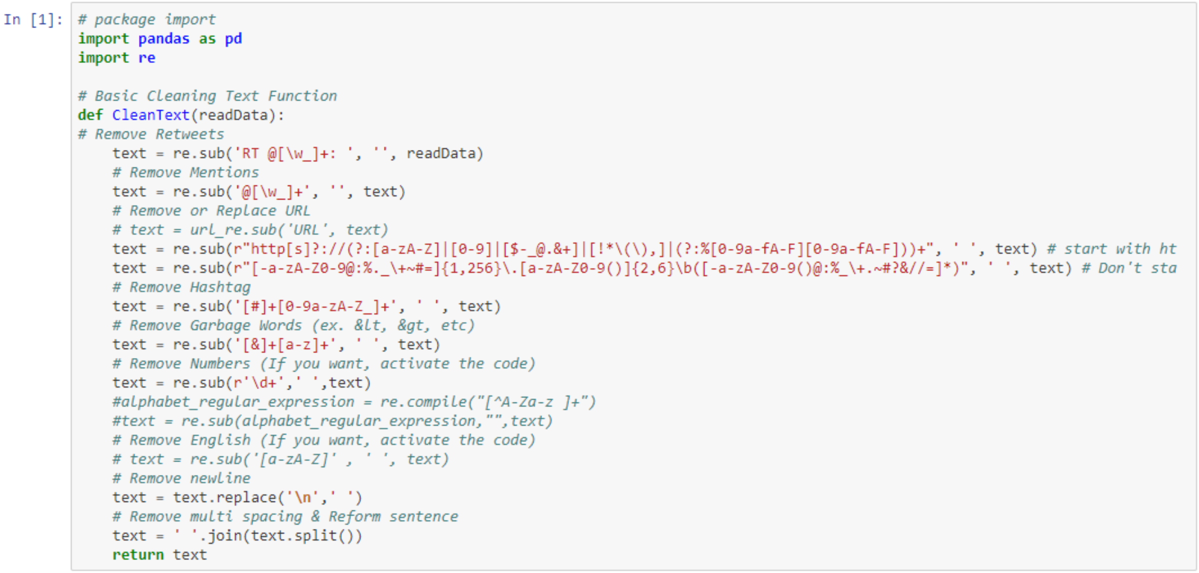
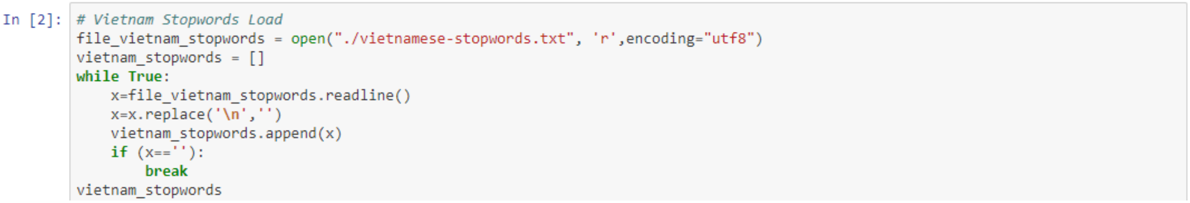
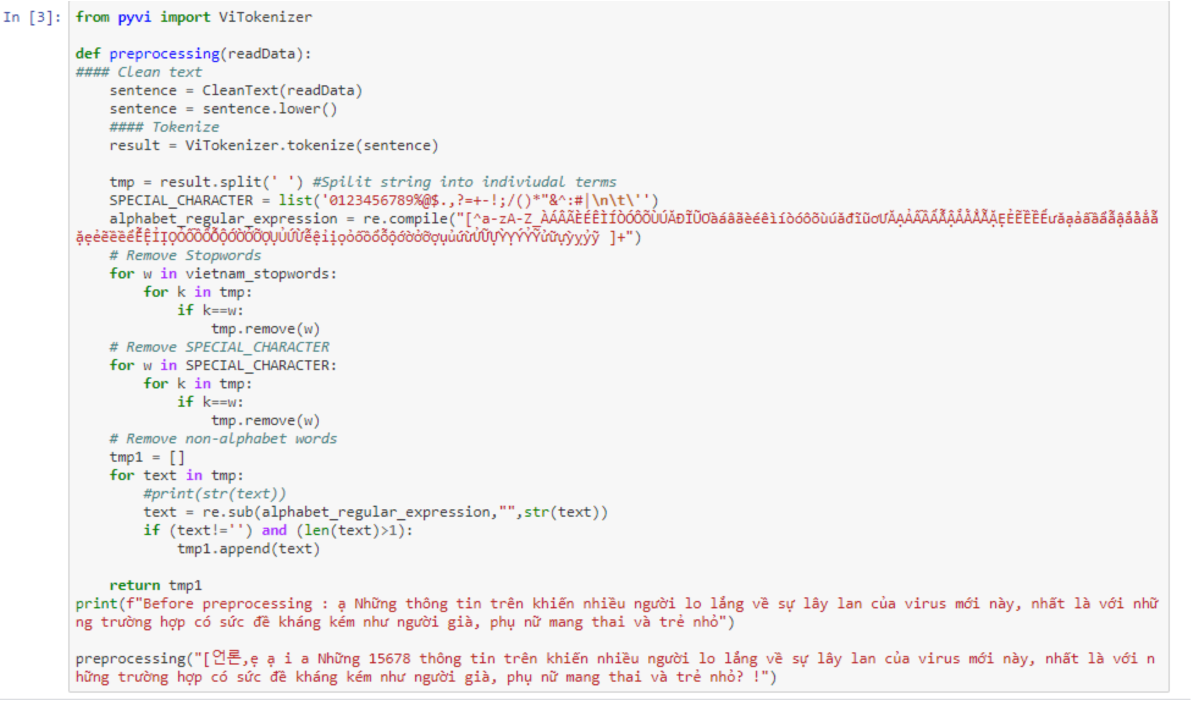


```
In [1]: # package import
        import pandas as pd
        import re

        # Basic Cleaning Text Function
        def CleanText(readData):
        # Remove Retweets
            text = re.sub('RT @[\w_]+: ', '', readData)
            # Remove Mentions
            text = re.sub('@[\w_]+', '', text)
            # Remove or Replace URL
            # text = url_re.sub('URL', text)
            text = re.sub(r"http[s]?://(?:[a-zA-Z]|[0-9]|[$-_@.&+]|[!*\(\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+", ' ', text) # start with ht
            text = re.sub(r"[-a-zA-Z0-9@:%._\+~#=]{1,256}\.[a-zA-Z0-9()]{2,6}\b([-a-zA-Z0-9()@:%_\+.~#?&//=]*)", ' ', text) # Don't sta
            # Remove Hashtag
            text = re.sub('[#]+[0-9a-zA-Z_]+', ' ', text)
            # Remove Garbage Words (ex. &lt, &gt, etc)
            text = re.sub('[&]+[a-z]+', ' ', text)
            # Remove Numbers (If you want, activate the code)
            text = re.sub(r'\d+',' ',text)
            #alphabet_regular_expression = re.compile("[^A-Za-z ]+")
            #text = re.sub(alphabet_regular_expression,"",text)
            # Remove English (If you want, activate the code)
            # text = re.sub('[a-zA-Z]' , ' ', text)
            # Remove newline
            text = text.replace('\n',' ')
            # Remove multi spacing & Reform sentence
            text = ' '.join(text.split())
            return text
```



```
In [2]: # Vietnam Stopwords Load
        file_vietnam_stopwords = open("./vietnamese-stopwords.txt", 'r',encoding="utf8")
        vietnam_stopwords = []
        while True:
            x=file_vietnam_stopwords.readline()
            x=x.replace('\n','')
            vietnam_stopwords.append(x)
            if (x==''):
                break
