Text Mining Tutorial 2:

Part-of-Speech (POS) Tagging

Named Entity Recognition (NER)

Visualization

Prof. Hsing-Kuo Pao
Teaching Assistant: Ghaluh Indah Permata Sari

TAIWAN TECH
NATIONAL TAIWAN UNIVERSITY OF SCIENCE AND TECHNOLOGY

Outline

- Part-of-Speech (POS) Tagging
- Name Entity Recognition (NER)
- Visualization

Part-of-Speech (POS) Tagging

a.k.a grammatical tagging

Objective: to *assign* a grammatical category (part of speech) to each word in a sentence based on its syntactic role and context within the sentence.

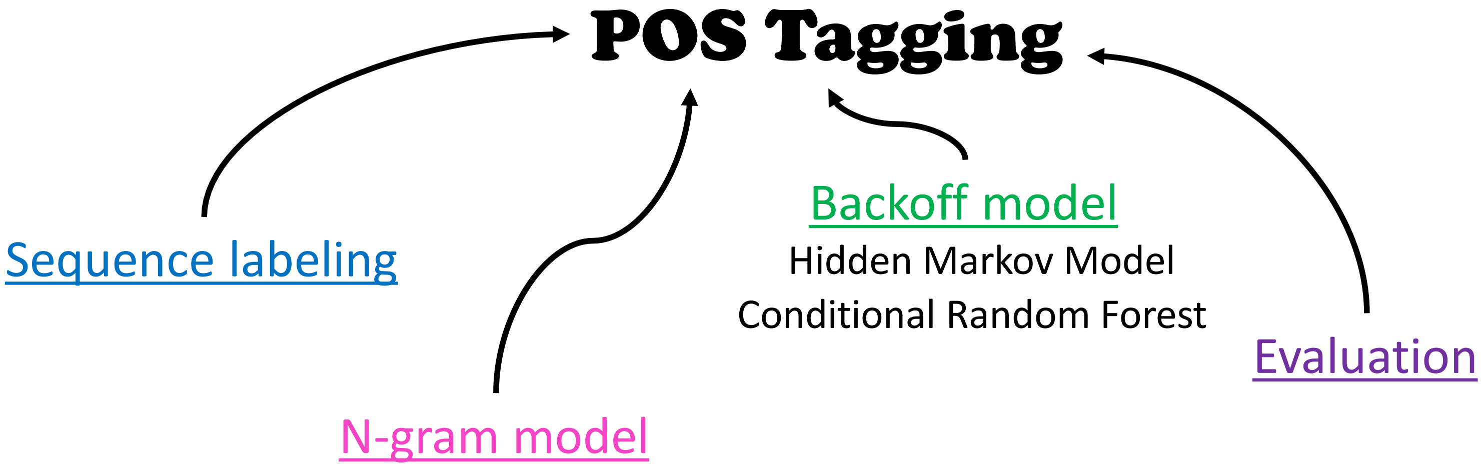
Purpose:

- Linguistic analysis
- Text understanding
- Feature extraction
- Parsing and syntax analysis
- Machine learning and language modeling

Part-of-Speech (POS) Tagging

Approaches:

- Statistical models and Machine learning
- Rule-based models



POS Tagging

Sequence labeling

N-gram model

Backoff model

Hidden Markov Model

Conditional Random Forest

Evaluation

Part-of-Speech (POS) Tagging: Pros & Cons

Rule-based model

- + Simple to implement and understand.
- + It doesn’t require a lot of computational resources or training data.
- + It can be easily customized to specific domains or languages.

- – Less accurate than statistical taggers
- – Limited by the quality and coverage of the rules
- – It can be difficult to maintain and update

Statistical model

- + More accurate than rule-based taggers.
- + Don’t require a lot of human-written rules.
- + Can learn from large amounts of training data.

- – Requires more computational resources and training data
- – It can be difficult to interpret and debug
- – Can be sensitive to the quality and diversity of the training data

Part-of-Speech (POS) Tagging: Steps

Rule-based

Example: using pos_tag from NLTK library

1. Tokenize the text
2. Apply Rules → pos_tag
3. Interpret the results

Machine Learning

Example: Hidden Markov Model (HMM)

1. Define the HMM
2. Instantiate the HMM
3. Prepare the input data
4. Apply the Viterbi algorithm
5. Interpret the results

Named Entity Recognition (NER)

Objective: to automatically identify and classify named entities within a text into predefined categories such as persons, organizations, locations, dates, quantities, and more.

