

Desarrollo de la Solución

1. Adquisición de datos

Podemos utilizar datos de alguna red social como Tweeter donde los usuarios dan opiniones sobre diferentes cuestiones y productos y usar estos datos para entrenar al modelo. En este caso se puede usar la librería Tweepy de Python (<https://www.tweepy.org/>) para acceder al API de Twitter y recopilar tweets. Estos datos los podemos guardas en un documento csv.

2. Etiquetado de los datos

Cada uno de los textos tendría que ser etiquetado y relacionado con los distintos sentimientos.

3. Pre-Procesamiento de los datos

Para que los datos, en el caso los textos, puedan ser utilizados por algoritmo sería necesario hacer algunas transformaciones, entre ellas:

- Convertir el texto a letras minúsculas
- Eliminar símbolos extraños (URLs, números, emojis...)
- Eliminar signos de puntuación
- Eliminar palabras sin sentido semántico (stopwords)
- Tokenización o división del texto en palabras o frases (se puede considerar como unidad una palabra o una frase)
- Lematización o relacionar una palabra flexionada o derivada con su *forma canónica* o *lema*. Y un lema no es otra cosa que la forma que tienen las palabras cuando las buscas en el diccionario
- Stemming (conversión de las palabras en su raíz)

Para llevar a cabo esta tarea podemos utilizar algunas de las librerías de python como [NLTK](#) o [spaCy](#).

4. Entrenamiento del modelo con el algoritmo elegido

El algoritmo analizará los datos e identificará patrones con los que poder aprender.

5. Validación del algoritmo

El modelo aplicará lo aprendido con los datos etiquetados para la clasificación de nuevos datos.

6. Evaluación del modelo

Finalmente tendremos que llevar a cabo el análisis de los errores de nuestro modelo que nos ayudará a entender cómo podemos mejorarlo, algunas de las opciones serían:

- usar un modelo más complejo
- usar un modelo más simple
- darnos cuenta de que necesitamos más datos y / o más características

En [kaggle](#) se pueden encontrar diferentes datasets para entrenar y validar nuestro modelo