| | | | |
|---|---|---|---|
| **Course Name:** | **Data Analysis Laboratory (216H03L501 )** | **Semester:** | **V** |
| **Date of Performance:** | **27/10/2025** | **DIV/ Batch No:** | **DA_4** |
| **Student Name:** | **Shreyans Tatiya** | **Roll No:** | **16010123325** |

**TITLE :** NLP on clinical data (Finding data from Literature)

**AIM:** Implement NLP on clinical data

**Expected Outcome of Experiment:**

**Books/ Journals/ Websites referred:**

**Sample case Study**
https://towardsdatascience.com/clinical-named-entity-recognition-using-spacy-5ae9c002e86f

**Theory**

**Named entity recognition (NER)** is a natural language processing (NLP) method that extracts information from text. NER involves detecting and categorizing important information in text known as named entities. Named entities refer to the key subjects of a piece of text, such as names, locations, companies, events and products, as well as themes, topics, times, monetary values and percentages.

NER is also referred to as entity extraction, chunking and identification. It's used in many fields in artificial intelligence (AI), including machine learning (ML), deep learning and neural networks. NER is a key component of NLP systems, such as chatbots, sentiment analysis tools and search engines. It's used in healthcare, finance, human resources (HR), customer support, higher education and social media analysis.

**The purpose of NER**

NER identifies, categorizes and extracts the most important pieces of information from unstructured text without requiring time-consuming human analysis. It's particularly

useful for quickly extracting key information from large amounts of data because it automates the extraction process.

As NER models improve their ability to correctly identify important information, they are helping improve AI systems in general. These systems are enhancing AI language comprehension capabilities in areas such as summarization and translation systems and the ability of AI systems to analyze text.
NER uses algorithms that function based on grammar, statistical NLP models and predictive models. These algorithms are trained on data sets that people label with predefined named entity categories, such as people, locations, organizations, expressions, percentages and monetary values. Categories are identified with abbreviations; for example, LOC is used for location, PER for persons and ORG for organizations.

**Dataset:**

**https://www.mtsamples.com**

**https://www.kaggle.com/datasets/tboyle10/medicaltranscriptions Language:**

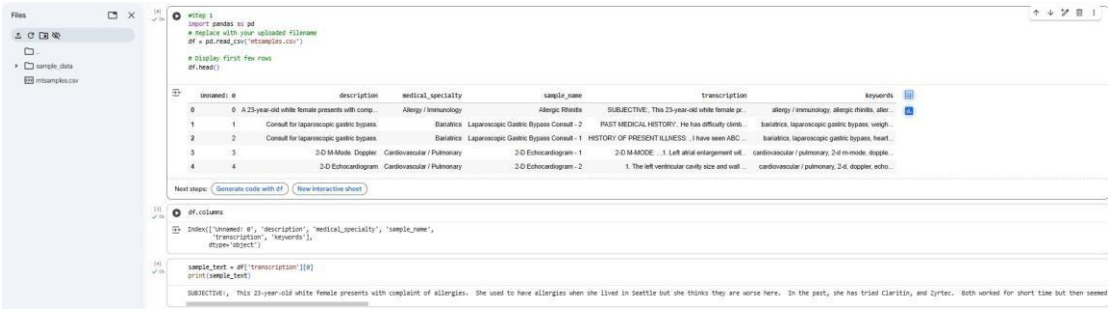
**C/C++/Java/Python/other – Choice of student**

**Steps**: 1. Download dataset (Given).
2. Pre-process the dataset if needed.
3. Use spaCy library or any other similar one
4. Visualize and Display result

**List out the library used with justification**

1. spaCy – For performing Named Entity Recognition (NER) using the pre-trained model en_core_web_sm.
2. en_core_web_sm model – A lightweight English NLP model trained for tasks such as tokenization, part-of-speech tagging, dependency parsing, and NER.
3. spacy.explain() – Used to provide human-readable explanations for recognized entity labels.

