REVA UNIVERSITY
Bengaluru, India

SCHOOL OF COMPUTING AND INFORMATION TECHNOLOGY

B22EJS604 and B22EA0602-Natural Language Processing
Question bank

UNIT-I

1. Explain the role of NLTK in Natural Language Processing. Discuss its key features and components, and evaluate its advantages in NLP tasks.
2. Explain the process of tokenizing, normalizing, and segmenting raw text. Using Python and regular expressions, write a script to clean and tokenize a paragraph of raw text.
3. Write a Python program to perform the following operations on a text file:
   - Read the file
   - Remove all punctuation
   - Convert text to lowercase
   - Count the frequency of each word
   - Display the top 5 most frequent words
4. Describe Conditional Frequency Distributions in NLTK. Write a Python program that uses a conditional frequency distribution to analyze word usage across multiple texts.
5. What are Collocations? How can they be identified using NLTK?
6. List and explain any five functions provided by the re (regular expression) module in Python for text processing.
7. Define segmentation in NLP. Describe the challenges in sentence and word segmentation. Provide code examples using NLTK’s sent_tokenize and word_tokenize functions.
8. What do you mean by automatic natural language understanding? Explain various language understanding technologies in detail.
9. Write a python snippet to tokenize a sentence into words using NLTK.
10. Define segmentation in NLP. Describe the challenges in sentence and word segmentation. Provide code examples using NLTK’s sent_tokenize and word_tokenize functions.
11. List two applications of regular expressions in Natural Language Processing.
12. What is the purpose of Stopwords in NLP? How are they used in text preprocessing?
13. a) Explain the steps involved in preprocessing the raw text using Python and NLTK. Your solution should include normalization, tokenization, removal of stopwords, and word frequency analysis.

    b) Create a basic text analysis tool using NLTK. Load the text1 corpus, convert it into a list of words, compute both the total number of words and the number of unique words. Generate a frequency distribution and display the 10 most common words. Finally, use WordNet to find the synonyms and definitions of the word “sleep”.

c) Given a long paragraph of text and asked to extract only dates and email addresses. Write a Python code using regular expressions.

14. a) You are preparing input text for training a language model. Explain the purpose and process of normalizing text before NLP processing.

b) Define a string and assign it to a variable, Ex: my_string = 'My String'. Print the contents of this variable in two ways, first by simply typing the variable name and pressing Enter, then by using the print statement. Try adding the string to itself using my_string + my_string, or multiplying it by a number, Ex: my_string * 3. Notice that the strings are joined together without any spaces. How could you fix this?

c) Define sent to be the list of words ['she', 'sells', 'sea', 'shells', 'by', 'the', 'sea', 'shore']. Write code to perform the following tasks:

i. Print all words beginning with sh.

ii. Print all words longer than four characters.

iii. Count & print the number of times the word 'sea' appears in the list.

iv. Create a new list with all unique words from sent, and print it.

15. a) Explain the steps involved in preprocessing the raw text using Python and NLTK. Your solution should include normalization, tokenization, removal of stopwords, and word frequency analysis. Illustrate the application using python code.

b) Demonstrate how Conditional Frequency Distributions in NLTK can be used to compare word frequencies across multiple texts.

c) You're designing a simple word suggestion tool that uses synonyms. Show how to access WordNet in NLTK and extract synonyms and definitions of a given word like "bright".

16. a) You are building a feature to process and summarize text articles from websites. The articles must be cleaned and tokenized before summarization. Describe how you would access web text using Python, normalize it, and tokenize it using NLTK. Provide example code.

b) You need to segment a paragraph into sentences and words as part of a larger NLP project. Describe the segmentation process using regular expressions and NLTK methods. Include challenges and how you would handle punctuation or abbreviations.

c) A user asks for the definition and hypernyms of the word "car." Mention the NLTK module used and one function to retrieve this information. Which function will convert the list into a sentence string str= ['The', 'sky', 'is', 'blue']?

17. A) Define NLP. List out the various merits and demerits of using NLP for real-world applications.

B) What are Lists in python? Illustrate with python code to create a list with five members and perform the following tasks: Display the content of the list in reverse order 2) Find the Length of the list 3) Display the list of first two elements 4) Append the new word "REVA" to the list.

C) Describe the class of strings matched by the following regular expressions:

1. [a-zA-Z]+ 2. [A-Z][a-z]*

3. p[aeiou]{,2}t 4. \d+(\.\d+)?

