BL.EN.U4CSE23145

R.V.MDEEKSHITH.M

Question 1

import pandas as pd

import nltk

from nltk.tokenize import word\_tokenize

nltk.download('punkt')

df = pd.read\_excel(r'C:\Users\year3\Desktop\NLP\Lab\_4\Data.xlsx', sheet\_name='Sheet1', engine='openpyxl')

documents = df['Report'].astype(str).tolist()

Documents = [word\_tokenize(doc) for doc in documents]

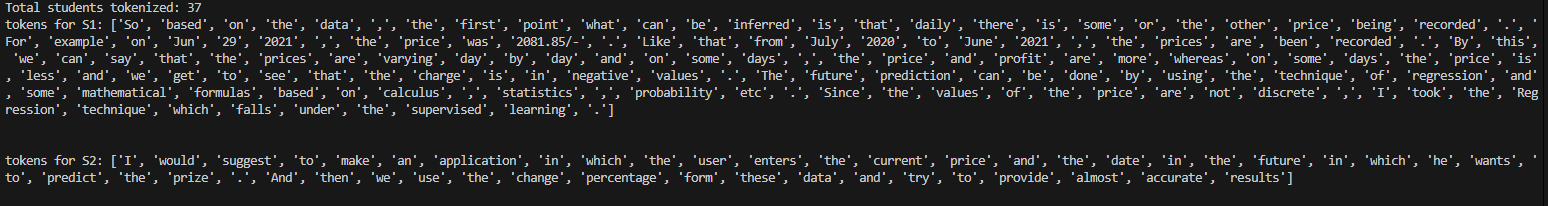
print(f'Total students tokenized: {len(Documents)}')

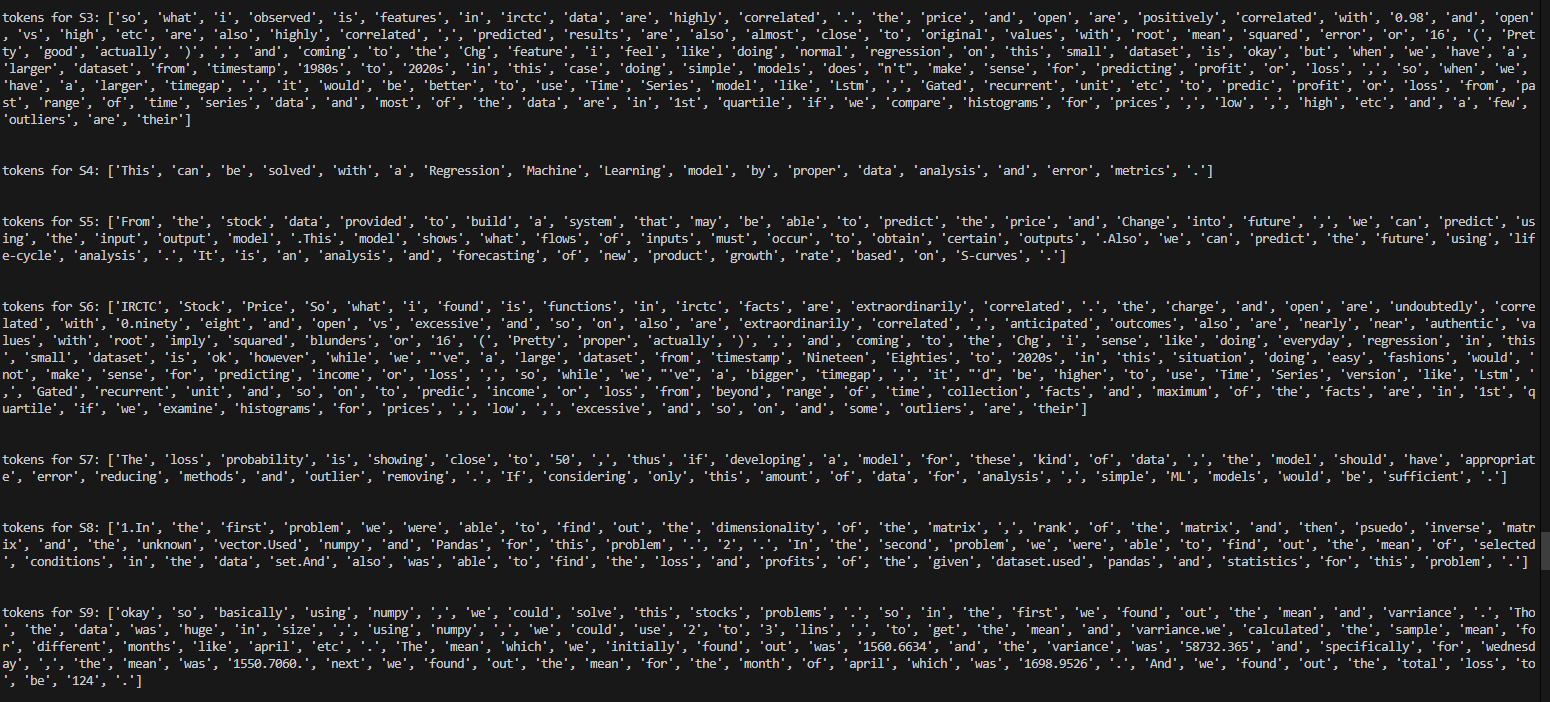
for i in range(37):

