Análisis de sentimientos con tweets

Carolina Acosta,
Edgar Bazo
& Elena Villalobos.

ITAM MCD

Objetivo

El objetivo de este proyecto es construir un clasificador que aprenda a distinguir entre tweets serios o negativos y positivos.

- Conjunto de datos*:
  - 1.6 millones de observaciones
  - Variables:
    - Etiqueta positiva o negativa
    - Texto del tweet

*Obtenido de H20, 2009

ITAM MCD

Trabajos relacionados

73%

NLTK & Naive Bayes

Clasificación de sentimientos positivos y negativos

- Corpus de reviews de películas de NLTK.
- 2,000 datos (75 - 25).
- Poca manipulación de los datos.
- Clasificador de Naive Bayes de NLTK

80%

NLTK & Naive Bayes

Clasificación de sentimientos positivos y negativos

- Propio dataset
- 1,200 datos de entrenamiento
- Eliminan palabras de 2 o menos caracteres.
- No hashtags, no menciones y no emojis, todo en minúscula.
- Clasificador de Naive Bayes de NLTK

NA

H20 Gradient Boosting

Clasificación de sentimientos positivos y negativos, y utilizar el clasificador para rankeo basado en un porcentaje de severidad.

- 1.6 M datos
- No emojis.
- TF-IDF
- H20 Gradient Boosting

ITAM MCD

Solución

Nuestros Datos

- 1.6 millones de datos (sin emojis).

Limpieza:

- ➔ Convertir a minúsculas.
- ➔ Quitar caracteres codificados en html (&, " < >).
- ➔ Quitamos URLs, menciones de usuarios y RT.
- ➔ Quitamos # y mantuvimos la palabra.
- ➔ Quitamos espacios o puntos extras.
- ➔ Quitamos caracteres especiales (’"?!,.():;-).
- ➔ Quitamos letras repetidas (sooooo haaappy).
- ➔ Quitamos caracteres ascii y desciframos a utf-8.
- ➔ Separamos abreviaciones.
- ➔ Quitamos stopwords.
- ➔ Hicimos *stemming* y lematización.
- ➔ Modificamos etiquetas (0 = negativo, 1 = positivo; muestra balanceada).

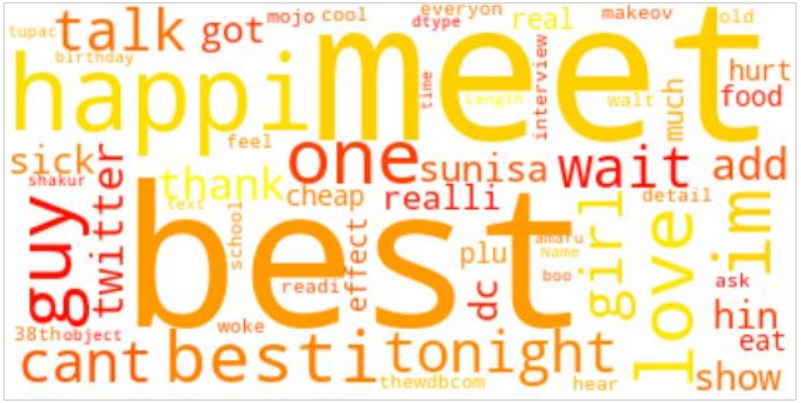
ITAM MCD

Ejemplo de limpieza:

Target	Texto	Texto limpio
0	@switchfoot http://twitpic.com/2y1zl - awww, that's a bummer. you shoulda got david carr of third day to do it. ;d	aww bummer shoulda got david carr third day
1	@RunningGolfer Glad you picked it up...she didn't	glad pick upsh
1	@i140 I'm excited I made it on your list. Thnx, Jason.	excit made list thnx jason
0	@johncmayer Where is that Belgian concert you were talking about? I can't even find it on google	belgian concert talk even find googl
1	sitting on a field with Emma and Miri watching Sarah and Naomi running around searchinf for elves	sit field emma miri watch sarah naomi run around searchinf elv
0	Couldn't decide if I wanted to go to Family Fortunes on Sunday but train is Â£45 now Its the Christmas special being filmed on that day!!	could decid want go famili fortun sunday train 45 christma special film day

Nubes de palabras

Etiqueta negativa

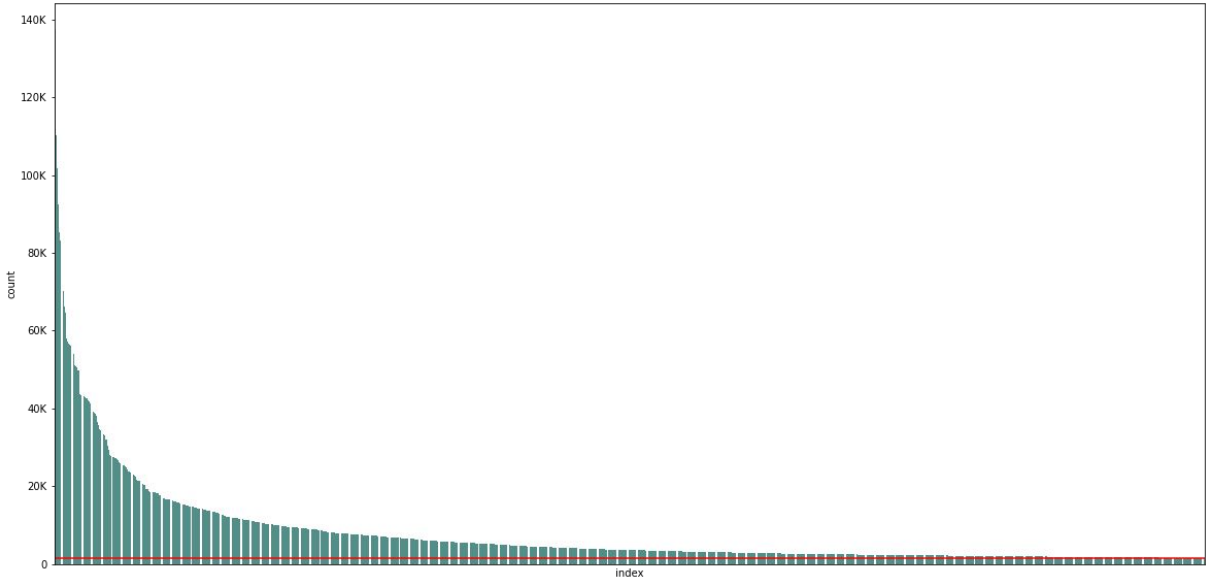


Etiqueta positiva

ITAM MCD

Tokenizer

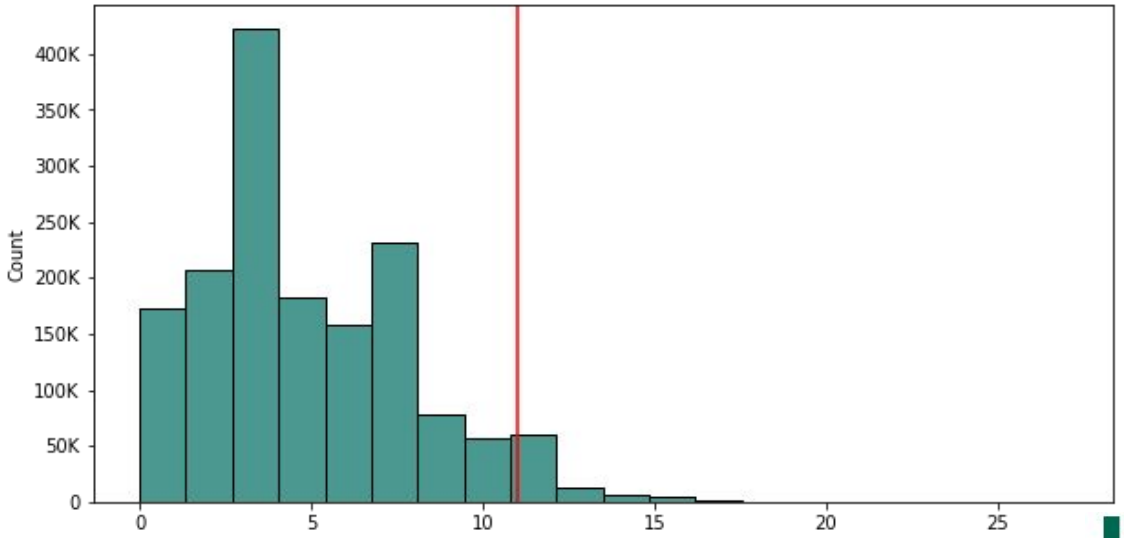
- ➔ 330,671 palabras en total.
- ➔ Número de palabras en nuestro vocabulario = 1,000.



