Escuela Superior
de Ingeniería y Tecnología
Universidad de La Laguna

Inteligencia Artificial Avanzada

Proyecto

Daniel Hernández de León
(alu0101331720@ull.edu.es)

1. Preprocesamiento.

Para el preprocesamiento los mejores valores utilizados han sido.
Quitar números, quitar palabras de más de 20 carácteres, quitar signos de puntuación, no quitar stopwords, los emojis ni mejora ni empeora, quitar url html y hashtags, utilizar truncamiento y no lematización y correción ortográfica.

2. Librerías utilizadas.

Utilizo NLTK para el truncamiento y la lematización con PorterStemmer y WordNetLemmatizer. https://www.nltk.org/
SymspellPy para la corrección ortográfica. https://pypi.org/project/symspellpy/
Pandas para la lectura de los ficheros excel. https://pandas.pydata.org/
Y emoji para poder pasar a palabras los emojis. https://pypi.org/project/emoji/

3. Implementación.

Se han creados 3 ficheros en python, uno para la creación del vocabulario con una clase Vocabulary que tokeniza todo el input y genera un fichero vocabulario.txt y un json de configuración de los parámetros de preprocesado utilizados.
Otro para la creación de los modelos de lenguajes positivos y negativos que utiliza el vocabulario generado anteriormente y los parámetros con las probabilidades logarítmicas de cada token y su frecuencia.
Por último un clasificador que dado un input de testeo lo preprocesa y luego lo clasifica como positivo o negativo según las probabilidades de sus tokens y de la clase positivo o negativo.

4. Error.

He probado con bastantes opciones de configuración y el mejor porcentaje de acierto que he logrado ha sido de 65%.
En el fichero report.md están todas las configuraciones probadas con el conjunto de testo y sus porcentajes de aciertos.