FACULTY OF COMPUTER SCIENCE AND ENGINEERING

Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi

Lab Duration:3 hr. AI361L NLP Lab Marks: 10 Lab No: 1 Instructor: Memoona Saleem

**Lab Activity**

**Lab Task: Communicating with ChatGPT and Exploring NLP Applications**

This lab focuses on interacting with ChatGPT to understand its capabilities in processing natural language, especially when dealing with ambiguous sentences. Additionally, students will explore different practical applications of Natural Language Processing (NLP) to gain insights into the field's breadth and depth.

**1. Interact with ChatGPT:**

Begin by crafting a series of sentences that you will use to communicate with ChatGPT. Aim for a mix of straightforward and ambiguous sentences to test the model's understanding and contextual interpretation capabilities.

- Examples of ambiguous sentences might include:

- "Can you tell me the time the clock stopped?"

- "I saw the man with a telescope."

- "They are cooking apples."

- Record ChatGPT's responses to each sentence, noting particularly how it handles the ambiguous ones.

**2. Analyze ChatGPT's Performance:**

For each interaction, analyze whether ChatGPT correctly understood the context of your sentences, especially the ambiguous ones. Identify any patterns in responses where the model may misinterpret the intended meaning.

1. **Explore NLP Practical Applications**

For each application, consider the following:

- The specific NLP techniques and models used.

- The benefits and limitations of applying NLP in this context.

- Any potential future developments or improvements in NLP that could enhance it

FACULTY OF COMPUTER SCIENCE AND ENGINEERING

Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi

Lab Duration:3 hr. AI361L NLP Lab Marks: 10 Lab No: 2 Instructor: Memoona Saleem

**Task Statement:**

In this NLP lab task, you will work with a text file containing a collection of recipes. Your objective is to extract four different types of information from each recipe: the amount, measure type, ingredient, and instructions.

**Instructions:**

**1. Read Text File:**

- Begin by reading the text file containing the collection of recipes. You can use Python's file handling capabilities or any suitable library for text file processing.

**2. Extract Information:**

- For each recipe in the text file, extract the following four types of information:

- Amount: The quantity of the ingredient (e.g., "1", "2.5").

- Measure Type: The unit of measurement for the ingredient (e.g., "cup", "teaspoon").

- Ingredient: The name of the ingredient (e.g., "flour", "sugar").

- Instructions: The cooking instructions or steps for preparing the recipe.

**3. Data Processing:**

- Utilize natural language processing (NLP) techniques, such as tokenization and part-of-speech tagging, to extract the required information accurately.

- Consider using regular expressions to identify patterns related to amounts, measure types, and ingredients within the recipe text.

**4. Output:**

- Organize the extracted information for each recipe into a structured format, such as a list of dictionaries or a pandas DataFrame.

FACULTY OF COMPUTER SCIENCE AND ENGINEERING

Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi

Lab Duration:3 hr. AI361L NLP Lab Marks: 10 Lab No: 3 Instructor: Memoona Saleem

**Information Extraction Using Regular Expressions**

Wikipedia contains a wealth of information in a semi-structured format. While reading a Wikipedia article, we can easily identify key pieces of information such as a person's name, age, date of birth, spouse, and net worth. However, programmatically extracting this information requires parsing the unstructured text and capturing the required details.

**Task 1**

**1. Data Collection:**

- Choose a public figure's Wikipedia page (e.g., Elon Musk, Bill Gates, etc.).

- Copy a portion of the text that includes the individual's name, age, date of birth, spouse(s), and net worth and detail about their life

**2. Information Extraction:**

- Write a Python script using regular expressions to extract the following information from the text you copied:

- Full Name

- Age

- Date of Birth

- Spouse(s) Name(s)

- Net Worth

And 5 more

- Pay attention to the different ways this information can be presented in text. For example, the date of birth might be in the format "June 28, 1971" or "1971-06-28".

- Net worth might be presented with various currencies or in a range (e.g., "USD $20 billion" or "20–30 billion dollars").

FACULTY OF COMPUTER SCIENCE AND ENGINEERING

Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi

Lab Duration:3 hr. AI361L NLP Lab Marks: 10 Lab No: 4 Instructor: Memoona Saleem

**Task: Creating a Simple Search Engine using Bag of Words**

The bag of words model is a way of representing text data when modeling text with machine learning algorithms. The model ignores the order of words but maintains multiplicity.

**Preparation:**

- Provide students with a dataset consisting of various text documents. This could be anything from news articles, plot summaries of movies, tweets, etc.