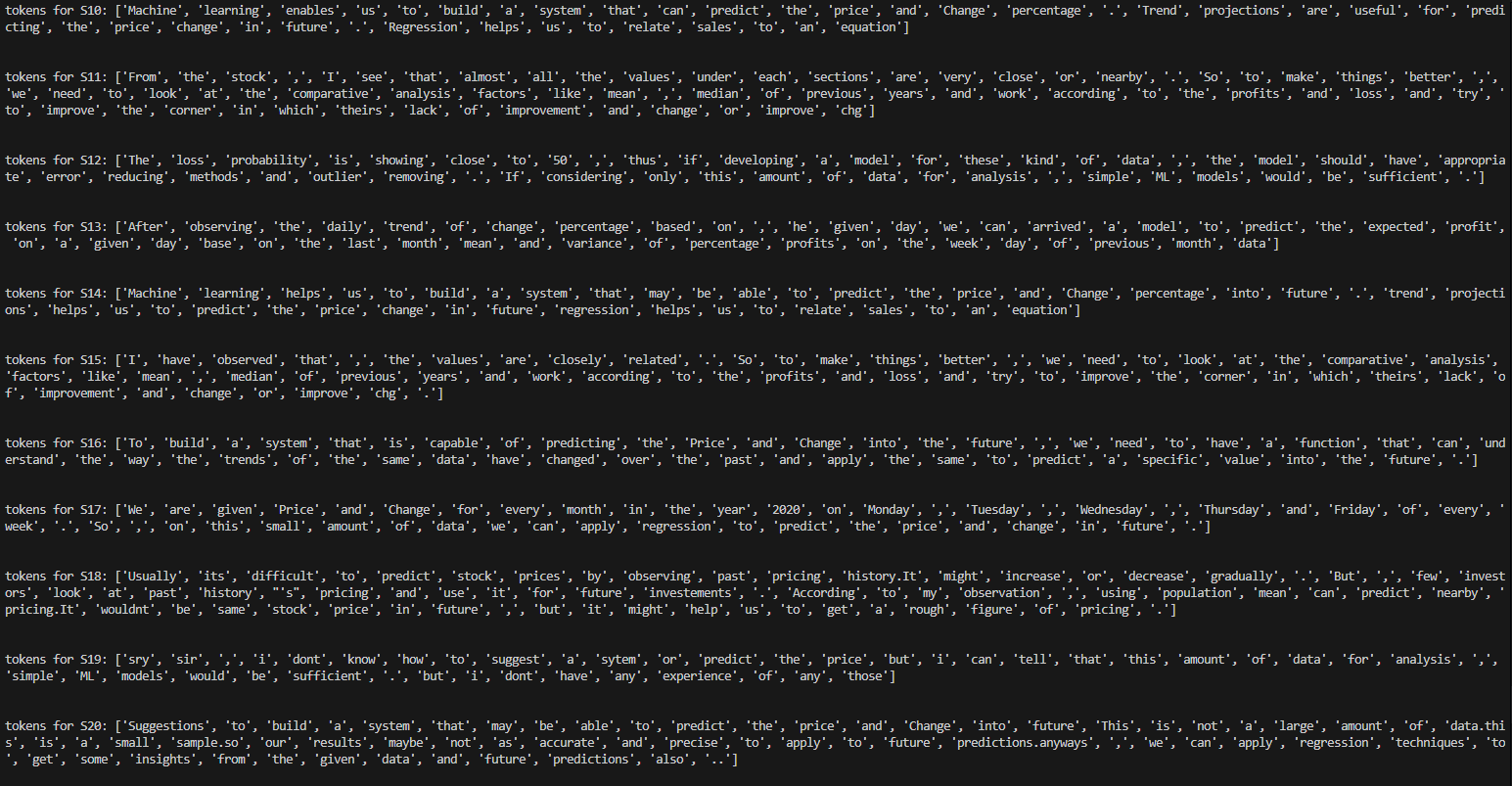
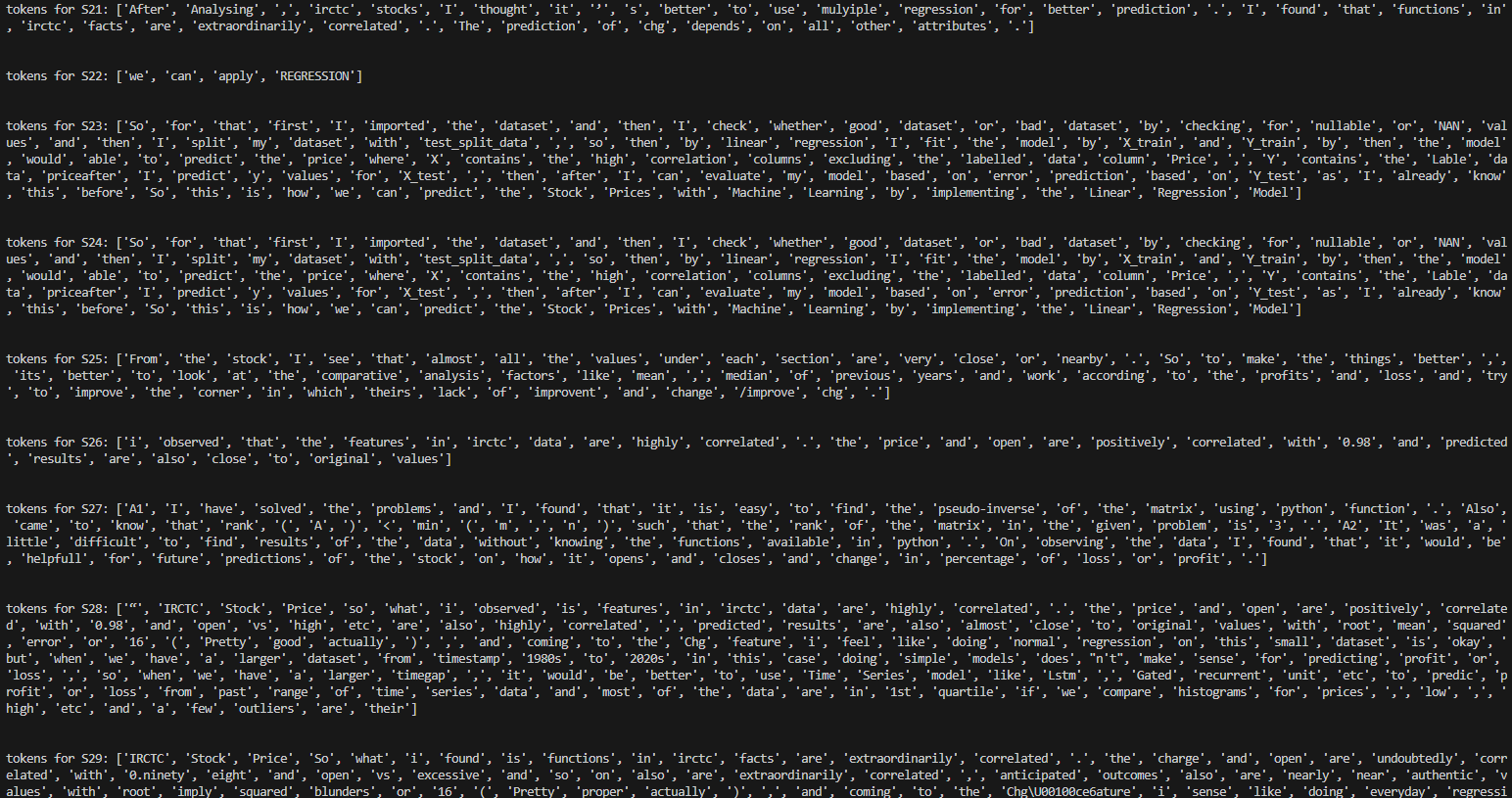
    print(f'tokens for S{i+1}:',Documents [i][:])

