Stemming vs Lemmatization in NLP: Must-Know Differences

BEGINNER NLP TECHNIQUE USE CASES

This article was published as a part of the Data Science Blogathon.

Introduction

In the field of Natural Language Processing i.e., NLP, Lemmatization and Stemming are **Text Normalization** techniques. These techniques are used to prepare words, text, and documents for further processing.

Languages such as English, Hindi consists of several words which are often derived from one another. Further, **Inflected Language** is a term used for a language that contains derived words. For instance, word "historical" is derived from the word "history" and hence is the derived word.

There is always a common root form for all inflected words. Further, degree of inflection varies from lower to higher depending on the language.

To sum up, root form of derived or inflected words are attained using Stemming and Lemmatization.

The package namely, *nltk.stem* is used to perform stemming via different classes. We import PorterStemmer from *nltk.stem* to perform the above task.

For instance, *ran, runs, and running* are derived from one word i.e., *run*, therefore the lemma of all three words is *run*. Lemmatization is used to get valid words as the actual word is returned.

WordNetLemmatizer is a library that is imported from *nltk.stem* which looks for lemmas of words from the WordNet Database.

Note: Before using the WordNet Lemmatizer, WordNet corpora has to be downloaded from NLTK downloader.

- Lemmatization and Stemming, both are used to generate root form of derived (inflected) words. However, lemma is an actual language word, whereas stem may not be an actual word.
- Lemmatization uses corpus for stop words and WordNet corpus to produce lemma. Moreover, parts-of-speech also had to be defined to obtain correct lemma.
- So, how to decide when to use what! If speed is important, use stemming as lemmatization scan the entire corpus which is a time-consuming task. Secondly, whether stemmers or lemmatizers should be used depends on the application we are working. Finally, if language is important while building a language application, lemmatization is used which scans a corpus to match root forms.

Stemming

It is the process of reducing infected words to their stem. For instance, in figure 1, stemming with replace words "history" and "historical" with *"histori"*. Similarly, for the words finally and final.

Stemming is the process of removing the last few characters of a given word, to obtain a shorter form, even if that form doesn't have any meaning.

Figure 1 showing Stemming

Why we need Stemming?

In NLP use cases such as sentiment analysis, spam classification, restaurant reviews etc., getting base word is important to know whether the word is positive or negative. Stemming is used to get that base word.

Code for Stemming Explained

This section will help you in stemming of paragraph using NLTK which can be used in various use cases such as sentiment analysis, etc.

So let's get started:

Note: *It is highly recommended to use google colab to run this code.*

#1. Import the libraries

Import libraries that will be required for stemming.

```
import nltk nltk.download('stopwords') nltk.download('punkt') from nltk.corpus import stopwords from nltk.stem import PorterStemmer
```

#2. Get the input

The paragraph will be taken as input and used for stemming.

```
paragraph = """I have three visions for India. In 3000 years of our history, people from all over the world have come and invaded us, captured our lands, conquered our minds. From Alexander onwards, the Greeks, the Turks, the Moguls, the Portuguese, the British, the French, the Dutch, all of them came and looted us, took over what was ours. Yet we have not done this to any other nation. We have not conquered anyone. We have not grabbed their land, their culture, their history and tried to enforce our way of life on them. """
```

#3. Tokenization (step before stemming)

Before, stemming, tokenization is done so as to break text into chunks. In this case, paragraph to sentences for easy computation.

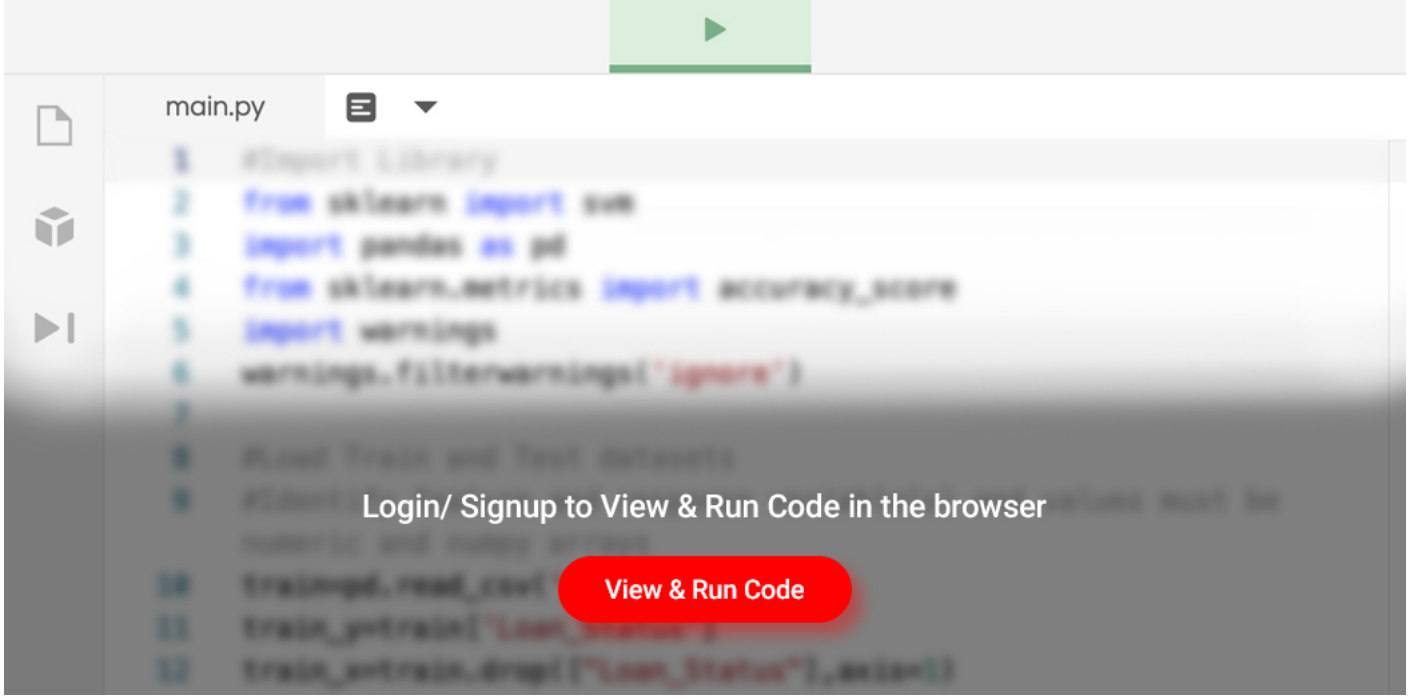
As can be seen from output paragraph is divided into sentences based on "." .

