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

List of Figures

List of Tables

inferential analysis (part 2)

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

# Load sentence_df from the TSV file
input_file_path = '/home/sol-nhl/rnd/d/quarto/osm-cca-nlp/csv/sentence_data.tsv'
input_file_path = '/content/osm-cca-nlp/csv/sentence_data.tsv'
sentence_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 token data
token_data = []

# Iterate over the sentences in the sentence_df DataFrame
for index, row in sentence_df.iterrows():
    doc = nlp(row['sentence_text'])  # Process the sentence text with spaCy

    # Iterate over the tokens in the sentence
    for j, token in enumerate(doc):
        token_data.append({
            'id': row['id'],                        # Original text ID
            'sentence_number': row['sentence_number'],  # Sentence number
            'token_number': j + 1,                  # Token number (starting from 1)
            'token_text': token.text,               # Token text
            'token_lemma': token.lemma_,            # Token lemma
            'token_pos': token.pos_,                # Token part of speech
            'token_entity': token.ent_type_         # Token entity type (if any)
        })

# Create a new DataFrame with the token data
token_df = pd.DataFrame(token_data)

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

# Display the token DataFrame
print(token_df)
```