Note:

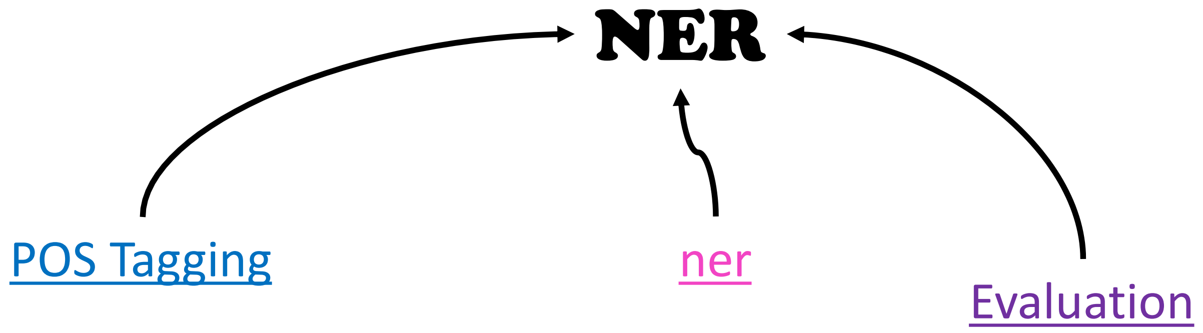
Named entities are specific words or phrases that refer to entities with unique names, such as people, places, organizations, and numerical expressions.

Named Entity Recognition (NER): Steps

NER

Example: using SpaCy library

1. Tokenize the text
2. POS Tagging
3. Feature Extraction
4. NER model
5. Output



NER

POS Tagging

ner

Evaluation

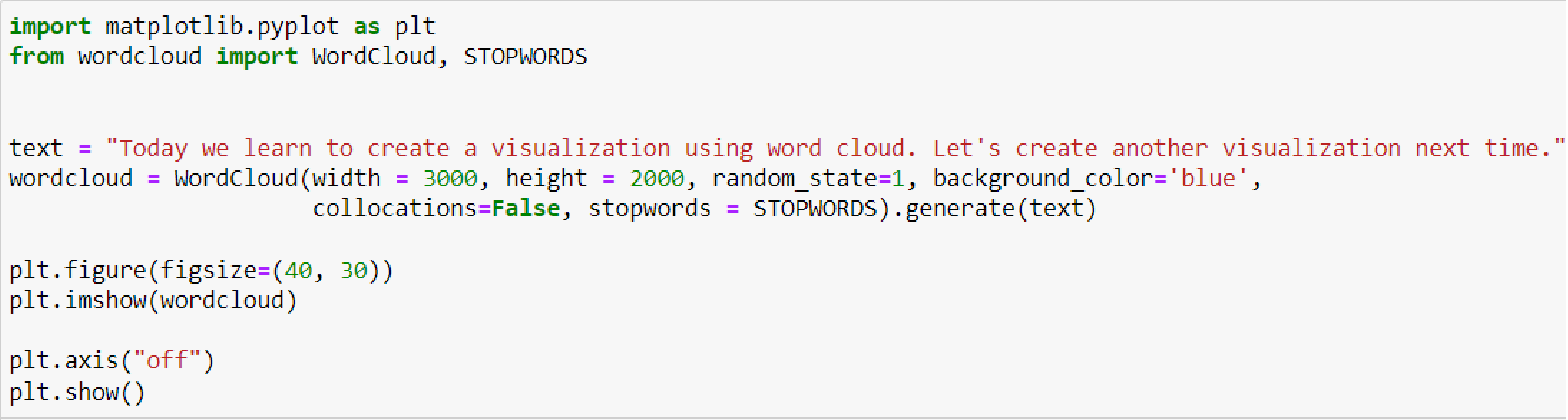
Visualization

- **Text visualization** is the technique of using graphs, charts, or word clouds to showcase written data in a visual manner
- **Visualization type**: word cloud, scatter text, wordnet, chart and histogram, map, etc.
- **Purpose:**
  1. Summarize large amounts of text.
  2. Make text data easy to understand.
  3. Discover hidden trends and patterns.
  4. Provides quick insight into the most relevant keywords in a text.