#4. Stemming

In the code given below, one sentence is taken at a time and word tokenization is applied i.e., converting sentence to words. After that, stopwords (such as the, and, etc) are ignored and stemming is applied on all other words. Finally, stem words are joined to make a sentence.

Note: Stopwords are the words that do not add any value to the sentence.

Python Code:



From the above output, we can see that stopwords such as have, for have been removed from sentence one. The word “visions” have been converted to “vision, “history” to “histori” by stemming.

Lemmatization

The purpose of lemmatization is same as that of stemming but overcomes the drawbacks of stemming. In stemming, for some words, it may not give may not give meaningful representation such as “Histori”. Here, lemmatization comes into picture as it gives meaningful word.

Lemmatization takes more time as compared to stemming because it finds meaningful word/ representation. Stemming just needs to get a base word and therefore takes less time.

Stemming has its application in Sentiment Analysis while Lemmatization has its application in Chatbots, human-answering.

Code for Lemmatization Explained

On similar lines of stemming, we will import libraries get input for lemmatization.

#1. Import the libraries

```
import nltk nltk.download('stopwords') nltk.download('punkt') nltk.download('wordnet') from nltk.stem import WordNetLemmatizer from nltk.corpus import stopwords
```

#2. Get the input

```
paragraph = """I have three visions for India. In 3000 years of our history, people from all over the world have come and invaded us, captured our lands, conquered our minds. From Alexander onwards, the Greeks, the Turks, the Moguls, the Portuguese, the British, the French, the Dutch, all of them came and looted us, took over what was ours. Yet we have not done this to any other nation. We have not conquered anyone. We have not grabbed their land, their culture, their history and tried to enforce our way of life on them. """
```

#3. Tokenization (step before stemming)

```
sentences = nltk.sent_tokenize(paragraph) print(sentences)
```

Output:

```
['I have three visions for India.', 'In 3000 years of our history, people from all over \n
```

#4. Lemmatization

The difference between stemming and lemmatization comes in this step where WordNetLemmatizer() is used instead of PorterStemmer(). Rest of steps are the same.

```
lemmatizer = WordNetLemmatizer() # Lemmatization 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)
```

#5. Get the output

```
print(sentences)
```

Output:

In above output, it can be noticed that although word “visions” have been converted to “vision” but word “history” remained “history” unlike stemming and thus retained its meaning.

Stemming v/s Lemmatization

Stemming	Lemmatization
Stemming is a process that stems or removes last few characters from a word, often leading to incorrect meanings and spelling.	**Lemmatization** considers the context and converts the word to its meaningful base form, which is called Lemma.
For instance, stemming the word ‘**Caring**‘ would return ‘**Car**‘.	For instance, lemmatizing the word ‘**Caring**‘ would return ‘**Care**‘.
Stemming is used in case of large dataset where performance is an issue.	Lemmatization is computationally expensive since it involves look-up tables and what not.

Conclusion

One thing to note is that a lot of knowledge and understanding about the structure of language is required for lemmatization. Hence, in any new language, the creation of stemmer is easier in comparison to lemmatization algorithm.

Lemmatization and Stemming are the foundation of derived (inflected) words and hence the only difference between lemma and stem is that lemma is an actual word whereas, the stem may not be an actual language word.
Lemmatization uses a corpus to attain a lemma, making it slower than stemming. Further, to get the proper lemma, you might have to define a parts-of-speech. Whereas, in stemming a step-wise algorithm is followed making it faster.

The above points show that stemming should be used if speed is important since lemmatizers scan a corpus which is a time-consuming task. Further, the choice between lemmatizers and stemmers also depends on the problem you are working on.

Article Url - https://www.analyticsvidhya.com/blog/2022/06/stemming-vs-lemmatization-in-nlp-must-know-differences/

Saumyab271