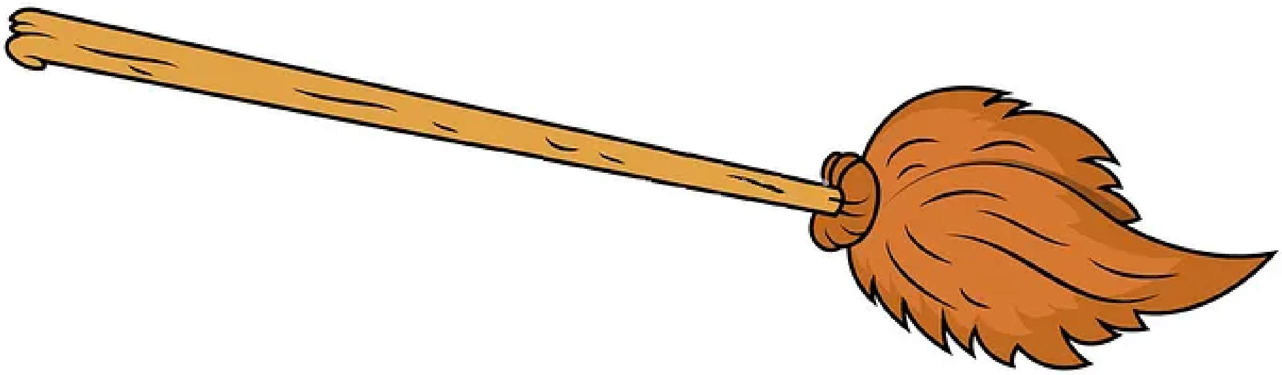
Text cleaning (using Regex) [Python]

Yash Jain · Follow
3 min read · Feb 18, 2022



Source: storyblocks.com

We need to learn how to work with unstructured data to be able to extract relevant information from it and make it useful. While working with text data it is very important to pre-process it before using it for predictions or analysis.

Let’s take an example

```
text = """@blogger It is possible to have an anaphor that has no
lexical\
zero anaphor realization at all, called \na zero anaphor or zero
pronoun, as in the https://medium.com following Italian\n\
and Japanese examples from Poesio et al. (2016):
(21.15) EN [John]i bla bla http://medium.com #NLP @blogger"""
```

this above text contains mention(@) , url, hashtag, numbers, reference in square brackets(`[]`), newline character (\n), these are some data that we don’t want in our text. Let’s tackle these one by one.

We’ll use `re.sub` -> **“Return the string obtained by replacing the leftmost non-overlapping occurrences of the pattern in string by the replacement repl. repl can be either a string or a callable; if a string, backslash escapes in it are processed. If it is a callable, it’s passed the Match object and must return a replacement string to be used.”**

Syntax

```
import re #-> regex library
re.sub(pattern, repl, string, count=0, flags=0)  ##syntax
## repl -> replacement string
```

> *Regex cheatsheet*

- **Removing mentions (@)**

We used pattern “@\S+” -> it suggests string group which starts with ‘@’ and followed by non-whitespace character(\S), ‘+’ means repeatition of preceding character one or more times, `\S+` → here it represents one or more non-whitespace characters.

```
import re
re.sub(r"@\S+", "",text) # removing @blogger
```

Output:

> ' It is possible to have an anaphor that has no lexicalzero anaphor realization at all, called \na zero anaphor or zero pronoun, as in the following Italian\nand Japanese examples from Poesio et al. (2016):\n(21.15) EN [John]i bla bla http://medium.com #NLP '

- **Removing urls (*http://*……)**

`?` → preceding character may or may not be present in the string,
`+` → 1 or more repetitions

```
re.sub("http[s]?\://\S+","",text) # removing http://medium.com
```

Output:

> '@blogger It is possible to have an anaphor that has no lexicalzero anaphor realization at all, called \na zero anaphor or zero pronoun, as in the following Italian\nand Japanese examples from Poesio et al. (2016):\n(21.15) EN [John]i bla bla #NLP @blogger'

- **Removing hashtag (#…)**

It is similar as removing mentions

```
re.sub(r"#\S+", "",text) # removing #NLP
```

Output

> '@blogger It is possible to have an anaphor that has no lexicalzero anaphor realization at all, called \na zero anaphor or zero pronoun, as in the https://medium.com following Italian\nand Japanese examples from Poesio et al. (2016):\n(21.15) EN [John]i bla bla http://medium.com @blogger'

- **Removing numbers (1,2,3..)**

[0–9] → represents range of numbers from 0 to 9

here we have replaced all numbers with empty string

```
re.sub(r"[0-9]", "",text) # removing 2016, 21 15
```

> ‘@blogger It is possible to have an anaphor that has no lexicalzero anaphor realization at all, called \na zero anaphor or zero pronoun, as in the https://medium.com following Italian\nand Japanese examples from

> Poesio et al. ():\n(.) EN [John]i bla bla http://medium.com #NLP @blogger’

- **Removing text in brackets ([…] or (…))**

```
re.sub(r"(\(.*\))|(\[.*\])", "",text)
# removes (21.15), [John],(2016)
```

`|` this pipe character represents or operator which includes both `()` and `[]` exclusion

Output:

> '@blogger It is possible to have an anaphor that has no lexicalzero anaphor realization at all, called \na zero anaphor or zero pronoun, as in the https://medium.com following Italian\nand Japanese examples from Poesio et al. :\n EN i bla bla http://medium.com #NLP @blogger'

