Procesamiento del lenguaje natural en tweets

Para la realización de este ejercicio, partimos de un dataset que encontré en internet de tweets clasificados por el tema de cada tweet.

(fuente: https://huggingface.co/datasets/cardiffnlp/tweet-topic-multi).

Este dataset contiene más de 5000 tweets para entrenamiento y otros 5000 para test. He tenido que realizar algo de preprocesamiento ya que los tweets están en inglés y a la hora de hacer pruebas quería hacerlas en español, pero no ha supuesto problema gracias a la librería deep_translator.

Además he tenido que borrar de cada tweet las menciones y los hashtags, así como los enlaces.

Importación y preprocesamiento de los datos:

```
X train =
pd.read_csv('C:/Users/Pablo/Desktop/IA-BD/MIA/Twitter/Twitter/datasets/x_train.csv')
pd.read_csv('C:/Users/Pablo/Desktop/IA-BD/MIA/Twitter/Twitter/datasets/y_train.csv')
pd.read_csv('C:/Users/Pablo/Desktop/IA-BD/MIA/Twitter/Twitter/datasets/x_test.csv')
pd.read csv('C:/Users/Pablo/Desktop/IA-BD/MIA/Twitter/Twitter/datasets/y test.csv')
X_train['text'] = X_train['text'].apply(lambda x: re.sub("@[A-Za-z0-9_]+","", x))
X_{\text{train}['\text{text'}]} = X_{\text{train}['\text{text'}]}.apply(lambda x: re.sub(r"\s*{.*}\s*", " ", x))
X_train['text'] = X_train['text'].apply(lambda x: re.sub("#[A-Za-z0-9_]+","", x))
X_train['text'] = X_train['text'].apply(lambda x: re.sub(r'https\S+', '', x))
X_train['text'] = X_train['text'].apply(lambda x: re.sub(r'http\S+', '', x))
```

El siguiente paso ha sido vectorizar los tweets haciendo uso de *TfidVectorizer()*.

Esto lo hacemos para el posterior entrenamiento en el que usaremos una máquina de soporte vectorial.

```
vectorizer = TfidfVectorizer()
vectorizer.fit(X_train['text'])
vectorizer.fit(X_test['text'])
#convertir a valores numéricos TF-IDF los datos de entrenamiento
X train Tfidf = vectorizer.transform(X train['text'])
```

Una vez vectorizados los datos, podemos proceder al entrenamiento. En este caso vamos a usar una máquina de soporte vectorial como ya hemos dicho. El kernel que mejores resultados ha dado y el que he elegido ha sido el kernel linear.

```
SVM = svm.SVC(kernel='linear', gamma=0.0001, random state=42, degree=12) #
SVM.fit(X_train_Tfidf,Y_train['topic']) # Fase de entrenamiento del modelo
X_test_Tfidf = vectorizer.transform(X_test['text']) #convertir a valores
numéricos TF-IDF los datos de test. Fase de uso
resultado = SVM.predict(X test Tfidf) #uso
```

Una vez entrenado el modelo, ya podemos probarlo y medir su rendimiento.

```
from deep translator import GoogleTranslator
def predecir(tweet):
   tweet eng = GoogleTranslator(source='auto', target='en').translate(tweet)
   prediction = SVM.predict(vectorizer.transform([tweet eng]))
   if prediction[0] == 'gaming':
       return (tweet, prediction[0])
   return (tweet, GoogleTranslator(source='auto',
target='es').translate(prediction[0].replace('_',
print(predecir("No sé a qué partido político debería votar"))
print(predecir("El pádel en sin duda mi deporte favorito"))
print(predecir("Este álbum me parece un antes y un después, recomendadísimo"))
print(predecir("Me encanta jugar a la xbox con mis amigos"))
```

En el código de arriba, vemos cómo traducimos los tweets de prueba, ya que el modelo está entrenado con tweets en inglés, y luego traducimos de vuelta el resultado. A continuación las predicciones de nuestro modelo:

```
('No sé a qué partido político debería votar', 'noticias y preocupación social')
('El pádel en sin duda mi deporte favorito', 'Deportes')
('Este álbum me parece un antes y un después, recomendadísimo', 'música')
('Me encanta jugar a la xbox con mis amigos', 'gaming')
```

