

Semantics meaning is not captured
Similar words should have similar kind of vectors.
Sparse matrix is generated
Increase no. of dimensions.

Page No.
Date

Q 23. Explain Word2Vec method in detail.

Word2Vec creates vectors of the words that are distributed numerical representations of word features.
These word features could comprise of words that represent the context of the individual words present in our vocabulary.
Word embeddings eventually help in establishing the association of a word with another similar meaning word through the created vectors.

Word2Vec consists of models for generating word embedding. These models are shallow two-layer neural networks having one input layer, one hidden layer and one output layer.

Word2Vec utilizes two architectures:-
- CBOW (Continuous Bag of Words)
- Skip Gram

Page No.
Date

Q.24. Explain the working of continuous bag of words in details with the help of suitable example.

The working of CBOW can be explained with the help of following example:-

Suppose we have the sentence "The quick brown fox jumped over the lazy dog".
We want to create word embeddings for each word using CBOW.
Let's assume a context window size of 3.

First, create the vocabulary.
the, quick, brown, fox, jumped, over, lazy, dog

Next, we create training instances for each target word in the sentence. For eg:, for the target word "fox", the context words would be "quick", "brown", "jumped", "over", "the".
Representing each word using one-hot encoding.

CW vector: [0, 1, 1, 0, 1, 1, 0, 0]
TW vector: [0, 0, 0, 1, 0, 0, 0, 0]

We repeat this process for all target words in the sentence and use this training data to train the CBOW model.

During training, the CBOW model learns to predict the target word based on the context words around it. It does this by mapping the input one-hot encoded vectors to a dense vector representation in the projection layer and then predicting the target word in the output layer.

Once the model is trained, we can extract the dense vector representation of each word in the vocabulary from the projection layer.

Then average these vectors to $\hat{V}$.
Generate a score vector $z = W^2 * \hat{V}$.
Turn the scores into probability $\hat{y} = \text{softmax}(z)$

Q.25. Explain the working of Skip Gram model in detail with the help of suitable example.

The skip-gram model is a simple neural network with one hidden layer trained in order to predict the probability of a given word being present when an input word is present.
The skip-gram model is the opposite of CBOW model. In this architecture, it takes the curent word as input and tries to accurately predict the words before and after its this current word.
This model essentially tries to learn and predict the context words around the specified input word.

Page No.
Date

Q.26

	Doc	Words	Class
Training	1	Chinese Beijing chinese	c
	2	Chinese Chinese Shangai	c
	3	Chinese Macao	c
	4	Tokyo Japan Chinese	j
Test	5	Chinese Chinese Chinese Tokyo Japan	?

Priors:

$P(c) = \frac{3}{4}$; $P(j) = \frac{1}{4}$

$V = \{$Chinese, Beijing, Shangai, Macao, Tokyo, Japan$\}$

Conditional Probabilities :-

$P(\text{Chinese}|c) = (3+1)/(8+6) = 0.2857$

$P(\text{Tokyo}|c) = (0+1)/(8+6) = 0.0714$

$P(\text{Japan}|c) = (0+1)/(8+6) = 0.0714$

$P(\text{Chinese}|j) = (1+1)/(3+6) = 0.2222$

$P(\text{Tokyo}|j) = (1+1)/(3+6) = 0.2222$

$P(\text{Japan}|j) = (1+1)/(3+6) = 0.2222$

$P(c|d5) = \frac{3}{4} \times (0.2857)^3 \times 0.0714 \times 0.0714 = 8.9163 \times 10^{-5}$

$P(j|d5) = \frac{1}{4} \times (0.2222)^3 \times 0.2222 \times 0.2222 = 1.3541 \times 10^{-4}$

The predicted class is 'j'

Page No.
Date

Q27. Explain how to perform WSD using Random Walk Algorithm using suitable example

The church bells no longer rung on Sundays.

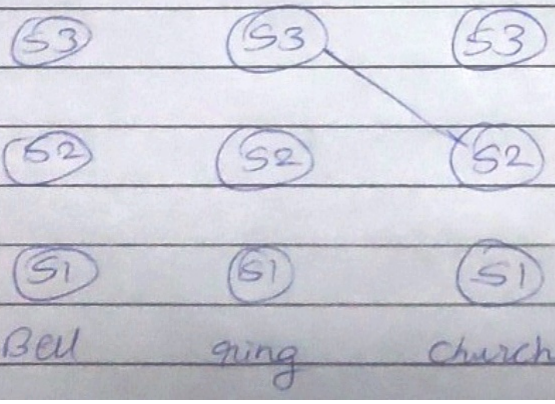
church - one of the group of Christians
a place for public worship
a service conducted in church

bell - a device
push button at an door
sound of a bell

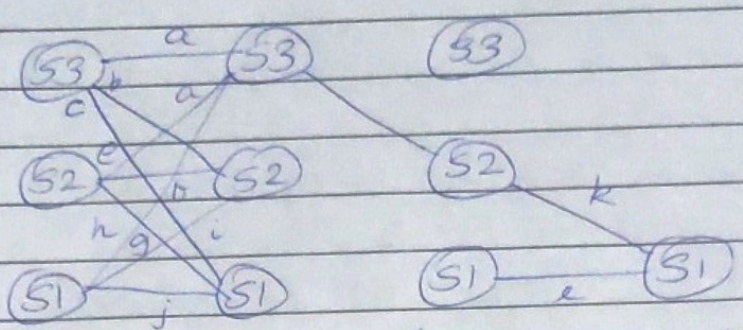
ring - make a ringing sound
ring or echo with sound
make (bells) ring

Sunday - first day of week.

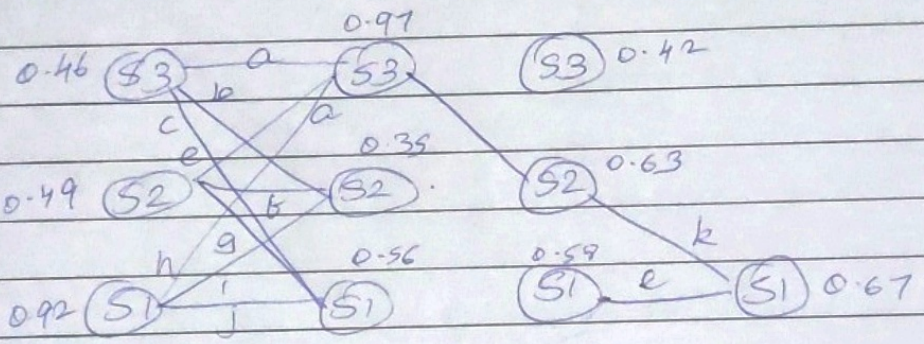
Step 1 :- Add a vertex for each possible sense of each word in the text



Step 2: Add weighted edges using definition based semantic similarity (Lesk's method)



Step 3: Apply graph-based ranking algorithm to find score of each vertex (i.e., for each word sense)



Step 4: Select the vector (sense) which has highest score.