# Using Huggingface language models

NLP master course 2022-2023

Mariano Rico (mariano.rico@upm.es)

Document created on 2023-01-05

## Table of contents

# 1 Using HuggingFace models from R

The key is the `reticulate` package. This package allows you to use any python package from R. You have the documentation here.

## 1.1 Checking the python environment

If you have installed the `keras` R package (the installation is not obvious) you will have a *Python virtual environment* `r-reticulate`. You can check it with this:

```
library(reticulate)

virtualenv_list() #List all available virtualenvs
```

```
[1] "r-reticulate"
```

```
#conda_list() #List all available (mini)conda envs. In this case, none.
```

To know all the details of the Python that is being used, we will do:

```
py_config() #What python(s) is/are installed and where is the ejecutable
```

```
python:         /home/rstudio/.local/share/r-miniconda/envs/r-reticulate/bin/python
libpython:      /home/rstudio/.local/share/r-miniconda/envs/r-reticulate/lib/libpython3.8.so
pythonhome:     /home/rstudio/.local/share/r-miniconda/envs/r-reticulate:/home/rstudio/.local/share/r-mi
version:        3.8.13 | packaged by conda-forge | (default, Mar 25 2022, 06:04:18)  [GCC 10.3.0]
numpy:          /home/rstudio/.local/share/r-miniconda/envs/r-reticulate/lib/python3.8/site-packages/nu
numpy_version:  1.23.4
```

If you are using (mini)conda you have to use a different function:

```
py_discover_config() #Use this for (mini)conda
```

```
python:         /home/rstudio/.local/share/r-miniconda/envs/r-reticulate/bin/python
libpython:      /home/rstudio/.local/share/r-miniconda/envs/r-reticulate/lib/libpython3.8.so
pythonhome:     /home/rstudio/.local/share/r-miniconda/envs/r-reticulate:/home/rstudio/.local/share/r-mi
version:        3.8.13 | packaged by conda-forge | (default, Mar 25 2022, 06:04:18)  [GCC 10.3.0]
numpy:          /home/rstudio/.local/share/r-miniconda/envs/r-reticulate/lib/python3.8/site-packages/nu
numpy_version:  1.23.4
```

## 1.2 Installing the HuggingFace module in Python

Using the `reticulate` package we can install the module/library/package Python `transformers` with:

```
use_virtualenv("r-reticulate")#Use a given virtualenv
packs <- py_list_packages(type = "virtualenv") #Returns a df
if(!"transformers" %in% packs$package){  #If not in the list of instaled Python libs
  py_install("transformers", pip = TRUE) #Install it
}
```

### 1.3 Testing NLP tasks

We have to load the `transformers` Python module. The R package `reticulate` can use the methods of the module:

```
transformers <- reticulate::import("transformers")
```

The `transformers` R object has a function `pipeline` that we can use with `transformers$pipeline()`. The `pipeline` function has and argument `task` to specify the kind of NLP task to do.

## 2 Text classification

We can call the "text-classification" task without specifying a model. In this case, a default model (in English) will be used, which occupies 268MB on disk:

```
classifier <- transformers$pipeline(task = "text-classification")
```

As the warning messages indicate, it is recommended to indicate the model that you want to use. We can use the model we just obtained:

```
classifier("I have a serious problem")
```

```
[[1]]
[[1]]$label
[1] "NEGATIVE"

[[1]]$score
[1] 0.9993677
```

```
classifier("I am filling good")
```

```
[[1]]
[[1]]$label
[1] "POSITIVE"

[[1]]$score
[1] 0.999871
```

As you can see, this model (at least at the time of execution of this source code) classifies text as positive or negative (it's a binary classifier).

### 2.1 Now in Spanish

If you try to do it for Spanish using something like this:

```
classifier <- transformers$pipeline(task = "text-classification",
                                    model= "PlanTL-GOB-ES/roberta-base-bne")
```

You will get an error message like this:

```
Downloading: 100%|      | 613/613 [00:00<00:00, 574kB/s]
Error in py_call_impl(callable, dots$args, dots$keywords) :
 ValueError: Could not load model PlanTL-GOB-ES/roberta-base-bne with any of the
 following classes: (
 <class 'transformers.models.auto.modeling_tf_auto.TFAutoModelForSequenceClassification'>,
 <class 'transformers.models.roberta.modeling_tf_roberta.TFRobertaForMaskedLM'>.
```

This error is because some models are written in PyTorch, so we need to have PyTorch (and some other classes) installed. The easiest is to execute the following:

```
reticulate::py_install("pytorch-pretrained-bert", pip = TRUE)#torch (887MB) and others
```

After the installation of the python packages, the R session must be restarted so that the values of some python environment variables are updated.
To restart the R session, click on the RStudio menu: Session–>Restart R.