5. ([^aeiou][aeiou][^aeiou])* 6. \w+|[^\w\s]

D) What is Bigrams?, Demonstrate using NLTK provides built-in support to find bigram for the following list. Str1="An apple per day keep a doctor away"

18. a) What do you understand by NLTK in Natural Language Processing? List the some libraries of the NLTK package that are often used in NLP and demonstration how to download and Counting number of Vocabulary of any text file.
    b) What are Spoken Dialogue Systems? Discuss in brief a Simple pipeline architecture for a spoken dialogue system.
    c) What Is Unicode? Discuss with a neat diagram for Text Processing with Unicode decoding and encoding.
    d) Write a python code to access a favorite web page and extract some text from it. For example, access a weather site and extract the forecast top temperature for your town or city today.

19. a) What is the significance of text corpora in NLP? How does NLTK facilitate access to these corpora? Illustrate with examples

b) Write a Python program to perform the following operations on a text file:

i. Read the file,

ii. Remove all punctuation,

iii. Convert text to lowercase,

iv. Count the frequency of each word,

v. Display the top 5 most frequent words

c) Illustrate with python code the purpose of stop-words in NLP with python code? How are they used in text preprocessing?

20. a) Define the terms from NLP pipeline with example

i. Sentence segmentation

ii. lemmatization/stemming

iii. POS tagging

iv. Named Entity Recognition (NER)

v. Chunking

b) Describe Conditional Frequency Distribution functions in NLTK. Write a Python program that uses a conditional frequency distribution to analyze word usage across multiple texts.

c) List two applications of regular expressions in natural language processing

UNIT – II

1. Define a Decision Tree in the context of text classification.
2. Write a complete python program that performs supervised classification of movie reviews using NLTK's Naive Bayes Classifier. Your program should:
    - Load the data
    - Extract features (bag-of-words)

- Train the model
- Evaluate and print accuracy

3. Explain N-Gram tagging. How does increasing the value of n affect tagging performance and complexity? Illustrate with an example using unigram and bigram taggers in NLTK.
4. Explain how mapping words to properties using Python dictionaries can assist in text categorization. Include a code snippet to support your answer.
5. You are given a sentence: "The quick brown fox jumps over the lazy dog." Demonstrate how to apply POS tagging using default, unigram, and bigram taggers in NLTK. Explain the role and limitations of each method.
6. Explain the role of evaluation in supervised learning. Implement code to split a labeled dataset into training and testing sets and evaluate accuracy of a classifier.
7. What is the role of a Part-of-Speech (POS) tagger in NLP?
8. Write a python program that:
    - Tags a sentence using the pos_tag() function from NLTK
    - Extracts all nouns from it
    - Prints the frequency distribution of those nouns.
9. Describe how to evaluate a text classifier. Explain accuracy, precision, recall, and F1-score. Show how to compute these metrics using an example confusion matrix and classification report.
10. Describe how a Unigram and Bigram Tagger works. Implement both using NLTK and compare their accuracy.
11. Briefly explain how decision trees can be used for text classification. What are the advantages and limitations?
12. a) List any 10 POS tags from the Penn Treebank tag set with their descriptions and examples.

    b) Discuss how mapping words to properties using Python dictionaries works for text processing. How can Python dictionaries be used in conjunction with taggers to map words to specific categories? Provide a practical example.

    c) Explain n-gram models in text classification. How does the use of n-grams (Ex: bigrams, trigrams) improve the classification of text data?
13. a) What is the difference between a Regular Expression Tagger and an Automatic POS Tagger in Python? Explain both with examples and code.

    b) What is the difference between supervised and unsupervised classification techniques in text classification? Provide an example of a supervised classification method, such as Naive Bayes or Decision Trees, and explain the evaluation metrics used for performance analysis.

    c) Define segmentation in NLP. Explain the challenges in word and sentence segmentation.
14. a) Explain the process of automatic word tagging using a tagger in NLTK. Demonstrate how to use an NLTK tagger to assign part-of-speech (POS) tags to a given sentence. What are the different types of taggers available in NLTK, and how do you choose the appropriate one for a task?

    b) Describe how n-gram tagging works in Python. Write code to implement a bigram tagger using NLTK and explain how it improves word categorization compared to simple taggers. Discuss the advantages and limitations of n-gram tagging.

c) Briefly explain how decision trees can be used to classify text. What are the advantages of decision trees over other classification methods?

15. a) Write Python code to perform the following tasks using a dictionary.

words = ['run', 'jump', 'blue', 'sky', 'run']

properties = ['verb', 'verb', 'adjective', 'noun', 'verb']

i. Create a dictionary that maps each unique word to its first observed property.

ii. Count how many times each word appears in the list and store the result in another dictionary.

iii. Print all words that are classified as verbs.

iv. Print the word that has the highest frequency.

b) Define tagging in NLP. What is the purpose of using a tagger? Give one example.

c) Explain the Naive Bayes classifier for text classification. Describe the process of applying this classifier to categorize a text document. How do you evaluate the performance of a Naive Bayes classifier, and what are its limitations in text classification?

