```
import re, string
```

Double-click (or enter) to edit

Loading the data

```
filename = '/content/novel.txt'
file = open(filename, 'rt', encoding='utf-8')
text = file.read()
file.close()
```

Split the data into words by whitespace

```
words = text.split()
print(words[:120])
```

```
['/:;<=@', 'One', 'morning,', 'when', 'Gregor', 'Samsa', 'woke', 'from', 'troubled', 'dreams,', 'he', 'found', 'himself', 'transformed', 'in', 'his', 'bed', 'into', 'a'
```

here we are seeing that the punctuation is preserved (e.g. armour-like and wasn't), which is nice. We can also see that end of sentence punctuation is kept with the last word (e.g. thought.) which is not great.

so this time let's try to split the words at non-word character

+ Code + Text

```
words = re.split(r'\W+', text)
print(words[:120])
```

```
['', 'One', 'morning', 'when', 'Gregor', 'Samsa', 'woke', 'from', 'troubled', 'dreams', 'he', 'found', 'himself', 'transformed', 'in', 'his', 'bed', 'into', 'a', 'horrit
```

Here we are seeing that the words like thought. have been converted into thought. But the problem is that the words like wasn't are converted into two words like wasn and t. We need to fix it.

In python, we can use string.punctuation to get bunch of punctuations at once. We will use that to remove punctuations from our text

```
print(string.punctuation)
```

```
!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~
```

So now we will split the words by whitespace and then remove all the punctuations which have been recorded in the data

```
words = text.split()
re_punc = re.compile('[%s]' % re.escape(string.punctuation))
stripped = [re_punc.sub('', word) for word in words]
print(stripped[:120])
```

```
['', 'One', 'morning', 'when', 'Gregor', 'Samsa', 'woke', 'from', 'troubled', 'dreams', 'he', 'found', 'himself', 'transformed', 'in', 'his', 'bed', 'into', 'a', 'horrit
```

Here we can see that we don't have the words like thought. but we have words like wasn't

Sometimes the text also contains the characters which are not printable. We need to filter those out too. To do this, we can use python string.printable which gives us bunch of characters that can be printed. So, we will remove the characters which are not present in this.

```
re_print = re.compile('[^%s]' % re.escape(string.printable))
result = [re_print.sub('', word) for word in stripped]
```

```
print(result[:120])
```

```
['', 'One', 'morning', 'when', 'Gregor', 'Samsa', 'woke', 'from', 'troubled', 'dreams', 'he', 'found', 'himself', 'transformed', 'in', 'his', 'bed', 'into', 'a', 'horrit
```

Now let's make all the words into lowercase. This will reduce our vocabolary. But this has some disadvantages also. After doing this, two words such as Apple as in company and apple as a fruit will be considered a same entity.

```
result = [word.lower() for word in result]
print(result[:120])
```

```
['', 'one', 'morning', 'when', 'gregor', 'samsa', 'woke', 'from', 'troubled', 'dreams', 'he', 'found', 'himself', 'transformed', 'in', 'his', 'bed', 'into', 'a', 'horribl
```

Also, words with one character won't contribute to most of the NLP tasks. So we will be removing those too.

```
result = [word for word in result if len(word) > 1]
```

```
print(result[:120])
```

```
['one', 'morning', 'when', 'gregor', 'samsa', 'woke', 'from', 'troubled', 'dreams', 'he', 'found', 'himself', 'transformed', 'in', 'his', 'bed', 'into', 'horrible', 'ver
```

Colab paid products - Cancel contracts here

✓ 0s completed at 11:14 AM