module03-analyzing-text-content-natural-language-processing-30pct/3-2-2-computer-lab.qmd

List of Figures

List of Tables

inferential analysis

```python
import pandas as pd
import spacy
import os

# Load text_df from the TSV file
input_file_path = '/home/sol-nhl/rnd/d/quarto/osm-cca-nlp/csv/text_data.tsv'
input_file_path = '/content/osm-cca-nlp/csv/text_data.tsv'
text_df = pd.read_csv(input_file_path, sep='\t')

# Load the spaCy model (small English model is used here)
nlp = spacy.load("en_core_web_sm")

# Initialize an empty list to store sentence data
sentence_data = []

# Iterate over the cleaned text in the DataFrame
for index, row in text_df.iterrows():
    doc = nlp(row['cleaned_text'])  # Process the cleaned text with spaCy

    # Iterate over the sentences in the document
    for i, sentence in enumerate(doc.sents):
        sentence_data.append({
            'id': row['id'],           # Original text ID
            'sentence_number': i + 1,  # Sentence number (starting from 1)
            'sentence_text': sentence.text.strip()  # Sentence text
        })

# Create a new DataFrame with the sentence data
sentence_df = pd.DataFrame(sentence_data)

# Save the sentence_df DataFrame as a TSV file
output_file_path = '/home/sol-nhl/rnd/d/quarto/osm-cca-nlp/csv/sentence_data.tsv'
output_file_path = '/content/osm-cca-nlp/csv/sentence_data.tsv'
sentence_df.to_csv(output_file_path, sep='\t', index=False)
```

```python
# Display the sentence DataFrame
print(sentence_df)
```

code discussion

This code chunk outlines a procedure for processing text data stored in a TSV file using Python, with a specific focus on sentence extraction and content analysis through natural language processing (NLP) techniques.

1. **Loading the Text Data**:
   - The process begins by loading a TSV file containing text data into a pandas DataFrame using `pd.read_csv()`. The file is located at a specified path (`input_file_path`), and the `sep='\t'` parameter indicates that the file is tab-separated. This DataFrame, `text_df`, holds the cleaned text data along with associated metadata like unique IDs.
2. **Initializing the NLP Model**:
   - The `spaCy` library, a powerful NLP tool, is loaded using `spacy.load("en_core_web_sm")`. This initializes a small English model that will be used to process the text data, allowing for sophisticated linguistic analysis such as tokenization, lemmatization, and sentence segmentation.
3. **Processing Text and Extracting Sentences**:
   - The core of the procedure involves iterating over each row of the `text_df` DataFrame. For each row, the cleaned text is passed through the spaCy model, which processes the text and divides it into sentences (`doc.sents`). Each sentence is then extracted and stored in a list called `sentence_data` along with its corresponding unique ID and sentence number. This step is crucial for breaking down the text into manageable units, facilitating more granular content analysis.
4. **Creating and Saving the Sentence Data**:
   - After all sentences have been extracted, the `sentence_data` list is converted into a new DataFrame, `sentence_df`. This DataFrame organizes the sentences with their associated metadata, making it easier to analyze or manipulate the content on a sentence-by-sentence basis.

    - The `sentence_df` DataFrame is then saved as a TSV file at a specified output path (`output_file_path`) using `sentence_df.to_csv()`, ensuring the data is stored in a structured and accessible format for future use.
5. **Displaying the Result**:
    - Finally, the `sentence_df` DataFrame is printed to the console, allowing for a quick inspection of the extracted sentence data. This step helps verify that the sentence extraction and data storage processes have been executed correctly.

This procedure effectively leverages Python and NLP techniques to transform raw text data into a structured format, breaking it down into sentences that can be further analyzed for various content analysis tasks. The use of `spaCy` allows for accurate sentence segmentation, which is a foundational step in many NLP workflows.