Họ tên: Lưu Hoài Linh

MSSV: 17021284

Nhóm 1

NLP task - report

Github:

1. Trong folder NLP Task gồm những files sau:

|  |  |
| --- | --- |
| * NLP Task.ipynb: | code  code |
| * Nlptask.py: |
| * NLP\_report.docx: | report |
| * Trainning.txt: | input  input  input |
| * Testing.txt: |
| * Labels.txt: |
| * Stopwords.txt: | tổng hợp các stop words trong tiếng Việt |
| * Predicts.txt: | outputs: label của dữ liệu file testing.txt |

2. Xây dựng mô hình phân lớp gồm 4 bước

* Pre-processing data

+ Load dữ liệu

+ Tách riêng phần label và data trong file trainning.txt

+ Xử lý tiếng Việt

+ Loại bỏ stop words, những thành phần không phải là chữ cái

* Extracting feature

+ Sử dụng kĩ thuật *tfidf* đánh giá độ quan trọng của từ

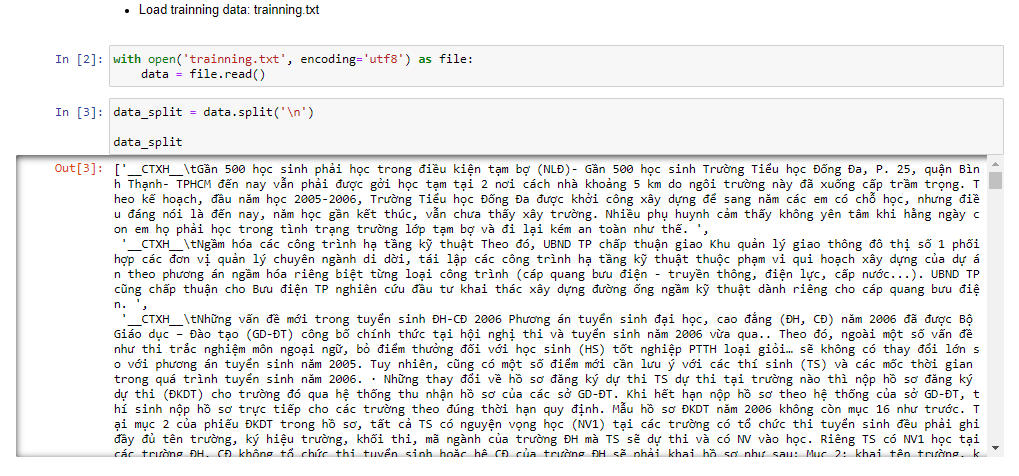
* Algorithrms

+ Đánh giá Naïve Bayes và SVM và sử dụng các tham số khác nhau

* Classify

2.1: Data pre-processing:

* Sau khi load trainning.txt và split từng phần tử:



* Gỡ bỏ các label (có dạng *\_\_labelname\_\_*) đứng trước từng phần tử để lấy được training data, bằng cách sử dụng regex trong python:

