In [1]:
```
import nltk
import numpy
from nltk import punkt
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize,sent_tokenize
from nltk.stem import PorterStemmer
from sklearn.feature_extraction.text import TfidfVectorizer
```

In [2]:
```
file1=open('bag.txt','r')
text=file1.read()
print(text)
```

```
I am eating orange icecream but I love chocolate icecream.
```

In [3]:
```
doc=sent_tokenize(text)
stop_words=set(stopwords.words('english'))
ps=PorterStemmer()
i=0
while(i<len(doc)):
    tokens=word_tokenize(doc[i])
    for token in tokens:
        token=token.lower()
        if token.isalpha():
            doc[i] = doc[i].replace(token,ps.stem(token))
        else:
            doc[i] = doc[i].replace(token,"")
    i+=1
print('The documents are:\n',doc)
```

```
The documents are:
 ['I am eat orang icecream but I love chocol icecream']
```

In [4]:
```
vectorizer=TfidfVectorizer(
    tokenizer=word_tokenize,
    stop_words=stop_words
)
tfidf=vectorizer.fit_transform(doc)
print("The vocabulary is:\n",vectorizer.get_feature_names_out())
print("\nThe TF-IDF is:\n",tfidf.toarray())
```

```
The vocabulary is:
 ['chocol' 'eat' 'icecream' 'love' 'orang']

The TF-IDF is:
 [[0.35355339 0.35355339 0.70710678 0.35355339 0.35355339]]

C:\ProgramData\Anaconda3\lib\site-packages\sklearn\feature_extraction\text.py:396: UserWarning: Your
stop_words may be inconsistent with your preprocessing. Tokenizing the stop words generated tokens
["'d", "'ll", "'re", "'s", "'ve", 'could', 'might', 'must', "n't", 'need', 'sha', 'wo', 'would'] not
in stop_words.
  warnings.warn(
```

In [ ]: