

Python | Lemmatization with TextBlob

Lemmatization is the process of grouping together the different inflected forms of a word so they can be analyzed as a single item. Lemmatization is similar to stemming but it brings context to the words. So it links words with similar meanings to one word.

Text preprocessing includes both [Stemming](#) as well as Lemmatization. Many times people find these two terms confusing. Some treat these two as the same. Actually, lemmatization is preferred over Stemming because lemmatization does morphological analysis of the words.

Applications of lemmatization are:

- Used in comprehensive retrieval systems like search engines.
- Used in compact indexing.

Examples of lemmatization :

```
-> rocks : rock  
-> corpora : corpus  
-> better : good
```

One major difference with stemming is that lemmatize takes a part of speech parameter, "pos" If not supplied, the default is "noun."

Below is the implementation of lemmatization words using TextBlob:

Python3

```
from textblob import Word  
  
u = Word("`"rocks"`)  
  
print``("`"rocks :"`", u.lemmatize())  
  
v = Word("`"corpora"`)  
  
print``("`"corpora :"`", v.lemmatize())  
  
w = Word("`"better"`)  
  
print``("`"better :"`", w.lemmatize("`"a""))
```

****Output 🤖****

```
rocks : rock  
corpora : corpus  
better : good
```