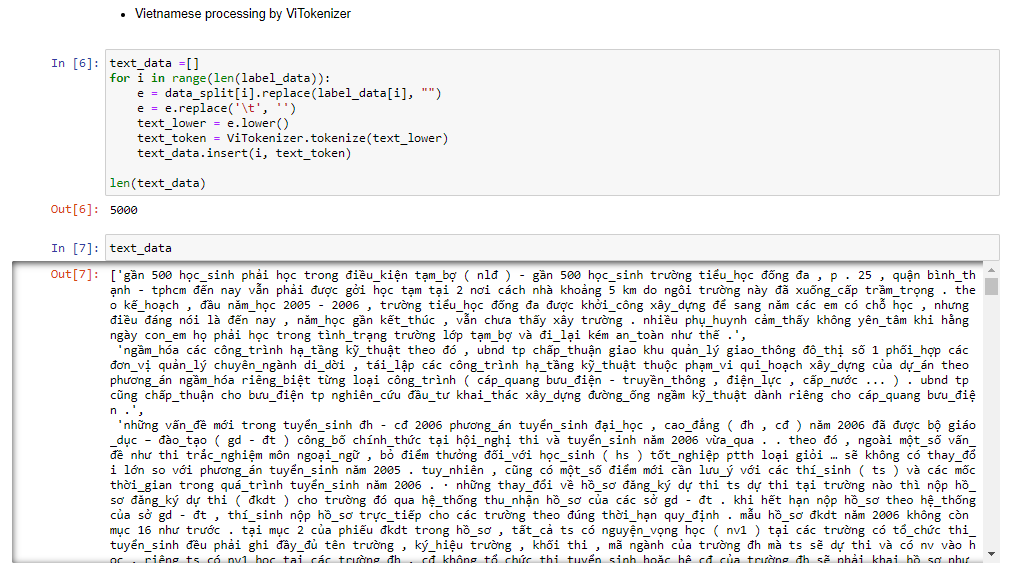


* Sử dụng lib pyvi để xử lý tiếng Việt: do trong tiếng Việt 1 từ có thể được ghép bởi 2 chữ, nếu đứng rời sẽ ảnh hưởng đến chất lượng phân lớp.

(ví dụ: *đất nước* nếu tách riêng *đất* và *nước* vẫn có nghĩa nhưng đã bị sai nghĩa)

* Sau khi loại bỏ thành phần phụ, chuyển về *lower text*, dữ liệu được tách từ bằng

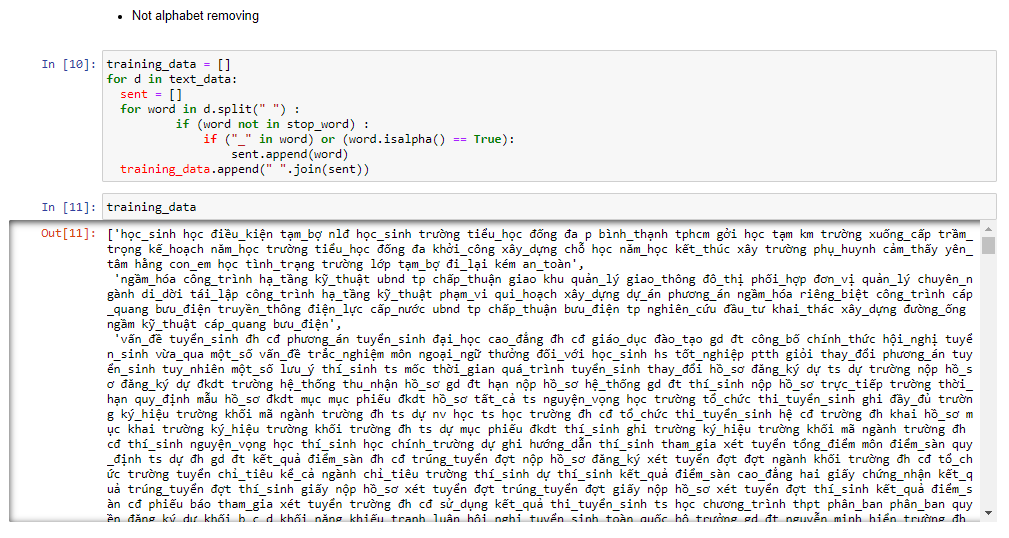
*ViTokenizer.tokenize*, các từ 2 chữ sẽ được nối bởi dấu gạch (\_)



* Loại bỏ các stop words trong tiếng Việt (tổng hợp trong file stopwords.txt), do có thể trở thành nhiễu trong việc training mô hình phân lớp