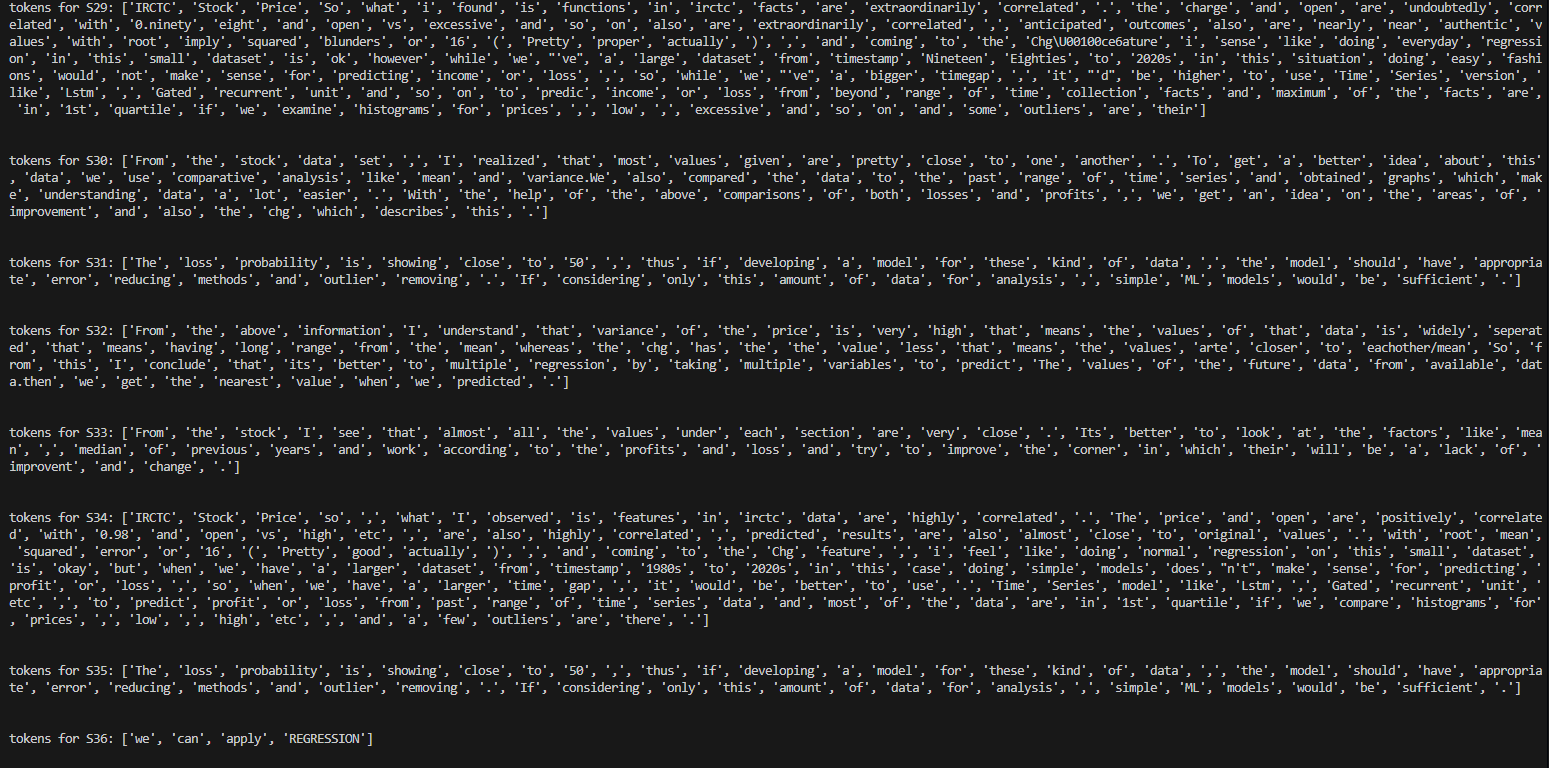
    print('\n')

