| EXP NO: 5b | INFORMATION EXTRACTION USING SPACY |
|---|---|
| DATE: 4/9/25 | |

**Aim:**

To perform tokenization, part-of-speech tagging, and named entity recognition (NER) using the spaCy natural language processing library.

## Program:

**Step 1: Import Required Libraries**

```
import kagglehub import pandas
as pd import spacy import re
```

**Step 2: Download and Load Dataset**

```
# Download dataset
path = kagglehub.dataset_download("snap/amazon-fine-food-reviews")

# Load the reviews CSV file import
os
df = pd.read_csv(os.path.join(path, "Reviews.csv"))


# Select only the 'Text' column and limit to 1000 reviews reviews
= df['Text'].dropna()[:1000]
```

**Step 3: Install and Load spaCy Model**

```
!pip install spacy pandas
!python -m spacy download en_core_web_sm
```

Load the **English language model**: `nlp =`

`spacy.load("en_core_web_sm")` **Step**

**4: Text Preprocessing**

```
def preprocess(text):
                          text = text.lower()                        # convert
to lowercase     text = re.sub(r'[^a-z\s]', '', text)     # remove
punctuation and numbers     return text
 reviews_cleaned =
reviews.apply(preprocess)
```

**Step 5: Tokenization**

```
def tokenize(text):
doc = nlp(text)
tokens = [token.text
```

```
for token in doc if
token.is_alpha and not
token.is_stop]
return tokens

tokens_sample = tokenize(reviews_cleaned.iloc[0]) print(tokens_sample[:20])
```

**Step 6: Part-of-Speech (POS) Tagging**

```
doc = nlp(reviews_cleaned.iloc[50])
pos_tags = [(token.text, token.pos_) for token in doc if token.is_alpha and not
token.is_stop] print("POS tags:", pos_tags[:15]) Step 7: Named Entity Recog
```

**(NER)**

```
ner_entities = [] for review
in reviews_cleaned:
    doc = nlp(review)
    ner_entities.extend([(ent.text, ent.label_) for ent in doc.ents])
 print("NER                    entities:",
ner_entities[:10]) Step 8: Visualize Named
```

**Entities**

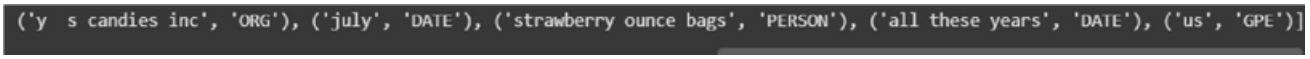
```
from spacy import displacy
displacy.render(doc, style="ent", jupyter=True)
```

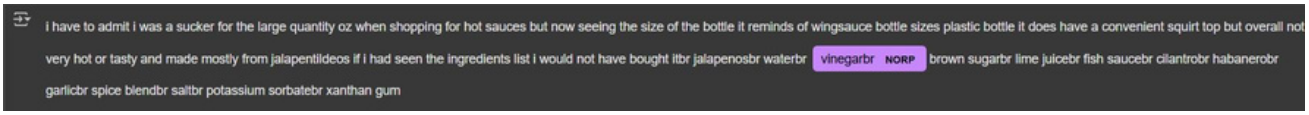
## Output:







**Result:**

The text data was successfully processed using spaCy for tokenization, POS tagging, and NER, providing structured information about entities, word types, and grammatical relationships.