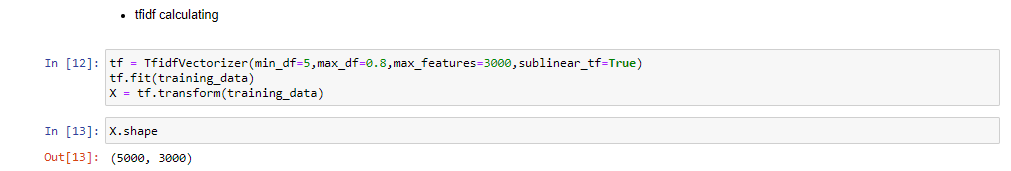


* Loại bỏ những thành phần không phải chữ cái



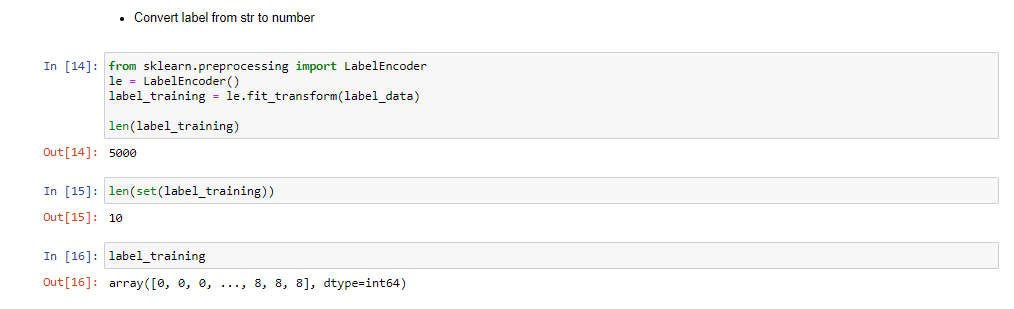
2.2. Extracting feature

* Sử dụng kĩ thuật *tfidf* đánh giá độ quan trọng của từ



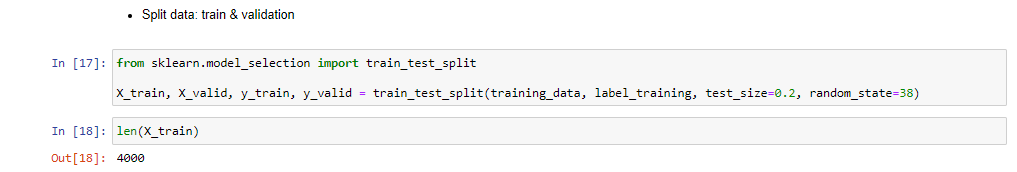
2.3. Naïve Bayes & SVM

* Chuyển dạng label từ *string* sang *number* (kiểm tra đúng số lượng label: 10)

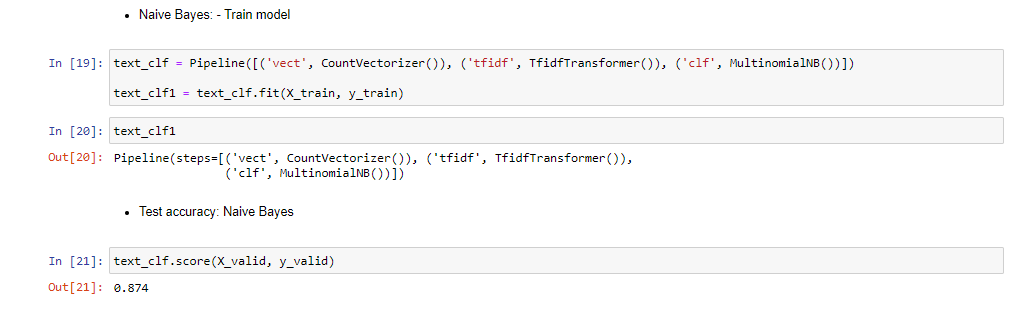


* Chia tập training data thành 2 phần: trainning và validation để kiểm tra độ chính xác của mô hình (*X\_train, X\_valid, y\_train, y\_valid*)

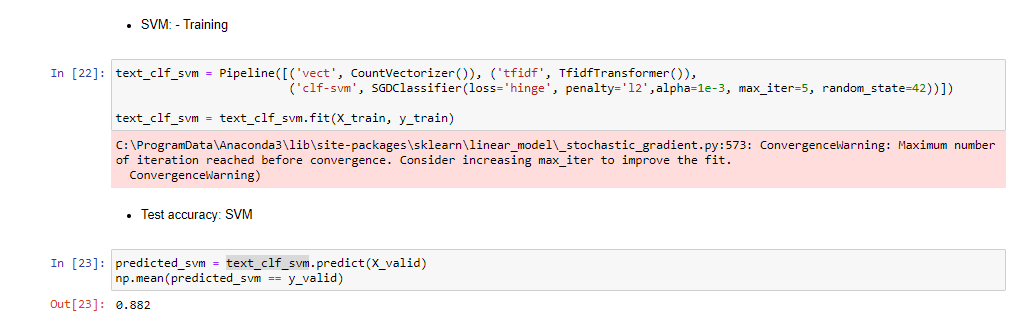
+ Với test\_size = 0.2 (4000/5000 dữ liệu) và random\_state = 38 (sau khi thử các giá trị trong khoảng từ 0 – 42)



