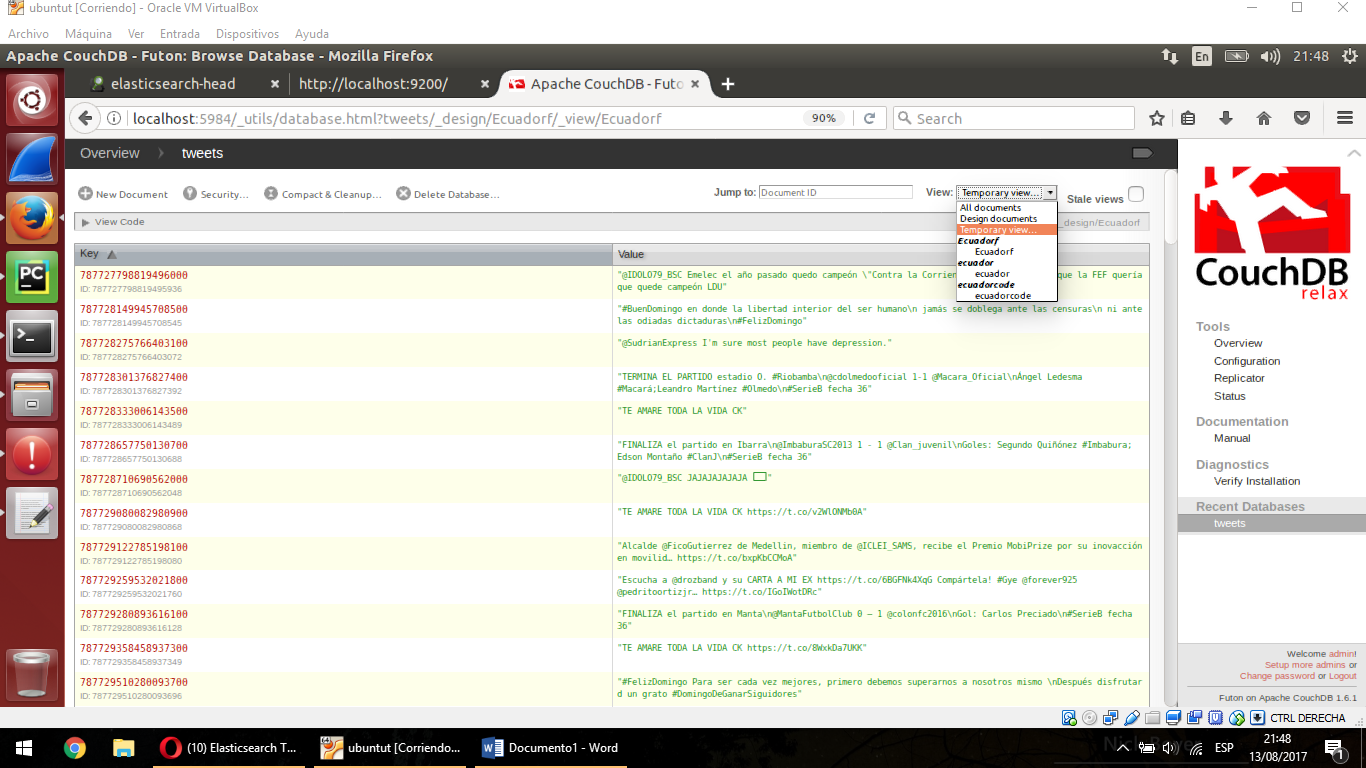
Tweets clasificador

Pasos para el clasificación y filtrado de tweets

Se crea una vista para filtrar tweets

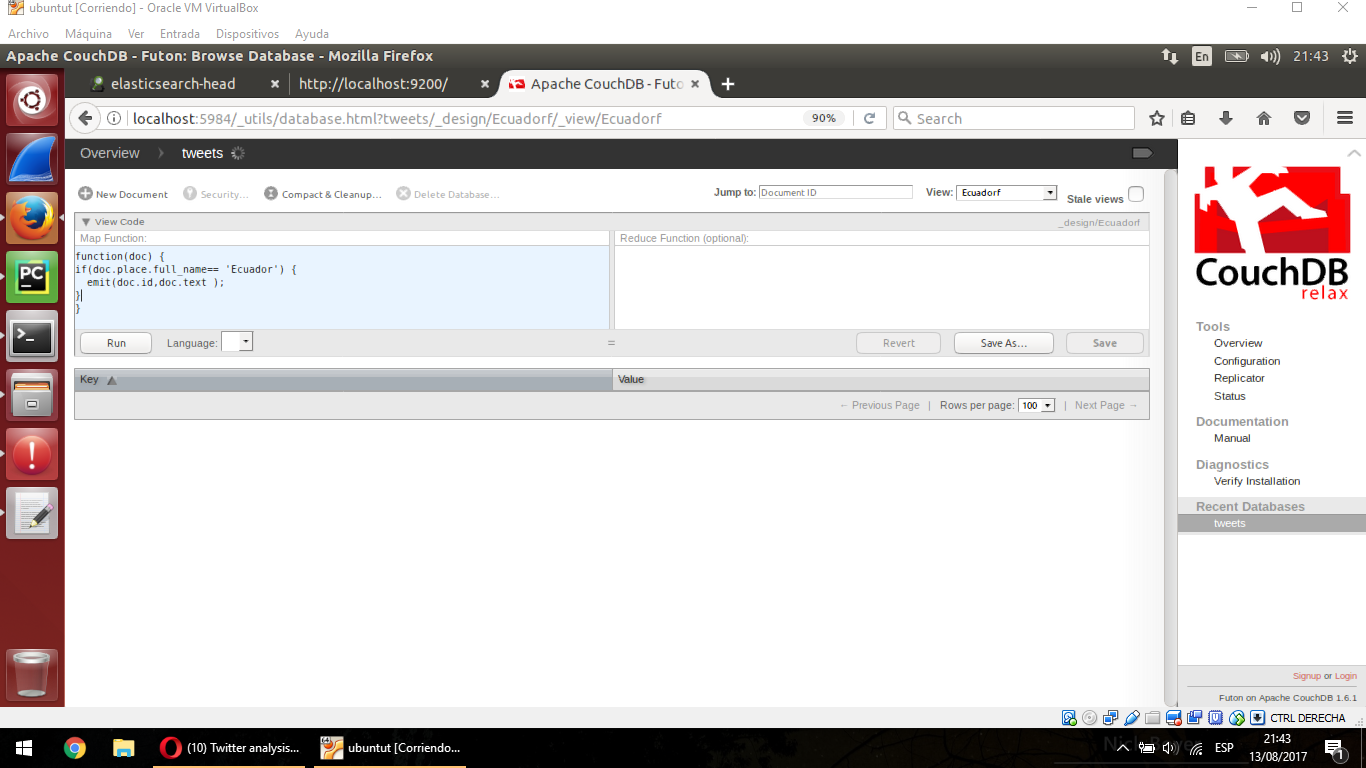


Se procede a crear una vista filtrando los tweets de la región de Ecuador ya que durante la recolección existió tweets que pertenecían a otras regiones del mundo la más común fue Perú

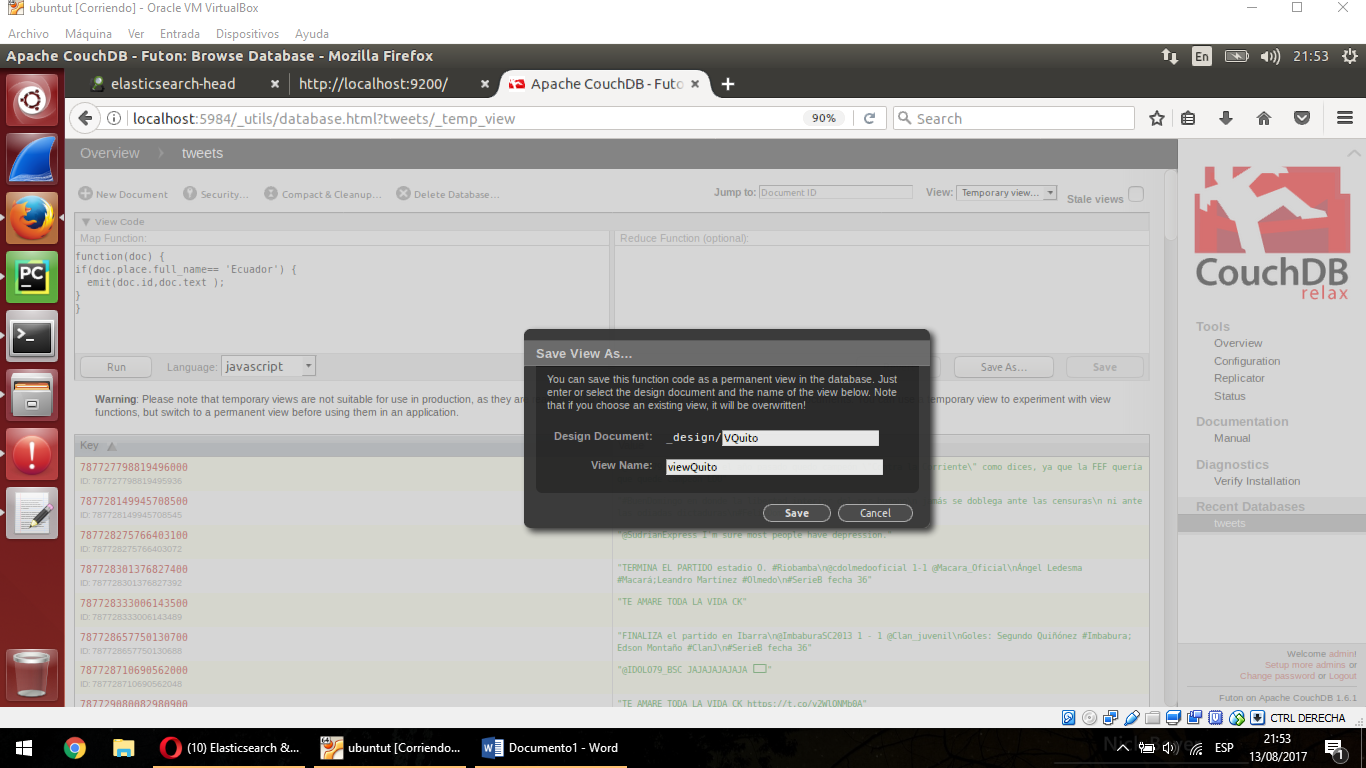
Este código empleamos para filtrar los tweets en función del full\_name=Ecuador

function(doc) {

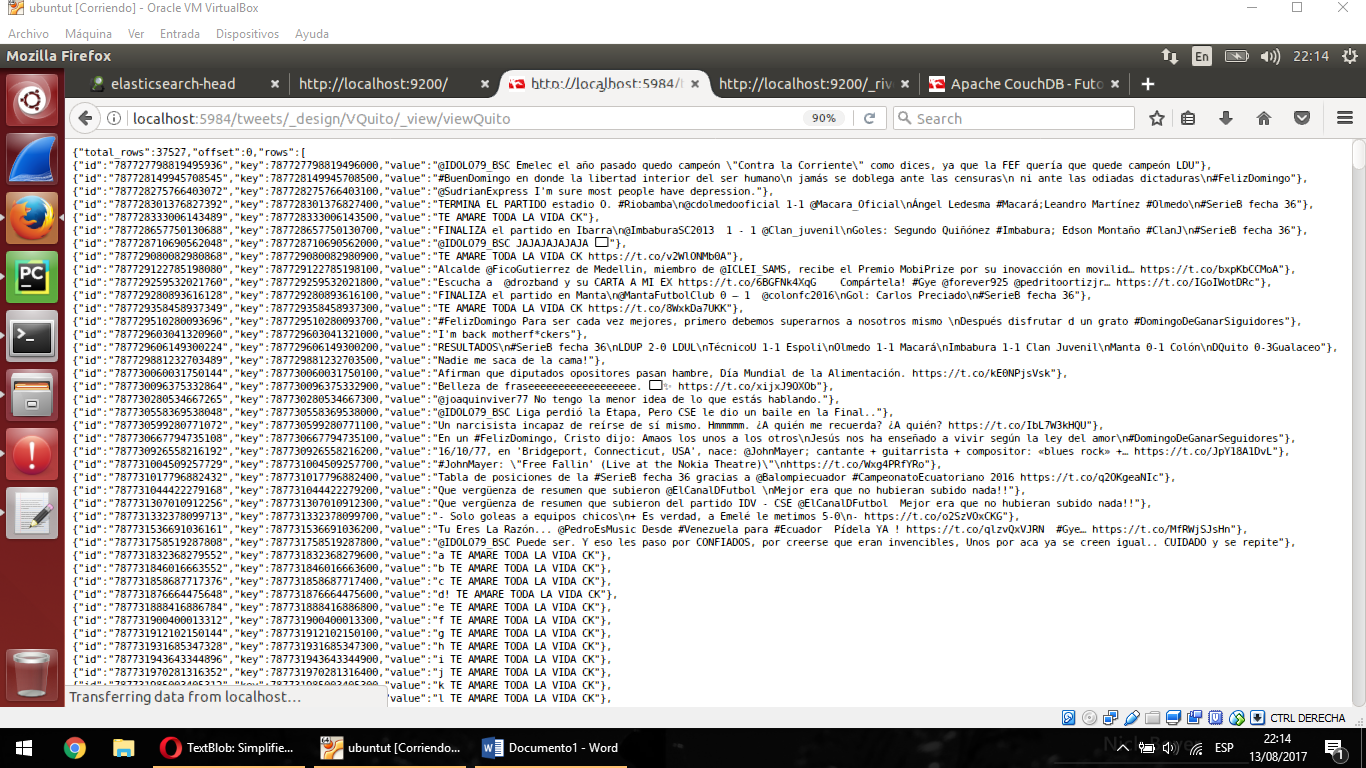
if(doc.place.full\_name== 'Ecuador') {

emit(doc.id,doc.text );}} 

Procedemos a guardar la vista

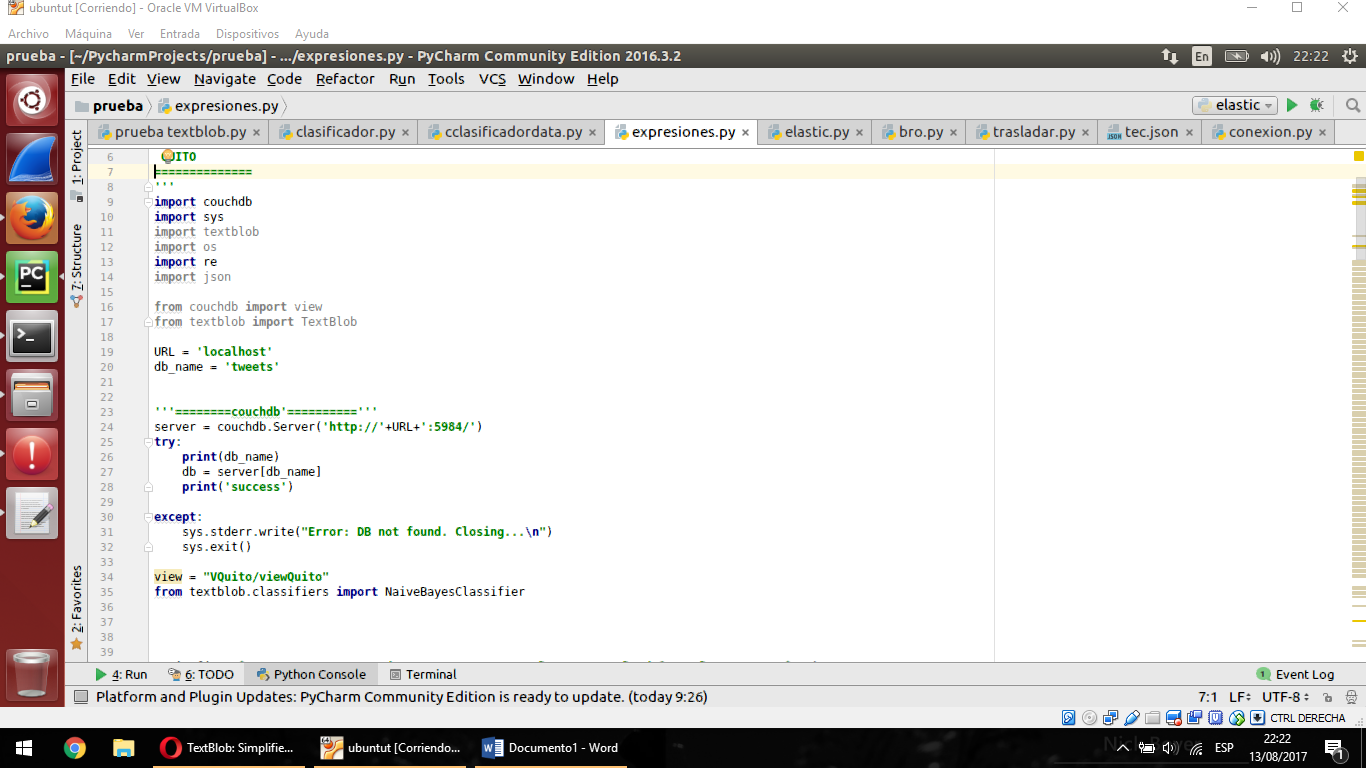
Para poder acceder a nuestra vista nos dirigimos al siguiente enlace aquí observamos que los tweets no están clasificados.