If R is not restarted you will get the following error (we saw it before):

```
Downloading: 100%|      | 613/613 [00:00<00:00, 407kB/s]
Error in py_call_impl(callable, dots$args, dots$keywords) :
  ValueError: Could not load model PlanTL-GOB-ES/roberta-base-bne with any of the
  following classes: (
  <class 'transformers.models.auto.modeling_tf_auto.TFAutoModelForSequenceClassification'>,
  <class 'transformers.models.roberta.modeling_tf_roberta.TFRobertaForMaskedLM'>
  ).
```

IMPORTANT: Even if R is restarted, the downloaded models are NOT lost, so they could grow up quickly and require a lot of disk space. The same goes for installed python packages. Therefore, check from time to time the size occupied by:

1) HuggingFace models. On linux they are stored in `/home/rstudio/.cache/huggingface/hub` (with `rstudio` being the RStudio user).
2) the python packages. On linux they are stored in `/home/rstudio/.virtualenvs/r-reticulate/lib/python3.8/site-packages` (with `rstudio` being the RStudio user), with directories `transformers`, `pytorch_pretrained_bert`, etc., each with multiple `.py` scripts.

Now we can execute the following code (it takes a while to download the 600MB model):

```
library(reticulate)
use_virtualenv("r-reticulate")
packs <- py_list_packages(type = "virtualenv") #Returns a df
if(!"transformers" %in% packs$package){
  py_install("transformers", pip = TRUE)
}
if(!"pytorch-pretrained-bert" %in% packs$package){
  py_install("pytorch-pretrained-bert", pip = TRUE)
}
transformers <- reticulate::import("transformers")
classifier <- transformers$pipeline(task = "text-classification",
                                    model="PlanTL-GOB-ES/roberta-base-bne")
```

Despite the intimidanting warning, we continue. This is the warning:

```
Downloading: 100%|     | 499M/499M [00:49<00:00, 10.1MB/s]
Some weights of the model checkpoint at PlanTL-GOB-ES/roberta-base-bne were not used when
initializing RobertaForSequenceClassification: ['lm_head.decoder.weight', 'lm_head.bias',
'lm_head.dense.bias', 'lm_head.dense.weight', 'lm_head.decoder.bias',
'lm_head.layer_norm.weight', 'lm_head.layer_norm.bias']
- This IS expected if you are initializing RobertaForSequenceClassification from the
checkpoint of a model trained on another task or with another architecture (e.g. initializing
a BertForSequenceClassification model from a BertForPreTraining model).
- This IS NOT expected if you are initializing RobertaForSequenceClassification from the
checkpoint of a model that you expect to be exactly identical (initializing a
BertForSequenceClassification model from a BertForSequenceClassification model).
Some weights of RobertaForSequenceClassification were not initialized from the model
checkpoint at PlanTL-GOB-ES/roberta-base-bne and are newly initialized: ['classifier.out_proj.bias', 'cl
'classifier.dense.weight']
You should probably TRAIN this model on a down-stream task to be able to use it for
predictions and inference.
Downloading: 100%|     | 1.39k/1.39k [00:00<00:00, 1.23MB/s]
Downloading: 100%|     | 851k/851k [00:00<00:00, 1.59MB/s]
Downloading: 100%|     | 509k/509k [00:00<00:00, 1.03MB/s]
Downloading: 100%|     | 2.21M/2.21M [00:00<00:00, 2.64MB/s]
Downloading: 100%|     | 957/957 [00:00<00:00, 695kB/s]
```

We can use the model:

```
classifier("Tengo un problema serio")
```

```
[[1]]
[[1]]$label
[1] "LABEL_1"

[[1]]$score
[1] 0.53885
```

```
classifier("Me siento fenomenal")
```

```
[[1]]
[[1]]$label
[1] "LABEL_1"

[[1]]$score
[1] 0.5383498
```

Clearly, the model is not working properly.

This is because this model is prepared for another task: “fill-mask” (you can see the model documentation here) .

```
unmasker <- transformers$pipeline(task = "fill-mask",
                                  model="PlanTL-GOB-ES/roberta-base-bne")
unmasker("Gracias a los datos de la BNE se ha podido <mask> este modelo del lenguaje.")
```

```
[[1]]
```

```
[[1]]$score
[1] 0.08422067

[[1]]$token
[1] 3832

[[1]]$token_str
[1] " desarrollar"

[[1]]$sequence
[1] "Gracias a los datos de la BNE se ha podido desarrollar este modelo del lenguaje."


[[2]]
[[2]]$score
[1] 0.06348325

[[2]]$token
[1] 3078

[[2]]$token_str
[1] " crear"

[[2]]$sequence
[1] "Gracias a los datos de la BNE se ha podido crear este modelo del lenguaje."


[[3]]
[[3]]$score
[1] 0.0614842

[[3]]$token
[1] 2171

[[3]]$token_str
[1] " realizar"

[[3]]$sequence
[1] "Gracias a los datos de la BNE se ha podido realizar este modelo del lenguaje."


[[4]]
[[4]]$score
[1] 0.05621832

[[4]]$token
[1] 10880

[[4]]$token_str
[1] " elaborar"

[[4]]$sequence
[1] "Gracias a los datos de la BNE se ha podido elaborar este modelo del lenguaje."
```

```
[[5]]
[[5]]$score
[1] 0.05133353

[[5]]$token
[1] 31915

[[5]]$token_str
[1] " validar"

[[5]]$sequence
[1] "Gracias a los datos de la BNE se ha podido validar este modelo del lenguaje."
```