        vietnam_stopwords
```



```
In [3]: from pyvi import ViTokenizer

        def preprocessing(readData):
        #### Clean text
            sentence = CleanText(readData)
            sentence = sentence.lower()
            #### Tokenize
            result = ViTokenizer.tokenize(sentence)

            tmp = result.split(' ') #Spilit string into indiviudal terms
            SPECIAL_CHARACTER = list('0123456789%@$.,?=+-!;/()*"&^:#|\n\t\'')
            alphabet_regular_expression = re.compile("[^a-zA-Z_ÀÁÂÃÈÉÊÌÍÒÓÔÕÙÚĂĐĨŨƠàáâãèéêìíòóôõùúăđĩũơƯĂẠẢẤẦẨẪẬẮẰẲẴẶẸẺẼỀỀỂưăạảấầẩẫậắằẳẵ
        ặẹẻẽềềếỄỆỈỊỌỎỐỒỔỖỘỚỜỞỠỢỤỦỨỪễệỉịọỏốồổỗộớờởỡợụủứừỬỮỰỲỴÝỶỸửữựỳỵỷỹ ]+")
            # Remove Stopwords
            for w in vietnam_stopwords:
                for k in tmp:
                    if k==w:
                        tmp.remove(w)
            # Remove SPECIAL_CHARACTER
            for w in SPECIAL_CHARACTER:
                for k in tmp:
                    if k==w:
                        tmp.remove(w)
            # Remove non-alphabet words
            tmp1 = []
            for text in tmp:
                #print(str(text))
                text = re.sub(alphabet_regular_expression,"",str(text))
                if (text!='') and (len(text)>1):
                    tmp1.append(text)

            return tmp1
        print(f"Before preprocessing : ạ Những thông tin trên khiến nhiều người lo lắng về sự lây lan của virus mới này, nhất là với nhữ
        ng trường hợp có sức đề kháng kém như người già, phụ nữ mang thai và trẻ nhỏ")

        preprocessing("[언론,ẹ ạ i a Những 15678 thông tin trên khiến nhiều người lo lắng về sự lây lan của virus mới này, nhất là với n
        hững trường hợp có sức đề kháng kém như người già, phụ nữ mang thai và trẻ nhỏ? !")
```

- For Vietnamese, Pyvi was used to tokenize.

- Used stopwords from dictionaries and customed ones.
- Add text normalization codes