

---

# **UE10 Extraction et programmation statistique de l'information**

**LAI Khang Duy**  
**Mazen TEBIB**

*Université Paris-Saclay*  
*Faculté des sciences d'Orsay*

**université**  
**PARIS-SACLAY**

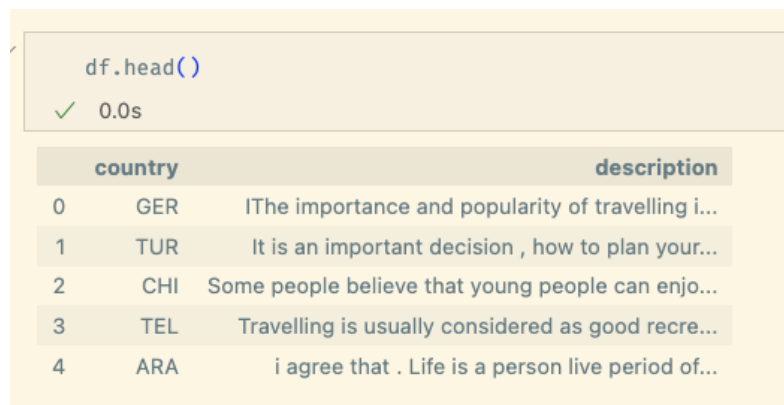
23-03-2023

## Contents

<b>1</b>	<b>Data exploration</b>	<b>2</b>
<b>2</b>	<b>Data preprocessing</b>	<b>2</b>
2.1	Clean raw texts . . . . .	3
2.2	TF-IDF Vectorizer . . . . .	3
2.3	Encoding labels . . . . .	4
2.4	Extracting features . . . . .	4
2.4.1	Lexical features . . . . .	4
2.4.2	Syntatic features . . . . .	4
<b>3</b>	<b>Models</b>	<b>4</b>
3.1	Model selection . . . . .	4
3.2	Cascade method . . . . .	6
<b>4</b>	<b>Hyperparameter tuning</b>	<b>7</b>

## 1 Data exploration

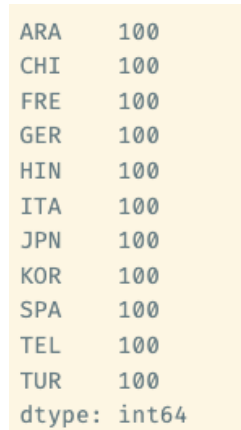
First, we want to see which exactly in the corpus in order to further do the data preprocessing. As you can see in the image of Raw data set, we have 2 columns in the corpus, `country` and the `description`. `country` is functioned as the label `y`, whereas `description` is the data itself.



	country	description
0	GER	IThe importance and popularity of travelling i...
1	TUR	It is an important decision , how to plan your...
2	CHI	Some people believe that young people can enjo...
3	TEL	Travelling is usually considered as good recre...
4	ARA	i agree that . Life is a person live period of...

**Figure 1:** Raw data set

We have 10 labels stand for 10 different native languages. The number of data points in each label is equal.



ARA	100
CHI	100
FRE	100
GER	100
HIN	100
ITA	100
JPN	100
KOR	100
SPA	100
TEL	100
TUR	100

dtype: int64

**Figure 2:** Number of data point in each class

## 2 Data preprocessing

In this step, we apply a few data preprocessing techniques before feed the processed data to the model. These steps include cleaning the texts, using IF-IDF to vectorized it, encode the labels in the

column `country` and Extract important features.

## 2.1 Clean raw texts

Normally, this process includes also removing stop words, lemmatization and stemming. However, due to the fact that removing such thing might remove very important features when the people who use English as second language could do such mistake too, so in this case we are not going to include it. We only include:

- Lowering all texts
- Remove numbers
- Remove punctuations
- Remove special characters

	country	description	cleaned_description
0	GER	IThe importance and popularity of travelling i...	ithe importance and popularity of travelling i...
1	TUR	It is an important decision , how to plan your...	it is an important decision how to plan your s...
2	CHI	Some people believe that young people can enjo...	some people believe that young people can enjo...
3	TEL	Travelling is usually considered as good recre...	travelling is usually considered as good recre...
4	ARA	i agree that . Life is a person live period of...	i agree that life is a person live period of t...