- The dataset should be preprocessed to remove any HTML tags, special characters, and should be case-normalized.

**Task Steps:**

**1. Tokenization:**

- Ask students to write a function to tokenize the documents into words. Each word represents a feature of the document.

**2. Stop Words Removal:**

- Provide a list of stop words.

- Students should filter out these common words that add no significant value to the meaning of the document.

**3. Frequency Distribution:**

- Students will create a frequency distribution for each document, counting how often each word appears.

**4. Bag of Words Model Creation:**

Using the frequency distributions, students will construct a bag of words model for the entire dataset.

- They will create a matrix where each row represents a document, each column represents a word in the dataset's vocabulary, and the entries are the word frequencies.



**5. Implementing the Search Function:**

- Students will write a function that takes a query as input and converts it into its bag of words representation.

- The function will then compare the query's bag of words with the dataset's bag of words to find the most relevant documents.

- Relevance can be determined by the number of matching words and their frequencies.

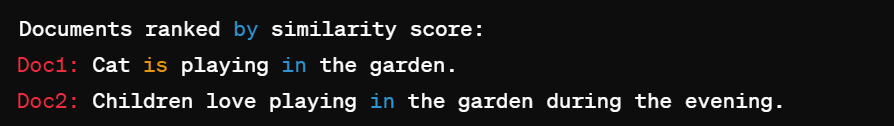
**6. Testing the Search Engine:**

- Each student will test their search engine with a set of queries to retrieve relevant documents.

- They should analyze which queries worked well and which didn’t and discuss why.

**Sample input & Output :**





FACULTY OF COMPUTER SCIENCE AND ENGINEERING

Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi

Lab Duration:3 hr. AI361L NLP Lab Marks: 10 Lab No: 5 Instructor: Memoona Saleem

**1. TF-IDF Analysis Task:**

Create a mini search engine for a set of documents.

**-Task:**

- you should come up with a corpus of documents (these could be articles, book chapters, or any collection of text).

- you need to write a python program that calculates the TF-IDF score for each word in each document.

- you should then create a simple search function that uses the TF-IDF scores to find and rank documents based on a query of one or more words.

**-Extension:**

- You could expand the search engine to include basic preprocessing of the text such as tokenization, stopping, stemming, or lemmatization.

- Discuss how different preprocessing steps might affect the TF-IDF scores and the search results.

1. **N-gram Model Task:**

- come up with a text dataset (like a set of tweets, sentences from books, or movie subtitles).or you can use previously used data

- you will create an N-gram model that predicts the next word(s) based on the previous word(s) in a sequence.

- you should then use their model to generate new sentences based on a given starting sequence of words.

**- Extension:**

- Explore the effects of smoothing techniques on the model's performance.

**Deliverable:** submit your code file via email

FACULTY OF COMPUTER SCIENCE AND ENGINEERING

Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi

Lab Duration:3 hr. AI361L NLP Lab Marks: 10 Lab No: 6 Instructor: Memoona Saleem

**1. Word Similarity and Analogy Task**

**- Objective:**

To explore and understand the semantic relationships captured by GloVe and Word2Vec embeddings.

**- Task:**

Students will use pre-trained GloVe and Word2Vec models to find words that are most similar to a given set of words (e.g., "king," "computer," "Paris") and solve word analogies (e.g., "king - man + woman = ?").

**- Outcome:**

This task will help students grasp how word embeddings capture semantic relationships and analogies, illustrating the models' ability to understand context and meaning.

**2. Visualization of Word Embeddings**

**- Objective:**

To visualize the high-dimensional word embeddings in a two-dimensional space.

**- Task:**

Students will use dimensionality reduction techniques like PCA (Principal Component Analysis) or t-SNE to visualize the word embeddings generated by GloVe and Word2Vec. They can then explore how similar words cluster together.

**- Outcome:**

This visual task will aid in understanding the geometric relationships between words in the embedding space, highlighting how similarity and context are represented.

**3. Text Classification Using Word Embeddings**

**- Objective:**

To apply word embeddings in a practical NLP task.

**- Task:** Students will build a simple text classification model (e.g., for sentiment analysis or topic categorization) using features derived from GloVe or Word2Vec embeddings. They can compare the performance of models using raw text features, Word2Vec embeddings, and GloVe embeddings.

**Submission Guidelines:**

Write code in python .

Submit jupyter notebook via email.

Compile your findings, code snippets (if any), and discussions into a Word document.

Ensure your document is well-organized, with clear headings for each part and task.

Include any references used for your theoretical explanations and practical experiments.