Tarea 2
Análisis de sentimiento

Kin Nevárez
26 de mayo de 2022
Maestría en Ciencia de Datos
Facultad de Ciencias Físico Matemáticas

[1]

***Resumen*—El reporte a continuación es una ejemplificación de los tipos de pre procesamiento que se pueden realizar para un correcto y oportuno análisis de texto, el cual, se puede observar en gráficas de frecuencia y estadísticas de palabras más usadas. Posteriormente, se realiza un análisis de sentimiento con tres módulos distintos y se comparan los resultados de dicho análisis.**

Palabras clave: *análisis de sentimiento, lematización, stopwords, pre procesamiento, nltk.*

I. Introducción

El análisis de sentimiento es una técnica analítica que utiliza la estadística, el procesamiento del lenguaje natural y el aprendizaje automático para determinar el significado emocional de las comunicaciones.

Se puede usar en muchos ámbitos como:

- *Negocios:* En el campo del marketing, las empresas lo utilizan para desarrollar sus estrategias, comprender los sentimientos de los clientes hacia los productos o la marca, cómo las personas responden a sus campañas o lanzamientos de productos y por qué los consumidores no compran algunos productos.
- *Política:* En el ámbito político, se utiliza para realizar un seguimiento de la visión política, para detectar consistencia e inconsistencia entre declaraciones y acciones a nivel de gobierno. ¡También se puede usar para predecir los resultados de las elecciones!
- *Acciones públicas:* El análisis de sentimientos también se utiliza para monitorear y analizar los fenómenos sociales, para detectar situaciones potencialmente peligrosas y determinar el estado de ánimo general de la blogósfera.

El objetivo del presente análisis es obtener la reacción de los interlocutores en un debate presidencial por medio de redes sociales, en específico Twitter.

[1]

II. Datos

La base de datos utilizada tiene el nombre "First GOP Debate Twitter Sentiment" y fue obtenida del repositorio de Kaggle. Esta base de datos contiene más de 10,000 tweets sobre el primer debate presidencial de 2016 celebrado en Ohio.

III. Metodología

A continuación, se presenta el planteamiento propuesto para dar solución a la problemática descrita en el apartado anterior.

Es importante destacar que todos los análisis fueron realizados con ayuda del lenguaje de programación Python, en la herramienta de Google Colab.

A. Pre procesamiento

1. Limpieza

Al estar trabajando con tweets, es necesario hacer limpieza de datos muy específica, la cual contempla los siguientes pasos:

- Se convierte todo el texto a minúsculas
- Se eliminan las páginas web (palabras que empiezan por "http")
- Se eliminan los signos de puntuación
- Se eliminan los números
- Se eliminan los espacios en blanco múltiples
- Se tokenizan las palabras
- Se elimnan los tokens con longitud < 2
- Se vuelven a unir las palabras que conforman el comentario

Fig. 1. Ejemplificación limpieza y tokenización

```
--------------------------------------------------------------------------------------------------------------
Texto crudo:
RT @NancyLeeGrahn: How did everyone feel about the Climate Change question last night? Exactly. #GOPDebate
--------------------------------------------------------------------------------------------------------------
Texto limpio:
rt nancyleegrahn how did everyone feel about the climate change question last night exactly gopdebate
```

Fuente: Elaboración propia

Después de aplicar la función a todo el conjunto de datos, se observa de la siguiente manera:

Fig. 2. Ejemplificación limpieza y tokenización - conjunto completo

	name	text	clean_text
0	I_Am_Kenzi	RT @NancyLeeGrahn: How did everyone feel about...	rt nancyleegrahn how did everyone feel about t...
1	PeacefulQuest	RT @ScottWalker: Didn't catch the full #GOPdeb...	rt scottwalker didn catch the full gopdebate l...
2	PussssyCroook	RT @TJMShow: No mention of Tamir Rice and the ...	rt tjmshow no mention of tamir rice and the go...
3	MattFromTexas31	RT @RobGeorge: That Carly Fiorina is trending ...	rt robgeorge that carly fiorina is trending ho...
4	sharonDay5	RT @DanScavino: #GOPDebate w/ @realDonaldTrump...	rt danscavino gopdebate realdonaldtrump delive...

Fuente: Elaboración propia

Fueron eliminados los caracteres especiales, sobre todo los más frecuentes como el "@" y el "#" utilizados para mencionar usuarios y crear tags respectivamente.

2. *Stopwords*

Se eliminan las palabras más frecuentes en el idioma inglés. Éstas no proporcionan información relevante en los textos analizados. A diferencia de otras ocasiones, se usa módulo gensim pues contiene un listado de stopwords más completo que el comúnmente utilizado en nltk.

Aunado a esto, se agregan palabras que en el contexto actual no son relevantes, como "rt" que indica un retweet y está presente en muchos textos, asícomo "gopdebate" que está incluido como etiqueta #GOPDebate.

Fig. 3. Ejemplificación eliminación de stopwords

```
Texto limpio con stopwords:
rt nancyleegrahn how did everyone feel about the climate change question last night exactly gopdebate

Texto limpio sin stopwords:
nancyleegrahn feel climate change question night exactly
```

Fuente: Elaboración propia

3. *Origen de palabras y lematización*

Se hace uso de un diccionario que contiene el tipo de palabra utilizada en la oración: adjetivo, verbo, sustantivo y adverbio para poder obtener la palabra raíz de ésta, por medio de la lematización.

Fig. 4. Ejemplificación etiquetado de tipo de palabra y lematización

```
Texto limpio sin stopwords, sin tag:
nancyleegrahn feel climate change question night exactly

Texto limpio sin stopwords, con tag:
[('nancyleegrahn', 'a'), ('feel', 'n'), ('climate', 'n'), ('change', 'n'), ('question', 'n'), ('night', 'n'), ('exactly', 'r')]

Texto limpio sin stopwords, con tag, sin lematización:
[('going', 'v'), ('msnbc', 'a'), ('live', 'a'), ('thomasaroberts', 'n'), ('pm', 'v'), ('et', 'n')]

Texto limpio sin stopwords, con tag, con lematización:
  go msnbc live thomasaroberts pm et
```

Fuente: Elaboración propia

Al final lo que se observa es el texto limpio sin caracteres especiales, sin términos irrelevantes y con la palabra raíz, lo cual hará mucho más sencillo el análisis de sentimientos posterior.

Fig. 5. Texto después de limpieza

	text	lemma_text
0	RT @NancyLeeGrahn: How did everyone feel about...	nancyleegrahn feel climate change question n...
1	RT @ScottWalker: Didn't catch the full #GOPdeb...	scottwalker catch night scott best line seco...
2	RT @TJMShow: No mention of Tamir Rice and the ...	tjmshow mention tamir rice hold cleveland wow
3	RT @RobGeorge: That Carly Fiorina is trending ...	robgeorge carly fiorina trending hour debate...
4	RT @DanScavino: #GOPDebate w/ @realDonaldTrump...	danscavino realdonaldtrump deliver high rati...

Fuente: Elaboración propia

B. *Análisis de Sentimiento*

1. *Con la librería SentiWordNet*

Para poder usar las librerías, es necesario conocer cómo funciona el análisis de sentimiento. De acuerdo con SentiWordNet, cada término tiene asociados una puntuación positiva, negativa y neutral:

$$Post(t) + Neg(t) + Neu(t) = 1 \qquad (1)$$

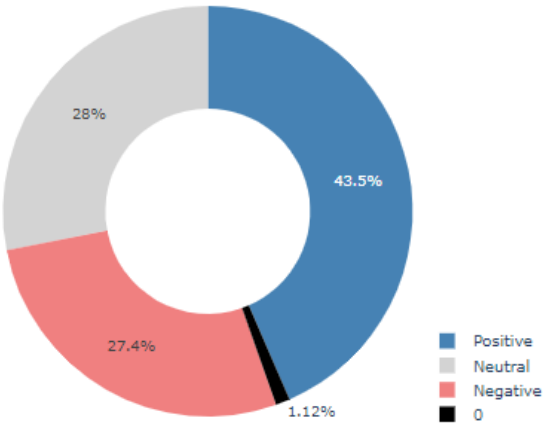
Fig. 6. Ejemplificación Polaridad - SentiWordNet

```
texto:
  nancyleegrahn feel climate change question night exactly
-----------------------------------
Polaridad según SentiWordNet
<precisely.r.01: PosScore=0.125 NegScore=0.0>
```