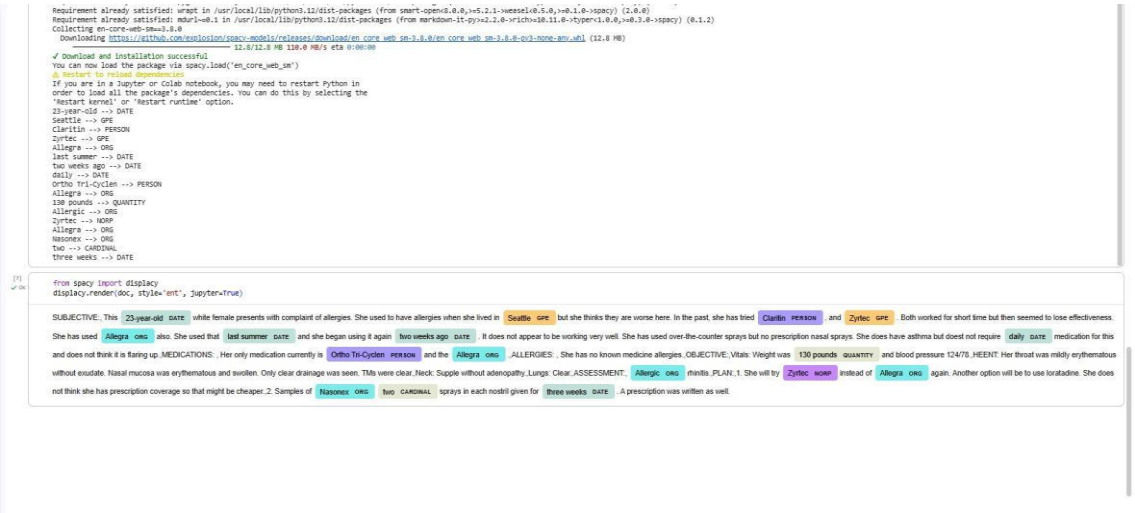
**Implementation details (recommended provide the comments)**



```
[6]
✓ 25s
# Step 3: NER with spaCy
!pip install spacy
!python -m spacy download en_core_web_sm
import spacy
nlp = spacy.load("en_core_web_sm")
doc = nlp(clean_text)

for ent in doc.ents:
    print(f"{ent.text} --> {ent.label_}")
```

```
Requirement already satisfied: spacy in /usr/local/lib/python3.12/dist-packages (3.8.7)
Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.11 in /usr/local/lib/python3.12/dist-packages (from spacy) (3.0.12)
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (1.0.5)
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (1.0.13)
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.0.11)
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in /usr/local/lib/python3.12/dist-packages (from spacy) (3.0.10)
Requirement already satisfied: thinc<8.4.0,>=8.3.4 in /usr/local/lib/python3.12/dist-packages (from spacy) (8.3.6)
Requirement already satisfied: wasabi<1.2.0,>=0.9.1 in /usr/local/lib/python3.12/dist-packages (from spacy) (1.1.3)
Requirement already satisfied: srsly<3.0.0,>=2.4.3 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.5.1)
Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.0.10)
Requirement already satisfied: weasel<0.5.0,>=0.1.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (0.4.1)
Requirement already satisfied: typer<1.0.0,>=0.3.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (0.20.0)
Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (4.67.1)
Requirement already satisfied: numpy>=1.19.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.0.2)
Requirement already satisfied: requests<3.0.0,>=2.13.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.32.4)
Requirement already satisfied: pydantic!=1.8,!=1.8.1,<3.0.0,>=1.7.4 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.11.10)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.12/dist-packages (from spacy) (3.1.6)
Requirement already satisfied: setuptools in /usr/local/lib/python3.12/dist-packages (from spacy) (75.2.0)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (25.0)
Requirement already satisfied: langcodes<4.0.0,>=3.2.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (3.5.0)
Requirement already satisfied: language-data>=1.2 in /usr/local/lib/python3.12/dist-packages (from langcodes<4.0.0,>=3.2.0->spacy) (1.3.0)
Requirement already satisfied: annotated-types>=0.6.0 in /usr/local/lib/python3.12/dist-packages (from pydantic!=1.8,!=1.8.1,<3.0.0,>=1.7.4->spacy) (0.7.0)
Requirement already satisfied: pydantic-core==2.33.2 in /usr/local/lib/python3.12/dist-packages (from pydantic!=1.8,!=1.8.1,<3.0.0,>=1.7.4->spacy) (2.33.2)
Requirement already satisfied: typing-extensions>=4.12.2 in /usr/local/lib/python3.12/dist-packages (from pydantic!=1.8,!=1.8.1,<3.0.0,>=1.7.4->spacy) (4.15.0)
Requirement already satisfied: typing-inspection>=0.4.0 in /usr/local/lib/python3.12/dist-packages (from pydantic!=1.8,!=1.8.1,<3.0.0,>=1.7.4->spacy) (0.4.2)
Requirement already satisfied: charset_normalizer<4,>=2 in /usr/local/lib/python3.12/dist-packages (from requests<3.0.0,>=2.13.0->spacy) (3.4.4)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.12/dist-packages (from requests<3.0.0,>=2.13.0->spacy) (3.11)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.12/dist-packages (from requests<3.0.0,>=2.13.0->spacy) (2.5.0)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.12/dist-packages (from requests<3.0.0,>=2.13.0->spacy) (2025.10.5)
