Bag of words Intuition

→ Binary
→ Normal

S_1 → ~~He~~ ~~is~~ a good boy.
S_2 → ~~She~~ ~~is~~ a good girl.
S_3 → Boy ~~and~~ girl ~~are~~ good.

stop keywords → S_1 good boy
removing → S_2 good girl
sentences S_3 Boy girl good.

words	frequency.
good	3
boy	2
girl	2

vectors ⇒ Bag of words

(Histogram)

⇓ we need to sort it in descending order.

independent features | dependent feature

.	f_1 good	f_2 boy	f_3 girl	o/p
S_1	1	1	0	
S_2	1	0	1	
S_3	1	1	1	

BOW.py.

```
import re
        ↳ regular expression.
from nltk.stem import PorterStemmer
from nltk.stem import WordNetLemmatizer
from nltk.corpus import stopwords
# cleaning the text.
ps = PorterStemmer()
lr = WordNetLemmatizer()
sentences = nltk.sent_tokenize(Paragraph)
corpus = []

for i in rang(len(sentences))
    review = re.sub('[^a-z A-Z]', '', sentences[i])
    review = review.lower()
    review = review.split()
    review = [ps.stem(word) for word in review if
      not in word in set(stopwords.words('english'))
    review = ' '.join(review)
    corpus.append(review)
```

Creating the bag of words model.

```
from sklearn.feature_extraction.text import CountVectorizer
cv = countVectorizer(max_features = 1500)
x = cv.fit_transform(corpus).toarray().
```