TASK-01 :Parts of Speech (POS)

NLTK uses the Penn Treebank tag seT

Double-click (or enter) to edit

```
import nltk
nltk.download('punkt')
nltk.download('averaged_perceptron_tagger_eng')
nltk.download('punkt_tab') # Added to fix the LookupError
sentence = "Students are learning Natural Language Processing"
tokens = nltk.word_tokenize(sentence)
pos_tags = nltk.pos_tag(tokens)
print(pos_tags)
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package averaged_perceptron_tagger_eng to
[nltk_data]     /root/nltk_data...
[nltk_data]   Package averaged_perceptron_tagger_eng is already up-to-
[nltk_data]       date!
[nltk_data] Downloading package punkt_tab to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt_tab.zip.
[('Students', 'NNS'), ('are', 'VBP'), ('learning', 'VBG'), ('Natural', 'NNP'), ('Language', 'NNP'), ('Processing', 'NNP')]
```

```
import spacy
nlp = spacy.load("en_core_web_sm")
doc = nlp("Students are learning Natural Language Processing")
for token in doc:
    print(token.text, token.pos_)
```

```
Students NOUN
are AUX
learning VERB
Natural PROPN
Language PROPN
Processing NOUN
```

```
import spacy
nlp = spacy.load("en_core_web_sm")
doc = nlp("Apple is looking at buying a startup in India.")
for token in doc:
    print(token.text, token.pos_, token.tag_)
```

```
Apple PROPN NNP
is AUX VBZ
looking VERB VBG
at ADP IN
buying VERB VBG
a DET DT
startup NOUN NN
in ADP IN
India PROPN NNP
. PUNCT .
```

POS Tagging for Social Media Text

```
import spacy
from collections import Counter
nlp = spacy.load("en_core_web_sm")
text = """Loving the new AI features
👍
👍
 #AI #MachineLearning"""
doc = nlp(text)
nouns = []
verbs = []
for token in doc:
    if token.pos_ in ["NOUN", "PROPN"]:
        nouns.append(token.text)
    elif token.pos_ == "VERB":
        verbs.append(token.text)
noun_freq = Counter(nouns)
verb_freq = Counter(verbs)
print("Noun Frequency:", noun_freq)
print("Verb Frequency:", verb_freq)
```

```
Noun Frequency: Counter({'AI': 2, '👍': 2, 'MachineLearning': 1})
Verb Frequency: Counter({'Loving': 1, 'features': 1})
```

TASK2: TAGGING PARTS OF SPEECH

```
nltk.download('averaged_perceptron_tagger')
from nltk.tokenize import word_tokenize
SRUniversity = """The SR University campus is located in Ananthasagar village of Hasanparthy Mandal in Warangal, Telangana,
It is in 150 acres, with both separate hostel facilities for boys and girls.
There is a huge central library along with Indias largest Technology Business Incubator (TBI) in tier 2 cities."""
words = word_tokenize(SRUniversity)
nltk.pos_tag(words)
```

```
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!
[('The', 'DT'),
 ('SR', 'NNP'),
 ('University', 'NNP'),
 ('campus', 'NN'),
 ('is', 'VBZ'),
 ('located', 'VBN'),
 ('in', 'IN'),
 ('Ananthasagar', 'NNP'),
 ('village', 'NN'),
 ('of', 'IN'),
 ('Hasanparthy', 'NNP'),
 ('Mandal', 'NNP'),
 ('in', 'IN'),
 ('Warangal', 'NNP'),
 (',', ','),
 ('Telangana', 'NNP'),
 (',', ','),
 ('India', 'NNP'),
 ('.', '.'),
 ('It', 'PRP'),
 ('is', 'VBZ'),
 ('in', 'IN'),
 ('150', 'CD'),
 ('acres', 'NNS'),
 (',', ','),
 ('with', 'IN'),
 ('both', 'DT'),
 ('separate', 'JJ'),
 ('hostel', 'NN'),
 ('facilities', 'NNS'),
 ('for', 'IN'),
 ('boys', 'NNS'),
 ('and', 'CC'),
 ('girls', 'NNS'),
 ('.', '.'),
 ('There', 'EX'),
 ('is', 'VBZ'),
 ('a', 'DT'),
 ('huge', 'JJ'),
 ('central', 'JJ'),
 ('library', 'NN'),
 ('along', 'IN'),
 ('with', 'IN'),
 ('Indias', 'NNP'),
 ('largest', 'JJS'),
 ('Technology', 'NN'),
 ('Business', 'NNP'),
 ('Incubator', 'NNP'),
 ('(', '('),
 ('TBI', 'NNP'),
 (')', ')'),
 ('in', 'IN'),
 ('tier', '$'),
 ('2', 'CD'),
 ('cities', 'NNS'),
 ('.', '.')]
```

```
nltk.download('tagsets')
nltk.download('tagsets_json') # Added to download the missing resource
nltk.help.upenn_tagset()
```

```
    low more off on open out over per pic raising start teeth that through
    low more off on open out over per pie raising start teeth that through
    under unto up up-pp upon whole with you
SYM: symbol
    % & ' '' ''. ) ). * + ,. < = > @ A[fj] U.S U.S.S.R * ** ***
TO: "to" as preposition or infinitive marker
    to
UH: interjection
    Goodbye Goody Gosh Wow Jeepers Jee-sus Hubba Hey Kee-reist Oops amen
    huh howdy uh dammit whammo shucks heck anyways whodunnit honey golly
    man baby diddle hush sonuvabitch ...
VB: verb, base form
    ask assemble assess assign assume atone attention avoid bake balkanize
    bank begin behold believe bend benefit bevel beware bless boil bomb
    boost brace break bring broil brush build ...
VBD: verb, past tense
    dipped pleaded swiped regummed soaked tidied convened halted registered
    cushioned exacted snubbed strode aimed adopted belied figgered
    speculated wore appreciated contemplated ...
VBG: verb, present participle or gerund
    telegraphing stirring focusing angering judging stalling lactating
    hankerin' alleging veering capping approaching traveling besieging
    encrypting interrupting erasing wincing ...
VBN: verb, past participle
    multihulled dilapidated aerosolized chaired languished panelized used
    experimented flourished imitated reunifed factored condensed sheared
    unsettled primed dubbed desired ...
VBP: verb, present tense, not 3rd person singular
    predominate wrap resort sue twist spill cure lengthen brush terminate
    appear tend stray glisten obtain comprise detest tease attract
    emphasize mold postpone sever return wag ...
VBZ: verb, present tense, 3rd person singular
    bases reconstructs marks mixes displeases seals carps weaves snatches
    slumps stretches authorizes smolders pictures emerges stockpiles
    seduces fizzes uses bolsters slaps speaks pleads ...
WDT: WH-determiner
    that what whatever which whichever
WP: WH-pronoun
    that what whatever whatsoever which who whom whosoever
WP$: WH-pronoun, possessive
    whose
WRB: Wh-adverb
    how however whence whenever where whereby whereever wherein whereof why
``: opening quotation mark
    ` ``
[nltk_data] Downloading package tagsets to /root/nltk_data...
[nltk_data]   Package tagsets is already up-to-date!
[nltk_data] Downloading package tagsets_json to /root/nltk_data...
[nltk_data]   Unzipping help/tagsets_json.zip.
```

Assignment 3.2

Install and import libraries

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

```
equirement already satisfied: nltk in /usr/local/lib/python3.12/dist-packages (3.9.1)
equirement already satisfied: spacy in /usr/local/lib/python3.12/dist-packages (3.8.11)
equirement already satisfied: click in /usr/local/lib/python3.12/dist-packages (from nltk) (8.3.1)
equirement already satisfied: joblib in /usr/local/lib/python3.12/dist-packages (from nltk) (1.5.3)
equirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.12/dist-packages (from nltk) (2025.11.3)
equirement already satisfied: tqdm in /usr/local/lib/python3.12/dist-packages (from nltk) (4.67.1)
equirement already satisfied: spacy-legacy<3.1.0,>=3.0.11 in /usr/local/lib/python3.12/dist-packages (from spacy) (3.0.12)
equirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (1.0.5)
equirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (1.0.15)
equirement already satisfied: cymem<2.1.0,>=2.0.2 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.0.13)
equirement already satisfied: preshed<3.1.0,>=3.0.2 in /usr/local/lib/python3.12/dist-packages (from spacy) (3.0.12)
equirement already satisfied: thinc<8.4.0,>=8.3.4 in /usr/local/lib/python3.12/dist-packages (from spacy) (8.3.10)
equirement already satisfied: wasabi<1.2.0,>=0.9.1 in /usr/local/lib/python3.12/dist-packages (from spacy) (1.1.3)
equirement already satisfied: srsly<3.0.0,>=2.4.3 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.5.2)
equirement already satisfied: catalogue<2.1.0,>=2.0.6 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.0.10)
equirement already satisfied: weasel<0.5.0,>=0.4.2 in /usr/local/lib/python3.12/dist-packages (from spacy) (0.4.3)
equirement already satisfied: typer-slim<1.0.0,>=0.3.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (0.21.1)
equirement already satisfied: numpy>=1.19.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.0.2)
equirement already satisfied: requests<3.0.0,>=2.13.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (2.32.4)
equirement 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
equirement already satisfied: jinja2 in /usr/local/lib/python3.12/dist-packages (from spacy) (3.1.6)
equirement already satisfied: setuptools in /usr/local/lib/python3.12/dist-packages (from spacy) (75.2.0)
equirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.12/dist-packages (from spacy) (25.0)
equirement already satisfied: annotated-types>=0.6.0 in /usr/local/lib/python3.12/dist-packages (from pydantic!=1.8,!=1.8.1,<
equirement already satisfied: pydantic-core==2.41.4 in /usr/local/lib/python3.12/dist-packages (from pydantic!=1.8,!=1.8.1,<3
equirement already satisfied: typing-extensions>=4.14.1 in /usr/local/lib/python3.12/dist-packages (from pydantic!=1.8,!=1.8.
equirement already satisfied: typing-inspection>=0.4.2 in /usr/local/lib/python3.12/dist-packages (from pydantic!=1.8,!=1.8.1
equirement already satisfied: charset_normalizer<4,>=2 in /usr/local/lib/python3.12/dist-packages (from requests<3.0.0,>=2.13
equirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.12/dist-packages (from requests<3.0.0,>=2.13.0->spacy) (
equirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.12/dist-packages (from requests<3.0.0,>=2.13.0->sp
equirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.12/dist-packages (from requests<3.0.0,>=2.13.0->sp
equirement 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)
equirement 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->
equirement already satisfied: cloudpathlib<1.0.0,>=0.7.0 in /usr/local/lib/python3.12/dist-packages (from weasel<0.5.0,>=0.4.
equirement already satisfied: smart-open<8.0.0,>=5.2.1 in /usr/local/lib/python3.12/dist-packages (from weasel<0.5.0,>=0.4.2-
equirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.12/dist-packages (from jinja2->spacy) (3.0.3)
equirement already satisfied: wrapt in /usr/local/lib/python3.12/dist-packages (from smart-open<8.0.0,>=5.2.1->weasel<0.5.0,>
ollecting 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-a
    ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12.8/12.8 MB 88.4 MB/s eta 0:00:00
 Download and installation successful
u can now load the package via spacy.load('en_core_web_sm')
 Restart to reload dependencies
f you are in a Jupyter or Colab notebook, you may need to restart Python in
der to load all the package's dependencies. You can do this by selecting the
Restart kernel' or 'Restart runtime' option.
```

```
import json
import nltk
import spacy
from nltk.tokenize import TweetTokenizer
from nltk import pos_tag
from collections import Counter

nltk.download('averaged_perceptron_tagger')
```

```
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /root/nltk_data...
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!
True
```

Load the enriched_posts.json file

```
file_path = "/content/enriched_posts.json"

with open(file_path, "r", encoding="utf-8") as f:
    data = json.load(f)

type(data), len(data)
```

```
(list, 60)
```

Extract text from posts

```
# Try common keys used in social media JSON
texts = []

for item in data:
    if 'text' in item:
        texts.append(item['text'])
    elif 'caption' in item:
        texts.append(item['caption'])
    elif 'content' in item:
        texts.append(item['content'])

texts[:5]
```

```
["Just saw a LinkedIn Influencer with 'Organic Growth' written in the profile with 65K+ followers claiming that he can help
you in growing your platform, copying the posts from other influencers.",
 "Jobseekers, this one’s for you.\n Every application, every interview, every follow-up… the pressure is immense.\n And I
know what you're thinking: Am I not good enough? \n But let me tell you, this isn’t about you or your skills. It’s about a
broken system where 60% of applicants never hear back. \n Your mental health is not worth sacrificing for a system that
doesn’t acknowledge your worth. \n Please remember, taking care of yourself is the real priority. \n Your dream job will
come, but for now, breathe. 🌻",
 'Looking for jobs on LinkedIn is like online dating: Full of promises, but in the end, you’re just left ghosted.',
 "LinkedIn scams be like: 'Congratulations, you've been selected for a role you didn’t even apply for!' \n The catch? Pay
Rs. 50,000 for the honor.",
 "sapne dekhna achi baat hai,\nlekin job ka sapna dekh ke 'interested' likhna,\nyeh toh achi baat nahi hai na?"]
```

Tokenize informal text

```
tokenizer = TweetTokenizer(
    preserve_case=False,
    strip_handles=True,
    reduce_len=True
)
```

```
sample_tokens = tokenizer.tokenize(texts[0])
sample_tokens
```

```
['just',
 'saw',
 'a',
 'linkedin',
 'influencer',
 'with',
 "'",
 'organic',
 'growth',
 "'",
 'written',
 'in',
 'the',
 'profile',
 'with',
 '65k',
 '+',
 'followers',
 'claiming',
 'that',
 'he',
 'can',
 'help',
 'you',
 'in',
 'growing',
 'your',
 'platform',
 ',',
 'copying',
 'the',
 'posts',
 'from',
 'other',
 'influencers',
 '.']
```

