Ejercicio 2: Python y NLTK. POS tagging

Objetivo: Uso de NLTK y Python. Procesamiento de corpus, POS tagging.

Referencias:

http://www.nltk.org/

http://www.python.org/

1. Procesamiento del corpus **cess_esp** anotado con información morfosintáctica.
    - Descargar corpus usando NLTK.
    - Procesar el corpus para transformar la anotación de las etiquetas originales (289 etiquetas) a un conjunto reducido (66 etiquetas). Para realizar esta transformación utilizar los siguientes criterios: todas las etiquetas serán de longitud igual a 2 por defecto, salvo los verbos (v) y los signos de puntuación (F) que pueden ser de tres. También pueden existir etiquetas de longitud =1. En el conjunto transformado también se deben eliminar anotaciones de la forma: `(u'*0*', u'sn')`.
    - Nota: para entender el significado de las etiquetas se puede consultar el siguiente enlace: **https://freeling-user-manual.readthedocs.io/en/latest/tagsets/**
    - Dividir el corpus en dos partes: training (el 90% de las primeras frases) y de test (el 10% restante)

Ejemplo de transformación

ORIGINAL

```
[(u'*0*', u'sn.e-SUJ'), (u'Era', u'vsii3s0'), (u'el',
u'da0ms0'), (u'sustituto', u'ncms000'), (u'natural', u'aq0cs0'),
(u'de', u'sps00'), (u'Redondo', u'np0000p'), (u',', u'Fc'),
(u'pero', u'cc'), (u'las', u'da0fp0'), (u'discrepancias',
u'ncfp000'), (u'acabaron', u'vmis3p0'), (u'con', u'sps00'),
(u'su', u'dp3cs0'), (u'uni\xf3n', u'ncfs000'), (u'-', u'Fg'),
(u'.', u'Fp')]
```

TRANSFORMADO

```
[(u'Era', u'vsi'), (u'el', u'da'), (u'sustituto', u'nc'),
(u'natural', u'aq'), (u'de', u'sp'), (u'Redondo', u'np'), (u',',
u'Fc'), (u'pero', u'cc'), (u'las', u'da'), (u'discrepancias',
u'nc'), (u'acabaron', u'vmi'), (u'con', u'sp'), (u'su', u'dp'),
(u'uni\xf3n', u'nc'), (u'-', u'Fg'), (u'.', u'Fp')]
```

2. Uso de etiquetadores morfosintácticos (usar los modelos **hmm** y **tnt**).
    - Saber entrenar el etiquetador con la partición de entrenamiento previamente transformada
    - Saber etiquetar un conjunto de test con el modelo aprendido
    - Evaluar las prestaciones de un etiquetador
3. Hacer una evaluación de las prestaciones de etiquetado usando todo el corpus (*10-fold cross validation*). Se propone hacer las 10 particiones usando el corpus reducido en el orden original y barajándolo (sugerencia: se puede usar el método *shuffle* importándolo del módulo *random* "*from random import shuffle*"). Comprobar si al barajar el corpus se observan diferencias en los resultados de cada partición.
4. Se debe entregar en la tarea de poliformat el código en Python de la práctica y la tabla y/o gráfico con los resultados.