<http://localhost:5984/tweets/_design/VQuito/_view/viewQuito>



En Pycharm modificamos el código para el filtrado de texto de cada uno de los tweets

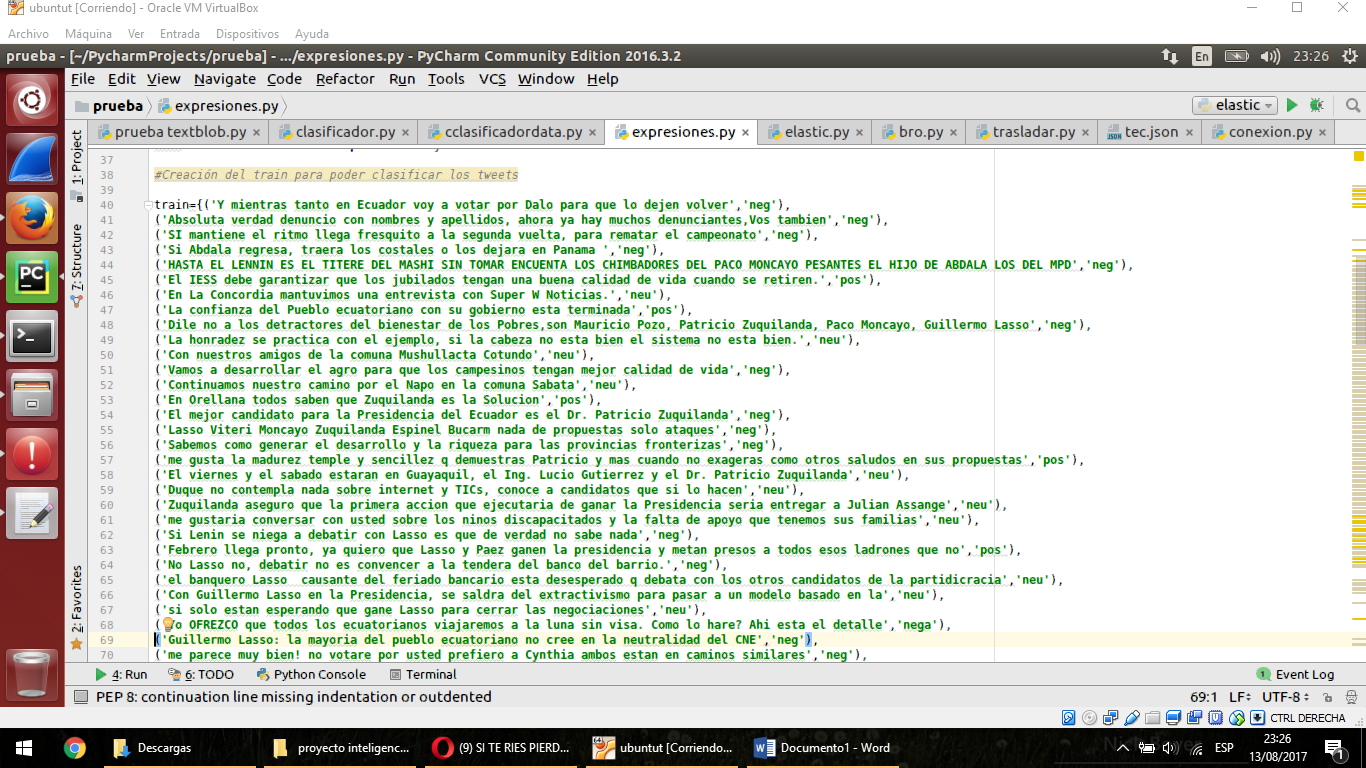
En esta linea de codigo accedemos a la vista que creamos anteriormente view = "VQuito/viewQuito"