- **Removing line or tab character (\n, \r, \t..)**

```
re.sub(r"\n", "",text) # removing \n
```

Output:

> '@blogger It is possible to have an anaphor that has no lexicalzero anaphor realization at all, called a zero anaphor or zero pronoun, as in the https://medium.com following Italianand Japanese examples from

> Poesio et al. (2016):(21.15) EN [John]i bla bla http://medium.com #NLP @blogger'

- **Let's combine url, square and round brackets, mentions and hashtag & `\n`**

```
re.sub(r"(http[s]?\://\S+)|([\[\(].*[\)\]])|([#@]\S+)|\n", "",text)
```

Output:

> ' It is possible to have an anaphor that has no lexicalzero anaphor realization at all, called a zero anaphor or zero pronoun, as in the following Italianand Japanese examples from Poesio et al. :i bla bla '

- **Removing extra space**

\s → matches any whitespace characters such as space and tab

```
text = 'VERY EXTRA      SPACE     '
re.sub('\s+',' ',text)
```

Output:

> 'VERY EXTRA SPACE '

I hope this helps in text cleaning in some way… You can learn regex expression and practice some interesting examples here.

Regex Python Cleaning String Removing

More from the list: "NLP"

Curated by Himanshu Birla

Jon Gi... in Towards Data ...

Characteristics of Word Embeddings

✦ · 11 min read · Sep 4, 2021

Jon Gi... in Towards Data ...

The Word2vec Hyperparameters

✦ · 6 min read · Sep 3, 2021

Jon Gi... in

The Word2ve

✦ · 15 min rea

View list

Written by Yash Jain

Follow

93 Followers

Data Scientist/ Data Engineer at IBM | Alumnus of @niituniversity | Natural Language Processing | Pronouns: He, Him, His

More from Yash Jain

Yash Jain

Spell check and correction[NLP, Python]

In Natural Language Processing it's important that spelling errors should be as less as...

4 min read · Feb 19, 2022

21 1

Yash Jain

Stopwords [NLP, Python]

Stop words are common words in any language that occur with a high frequency b...

6 min read · Feb 23, 2022

8

Yash Jain

POS Tagging [NLP, Python]

POS tagging is important to get an idea that which parts of speech does tokens belongs t...

5 min read · Feb 27, 2022

16 1

Yash Jain

Lemmatization [NLP, Python]

Lemmatization is the process of replacing a word with its root or head word called lemm...

4 min read · Feb 22, 2022

13 2

See all from Yash Jain

Recommended from Medium

Rashini Liyanarachchi in Text Mining Basics

RegEx for Text Mining — I

RegEx or Regular Expressions are used in Text Mining mainly for Basic Patterns or…

7 min read · Jul 25

1

Nimrita Koul

Natural Langauge Processing with Python Part 3: Text Preprocessing…

This article is the third in the series of my articles covering the sessions I delivered for…

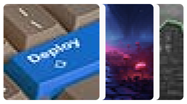
4 min read · Apr 26

2

Lists

Coding & Development

11 stories · 200 saves

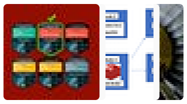


Predictive Modeling w/ Python

20 stories · 452 saves

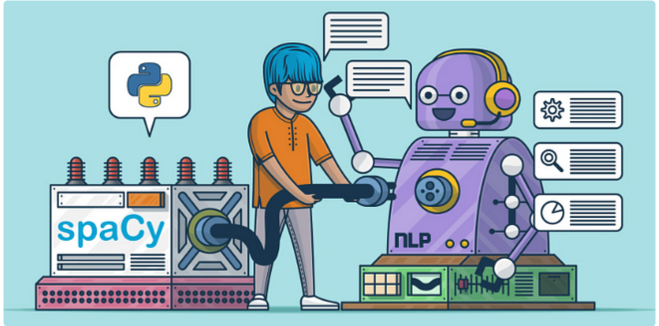
Practical Guides to Machine Learning

10 stories · 519 saves



New_Reading_List

174 stories · 133 saves



H HasancanÇakıcıoğlu

Comprehensive Text Preprocessing NLP (Natural Language…

Text preprocessing plays a crucial role in Natural Language Processing (NLP) by…

12 min read · Jul 9

S Sanjithkumar

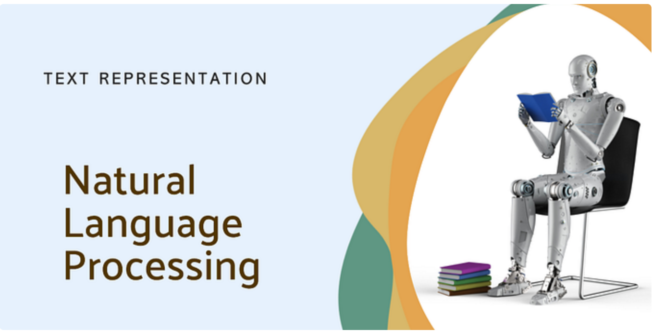
Text Preprocessing Part — 2

Text preprocessing is an integral part of Natural Language Processing as no machine…

5 min read · Sep 7

12

27 1



Susovan Dey

From Traditional to Modern: A Comprehensive Guide to Text...

Natural Language Processing (NLP) is a rapidly growing field that focuses on enablin...

6 min read · Apr 27

3

JYOTI KHETAN

A Practical Guide to TF-IDF and Term Frequency in Text Analysis

Introduction:

3 min read · May 14

37

See more recommendations