Requirement already satisfied: blis<1.4.0,>=1.3.0 in /usr/local/lib/python3.12/dist-packages (from thinc<8.4.0,>=8.3.4->spacy) (1.3.0)
Requirement already satisfied: confection<1.0.0,>=0.0.1 in /usr/local/lib/python3.12/dist-packages (from thinc<8.4.0,>=8.3.4->spacy) (0.1.5)
Requirement already satisfied: click>=8.0.0 in /usr/local/lib/python3.12/dist-packages (from typer<1.0.0,>=0.3.0->spacy) (8.3.0)
Requirement already satisfied: shellingham>=1.3.0 in /usr/local/lib/python3.12/dist-packages (from typer<1.0.0,>=0.3.0->spacy) (1.5.4)
Requirement already satisfied: rich>=10.11.0 in /usr/local/lib/python3.12/dist-packages (from typer<1.0.0,>=0.3.0->spacy) (13.9.4)
Requirement already satisfied: cloudpathlib<1.0.0,>=0.7.0 in /usr/local/lib/python3.12/dist-packages (from weasel<0.5.0,>=0.1.0->spacy) (0.23.0)
Requirement already satisfied: smart-open<8.0.0,>=5.2.1 in /usr/local/lib/python3.12/dist-packages (from weasel<0.5.0,>=0.1.0->spacy) (7.4.1)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.12/dist-packages (from jinja2->spacy) (3.0.3)
Requirement already satisfied: marisa-trie>=1.1.0 in /usr/local/lib/python3.12/dist-packages (from language-data>=1.2->langcodes<4.0.0,>=3.2.0->spacy) (1.3.1)
Requirement already satisfied: markdown-it-py>=2.2.0 in /usr/local/lib/python3.12/dist-packages (from rich>=10.11.0->typer<1.0.0,>=0.3.0->spacy) (4.0.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in /usr/local/lib/python3.12/dist-packages (from rich>=10.11.0->typer<1.0.0,>=0.3.0->spacy) (2.19.2)
Requirement already satisfied: wrapt in /usr/local/lib/python3.12/dist-packages (from smart-open<8.0.0,>=5.2.1->weasel<0.5.0,>=0.1.0->spacy) (2.0.0)
Requirement already satisfied: mdurl~=0.1 in /usr/local/lib/python3.12/dist-packages (from markdown-it-py>=2.2.0->rich>=10.11.0->typer<1.0.0,>=0.3.0->spacy) (0.1.2)
Collecting en-core-web-sm==3.8.0
  Downloading https://github.com/explosion/spacy-models/releases/download/en_core_web_sm-3.8.0/en_core_web_sm-3.8.0-py3-none-any.whl (12.8 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12.8/12.8 MB 110.0 MB/s eta 0:00:00
```



```
✓ Download and installation successful
You can now load the package via spacy.load('en_core_web_sm')
⚠ Restart to reload dependencies
If you are in a Jupyter or Colab notebook, you may need to restart Python in
order to load all the package's dependencies. You can do this by selecting the
'Restart kernel' or 'Restart runtime' option.
23-year-old --> DATE
Seattle --> GPE
Claritin --> PERSON
Zyrtec --> GPE
Allegra --> ORG
last summer --> DATE
two weeks ago --> DATE
daily --> DATE
Ortho Tri-Cyclen --> PERSON
Allegra --> ORG
130 pounds --> QUANTITY
Allergic --> ORG
Zyrtec --> NORP
Allegra --> ORG
Nasonex --> ORG
two --> CARDINAL
three weeks --> DATE
```

```
[7]
✓ 0s
from spacy import displacy
displacy.render(doc, style='ent', jupyter=True)
```




**Conclusion (Interpretation of result):**

The program successfully performed Named Entity Recognition (NER) on clinical text using the spaCy library.It identified both general entities such as organizations and dates, as well as domain-specific medical entities such as diseases and drugs through custom labeling.This demonstrates how NLP can be effectively applied in the healthcare domain to extract structured information from unstructured clinical data, improving the efficiency of data analysis and aiding medical research.