Visualization: code example

Visualization using Word cloud

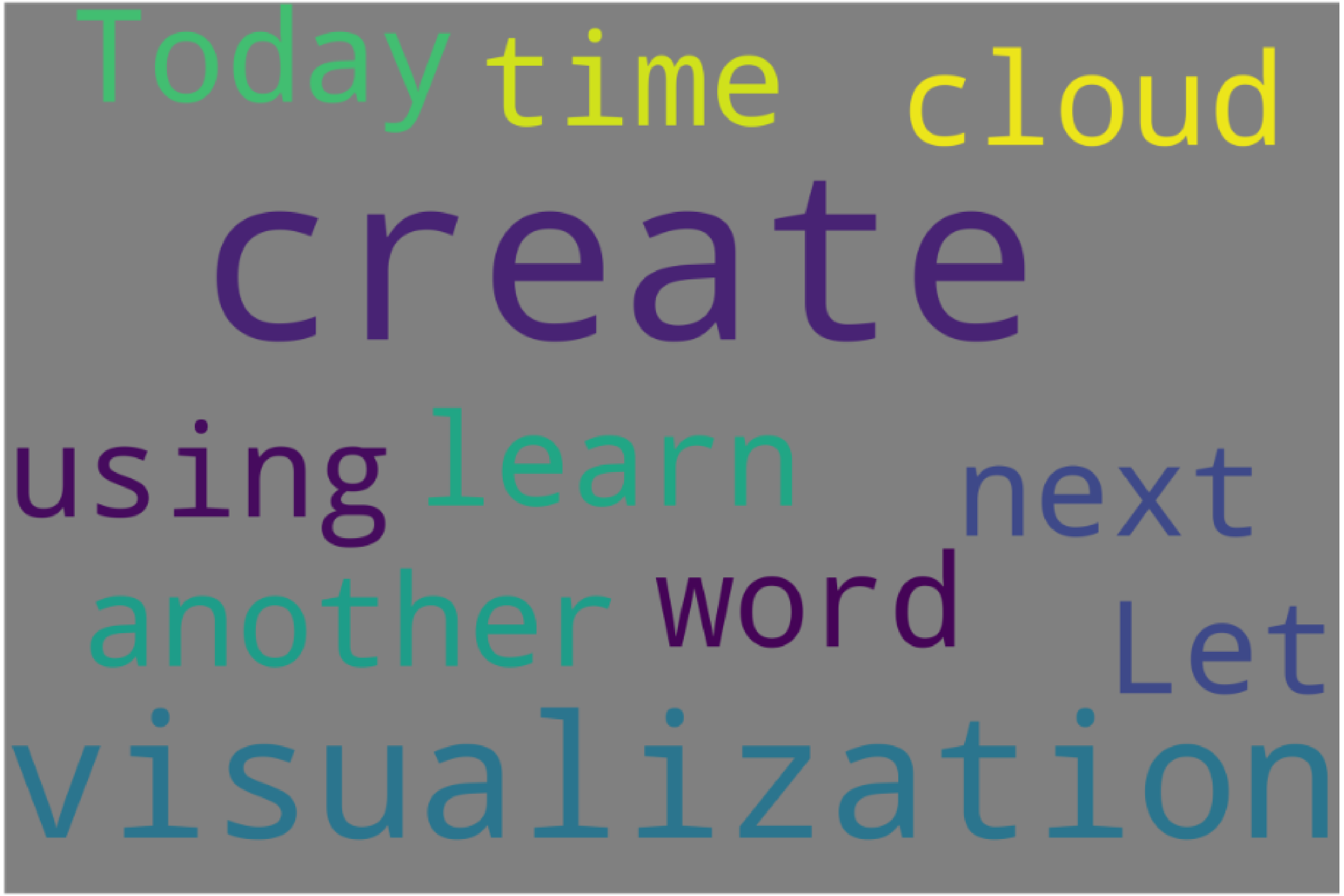
Code



```
import matplotlib.pyplot as plt
from wordcloud import WordCloud, STOPWORDS


text = "Today we learn to create a visualization using word cloud. Let's create another visualization next time."
wordcloud = WordCloud(width = 3000, height = 2000, random_state=1, background_color='blue',
                      collocations=False, stopwords = STOPWORDS).generate(text)

plt.figure(figsize=(40, 30))
plt.imshow(wordcloud)

plt.axis("off")
plt.show()
```

Visualization: code example

Output



Thank you
Q & A