Escuela Técnica Superior de Ingeniería Informática

Inteligencia Artificial Avanzada

Proyecto: Clasificación de Textos en Lenguaje Natural

Objetivo: Construir un sistema para clasificar la letra de canciones como rock o rap.

Contenido:

Parte 1 Corpus de trabajo

Se te proporcionará un corpus con las letras de canciones y el tipo (rock o rap).

El tipo de la canción se identificará mediante las etiquetas __label__rock__ y__label__rap__

Cada canción estará escrita en un línea individual, los saltos de línea se representarán mediante EOL

Ejemplo:

__label__rock__ Mi casa EOL Mi perro

__label__rap__ Mi amiga EOL Mi gato

Parte 2 Preprocesado, tokenización, creación del vocabulario y conjunto de entrenamiento

Debes preprocesar, si lo consideras necesario, el corpus proporcionado además de tokenizar, crear el vocabulario y el conjunto de entrenamiento.

Algunas tareas de preprocesado típico en clasificación son:

- Pasar a minúsculas.
- Eliminación de signos de puntuación.
- Eliminación de números
- Eliminación de palabras reservadas (stopwords).

- Eliminación de emojis y emoticonos o su conversión a palabras.
- Eliminación de URLs, etiquetas HTML, hashtags.
- Corrección ortográfica.
- Truncamiento: Reducir una palabra a su raíz (grito, grita, gritos, gritas ->grit).
- Lematización: Reducir una palabra a su forma canónica (dije,diré,dijéramos->decir).

Puedes utilizar cualquier librería para las tareas anteriores.

Parte 3 Creación de los modelos del Lenguaje

Debes crear clasificadores utilizando los modelos vistos en clase: n-gramas, redes neuronales PAD, redes neuronales recurrentes y transformers.

Parte 4 Clasificación

En esta parte se clasificarán las canciones como rock o rap.

Escribe un programa que reciba un corpus en el formato dado en Parte 1 y devuelva los resultados de la clasificación de cada canción en un **fichero csv con el siguiente formato**:

num_cancion,ngramas,pad,recurrente,transformer ← Cabecera

1,K,P,P,K

2,K,K,P,P ← 2,K,K,P,P-> Canción 2, ngramas y pad clasifican como rock, recurrente y transformer como rap.

3,K,K,K,R

…

El nombre del fichero de salida será **aluNUM.csv** con NUM tu número de alu.

Notas:

Las canciones clasificadas **deben estar en el mismo orden** de entrada.

Recuerda que cualquier preprocesamiento que hagas sobre los corpus de entrada para entrenamiento lo debes hacer también para testeo (incluyendo el corpus del profesor)

Se penalizará con un 20% de la evaluación del proyecto no ajustarse al nombre del fichero o al formato pedido.

Evaluación del Proyecto

- **Informe con la implementación (4/10)**
  - Detalle del Preprocesamiento.
  - Estimación del error en tu programa. Usa una parte del corpus (80%) como conjunto de entrenamiento y el resto (un 20%) de testeo
  - Dos cuadernos **ejecutables** en Google Colab con:
    - La implementación en código fuente de la clasificación para todos los modelos incluyendo el aprendizaje
    - La implementación de la clasificación para todos los modelo cargando las redes ya entrenadas por el cuaderno anterior.

- **Rendimiento del programa sobre el corpus que proporcionará el profesor (6/10)**
  - Para cada uno de los 4 clasificadores:
    - 97-100% del porcentaje de acierto del mejor clasificador: 1.5 puntos
    - 95-97% del porcentaje de acierto del mejor clasificador: 1.3 puntos
    - 90-95% del porcentaje de acierto del mejor clasificador: 1.1 puntos
    - 87-90% del porcentaje de acierto del mejor clasificador: 1.0 puntos
    - 83-87% del porcentaje de acierto del mejor clasificador 0.8: puntos
    - 75-83% del porcentaje de acierto del mejor clasificador 0.4: puntos
    - Menos del 75% del porcentaje de acierto del mejor clasificador 0.2 puntos

Fecha límite: 9 de Mayo (en ese día deberás subir el informe y ejecutar el programa con el corpus del profesor)