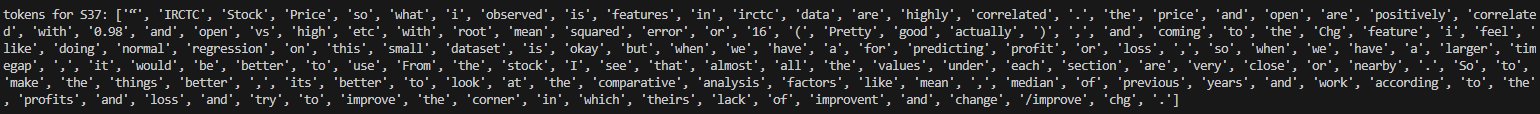
Output:









Question 2:

import pandas as pd

import nltk

from nltk.tokenize import word\_tokenize

nltk.download('punkt')

df = pd.read\_excel(r'C:\Users\year3\Desktop\NLP\Lab\_4\Data.xlsx', sheet\_name='Sheet1', engine='openpyxl')

documents = df['Report'].astype(str).tolist()

Documents = [word\_tokenize(doc) for doc in documents]

print(f'Total students tokenized: {len(Documents)}')

for i in range(37):

    print(f'tokens for S{i+1}:',Documents [i][:])

    print('\n')

all\_tokens = [token for doc in Documents for token in doc]

token\_population = list(set(all\_tokens))

V = len(token\_population)

print(f'The size of the token population (V) is: {V}')

Output:



Question 3 :

import nltk

import pandas as pd

from nltk.corpus import stopwords

from nltk.tokenize import word\_tokenize

nltk.download('stopwords')

nltk.download('punkt')

df = pd.read\_excel(r'C:\Users\year3\Desktop\NLP\Lab\_4\Data.xlsx', sheet\_name='Sheet1', engine='openpyxl')

documents = df['Report'].astype(str).tolist()

Documents = [word\_tokenize(doc) for doc in documents]

print(f'Total students tokenized: {len(Documents)}')

for i in range(37):

    print(f'tokens for S{i+1}:',Documents [i][:])

    print('\n')

token\_population = list(set([token for doc in Documents for token in doc]))

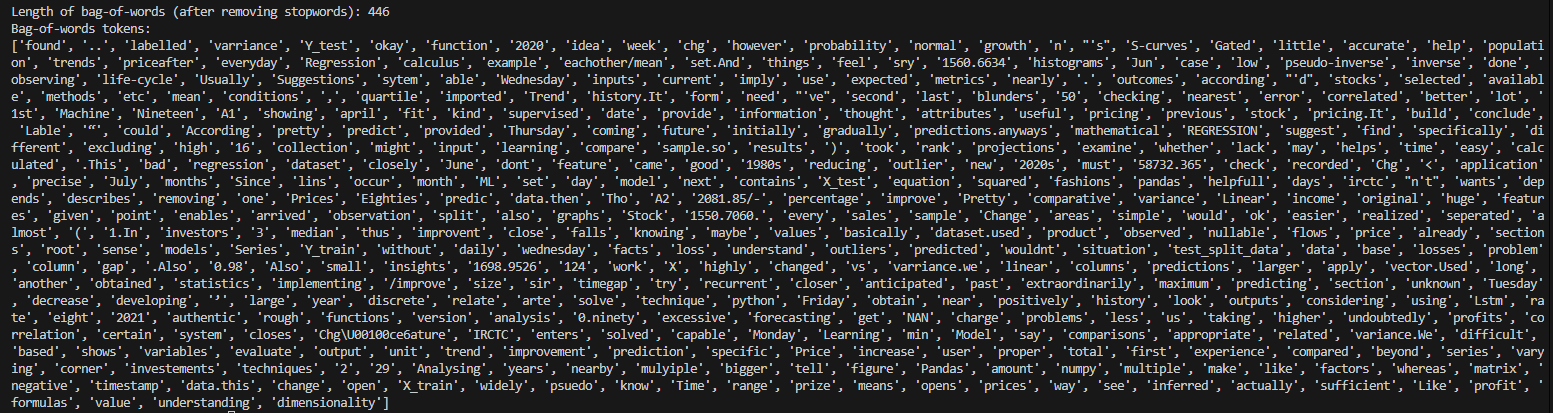
stop\_words = set(stopwords.words('english'))

bag\_of\_words = [token for token in token\_population if token.lower() not in stop\_words]

print(f'Length of bag-of-words (after removing stopwords): {len(bag\_of\_words)}')

print(f'Bag-of-words tokens:\n{bag\_of\_words}')