Posterior a esto importamos el clasificador de textblob para la posterior clasificación.

from textblob.classifiers import NaiveBayesClassifier

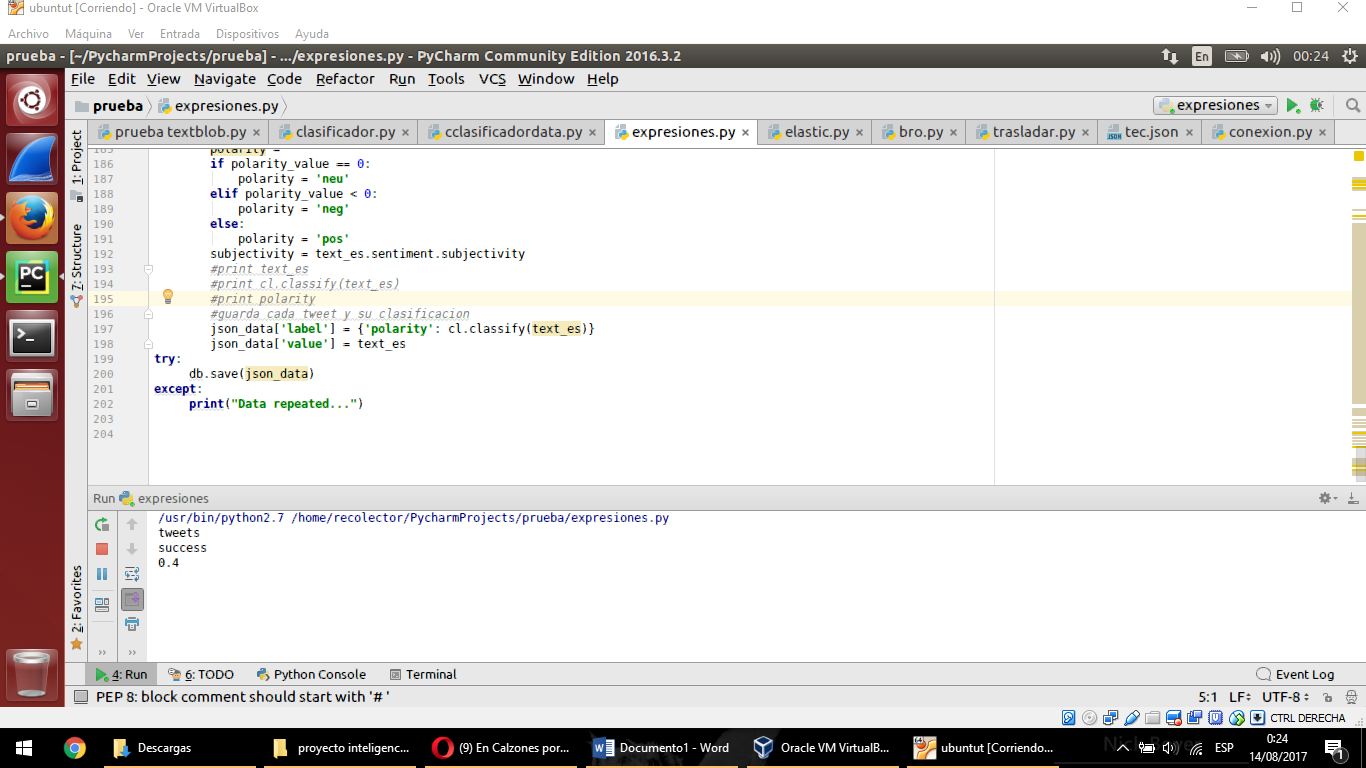
Creación del train para poder clasificar los tweets