16. A) Discuss POS tagging with example and demonstrate Tokenize and tagging using python code for the following sentence: “They wind back the clock, while we chase after the wind”. List the simplified parts-of-speech are involved?

B) In general, linguists use morphological, syntactic, and semantic clues to determine the category of a word. Discuss each of them with an example.

C) Demonstrate the Naïve Bayes classifier and how does it work for text classification?

17. A) Summarize Automatic Tagging. Discuss regular expression tagger and lookup tagger with an example for each.

B) Assume you are working on a NLP project that involves analyzing customer feedback for a restaurant. You want to categorize the feedback into positive, negative, or neutral. To do this, you decide to use n-gram tagging to identify the sentiment of the feedback. Explain how would you approach this problem of classifying reviews using n-gram tagging?

C) Discuss about machine learning technique classification. List and explain different types of classification models with an example for each.

18. a) Explain N-Gram tagging. How does increasing the value of n affect tagging performance and complexity? Illustrate with an example using unigram and bigram taggers in NLTK.

B)What is the role of a part-of-speech (POS) tagger in NLP? Write any 15 POS tags given by Penn Treebank with their meanings

c) How Naive Bayes Classifiers works for text classification? Explain in brief.

19. A) Explain the framework used in the representation of semantics with example.

b) Briefly explain how decision trees can be used for text classification. What are the advantages and limitations

c) Explain how mapping words to properties using Python dictionaries can assist in text categorization.

UNIT – III

1. Define lexical similarity in the context of Natural Language Processing. Discuss its different types and explain how each category is used to measure the similarity between texts or words.
2. Write a python program to compute similarity between documents using cosine similarity. Include preprocessing, vectorization, and similarity calculation.
3. Discuss the role of Term Frequency (TF), Inverse Document Frequency (IDF), and TF-IDF in understanding the importance of terms in a document corpus.
4. Define Term Similarity and give one method to compute it.
5. How distance metrics are used to compute and measure similarities? List various distance measures and explain any one of them by taking an example.
6. Explain the role of TF-IDF in extractive text summarization. Why is it important?
7. Explain the concept of lexical similarity along with its categorization in detail.
8. Define Keyphrase Extraction and list one real-life application.
9. What is Information Extraction? Explain its types and how it contributes to the summarization and understanding of text.
10. What are the text normalization steps required preprocessing the text in text summarization? explain with a small program.
11. Explain the process of Topic Modeling using Latent Semantic Indexing (LSI). How does it help in document summarization and understanding? Illustrate with an example.
12. What is topic modeling? How does it help in summarizing and understanding large corpora?
13. a) What is Topic Modeling in the context of NLP? Explain Latent Dirichlet Allocation (LDA) with its assumptions and output interpretation.

    b) What is feature engineering in text analysis? Give two examples of commonly used features.

    c) Illustrate how Term Frequency (TF), Inverse Document Frequency (IDF), and their combination (TF-IDF) are used for measuring term importance. How does this aid in keyphrase extraction?
14. a) List and briefly explain any one application of Information Retrieval systems.

    b) Evaluate the role of Feature Engineering in improving the performance of text similarity and clustering models. Discuss with examples.

    c) Explain any two-distance metrics in detail to find the text similarity with an example each.
15. a) Explain the process of Automated Document Summarization. Compare and contrast extractive and abstractive summarization techniques with examples.

    b) Discuss the process and importance of Topic Modelling in large document collections. How does Latent Semantic Indexing (LSI) work?

    c) Define Information Extraction and list its main components.
16. a) Explain the workflow of document clustering using unsupervised machine learning.

    b) Explain the TextRank algorithm and describe the steps involved in using it for text summarization.

    c) Describe the concept of text similarity. How do TF-IDF and cosine similarity help in determining document similarity? Include examples.
17. A) Discuss information extraction, document summarization, and topic modeling with own example with its application

B) Summarize the various ways of extract key information from textual data and ways of summarizing large documents

C) Identify the three approaches towards reducing information overload, including keyphrase extraction, topic models, and automated document summarization

18. A) What is topic modeling and briefly discuss any two of the following
i) latent semantic indexing
ii) latent Dirichlet allocation and
iii) non-negative matrix factorization

B) Text data is unstructured and highly noisy. Document clustering is an unsupervised learning process. Discuss several concepts related to text similarity, distance metrics, and unsupervised ML algorithms

C) Discuss clustering models like k-means, affinity propagation, and Ward’s hierarchical clustering to build, analyze, and visualize clusters for real-world example.

19. A) Explain the following concepts needed for text summarization in detail.

Feature extraction

Feature matrix

B) Illustrate the cosine similarity in detail. Compute the cosine similarity between given two sentences.

