NLP :- NLTK(for text processing)

Application:- google assistance , alexa ,gmail spam classifier

1. Tokenization:-
   * + - 1. sentences = nltk.sent\_tokenize(paragraph) ,
         2. words = nltk.word\_tokenize(paragraph)
2. Stopwords :- unnecessary words not much important (I, me, them , is , to etc)

From nltk.corpus import stopwords

1. Cleaning

//// caleaning --🡪 removing all character other then [a-zA-Z]

Clean\_stence = re.sub(‘ [a-zA-Z]’ , ’ ‘ ,sentences[i]) ////// replacing all other unnecessary character with space

1. Stemming :- Process of reducing word to there word stem.

Ex- history,historical ---------🡪histori

(no meaning)

Not take much time

From nltk.stem import PortStemmer

Stemmer = PortStemmer()

//////Code Stemming

For i in range(sentences):

Words=nltk.word\_tokenize(len(sentences[i]))

Words =[stemmer.stem(word) for word in words if word not in set(stopwords.words(english))]

Sentences[i]=” “.join(words)

/////// sentence with removed stopwords & perform Stemming

1. Lemmatization:- do same as stemming but meaningfull.

Ex:- history,historical---------🡪history

(meaningfull)

Take time.

/////code for lemmatization

From nltk.stem import WordNetLemmatizer

Lemmatizer = WordNetLemmatizer()

For i in range(len(sentences)):

Words=nltk.word\_tokenize(sentences[i])

Words =[lemmatizer.lemmatize(word) for word in words if word not in set(stopwords.words(english))]

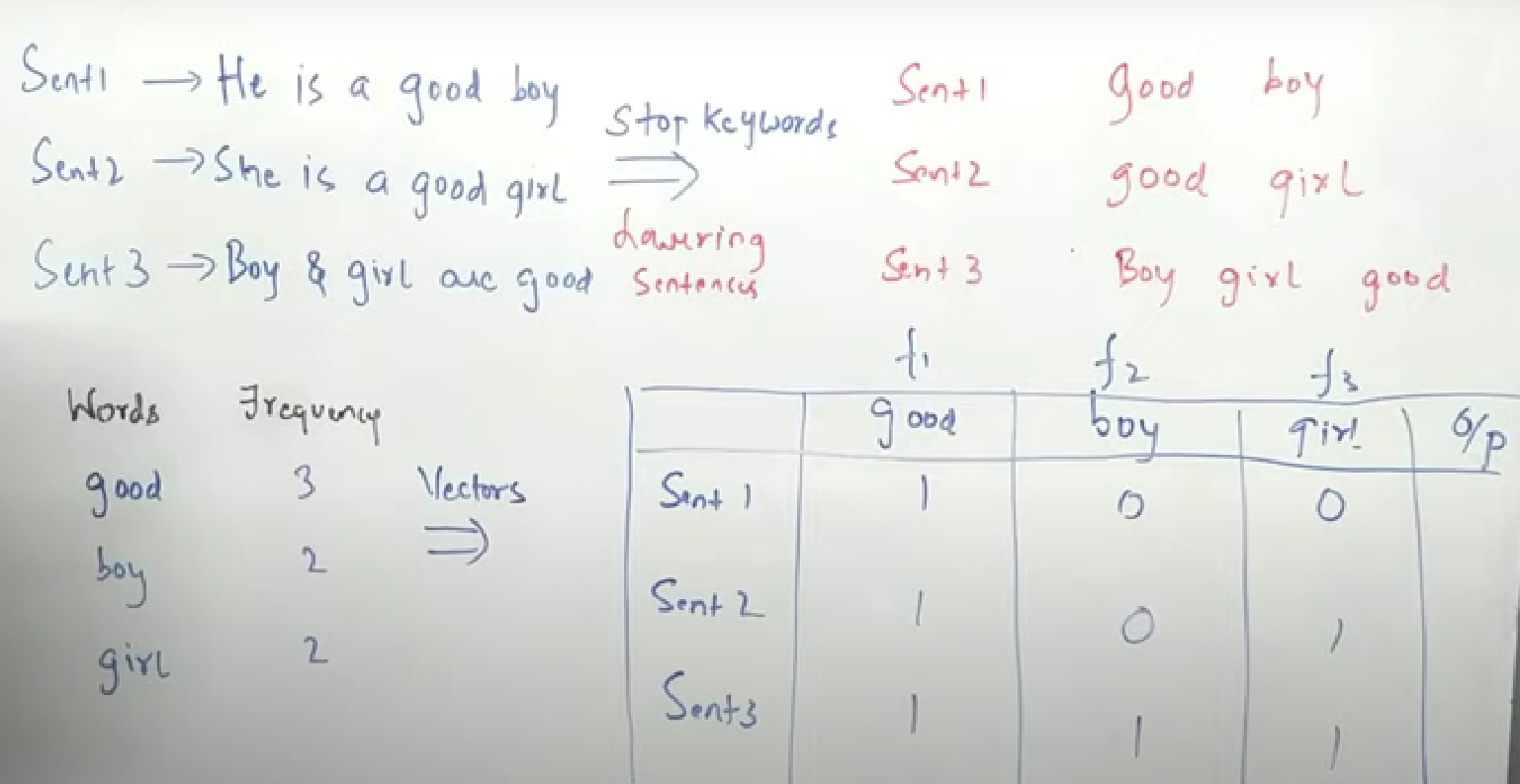
Sentences[i]=” “.join(words)

////// sentence with removed stopwords & perform lemmatization

1. Bag of words :- Technique convert sentence to vector (Senctence(text ) can not be feed directly to model)

From sklearn.feature\_extraction.text import CountVectorizer

* 1. remove stopwords from sentence
  2. find frequency of each words
  3. arrange word in table acc to frequency and find vector of each sentence



/////Here stemming is performed we can also perform Lemmatization(better give meaningfull words)

