Universidad del Valle de Guatemala
Data Science - Sección 20 - Luis Furlán
Laura Tamath Carné: 19365
Martín España Carné: 19258
Sofia Rueda Carné: 19099

Laboratorio #5

Análisis de sentimientos

1. Describa de forma detallada las actividades de preprocesamiento que llevó a cabo.

- **Convertir el texto a mayúsculas o a minúsculas.** Se convierte el texto a minúsculas utilizando .lower().
- **Quitar los caracteres especiales que aparecen como "#","@" o los apóstrofes.** Se hizo un replace en las palabras que incluyen los caracteres # y @ se reemplazan por un espacio vacío.
- **Quitar las URL.** Se realizó una función la cual se usa *re.findall()* la cual devuelve todas las coincidencias no superpuestas en una cadena, la cual se escanea de izquierda a derecha y se van devolviendo en el orden que van siendo encontradas. Dentro de este se encuentra `r"http[^\s]*"` lo cual indica que:
  - r: Se utiliza este prefijo para que cuando se utilice "\" se tome como un carácter real.
  - "http: busca en todo el archivo en dónde se encuentra http
  - [^\s]": [] indica cualquiera de los caracteres especificados dentro de este, en donde ^ indica una negación y \s indica respectivamente cualquier carácter, espacio en blanco y cualquier carácter a excepción del espacio en blanco.
  - *: indica que la repetición de un carácter una o más veces

  Por último se realiza un for para ejecutarlo en todo donde se encuentre un link, y este es reemplazado por un espacio en blanco.
- **Revisar si hay emoticones y quitarlos.** Se realizó una función la cual se utiliza *.join* el cual forma una cadena de caracteres con los elementos de una lista, estos elementos se guardarán en la cadena, separados por lo especificado.
  Se utilizó el paquete *emoji* el cual permite imprimir emojis a través de un programa en python, y al usar *.EMOJI_DATA* se hace referencia a cualquier emoji guardado en la data de esta librería.
  Cuando termina de realizarse todo el join, se reemplaza por un espacio en blanco.
- **Quitar los signos de puntuación** Se utilizó el paquete "string" del cual se hizo uso del método *.punctuation* el cual trae los siguientes caracteres: *!"#$%&'()*+, -./:;<=>?@[\]^_`{|}~* . Y se hace uso del *str.maketrans* que es utilizada para especificar una lista de caracteres específicos que deben reemplazarse en todos los caracteres de la cadena, que en este caso es en los textos que la contengan.
- **Quitar los artículos, preposiciones y conjunciones (stopwords).** Para comenzar se hace *from nltk.corpus import stopwords*, luego stopwords.words('english'), que básicamente trae palabras que se pueden sacrificar sin alterar el significado de la oración. El output trae [u'your', u'yours', u'yourself', u'yourselves', … , u'while', u'of',

u'at']. Y por último se hace un .join en el que si las palabras están incluidas en la variable que tiene guardada las stopwords, las guarda como un espacio en blanco.

- **Quitar números si considera que interferirá en la clasificación (quizá debería valorar si quitar o no el 911).** Se declaró una variable la cual busca en donde hayan números dentro del texto (busca desde el número 0 hasta al número 9). Luego se realiza una condición en donde si existe un número o más entra a un for en el cual verifica que estos números no sea 911 porque de ser así, lo reemplaza por un espacio vacío.

Como para eliminar cada característica especificada en este inciso, se colocaba un espacio en blanco en lugar de los caracteres ya existentes, se tuvo que importar el módulo re, realizar un *re.sub(' ', ' ', text)* el cual básicamente la función sub reemplaza las coincidencias por lo que le especifiques en el segundo argumento, es decir, en donde hay doble espacio, se convierte en un espacio afectando a todo lo que está incluido en text.

2. ¿Qué palabras cree que le servirán para hacer un mejor modelo de clasificación?¿Vale la pena explorar bigramas o trigramas para analizar el contexto?

Negativas		Positivas	
Palabra	Frecuencia	Palabra	Frecuencia
Like	348	fire	180
new	225	disaster	121
news	201	suicide	112
people	200	killed	95
video	166	hiroshima	93

Sí es necesario explorar bigramas, debido a que en ambas se repiten ciertas palabras, por eso es necesario un bigrama o trigrama para así comprender el contexto del porqué está en cada categoría, ya que con solo una palabra no siempre es suficiente.