En cuanto a rendimiento, este modelo tiene una tasa de aciertos de un 47%. Aquí abajo dejo las métricas detalladas y la Matriz de Confusión:

classification_report

	precision	recall	f1-score	support	
arts_&_culture	0.53	0.20	0.29	267	
business_&_entrepreneurs	0.50	0.14	0.22	258	
celebrity_&_pop_culture	0.36	0.48	0.41	754	
diaries & daily life	0.34	0.34	0.34	574	
family	0.57	0.06	0.11	64	
fashion_&_style	0.59	0.31	0.41	74	
film_tv_&_video	0.41	0.31	0.35	474	
fitness_&_health	0.66	0.12	0.21	185	
food_&_dining	0.57	0.13	0.21	61	
gaming	0.70	0.09	0.16	158	
learning_&_educational	1.00	0.01	0.03	69	
music	0.59	0.39	0.47	363	
news_&_social_concern	0.45	0.76	0.57	1078	
other_hobbies	0.00	0.00	0.00	72	
relationships	0.00	0.00	0.00	23	
science_&_technology	0.00	0.00	0.00	55	
sports	0.60	0.75	0.67	991	
travel_&_adventure	0.00	0.00	0.00	13	
youth_&_student_life	0.00	0.00	0.00	2	
accuracy			0.47	5535	
macro avg	0.41	0.22	0.23	5535	
weighted avg	0.48	0.47	0.43	5535	

Matriz de confusión

Matriz de confusión																						
	arts_&_culture -	54	1	56	35	1	0	28	0	0	0	0	6	69	0	0	0	17	0	0		- 800
b	usiness_&_entrepreneurs -	1	37	27	31	0	6	6	1	0	0	0	2	113	0	0	0	34	0	0		
	celebrity_&_pop_culture -	4	0	365	55	1	2	51	0	0	1	0	38	142	0	0	0	95	0	0		- 700
	diaries_&_daily_life -	15	2	121	196	1	6	24	1	3	2	0	7	142	0	0	0	54	0	0		
	family -	0	1	11	22	4	0	1	0	1	0	0	2	19	0	0	0	3	0	0		
	fashion_&_style -	1	2	13	5	0	23	3	3	0	0	0	0	15	0	0	0	9	0	0		- 600
	film_tv_&_video -	2	10	93	35	0	0	148	1	1	1	0	34	75	0	0	0	74	0	0		
	fitness_&_health -	1	0	9	24	0	0	3	23	0	0	0	1	106	0	0	0	18	0	0		- 500
	food_&_dining -	0	0	12	12	0	0	3	0	8	0	0	1	19	0	0	0	6	0	0		
True label	gaming -	0	1	21	19	0	0	10	0	0	14	0	2	28	0	0	0	63	0	0		- 400
T	learning_&_educational -	2	0	5	12	0	0	2	1	0	0	1	0	44	0	0	0	2	0	0		
	music -	2	1	104	18	0	1	21	0	0	0	0	141	54	0	0	0	21	0	0		
	news_&_social_concern -	5	9	73	55	0	0	24	5	1	1	0	5	819	0	0	0	81	0	0		- 300
	other_hobbies -	12	6	14	10	0	0	4	0	0	1	0	0	16	0	0	0	9	0	0		
	relationships -	1	0	11	4	0	0	1	0	0	0	0	0	3	0	0	0	3	0	0		- 200
	science_&_technology -	0	2	12	5	0	0	1	0	0	0	0	0	32	0	0	0	3	0	0		
	sports -	2	2	78	34	0	1	32	0	0	0	0	1	97	0	0	0	744	0	0		- 100
	travel_&_adventure -	0	0	3	2	0	0	0	0	0	0	0	0	6	0	0	0	2	0	0		
	youth_&_student_life -	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0		
	L	distress distress	, trepre	Treut's diag	Hes & daily	te family	ashion & styl	e hide	ness & heal	nood & dining	garring	educa	Harry Hear	S. Social co	ncern hobbies	elationships	techno	good got b	uel & advert	ith & Student	_ Lifte	⊥₀
	ર્જ	 Jejness	, delept	ital py	iles /	K	85. 8	itt. et	ue.	60,	le arni	nd &	Rent	₩, O	9.	gier	ie,	40	16, 10,	in,		
		dr.								Pre	edicted lal	bel										