

TP-3: Bag of Words (BoW)

1. Implement a bag of words algorithm with Python

Sample sentences:

```
sentence1 =" Welcome to NLP Learning , Now start learning"  
sentence2 =" Learning is a good practice"
```

Expected Output:

```
['welcome', 'to', 'nlp', 'learning', ',', 'now', 'start', 'learning']  
['learning', 'is', 'a', 'good', 'practice']  
['welcome', 'to', 'nlp', 'learning', ',', 'now', 'start', 'is', 'a', 'good', 'practice']  
['welcome', 'nlp', 'learning', 'now', 'start', 'good', 'practice']  
[1, 1, 2, 1, 1, 0, 0]  
[0, 0, 1, 0, 0, 1, 1]
```

2. Implement Bag of Words using SKLEARN

Sample sentences:

```
sentence1 ="This is a good job. I will not miss it for anything"  
sentence2 =" This is not good at all"
```

Expected Output:

	good	job	miss
0	1	1	1
1	1	0	0

3. Implement Bag of words using NLTK

Task:

- Importing the necessary libraries from NLTK.
- Define a list of sample documents.
- Tokenize the documents into words and convert them to lowercase.
- Remove stopwords and punctuation from the tokens.
- Create a vocabulary by collecting all unique words from the processed documents.
- Initialize a BoW dictionary with word counts, setting the initial count for each word to 0.
- Iterate through the filtered tokens and increment the count for each word in the BoW dictionary.
- Print the BoW representation, which shows the word counts for each word in the vocabulary.

Sample documents:

"I love natural language processing.",
"Text classification is an important NLP task.",
"NLTK provides useful tools for NLP.",

Expected Output:

Bag of Words (Bow) representation:

```
{'language': 1, 'love': 1, 'nlp': 2, 'useful': 1, 'text': 1, 'classification': 1, 'task': 1, 'important': 1, 'tools': 1, 'natural': 1, 'provides': 1, 'nltk': 1, 'processing': 1}
```

4. Classify movie review is **positive or negative using *Bag of words* for pre-processing the text (from Sklearn) and apply with any models (RF, DT)**

Dataset:

Link = <https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews?resource=download>

- This data consists of two columns. - review - sentiment
- Reviews are the statements given by users after watching the movie.
- sentiment feature tells whether the given review is positive or negative.

Expected Output:

	precision	recall	f1-score	support
0	0.82	0.85	0.83	1849
1	0.85	0.83	0.84	1951
accuracy			0.84	3800
macro avg	0.84	0.84	0.84	3800
weighted avg	0.84	0.84	0.84	3800