3. Documente todos los análisis

- **Investigar qué palabra se repite más en cada una de las categorías**

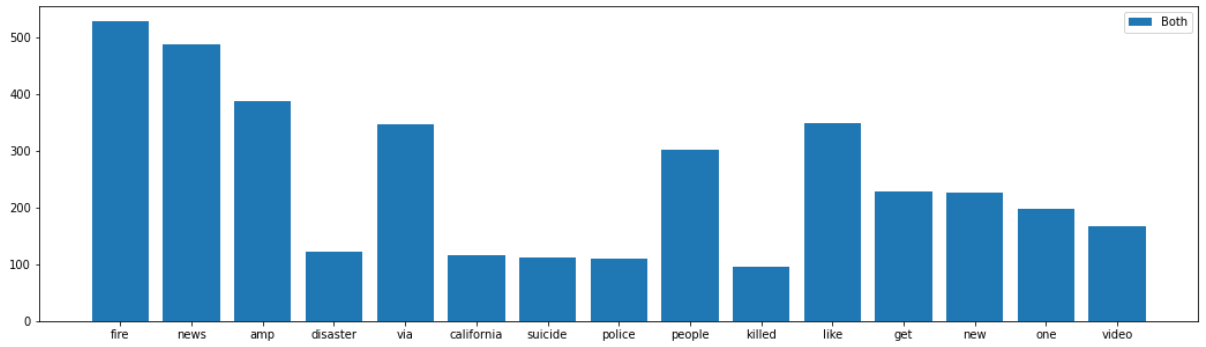
Las palabras que más se repiten en los tweets negativos son “like”, “amp”, “fire”, “get” y “new”. Por otro lado, las palabras que más se repiten en los tweets positivos son “fire”, “news”, “amp”, “disaster”, y “via”.

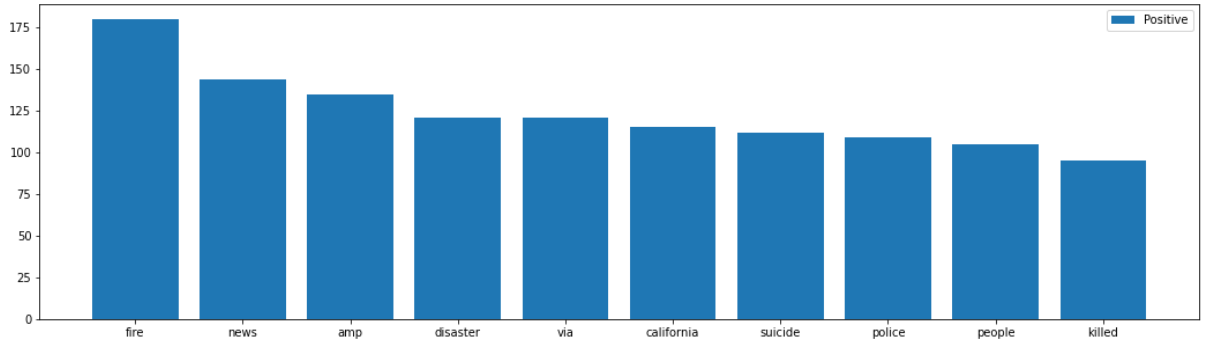
- **Hacer una nube de palabras para visualizar las que aparecen con más frecuencia**