* Với Naïve Bayes sử dụng *pipeline* để train mô hình phân lớp, sau đó tính *accuracy* của mô hình = *0.874*

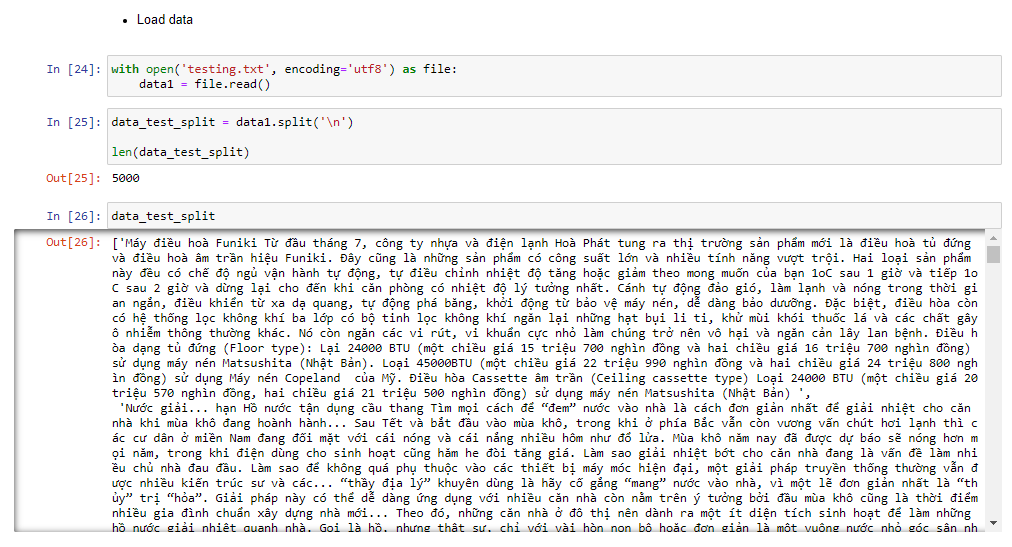


* Tương tự với SVM, độ *accuracy* cũng xấp xỉ NB, = *0.882*

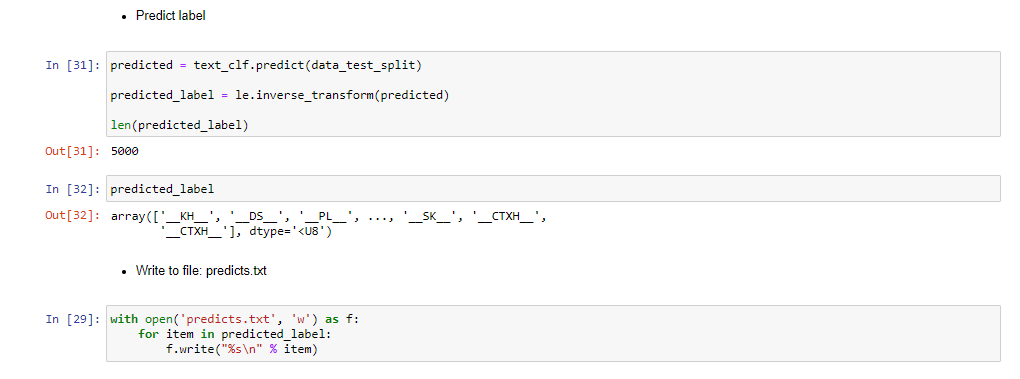


2.4. Classify

* Load test\_data gồm 5000 dòng



* Lựa chọn: Naïve Bayes để phân lớp
* Label dự đoán được lưu vào file predicts.txt



3. Kết quả

* Các label cần dự đoán nằm trong file: *predicts.txt*