Sentence 1: "I love NLP"

Sentence 2: "I enjoy learning NLP"

C) How does text summarization help address the problem of information overload?

20. A) Compute TF-IDF weights for each chunk in following example and return the top weighted phrases
Corpus:
"Artificial intelligence is a branch of computer science."
"Deep learning and machine learning are subsets of artificial intelligence."
"Natural language processing is an AI application."

B) Explain Levenshtein Edit Distance with three edit operations insert, delete and substitutio .compute the Levenshtein distance to convert “beleive” to “believe”

c) What is semantic similarity? Explain in brief with different categories .

UNIT – IV

1. How does sentiment analysis work on user-generated content like IMDb movie reviews? Describe the typical pipeline from text input to sentiment output.
2. What is Semantic Analysis in NLP? Discuss its key tasks, including Word Sense Disambiguation (WSD) and Named Entity Recognition (NER), with examples.
3. Discuss the working of a sentiment analysis system. How can machine learning models be trained for sentiment classification using IMDb movie reviews? Include steps such as preprocessing, feature extraction, model training, and evaluation.

4. Define Named Entity Recognition (NER). Explain its types and discuss its applications in information extraction.
5. Differentiate between semantics and sentiment in the context of text analysis.
6. Explain the framework used in the representation of semantics with example.
7. What are the key components of a sentiment analysis system?
8. List and explain any three applications of Semantic Analysis in real-world NLP tasks.
9. What is Named Entity Recognition (NER)? How does it differ from POS tagging? Explain how NER is implemented using NLTK or spaCy with an example.
10. Describe Word Sense Disambiguation (WSD). Why is it important in NLP? Explain any two approaches to perform WSD.
11. What are Homonyms, Homographs, Hyponyms, Hypernyms and Holonyms. Give examples of each.
12. What is WordNet? Describe how it helps in semantic analysis with an example using synonyms and hypernyms.
13. a) Explain the components of a Named Entity Recognition (NER) system. Demonstrate its working using a Python-based implementation with an example.

    b) Explain WordNet. How can it be used to find synonyms, antonyms, and hypernyms in NLP? Give suitable Python code using NLTK.

    c) What is the difference between polarity and subjectivity in sentiment analysis?
14. a) Perform sentiment analysis on IMDb movie reviews using any ML model. Explain preprocessing, model, and accuracy evaluation.

    b) How do 5 semantic analysis topics help a virtual assistant understand user commands? Use example like “Book a flight to Paris”.

    c) Define entailment in semantic analysis. Differentiate between the following pairs with suitable examples: (i) Homonyms vs Homographs, (ii) Hyponyms vs Hypernyms.
15. a) Explain Word Sense Disambiguation (WSD) using the Lesk Algorithm. Illustrate your explanation with an example and Python code implementation.

    b) Write Python code to perform a basic sentiment analysis using TextBlob.

    c) Implement a Named Entity Recognition (NER) task using spaCy. Analyze and explain the types of entities detected.
16. a) Discuss various semantic similarity measures. Use WordNet to compare semantic similarity between word pairs.

    b) What is semantic analysis? Explain its importance in NLP.

    c) Write a Python program to perform sentiment analysis on IMDb reviews using TF-IDF and a classifier. Briefly explain each step.
17. A) Text semantics specifically deals with understanding the meaning of text or language. Discuss the following topics under semantic analysis.

    i) Exploring WordNet and synsets

    ii) Analyzing lexical semantic relations iii) Word sense disambiguation

    iv) Named entity recognition

    B) Differentiate rule – based, machine learning based and deep learning based approaches for Named Entity recognition.

    C) List and explain any three applications of semantic analysis with real time example.
18. A) Sentiment analysis is to analyze a body of text for understanding the opinion expressed by it and other factors like mood and modality. Demonstrate sentiment analysis for dataset of movie reviews obtained from the Internet Movie Database (IMDb)to find text classification

such that the classes to predict here are positive and negative sentiment corresponding to the movie reviews.

B) Develop a python program to find similarity between two documents given below.
Document1=”Natural Language Processing with Python”
Document2=” Natural Language Processing with Java”

C) Compare semantics and sentiment in the context of text analysis.

19. A) Explain Word Sense Disambiguation in detail with an example including steps in Lesk algorithm.

B) With the help of IMDb Movie Reviews ,explain how sentiment analysis works .

C) Why NER is important? List different types of named entities.

20. A) Explain how word entailments are used in analysing Lexical Semantic Relations (ALSR)” in detail with code.

B) List different techniques of Semantic Representations. Explain logic-based representations in detail.

C)Define Hyponyms and Hypernyms with respect to word ”tree” including code snippet.