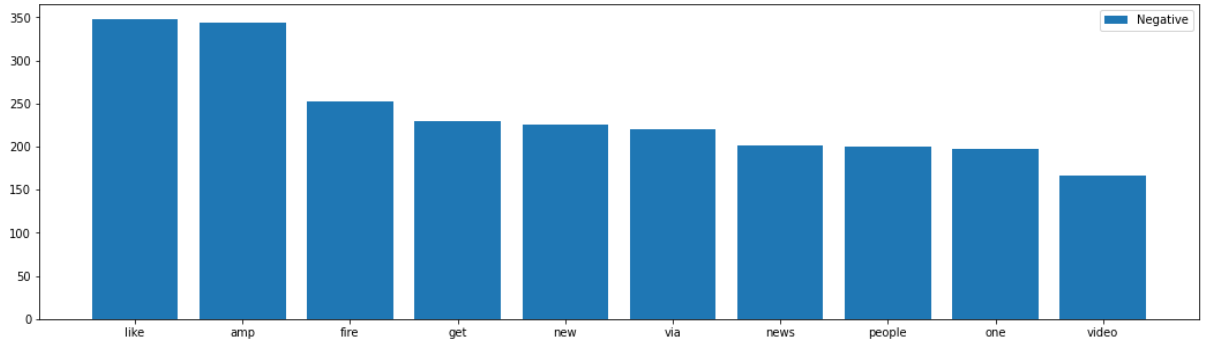
Nube para tweets positivos

Nube para tweets negativos

- **Hacer un histograma con las palabras que más se repiten**







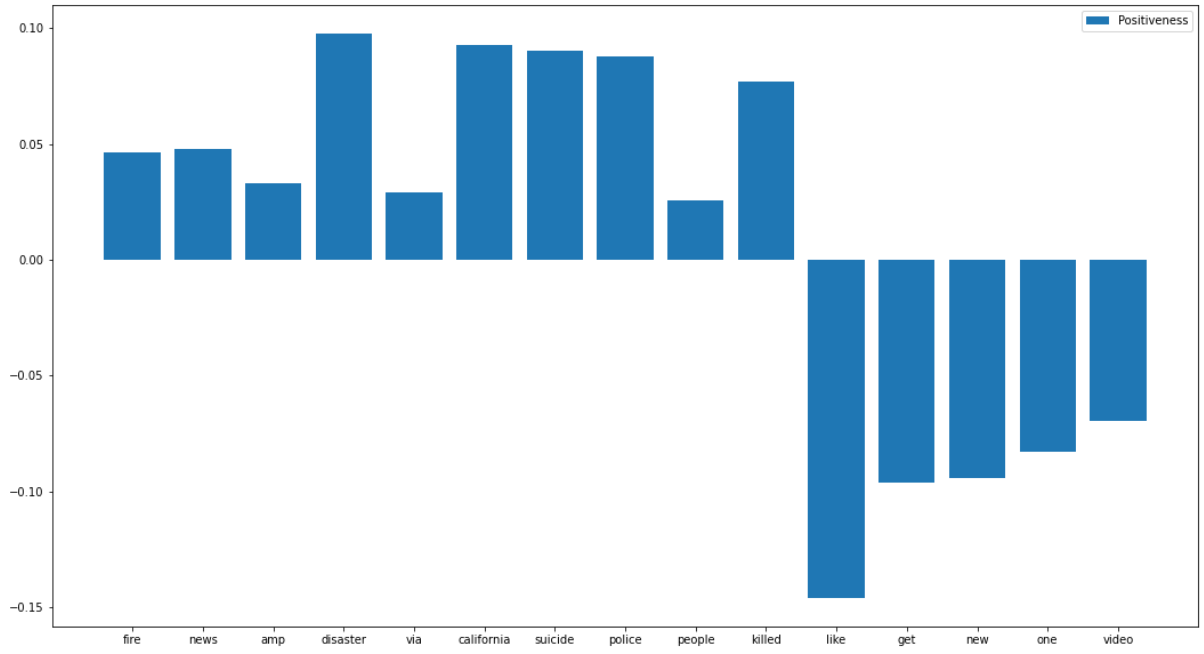
- **Discutir sobre las palabras que tienen presencia en todas las categorías.**

Respecto a las palabras que tienen presencia en todas las categorías, lo más práctico sería tomarlas como palabras neutrales, sin embargo, puede darse el caso de palabras que aparezcan en ambas categorías, pero tengan mayor presencia en una que en la otra. Por ello, la mejor solución podría ser evaluar ambas categorías, y las palabras que tengan en común, determinar su cuenta en ambos casos.

