Lab Report: **02**

Subject: CSE - 477

Section: 1

Submitted By:

Syed Musayedul Hussain

ID: 2021-2-60-015

Submitted to:

Amit Mandal
Lecturer

Department of Computer Science and Engineering ,East West University

Date: 12/07/2025

1. First, I used this video: https://www.youtube.com/watch?v=i_LwzRVP7bg&t=9270s and extracted the comments and captions by following the instructions from the last lab. Then, I installed all the necessary libraries

```
pip install pandas matplotlib nltk webvtt-py
```

```
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (2.2.2)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.11/dist-packages (3.10.0)
Requirement already satisfied: nltk in /usr/local/lib/python3.11/dist-packages (3.9.1)
Collecting webvtt-py
  Downloading webvtt_py-0.5.1-py3-none-any.whl.metadata (3.4 kB)
Requirement already satisfied: numpy>=1.23.2 in /usr/local/lib/python3.11/dist-packages (from pandas) (2.0.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas) (2025.2)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (1.3.2)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (4.58.5)
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (1.4.8)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (24.2)
Requirement already satisfied: pillow>=8 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (11.2.1)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.11/dist-packages (from matplotlib) (3.2.3)
Requirement already satisfied: click in /usr/local/lib/python3.11/dist-packages (from nltk) (8.2.1)
Requirement already satisfied: joblib in /usr/local/lib/python3.11/dist-packages (from nltk) (1.5.1)
Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.11/dist-packages (from nltk) (2024.11.6)
Requirement already satisfied: tqdm in /usr/local/lib/python3.11/dist-packages (from nltk) (4.67.1)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2->pandas) (1.17.0)
Downloading webvtt_py-0.5.1-py3-none-any.whl (19 kB)
Installing collected packages: webvtt-py
Successfully installed webvtt-py-0.5.1
```

```
[5] pip install scikit-learn
```

```
[6] import nltk
```

```
[13] nltk.download('punkt')
     nltk.download('stopwords')
     nltk.download('wordnet')
     nltk.download('omw-1.4')
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
[nltk_data] Downloading package omw-1.4 to /root/nltk_data...
[nltk_data]   Package omw-1.4 is already up-to-date!
True
```

```
nltk.download('punkt_tab')
```

```
[nltk_data] Downloading package punkt_tab to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt_tab.zip.
True
```

2. Python script processes a raw text file containing YouTube comments and structures the data into a clean and organized format using a DataFrame. The text file is assumed to contain usernames, timestamps (like "2 weeks ago"), and comment text in a sequential format. The script identifies each comment by detecting the username and matching the corresponding timestamp using a regular expression. It then collects the lines of text that belong to that comment while skipping irrelevant lines like "Reply" or "...more". Each comment is saved as a

dictionary containing the username, timestamp, and full comment text. Finally, all the structured data is returned as a pandas DataFrame, making it easier to analyze or export for further use.

```python
import pandas as pd
import re

def structure_comments_from_txt(filepath='/content/drive/MyDrive/CSE477/comment.txt'):
    with open(filepath, 'r', encoding='utf-8') as f:
        lines = f.readlines()
    comments_data = []
    time_regex = re.compile(r'.*(second|minute|hour|day|week|month|year)s? ago.*', re.IGNORECASE)
    i = 0
    while i < len(lines):
        line = lines[i].strip()
        if i + 1 < len(lines) and time_regex.match(lines[i+1].strip()):
            username = line
            timestamp = lines[i+1].strip()
            comment_text = []
            i += 2
            while i < len(lines) and not (i+1 < len(lines) and time_regex.match(lines[i+1].strip())):
                comment_line = lines[i].strip()
                if comment_line and not comment_line.lower() in ['reply', '...more']:
                    comment_text.append(comment_line)
                i += 1
            if comment_text:
                comments_data.append({
                    'username': username,
                    'timestamp_text': timestamp,
                    'comment_text': ' '.join(comment_text)
                })
        else:
            i += 1
```

```
                username       timestamp_text  \
0    @risebyliftingothers  2 years ago (edited)
1  @auliamardhatillah2240          2 years ago
2            @limwei2634          2 years ago
3         @nancykataria08         2 months ago
4                @jpbaugh          2 years ago

                                        comment_text
0  ₹100.00 Thanks for an amazingly simplified app...
1  Yesterday I click on a video called 'learning ...
2  I've been trying to learn ML for quite awhile ...
3  No fancy words, just simple English and the ri...
4  For anyone getting an error related to convert...
```

3. Extracts and organizes subtitle text from a `.vtt` caption file. It uses the `webvtt` library to read caption content or falls back to manual parsing if needed. The collected text is cleaned, joined, and then split into individual sentences based on punctuation. The final structured output is stored in a pandas DataFrame, which makes the caption sentences easy to view, manipulate, or use in further analysis such as natural language processing or sentiment analysis.

```python
import webvtt
import re
import pandas as pd

def structure_captions_from_vtt(filepath):
    try:
        full_text = ' '.join([caption.text.strip() for caption in webvtt.read(filepath)])
    except Exception as e:
        print(f"Error reading VTT file: {e}. Falling back to manual line reading.")
        captions = []
        with open(filepath, 'r', encoding='utf-8') as f:
            for line in f:
                line = line.strip()
                if '-->' not in line and line and not line.isdigit() and 'WEBVTT' not in line:
                    captions.append(line)
        full_text = ' '.join(captions)
    sentences = re.split(r'(?<=[.!?]) +', full_text)
    return pd.DataFrame(sentences, columns=['caption_sentence'])

captions_df = structure_captions_from_vtt('/content/drive/MyDrive/CSE477/Caption.vtt')
print(captions_df.head())
```

```
                                caption_sentence
0  kylie ying has worked at many kylie ying has w...
1             so let's actually just accuracy is 81.
2             so let's actually just accuracy is 81.
3  so let's actually just\nmake this five make th...
4                 of people that have covid is 53
```

What can I help you build?

4. Save clean comments and captions file in csv format for future labs-

```
[40] comments_df.to_csv('/content/drive/MyDrive/CSE477/comments_output.csv', index=False)
```

```
[41] captions_df.to_csv('/content/drive/MyDrive/CSE477/captions_output.csv', index=False)
```

5. **Complexity I faced:**
   The primary challenge with the caption data is its formatting. Since there are no defined start or end points, extracting meaningful lines becomes quite difficult.