**Figure 3:** After cleaning corpus

## 2.2 TF-IDF Vectorizer

Split the data into test set and train set before use tf-idf to vectorize it. If you learn tf-idf also on the test set you will have data leakage. In example, you won't have out-of vocabulary words in inference on the test set, what might happen on the real world.

```
1 #would be better if u split data before then fit then transform
2 tf = TfidfVectorizer(stop_words='english', ngram_range=(1,2))
3 tf_Xtrain = tf.fit_transform(X_train)
4 tf_Xtest = tf.transform(X_test)
```

We have tried to adjust the `ngram_range` between `ngram_range=(2,3)` and `ngram_range=(1,2)`. The best one is `(1,2)`.

## 2.3 Encoding labels

We cannot use the direct label to feed the model. Before feeding the model, it is necessary to encode it into the form of numbers. We added a new column in the table. `country` is the label and `y_country` is the encoded one, from 0 to 9 according to 10 labels.

country	y_country
---------	-----------

## 2.4 Extracting features

### 2.4.1 Lexical features

### 2.4.2 Syntactic features

sthg

## 3 Models

We have tried several method for the model of this subject. First, we tried to determine which is the best classification model. Second, we apply the hierarchy method (cascade method), in order to see if there are any improvements.

### 3.1 Model selection

In order to find the best model that work with the corpus, we must run several models and then select the model with the best score overall.

The list of models that we used including:

- Nearest Neighbors
- Linear SVC
- Random Forest
- Kmeans
- GaussianNB

```

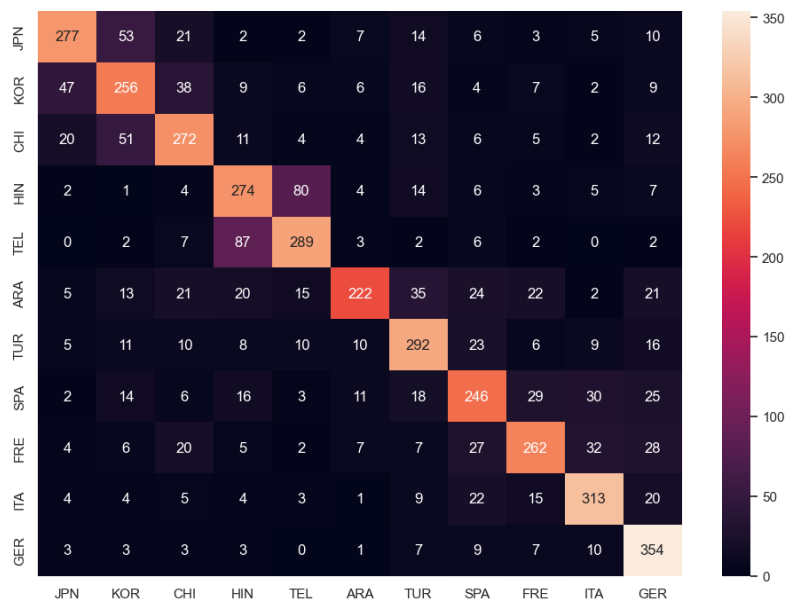
for classifier : Nearest Neighbors
accuracy 0.216363636363637
for classifier : Linear SVC
accuracy 0.7163636363636363
for classifier : Random Forest
accuracy 0.3990909090909091
for classifier : Kmeans

```

**Figure 4:** Model selection results

With the LinearSVC, we have the best score overall so in the next cascade method I will use LinearSVC as our model.

Here is the confusion matrix of LinearSVC with the average accuracy of 70% if we classify language directly.



**Figure 5:** LinearSVC confusion matrix

I re run the test with randomly train test split 4 times to take the average heat map confusion matrix.

### 3.2 Cascade method

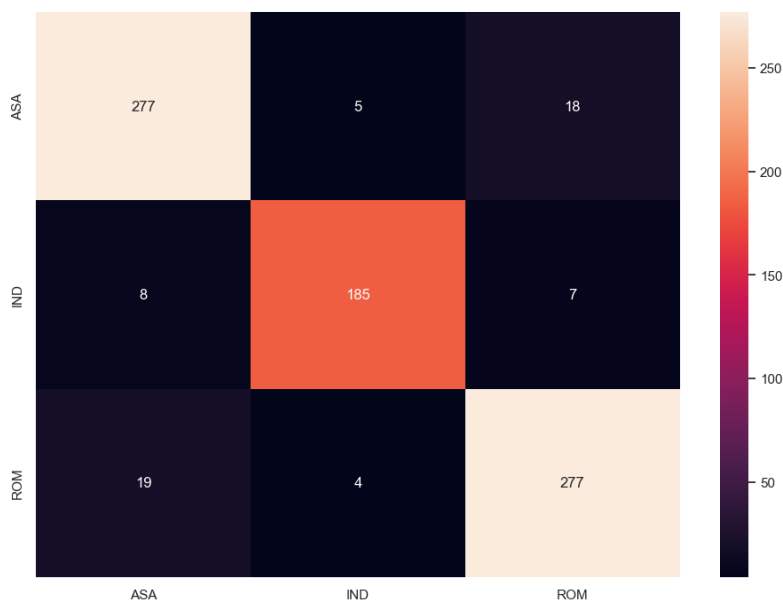
After we find the best model, we implement cascade method on the project. The idea is simple. We have 2 layers of models. First, we train the model so that it could well separate the language groups. We divided 10 languages into 4 language groups as below:

- Asian language group
- Roman language group
- Indian language group
- Other languages

accuracy 0.92375

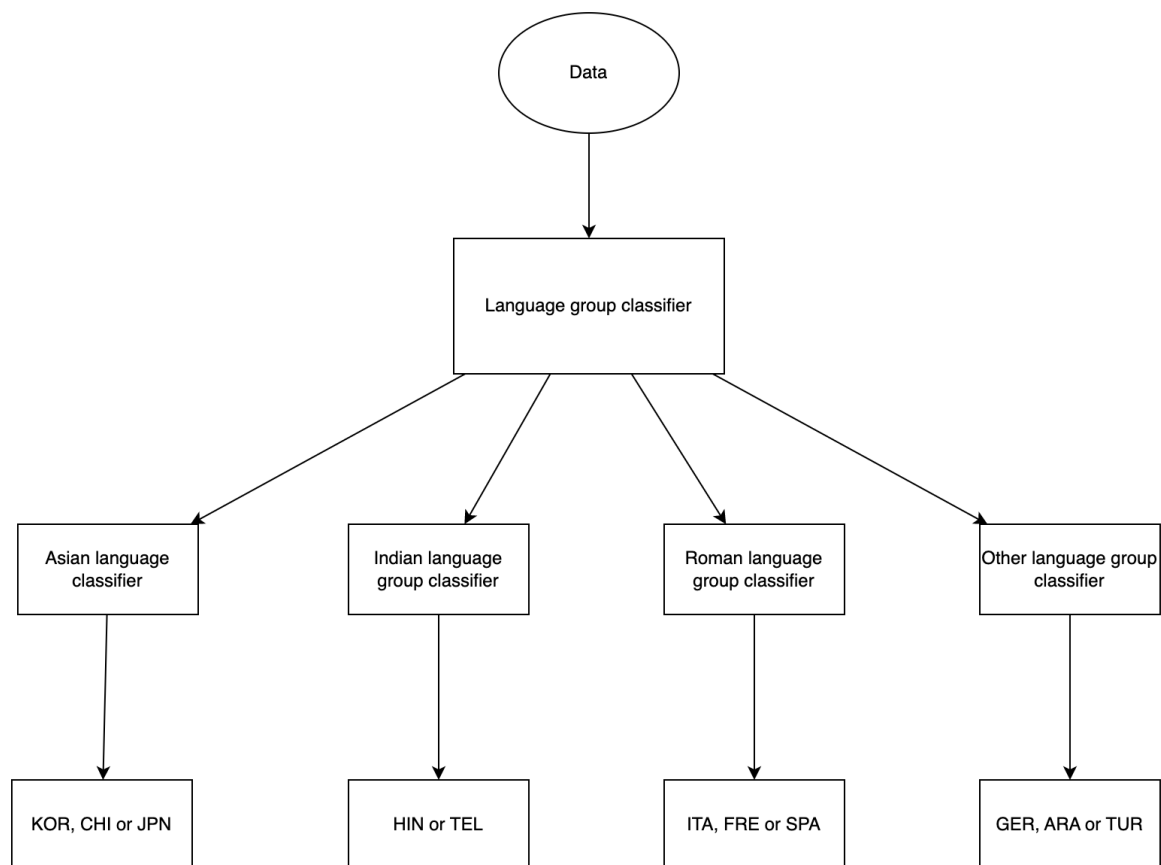
**Figure 6:** Language group confusion matrix

We can achieve the confident of 93% with group separation. Below is the confusion matrix between language groups.



**Figure 7:** Language group confusion matrix

After that, which data point have the highest confident we will pass it in to the language specific layer. You can see the visualization of it as the diagram below.



**Figure 8:** Cascade diagram

## 4 Hyperparameter tuning

We can change the threshold of the language group layer and language specific layer until we have the best score.

```
1 hierarchical_predict(X, group_clf, classifiers, threshold_group=0.8,  
    threshold_specific=0.5):
```

the `threshold_group` and the `threshold_specific` varies from 0.5 to 0.8, and the impact on the last score is minor. So we keep the score as 0.5.