- **Determinar las palabras positivas, negativas o neutras**

Para determinar la positividad de una palabra, se toma la frecuencia de la misma. Si esta en ambas categorías, entonces se cuenta con dos frecuencias. Luego se resta la frecuencia positiva de la negativa y se divide dentro del total de veces que aparece la palabra, ya sea en tweets positivos o negativos. Este número es la proporción respecto a ambas categorías. Sin embargo, puede que no sea una palabra muy frecuente en la categoría ganadora, entonces es

necesario calcular esta proporción y multiplicarla por el número obtenido. Este número por lo tanto es la proporción de veces que aparece en su categoría vs la cantidad de veces que aparece en ambas categorías. Por otro lado, si la palabra solamente se encuentra en una categoría entonces la positividad de la misma es la proporción de la cantidad de veces que aparece sobre esa misma categoría. También se consideró el caso en que la palabra tenga la misma frecuencia en categoría positiva o negativa por lo que la positividad sería 0. El valor de la positividad de cada palabra oscila en el rango entre -1 y 1. Siendo -1 totalmente negativo; 0 totalmente neutral; y 1 totalmente positivo. A continuación una gráfica con las 15 palabras más frecuentes y su respectiva positividad.



4. Determine qué tan positivo, negativo o neutral es el mismo.

Para calcular qué tan positivo, negativo o neutral es un tweet, se observan cada palabra de cada tweet y se revisa el índice de positividad de esa palabra. Si hay varias ocurrencias, se suman estos índices para observar el comportamiento y luego se divide dentro de la cantidad de ocurrencias para que el rango de valores permanezca entre -1 y 1.

	id	keyword	location	text	target	positiveness
7346	10518	wildfire	usa	california battling scariest wildfire far rock...	1	0.000552
7193	10305	weapon	usa	breaking obama officials gave muslim terrorist...	1	0.000240
3674	5229	fatality	enterprise, nv	fatality	0	0.000102
4832	6879	mass murder	usa	fredolsencruise please take faroeislands itine...	0	0.000228
2719	3905	devastate	republica dominicana	losdelsonido obama declares disaster typhoon d...	1	0.000418
2945	4235	drown	melbourne	hundreds feared drowned another mediterranean ...	1	0.000165
5860	8371	ruin	snapchat~ maddzz_babby	family ruin something actually going good	0	0.000155
3835	5458	first responder	usa	firefighters make gains rockyfire jerry brown ...	1	0.000142
5172	7377	obliterate	ondo	someone teaching obedience obliterate trials l...	1	0.000115
6817	9764	trap	shoujo hell	onihimedesu whole city trapped leave city supp...	1	0.000134
7211	10330	weapon	jayankondacholapuram.tamilnadu	day remember nuclear weapon power hiroshima th	1	0.000361
595	860	bioterror	pelham, al	thank fedex longer shipping live microbes depa...	0	0.000105
5894	8416	sandstorm	usa	watch airport get swallowed sandstorm minute	1	0.000310
5694	8126	rescue	unknow	migrants rescued boat capsizes libya	1	0.000215
7292	10433	whirlwind	unknow	past week absolute whirlwind athens bound	1	0.000106

5. Determine

- **¿Cuáles son los 10 tweets más negativos?¿En qué categoría están?**

	id	keyword	location	text	target	positiveness
113	163	aftershock	belgium	aftershock	0	-0.000315
131	190	aftershock	unknow	aftershock	0	-0.000315
820	1191	blizzard	unknow	stats	0	-0.000043
6745	9664	tornado	wherever i'm sent	ticklemeshawn evebrigid bet	0	-0.000043
255	363	annihilation	usa	souls punished withåêannihilation	0	-0.000029
4273	6071	heat wave	unknow	cherry print matching lipstick rediscovered na...	0	-0.000022
5564	7939	rainstorm	usa	nathanfillion hardly	0	-0.000021
2800	4027	disaster	lima, peru	‰û¢i architect disaster‰û¢	0	-0.000019
6306	9009	stretcher	unknow	freeze fruits veggies	0	-0.000019
19	28	nan	unknow	goooooooaaaaaal	0	-0.000014

Los 10 tweets más negativos son los que contienen las palabras claves de aftershock, blizzard, tornado, annihilation, heat wave, rainstorm, disaster y stretcher. Estos tweets tienen un puntaje de positividad entre -0.000315 a -0.000014.