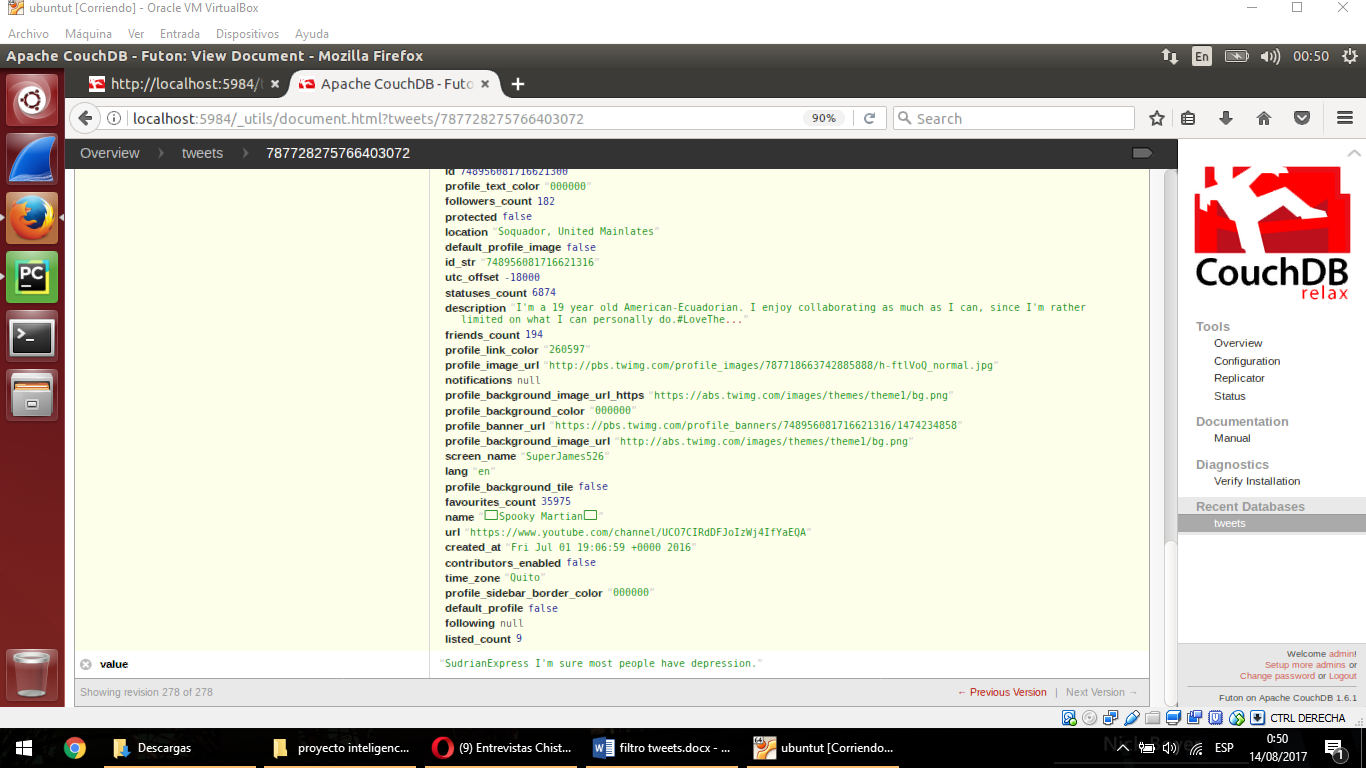
Ejecución del filtrado previo al análisis de tweets