## 3 Part of speech

You have a Part Of Speech neural model for Spanish: Warning! The model uses 1.4GB disc space.

```
nlp_pos <- transformers$pipeline(task = "token-classification",
                                 model="PlanTL-GOB-ES/roberta-large-bne-capitel-pos")

res <- nlp_pos(paste(
              "Festival de San Sebastián: Johnny Depp recibirá el premio Donostia",
              "en pleno rifirrafe judicial con Amber Heard"))
data.frame(entity=sapply(res, function(x){x$entity}),
           score =sapply(res, function(x){x$score}),
           index =sapply(res, function(x){x$index}),
           word  =sapply(res, function(x){x$word}),
           start =sapply(res, function(x){x$start}),
           end   =sapply(res, function(x){x$end})
          )
```

| entity | score | index | word | start | end |
|---|---|---|---|---|---|
| NOUN | 0.9996961 | 1 | ĠFestival | 0 | 8 |
| ADP | 0.9996126 | 2 | Ġde | 9 | 11 |
| PROPN | 0.9993528 | 3 | ĠSan | 12 | 15 |
| PROPN | 0.9884313 | 4 | ĠSebastiÃ¡n | 16 | 25 |
| PUNCT | 0.9493771 | 5 | : | 25 | 26 |
| PROPN | 0.9998086 | 6 | ĠJohnny | 27 | 33 |
| PROPN | 0.9978847 | 7 | ĠDe | 34 | 36 |
| PROPN | 0.8299615 | 8 | pp | 36 | 38 |
| VERB | 0.9997770 | 9 | ĠrecibirÃ¡ | 39 | 47 |
| DET | 0.9998410 | 10 | Ġel | 48 | 50 |
| NOUN | 0.9998456 | 11 | Ġpremio | 51 | 57 |
| PROPN | 0.9996608 | 12 | ĠDonostia | 58 | 66 |
| ADP | 0.9998453 | 13 | Ġen | 67 | 69 |
| ADJ | 0.8772469 | 14 | Ġpleno | 70 | 75 |
| NOUN | 0.9996101 | 15 | Ġrif | 76 | 79 |
| X | 0.7009529 | 16 | ir | 79 | 81 |
| X | 0.6158468 | 17 | ra | 81 | 83 |
| NOUN | 0.9970248 | 18 | fe | 83 | 85 |

| entity | score | index | word | start | end |
|---|---|---|---|---|---|
| ADJ | 0.9998363 | 19 | Ġjudicial | 86 | 94 |
| ADP | 0.9998705 | 20 | Ġcon | 95 | 98 |
| PROPN | 0.9998962 | 21 | ĠAmber | 99 | 104 |
| PROPN | 0.7110988 | 22 | ĠHe | 105 | 107 |
| X | 0.9438123 | 23 | ard | 107 | 110 |

# 4 Name Entity recognition (NER)

We will use the ner-plus model from PlanTL, that takes 500MB disc space. The ner model is heavier (1.5GB).

```
nlp_ner <- transformers$pipeline(task = "ner",
                                 model="PlanTL-GOB-ES/roberta-large-bne-capitel-ner")
res <- nlp_ner(paste(
              "Festival de San Sebastián: Johnny Depp recibirá el premio Donostia",
              "en pleno rifirrafe judicial con Amber Heard"))
data.frame(entity=sapply(res, function(x){x$entity}),
           score =sapply(res, function(x){x$score}),
           index =sapply(res, function(x){x$index}),
           word  =sapply(res, function(x){x$word}),
           start =sapply(res, function(x){x$start}),
           end   =sapply(res, function(x){x$end})
          )
```

| entity | score | index | word | start | end |
|---|---|---|---|---|---|
| B-OTH | 0.9795215 | 1 | ĠFestival | 0 | 8 |
| I-OTH | 0.9771883 | 2 | Ġde | 9 | 11 |
| I-OTH | 0.9894046 | 3 | ĠSan | 12 | 15 |
| E-OTH | 0.9754279 | 4 | ĠSebastiÃ¡n | 16 | 25 |
| B-PER | 0.9998310 | 6 | ĠJohnny | 27 | 33 |
| E-PER | 0.9994376 | 7 | ĠDe | 34 | 36 |
| E-PER | 0.9986013 | 8 | pp | 36 | 38 |
| S-OTH | 0.9964671 | 12 | ĠDonostia | 58 | 66 |
| B-PER | 0.9998432 | 21 | ĠAmber | 99 | 104 |
| E-PER | 0.9997476 | 22 | ĠHe | 105 | 107 |
| E-PER | 0.9975891 | 23 | ard | 107 | 110 |