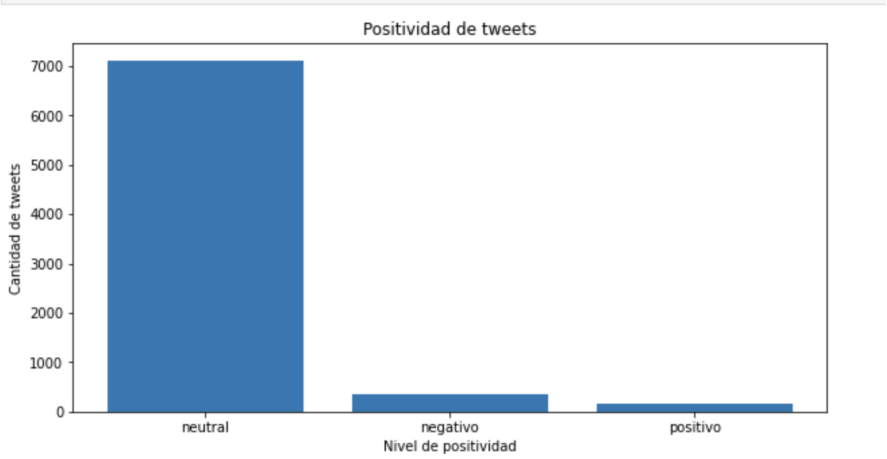
- **¿Cuáles son los 10 tweets más positivos? ¿En qué categoría están?**

	id	keyword	location	text	target	positiveness
247	352	annihilation	wild wild web	annihilating quarterstaff annihilation	1	0.466667
563	814	battle	unknow	young german stormtrooper engaged battle somme ìñ	1	0.463341
227	322	annihilated	uk	oryx symbol arabian peninsula annihilated hunters	1	0.460758
1100	1591	bombed	usa	cyhitheprynce bombed kanye elephantintheroom	1	0.455357
214	302	annihilated	unknow	annihilated abs	1	0.453704
1458	2102	casualty	usa	nowplaying dubstep hardstyle trap messy mix ev...	1	0.452433
969	1402	body bag	paignton	new ladies shoulder tote handbag faux leather ...	0	0.452029
981	1420	body bag	paignton	new ladies shoulder tote handbag faux leather ...	0	0.452029
974	1409	body bag	paignton	new ladies shoulder tote handbag faux leather ...	1	0.452029
234	334	annihilated	unknow	tomcatarts thus explaining annihilated case su...	1	0.451934

Los 10 tweets más positivos son los que contienen las palabras claves de annihilation, battle, annihilated, bombed, casualty y body bag. Estos tweets tienen un puntaje de positividad entre 0.466667 a 0.451934.

6. ¿La inclusión de esta variable mejoró los resultados del modelo de clasificación?

La inclusión de la variable de sentimientos, permitió medir los niveles de positividad por el contenido de cada tweet. Por lo tanto, fue posible clasificar los textos por positivo, negativo y neutral. A partir de eso, el modelo funciona ahora por estas tres categorías y la frecuencia de cada una en la base de datos.



Referencias

- Python - Remove Stopwords: https://www.tutorialspoint.com/python_text_processing/python_remove_stopwords.htm#:~:text=Stopwords%20are%20the%20English%20words,the%2C%20he%2C%20have%20etc.

- Emoji Data Python documentation: https://emoji-data-python.readthedocs.io/en/latest/
- Nikhilaggarwal3 (2022) Python regex: re.search() vs re.findall(). GeeksForGeeks: https://www.geeksforgeeks.org/python-regex-re-search-vs-re-findall/
- Expresiones regulares - split() y sub() - RegEx - Curso de Python desde cero - Capítulo 49: https://www.programacionfacil.org/cursos/python_basico/capitulo_49_expresiones_regulares_split_sub_python.html
- Parzibyte (2020) Python – Contar frecuencia de palabras. Parzibyte: https://parzibyte.me/blog/2020/10/16/python-contar-frecuencia-palabras/#:~:text=Para%20sacar%20la%20frecuencia%20de,un%20arreglo%2C%20separ%C3%A1ndola%20por%20espacios.
- Currie, Ian (2022). Sorting a Python Dictionary: Values, Keys, and More. Real Python: https://realpython.com/sort-python-dictionary/

Link de Repositorio en GitHub

https://github.com/lauratamath/HDT5_Data.git

Link de Documento en Google Drive

https://docs.google.com/document/d/11yTeQHJvkhRdYpRcrVlxuu-_jECOPLYIsJqrNGSJpUE/edit?usp=sharing