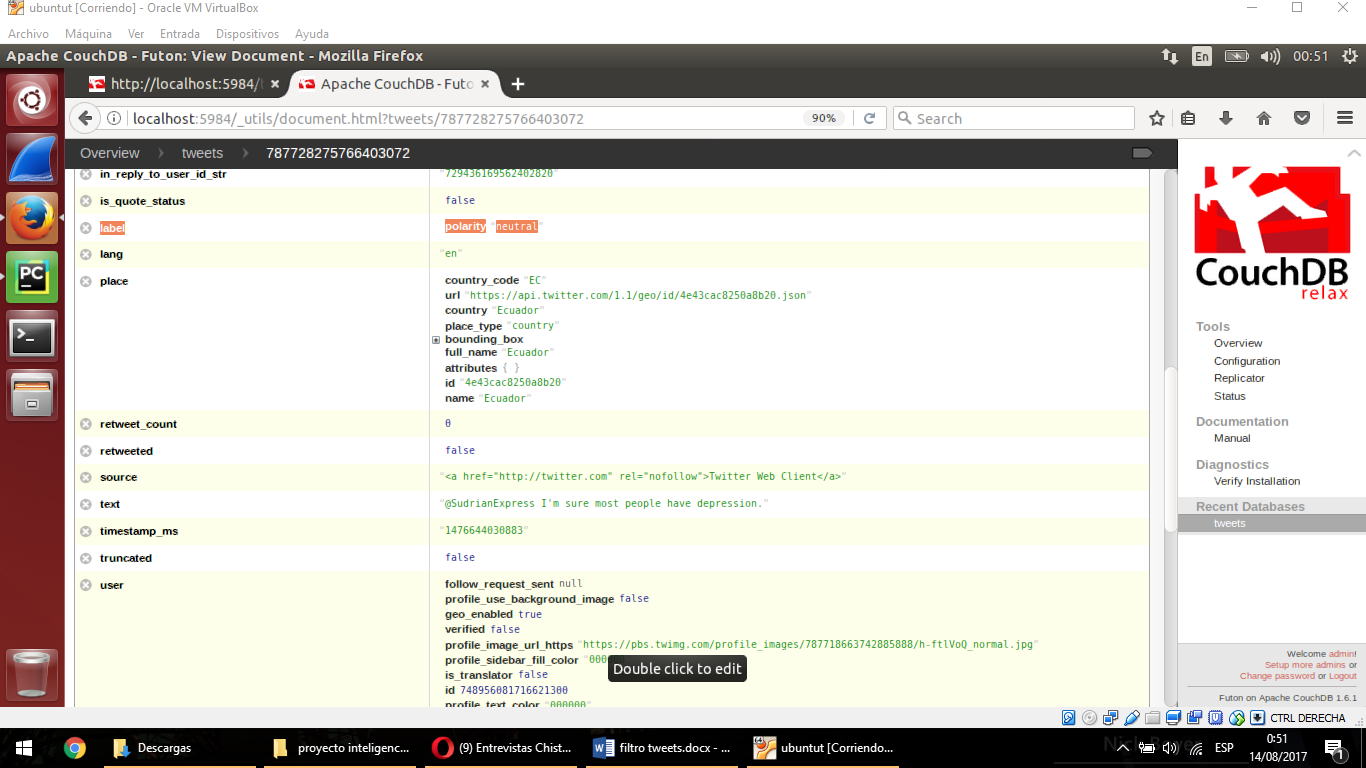


Procedemos a ejecutar el script



Se observa el tweet filtrado y la polaridad del tweet





Enlazamos los datos al Elastisearch con el siguiente comando  
sudo curl -X PUT '[127.0.0.1:9200/\_river/tweetsbi/\_meta](https://l.facebook.com/l.php?u=http%3A%2F%2F127.0.0.1%3A9200%2F_river%2Ftweetsbi%2F_meta&h=ATNXS3lKHpheD8K4R6WWx2zrhJ2TypiDusJFem9fB338DUQEdO31FX95A2eWpEZTXnk7dVVAIjtuNKxCoR3fdNWDBBZZ6eO-6EK9k22XjkKuOnYM5nk7pePMXGGOy7vPwKSN2h9Yt85K)' -d '{ "type" : "couchdb", "couchdb" : { "host" : "localhost", "port" : 5984, "db" : "tweets", "filter" : null }, "index" : { "index" : "tweets", "type" : "testdb", "bulk\_size" : "100", "bulk\_timeout" : "10ms" } }'