POS tagging using NLTK

```
nltk_pos = pos_tag(sample_tokens)
nltk_pos
```

```
[('just', 'RB'),
 ('saw', 'VBD'),
 ('a', 'DT'),
 ('linkedin', 'JJ'),
 ('influencer', 'NN'),
 ('with', 'IN'),
 ("'", "''"),
 ('organic', 'JJ'),
 ('growth', 'NN'),
 ("'", "''"),
 ('written', 'VBN'),
 ('in', 'IN'),
 ('the', 'DT'),
 ('profile', 'NN'),
 ('with', 'IN'),
 ('65k', 'CD'),
 ('+', 'JJ'),
 ('followers', 'NNS'),
 ('claiming', 'VBG'),
 ('that', 'IN'),
 ('he', 'PRP'),
 ('can', 'MD'),
 ('help', 'VB'),
 ('you', 'PRP'),
 ('in', 'IN'),
 ('growing', 'VBG'),
 ('your', 'PRP$'),
 ('platform', 'NN'),
 (',', ','),
 ('copying', 'VBG'),
 ('the', 'DT'),
 ('posts', 'NNS'),
 ('from', 'IN'),
 ('other', 'JJ'),
 ('influencers', 'NNS'),
 ('.', '.')]
```

POS tagging using spaCy

```
nlp = spacy.load("en_core_web_sm")
doc = nlp(texts[0])

[(token.text, token.pos_) for token in doc]
```

```
[('Just', 'ADV'),
 ('saw', 'VERB'),
 ('a', 'DET'),
 ('LinkedIn', 'NOUN'),
 ('Influencer', 'NOUN'),
 ('with', 'ADP'),
 ("'", 'PUNCT'),
 ('Organic', 'PROPN'),
 ('Growth', 'PROPN'),
 ("'", 'PUNCT'),
 ('written', 'VERB'),
 ('in', 'ADP'),
 ('the', 'DET'),
 ('profile', 'NOUN'),
 ('with', 'ADP'),
 ('65K+', 'PROPN'),
 ('followers', 'NOUN'),
 ('claiming', 'VERB'),
 ('that', 'SCONJ'),
 ('he', 'PRON'),
 ('can', 'AUX'),
 ('help', 'VERB'),
 ('you', 'PRON'),
 ('in', 'ADP'),
 ('growing', 'VERB'),
 ('your', 'PRON'),
 ('platform', 'NOUN'),
 (',', 'PUNCT'),
 ('copying', 'VERB'),
 ('the', 'DET'),
 ('posts', 'NOUN'),
 ('from', 'ADP'),
 ('other', 'ADJ'),
 ('influencers', 'NOUN'),
 ('.', 'PUNCT')]
```

Compare tag sets

```
print("NLTK POS Tags:")
print(nltk_pos)

print("\nspaCy POS Tags:")
print([(token.text, token.pos_) for token in doc])
```

```
NLTK POS Tags:
[('just', 'RB'), ('saw', 'VBD'), ('a', 'DT'), ('linkedin', 'JJ'), ('influencer', 'NN'), ('with', 'IN'), ("'", "''"), ('orgar

spaCy POS Tags:
[('Just', 'ADV'), ('saw', 'VERB'), ('a', 'DET'), ('LinkedIn', 'NOUN'), ('Influencer', 'NOUN'), ('with', 'ADP'), ("'", 'PUNCT
```

Extract Nouns & Verbs and frequency

```
# NLTK
nltk_nouns = [w for w, t in nltk_pos if t.startswith('NN')]
nltk_verbs = [w for w, t in nltk_pos if t.startswith('VB')]

# spaCy
spacy_nouns = [t.text for t in doc if t.pos_ == 'NOUN']
spacy_verbs = [t.text for t in doc if t.pos_ == 'VERB']

print("Top NLTK Nouns:", Counter(nltk_nouns).most_common(10))
print("Top NLTK Verbs:", Counter(nltk_verbs).most_common(10))
```

```
Top NLTK Nouns: [('influencer', 1), ('growth', 1), ('profile', 1), ('followers', 1), ('platform', 1), ('posts', 1), ('influe
Top NLTK Verbs: [('saw', 1), ('written', 1), ('claiming', 1), ('help', 1), ('growing', 1), ('copying', 1)]
```