ITAM MCD

Tokenizer

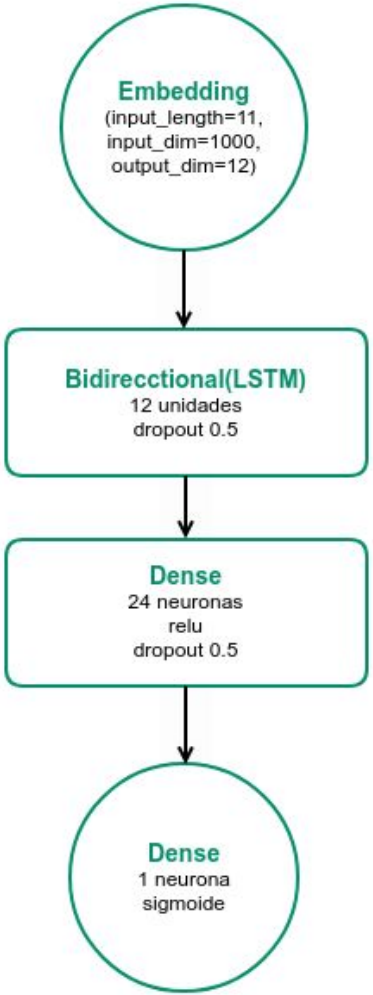
- ➔ 27 es el máx de número de tokens por secuencia.
- ➔ Para el ejercicio definimos la longitud de tokens donde estuviera el 95% de nuestros datos, y definimos que 11 sería nuestro máximo de longitud por secuencia.



ITAM MCD

Tokenizer

- ➔ Padding = ‘post’, dio mejores resultados.
- ➔ Después de la limpieza, datos balanceados utilizados 1,591,506 data points.
  - ◆ 70% de entrenamiento.
    - 20% de validación.
  - ◆ 30% de prueba.



Red del mejor modelo

- ➔ **Embedding**: Capa que asigna una representación numérica a las palabras.
- ➔ **LSTM bidireccional**: RNN que permite retroalimentación entre neuronas.
- ➔ Función ReLU para oculta.
- ➔ Función sigmoide para salida.

Hiper-parámetros:

Configuración del modelo:

- ➔ Optimizador: RMSprop.
- ➔ Función de pérdida: BinaryCrossEntropy.
- ➔ Métricas: Binary accuracy.

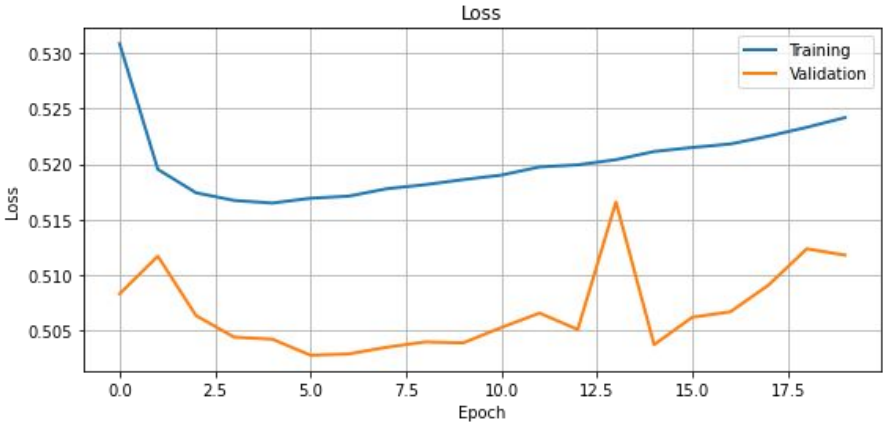
Configuración para entrenamiento:

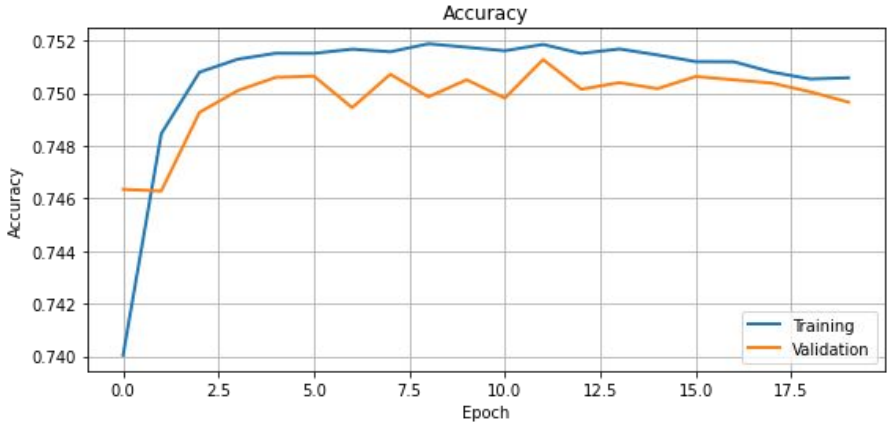
- ➔ Épocas: 20.
- ➔ Tamaño de lote: 64.
- ➔ 20% validación.