Ouput :



Question 4:

import pandas as pd

import nltk

import string

from nltk.tokenize import word\_tokenize

from nltk.corpus import stopwords

import numpy as np

nltk.download('punkt')

nltk.download('stopwords')

excel\_path = r'C:\Users\year3\Desktop\NLP\Lab\_4\Data.xlsx'

df = pd.read\_excel(excel\_path, sheet\_name='Sheet1', engine='openpyxl')

Documents = [ word\_tokenize(str(txt)) for txt in df['Report'] ]

stop\_words  = set(stopwords.words('english'))

punctuation = set(string.punctuation)

normalized\_docs = []

for tokens in Documents:

    clean = [

        t.lower()

        for t in tokens

        if t.lower() not in stop\_words

           and t not in punctuation

    ]

    normalized\_docs.append(clean)

vocab\_set  = {token for doc in normalized\_docs for token in doc}

vocab\_list = sorted(vocab\_set)

N = len(normalized\_docs)

M = len(vocab\_list)

V1 = np.zeros((N, M), dtype=int)

V2 = np.zeros((N, M), dtype=int)

for d\_idx, doc in enumerate(normalized\_docs):

    token\_counts = {}

    for token in doc:

        token\_counts[token] = token\_counts.get(token, 0) + 1

    for w\_idx, word in enumerate(vocab\_list):

        count = token\_counts.get(word, 0)

        V2[d\_idx, w\_idx] = count

        V1[d\_idx, w\_idx] = 1 if count > 0 else 0

print("Number of documents (N):", N)

print("Vocabulary size (M):", M)

print("V1 shape:", V1.shape)

print("V2 shape:", V2.shape)

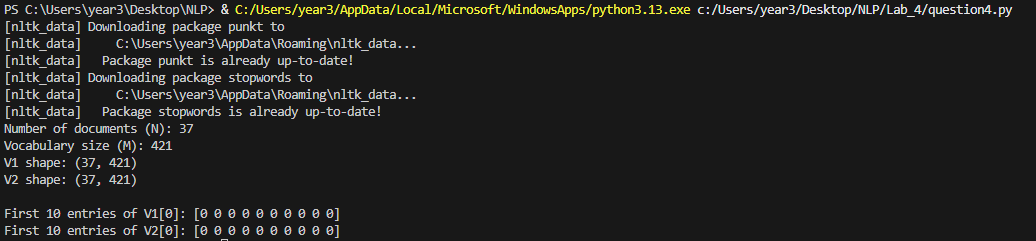
print("\nFirst 10 entries of V1[0]:", V1[0, :10])

print("First 10 entries of V2[0]:", V2[0, :10])

np.save('V1\_binary.npy', V1)

np.save('V2\_counts.npy', V2)

Output :



Question 5:

import pandas as pd

import numpy as np

from sklearn.metrics.pairwise import cosine\_similarity

import nltk

import string

from nltk.tokenize import word\_tokenize

from nltk.corpus import stopwords

nltk.download('punkt')

nltk.download('stopwords')

excel\_path = r'C:\Users\year3\Desktop\NLP\Lab\_4\Data.xlsx'

df = pd.read\_excel(excel\_path, sheet\_name='Sheet1', engine='openpyxl')

Documents = [ word\_tokenize(str(txt)) for txt in df['Report'] ]

stop\_words  = set(stopwords.words('english'))

punctuation = set(string.punctuation)

normalized\_docs = []

for tokens in Documents:

    clean = [t.lower() for t in tokens

             if t.lower() not in stop\_words

                and t not in punctuation]

    normalized\_docs.append(clean)

vocab = sorted({tok for doc in normalized\_docs for tok in doc})

M = len(vocab)

N = len(normalized\_docs)

V2 = np.zeros((N, M), dtype=int)

for i, doc in enumerate(normalized\_docs):

    counts = {}

    for tok in doc:

        counts[tok] = counts.get(tok, 0) + 1

    for j, term in enumerate(vocab):

        V2[i, j] = counts.get(term, 0)

cos\_sim\_matrix = cosine\_similarity(V2)

doc\_labels = [f"S{i}" for i in range(1, N+1)]

cosine\_df = pd.DataFrame(cos\_sim\_matrix, index=doc\_labels, columns=doc\_labels)

print("Cosine‐similarity between documents:")

print(cosine\_df)

Output :