Fuente: Elaboración propia

De esta forma, el score final, se encuentra en un rango entre [0,1], y se asigna el sentimiento de la puntuación más alta.

Fig. 7. Composición de sentimientos - SentiWordNet



Fuente: Elaboración propia

TABLA I
COMPOSICIÓN DE SENTIMIENTOS - SENTIWORDNET

Sentimiento	Frecuencia	Porcentaje
Positivo	6,029	43.5%
Neutral	3,889	28%
Negativo	3,797	27.4%
Nulo	156	1.12%

Casi el 44% de los comentarios tienen una connotación positiva, segun SentiWordNet, mientras que los sentimientos neutrales o negativos se reparten de manera equitativa en un 28% aproximadamente. SentiWordNet no logró clasificar correctamente el 1.12% de los comentarios. Revisando algunos de los comentarios no clasificados, se observa que después de la limpieza y lematización, la mayoría coincide en que son pocas palabras no legibles, como nombres de usuarios

o hashtagas, o palabras cortas:

Fig. 8. Ejemplificación de comentarios sin clasificar

	text	lemma_text	sentiment_swn
43	RT @brownblaze: PLEASE RT. #KKKorGOP #GOPDebat...	brownblaze kkkorgop	0
143	RT @peddoc63: Go Carly📢\nGo Carly📢\nGo Carly📢\...	peddoc carly📢 carly📢 carly📢 carlyfiorina car...	0
253	RT @JosephKapsch: And then hillaryclinton @bar...	josephkapsch hillaryclinton barackobama joeb...	0
385	At least he was honest about it. #GOPDebate h...	honest	0
411	RT @Squintz1983: #GOPDebate pretty much http:/...	squintz pretty	0

Fuente: Elaboración propia

2. *Con la librería BlobText*

La salida de TextBlob de polaridad es un flotante dentro del rango [-1 , 1] dónde -1.0 es una polaridad negativa y 1.0 es positivo. Esta puntuación también puede ser igual a 0, que representa una evaluación neutral de una declaración, ya que no contiene ninguna palabra del conjunto de entrenamiento.

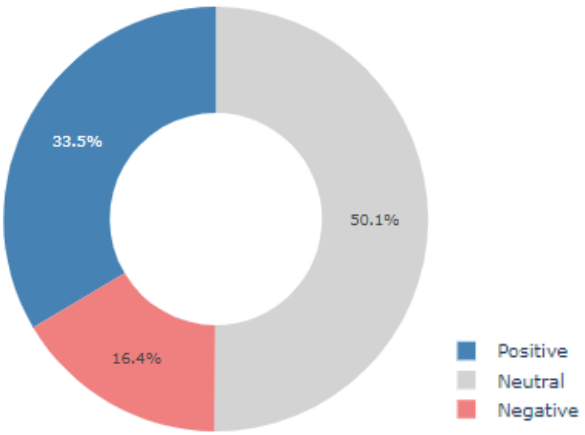
La librería se ha construido pensando, entre otras cosas, en el análisis particular de redes sociales.

Fig. 9. Ejemplificación Polaridad - TextBlob

```
texto:
  nancyleegrahn feel climate change question night exactly
------------------------------------
Polaridad según TextBlob
0.25
```

Fuente: Elaboración propia

Fig. 10. Composición de sentimientos - TextBlob



Fuente: Elaboración propia

TABLA II
COMPOSICIÓN DE SENTIMIENTOS - TEXTBLOB

Sentimiento	Frecuencia	Porcentaje
Positivo	4,652	33.5%
Neutral	6,949	50.1%
Negativo	2,270	16.4%

En el caso anterior, se observa que la cantidad de tweets con sentimiento positivo disminuye drásticamente, obteniendo apenas un 34% de los comentarios totales, mientras que los comentarios neutrales ascienden a la mitad. También se puede notar que TextBlob no tuvo problema en clasificar los tweets confusos o con pocas palabras, a diferencia de SentiWordNet.

Fig. 11. Ejemplificación de comentarios sin clasificar - Con TextBlob

	text	lemma_text	sentiment_swn	sentiment_tb
43	RT @brownblaze: PLEASE RT. #KKKorGOP #GOPDebat...	brownblaze kkkorgop	0	Neutral
143	RT @peddoc63: Go Carly📢\nGo Carly📢\nGo Carly📢\...	peddoc carly📢 carly📢 carly📢 carlyfiorina car...	0	Neutral
253	RT @JosephKapsch: And then hillaryclinton @bar...	josephkapsch hillaryclinton barackobama joeb...	0	Neutral
385	At least he was honest about it. #GOPDebate h...	honest	0	Positive
411	RT @Squintz1983: #GOPDebate pretty much http:/...	squintz pretty	0	Positive

Fuente: Elaboración propia

3. *Con la librería Vader*

VADER (Valence Aware Dictionary and sEntiment Reasoner) es una herramienta de análisis de sentimientos basada en reglas y léxico que se adapta específicamente a los sentimientos expresados en las redes sociales.

Al momento de ejecutar el código para obtener la polaridad, VADER arroja lo siguiente:

Fig. 12. Ejemplificación Polaridad - VADER

```
texto:
  nancyleegrahn feel climate change question night exactly
------------------------------------
Polaridad según Vader
{'compound': 0.0, 'neg': 0.0, 'neu': 1.0, 'pos': 0.0}
```

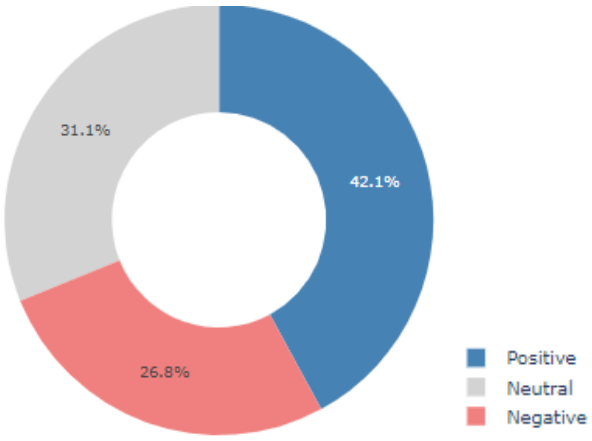
Fuente: Elaboración propia

El puntaje compuesto es una métrica que calcula la suma de todas las calificaciones de léxico que se han normalizado entre -1(más extremo negativo) y +1(más extremo positivo).

De acuerdo con la página de Python | Análisis de Sentimiento con VADER, se tomará el siguiente umbral para calcular el sentimiento:

- Sentimiento positivo:(puntaje compuesto> = 0.05)
- sentimiento negativo:(puntaje compuesto <= -0.05)
- sentimiento neutro: (puntaje compuesto> -0.05) y (puntaje compuesto <0.05)

Fig. 13. Composición de sentimientos - VADER



Fuente: Elaboración propia

TABLA III
COMPOSICIÓN DE SENTIMIENTOS - VADER

Sentimiento	Frecuencia	Porcentaje
Positivo	5,840	42.1%
Neutral	4,320	31.1%
Negativo	3,711	26.8%

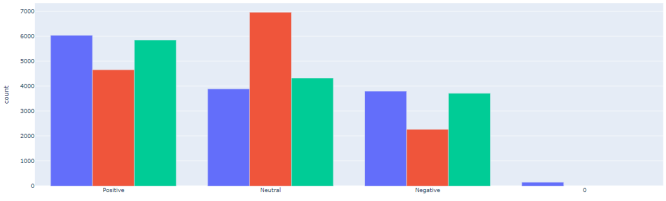
En el último caso, con el algoritmo de Vader, se comporta relativamente similar con el umbral seleccionado al algoritmo de SentiWordNet, con la diferencia de que Vader logró etiquetar correctamente la totalidad de los tweets, al igual que TextBlob. Esto sucede porque el léxico de estos últimos está adaptado para etiquetas de comentarios de redes sociales.

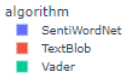
IV. VISUALIZACIÓN DE COMPARATIVAS

Se observa un desbalance significativo entre TextBlob y los otros dos algoritmos probados. Es importante destacar que el hecho de que SentiWordNet haya arrojado valores nulos en el sentimiento de algunos comentarios, indica que estos pueden descartarse por la poca cantidad que representan del total (1.12%), no obstante, los valores nulos también hacen ver que aún hay área de oportunidad en el pre procesamiento del texto y se podría realizar más limpieza de palabras, por ejemplo, agregando stopwords.

Para futuros análisis, se usará el sentimiento de VADER como el definitivo, puesto que existe más similitud con el de SentiWordNet y está mejor adaptado a redes sociales, además, éste último contiene valores nulos.

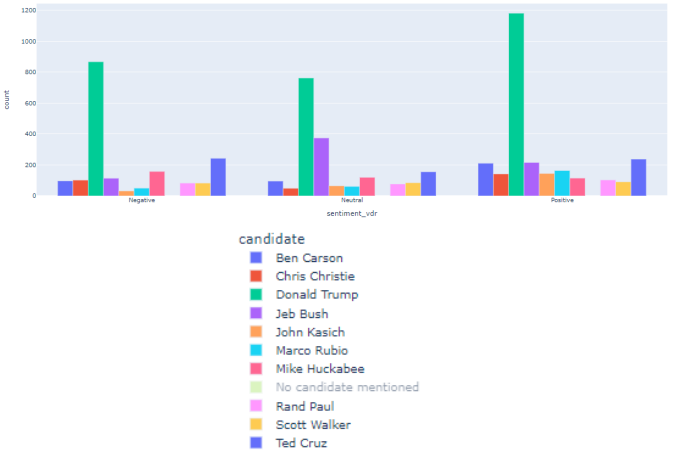
Fig. 14. Composición de sentimientos - VADER





Fuente: Elaboración propia

Fig. 15. Análisis de sentimiento por candidato



Fuente: Elaboración propia

Dejando fuera los tweets que no mencionaron a ningún candidato, el candidato más popular por mucho fue Donald Trump, con 1182 tweets positivos hacia él, 763 neutrales y 868 negativos.

V. CONCLUSIONES

El análisis de texto es una herramienta muy poderosa al momento de tener datos en crudo que aportan mucho valor. El análisis de sentimiento particularmente para redes sociales se ha vuelto muy relevante pues la cantidad inmensa de contenido que se genera ha derivado en la necesidad de realizar mayor limpieza y de categorizar o etiquetar los comentarios.

En este proyecto se observó que la técnica más efectiva para abordar el análisis de sentimiento fue la del algoritmo de VADER, por ser comentarios obtenidos de una red social.

REFERENCIAS

[1] Github Personal: https://github.com/KinMichelle/FCFM/blob/ca9df2c9245d4953853e37ebd9148ddaa2b65241/Procesa_y_clasif/Tarea_2/Tarea%202%20-%20KMNR%20v2.ipynb

[2] Conjunto de datos: First GOP Debate Twitter Sentiment. Analyze tweets on the first 2016 GOP Presidential Debate. https://www.kaggle.com/datasets/crowdflower/first-gop-debate-twitter-sentiment?resource=download

[3] En apoyo de: https://github.com/mayraberrones94/FCFM/blob/master/Semana_2_Analisis_de_sentimiento.ipynb