Resultados

Resultados obtenidos:

	Entrenamiento	Validación	Prueba
Precisión	75.09%	74.97%	75.18%
Pérdida	0.5236	0.5118	0.5102





Ejemplo*:

Etiqueta real	Etiqueta predicha	Texto	Texto limpio
Positive	Positive	how can you not love Obama? he makes jokes about himself.	love obama make joke
Negative	Negative	cant sleep... my tooth is aching.	cant sleep tooth ach
Negative	**Positive**	**I just created my first LaTeX file from scratch. That didn't work out very well. (See @amandabittner , it's a great time waster)**	**creat first latex file scratch work well see great time waste**

*Otro conjunto de prueba con 359 observaciones, del cual obtuvimos 76.6% de precisión

ITAM MCD

Comparaciones con otros modelos probados*

	Modelo A	Modelo B	Mejor modelo
Tokenizer	max_words = 1,000 max_len = **21** padding='post'	max_words = **10,000** max_len = 13 padding='post'	max_words = 1,000 max_len = **11** padding='post'
Red	Embedding(input_length=max_len, input_dim=max_words, output_dim=12) LSTM(units=12, dropout=0.5) Dense(units=24, activation='relu', dropout=0.5) Dense(units=12, activation='relu', dropout=0.5) Dense(units=1, activation='sigmoid')	Embedding(input_length=max_len, input_dim=max_words, output_dim=12) LSTM(units=**33**, dropout=0.5) **Dense(units=24, activation='relu', dropout=0.5)** Dense(units=8, activation='relu', dropout=0.5) Dense(units=1, activation='sigmoid')	Embedding(input_length=max_len, input_dim=max_words, output_dim=12) **Bidirectional**(LSTM(units=12, dropout=0.5)) Dense(units=24, activation='relu', dropout=0.5) Dense(units=1, activation='sigmoid')
Compile	BinaryCrossentropy(from_logits=True) optimizer = **adadelta** epochs=20, batch_size=16	BinaryCrossentropy(from_logits=True) optimizer = rmsprop epochs=15, batch_size=24	BinaryCrossentropy(from_logits=True) optimizer = rmsprop epochs=20, batch_size=64.
Resultados validación	loss: 0.69 binary_accuracy: 0.50 val_loss: 0.69 val_binary_accuracy: 0.50	loss: 0.47 binary_accuracy: 0.79 val_loss: 0.55 val_binary_accuracy: 0.72	loss: 0.47 binary_accuracy: 0.77 val_loss: 0.55 val_binary_accuracy: 0.0.72
Resultados de prueba	loss: 0.69 binary_accuracy: 0.51	loss: 0.55 binary_accuracy: 0.72	loss: 0.57 binary_accuracy: 0.73

*Se probaron ~30 modelos

Otras lecciones aprendidas...

- ➔ Uso de etiquetas [0,4] ó [0,1].
- ➔ Tamaño de datos:
  - ◆ Aumentó en 5%, el uso de los 1.6 millones de *tweets*.
- ➔ Vocabulario reducido.
- ➔ Dropouts.
  - ◆ Técnica de regularización.
  - ◆ Colocar más genera que la precisión de la validación vaya arriba del entrenamiento.

Conclusiones

Análisis de desempeño obtenido

- ➔ El uso de LSTM y pocas capas ocultas genera “buenos” para la naturaleza de nuestro problema.
- ➔ Limpieza adecuada del texto.

Problemas encontrados

- ➔ Difícil alcanzar una limpieza *ideal* del texto.
- ➔ Características de los datos parecieron no permitir mejores métricas.

Logros

- ➔ Se obtuvo un modelo relativamente “bueno” en los diferentes conjuntos de datos (75%). Un buen aprox. a lo que califica el humano (80%).
- ➔ Aprendizaje del comportamiento de modelos de redes neuronales que evalúan lenguaje.

Futuras investigaciones

- ➔ Agregar una categoría de clasificación, como neutral.
- ➔ Hacer el mismo análisis en el idioma español y con cierto *hashtag*.

¡Gracias por su atención!

Para mayor información en:

https://github.com/ElenaVillano/sentiment_analysis_tweets