In [2]:

```
!pip install nltk
```

```
Requirement already satisfied: nltk in c:\users\ashvini mahajan\miniconda3\lib\site-packages (3.8.1)
Requirement already satisfied: click in c:\users\ashvini mahajan\miniconda3\lib\site-packages (from nltk) (8.1.7)
Requirement already satisfied: joblib in c:\users\ashvini mahajan\miniconda3\lib\site-packages (from nltk) (1.2.0)
Requirement already satisfied: regex>=2021.8.3 in c:\users\ashvini mahajan\miniconda3\lib\site-packages (from nltk) (2023.12.25)
Requirement already satisfied: tqdm in c:\users\ashvini mahajan\miniconda3\lib\site-packages (from nltk) (4.65.0)
Requirement already satisfied: colorama in c:\users\ashvini mahajan\miniconda3\lib\site-packages (from click->nltk) (0.4.6)
```

Tokenization

sentence tokenization

In [3]:

```
import nltk
nltk.download('punkt')
from nltk import sent_tokenize
```

```
[nltk_data] Downloading package punkt to C:\Users\Ashvini
[nltk_data]     Mahajan\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
```

In [4]:

```
text="Hii there. I am studying in DYPCOE which is better than other colleges. I played basketball yesterday."
```

In [5]:

```
sent_tokens=sent_tokenize(text)
print(sent_tokens)
```

```
['Hii there.', 'I am studying in DYPCOE which is better than other colleges.', 'I played basketball yesterday.']
```

word tokenziation

In [6]:

```
from nltk import word_tokenize
```

In [7]:

```
word_tokens=word_tokenize(text)
print(word_tokens)
```

```
['Hii', 'there', '.', 'I', 'am', 'studying', 'in', 'DYPCOE', 'which', 'is', 'better', 'than', 'other', 'colleges', '.', 'I', 'played', 'basketball', 'yesterday', '.']
```

stemming

In [8]:

```
from nltk.stem import PorterStemmer
porter=PorterStemmer()
```

In [9]:

```
stem_words=[]
for i in word_tokens:
    stem_word=porter.stem(i)
    stem_words.append(stem_word)
```

In [10]:

```
print(stem_words)
```

```
['hii', 'there', '.', 'i', 'am', 'studi', 'in', 'dypco', 'which', 'is', 'better', 'than', 'other', 'colleg', '.', 'i', 'play', 'basketbal', 'yesterday', '.']
```

pos tagging

In [11]:

```
nltk.download('averaged_perceptron_tagger')
pos_tags=nltk.pos_tag(word_tokens)
print(pos_tags)
```

```
[('Hii', 'NNP'), ('there', 'RB'), ('.', '.'), ('I', 'PRP'), ('am', 'VBP'), ('studying', 'VBG'), ('in', 'IN'), ('DYPCOE', 'NNP'), ('which', 'WDT'), ('is', 'VBZ'), ('better', 'JJR'), ('than',
'IN'), ('other', 'JJ'), ('colleges', 'NNS'), ('.', '.'), ('I', 'PRP'), ('played', 'VBD'), ('basketball', 'NN'), ('yesterday', 'NN'), ('.', '.')]
```

```
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     C:\Users\Ashvini Mahajan\AppData\Roaming\nltk_data...
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!
```

stop words

let's see which are stop words

In [12]:

```
from nltk.corpus import stopwords
nltk.download('stopwords')
stop_words=stopwords.words('english')
print(stop_words)
```

```
['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've", "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his', 'himself', 'she', "s
he's", 'her', 'hers', 'herself', 'it', "it's", 'its', 'itself', 'they', 'them', 'their', 'theirs', 'themselves', 'what', 'which', 'who', 'whom', 'this', 'that', "that'll", 'these', 'those',
'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'have', 'has', 'had', 'having', 'do', 'does', 'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or', 'because', 'as', 'until', 'w
hile', 'of', 'at', 'by', 'for', 'with', 'about', 'against', 'between', 'into', 'through', 'during', 'before', 'after', 'above', 'below', 'to', 'from', 'up', 'down', 'in', 'out', 'on', 'off',
'over', 'under', 'again', 'further', 'then', 'once', 'here', 'there', 'when', 'where', 'why', 'how', 'all', 'any', 'both', 'each', 'few', 'more', 'most', 'other', 'some', 'such', 'no', 'nor',
'not', 'only', 'own', 'same', 'so', 'than', 'too', 'very', 's', 't', 'can', 'will', 'just', 'don', "don't", 'should', "should've", 'now', 'd', 'll', 'm', 'o', 're', 've', 'y', 'ain', 'aren',
"aren't", 'couldn', "couldn't", 'didn', "didn't", 'doesn', "doesn't", 'hadn', "hadn't", 'hasn', "hasn't", 'haven', "haven't", 'isn', "isn't", 'ma', 'mightn', "mightn't", 'mustn', "mustn't",
'needn', "needn't", 'shan', "shan't", 'shouldn', "shouldn't", 'wasn', "wasn't", 'weren', "weren't", 'won', "won't", 'wouldn', "wouldn't"]
```

```
[nltk_data] Downloading package stopwords to C:\Users\Ashvini
[nltk_data]     Mahajan\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

removing stop words form word_tokens

In [13]:

```
# before filtering out stop words
print(word_tokens)
```

```
['Hii', 'there', '.', 'I', 'am', 'studying', 'in', 'DYPCOE', 'which', 'is', 'better', 'than', 'other', 'colleges', '.', 'I', 'played', 'basketball', 'yesterday', '.']
```

In [14]:

```
filtered=[]
for i in word_tokens:
    if i.lower() not in stop_words:
        filtered.append(i)
print(filtered)
```

```
['Hii', '.', 'studying', 'DYPCOE', 'better', 'colleges', '.', 'played', 'basketball', 'yesterday', '.']
```

lemmetization

In [32]:

```
from nltk.stem import WordNetLemmatizer
nltk.download('wordnet')
lemmatizer=WordNetLemmatizer()


pos_map = {
    'J': 'a',  # Adjective
    'V': 'v',  # Verb
    'N': 'n',  # Noun
    'R': 'r'   # Adverb
}

lem=[]
for word,pos in pos_tags:
    wn_pos = pos_map.get(pos[0].upper(), 'n')  # Default to 'n' (noun) if not found
    word=lemmatizer.lemmatize(word,pos=wn_pos)
    lem.append(word)
print(lem)
```

```
['Hii', 'there', '.', 'I', 'be', 'study', 'in', 'DYPCOE', 'which', 'be', 'good', 'than', 'other', 'college', '.', 'I', 'play', 'basketball', 'yesterday', '.']
```

```
[nltk_data] Downloading package wordnet to C:\Users\Ashvini
[nltk_data]     Mahajan\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

Term Frequency

Inverse Document Frequency`

In [41]:

```
documents = [
    "Mumbai Pune Mumbai",
    "Pune Pune Pune",
    "Nashik Pune Mumbai",
    "Nashik Nashik Nashik",
]
from sklearn.feature_extraction.text import TfidfVectorizer
tfidf=TfidfVectorizer()
matrix=tfidf.fit_transform(documents)
print(matrix.toarray())
```

```
[[0.92693676 0.         0.3752176 ]
 [0.         0.         1.        ]
 [0.61366674 0.61366674 0.49681612]
 [0.         1.         0.        ]]
```

In [ ]: