

**Evaluación final de módulo**  
**© Ediciones Roble S. L.**

# Indice

<b>Evaluación final de módulo</b>	<b>3</b>
Instrucciones evaluación final del módulo	3
Caso práctico	3
Ejercicio 1	3
Ejercicio 2	6

## Evaluación final de módulo



Es importante leer las instrucciones antes de realizar la evaluación.

## Instrucciones evaluación final del módulo

La evaluación final consta de dos partes:

### Caso práctico

#### Formato de entrega de los ejercicios

La entrega de la práctica se realizará mediante una **tutoría**. Las condiciones de entrega de la práctica son las siguientes:

1. OBLIGATORIO: Archivo .py o .ipynb con el código desarrollado en la práctica. Si se desea, se puede realizar una memoria sobre la práctica implementada haciendo uso de la plantilla disponible en Recursos.
2. OPCIONAL: Archivo .html resultante de guardar los notebooks (si se realizan) como .html.

**Si se hace entrega de más de un archivo, se deberán entregar desde un archivo comprimido .zip o .rar.**

### Cuestionario tipo test

15 preguntas con cuatro opciones de respuesta, de las cuales solo una es la correcta. Cuentas con 20 minutos para realizarlo y dispones solo de tres intentos.

## Caso práctico

Se van a plantear **dos problemas** para ilustrar un análisis de datos sobre información procedente de Twitter utilizando Python.

## Ejercicio 1



- [Tweets.txt](#)
- [Sentimientos.txt](#)

Dado un archivo que contiene en cada línea una palabra o conjunto de palabras seguido de un valor numérico, denominado “sentimiento”, y un conjunto de tweets, se pide calcular para cada tweet un valor denominado “sentimiento del tweet”, que se obtiene como la suma de los “sentimientos” de los términos que aparecen en el tweet.



Obsérvese que:

- Cada línea del fichero Tweets.txt contiene una cadena JSON.
- No todas las líneas del archivo Tweets.txt contiene tweets.
- Cada tweet es representado por una cadena en formato JSON (figura 9.1.), en la que el tweet es almacenado en la clave text:

```
{
  "created_at": "Wed Sep 10 17:01:33 +0000 2014",
  "id": 509748524897292288,
  "id_str": "509748524897292288",
  "text": "@Brenamae_ I WHALE SLAP YOUR FIN AND TELL YOU ONE LAST TIME: GO AWHALE",
  "source": "\u003ca href=\"http://twitter.com/download/android\" rel=\"nofollow\"\u003eTwitter for Android\u003c/a\u003e",
  "truncated": false,
  "in_reply_to_status_id": 509748106015948800,
  "in_reply_to_status_id_str": "509748106015948800",
  "in_reply_to_user_id": 242563886,
  "in_reply_to_user_id_str": "242563886",
  "in_reply_to_screen_name": "Brenamae_",
  "user": {
    "id": 175160659,
    "id_str": "175160659",
    "name": "Butterfly",
    "screen_name": "VanessaLilyWan",
    "location": "Canada, Montreal",
    "url": "http://instagram.com/vanessalilywan",
    "description": "British youtubers. 'Nuff said.",
    "protected": false,
    "verified": false,
    "followers_count": 118,
    "friends_count": 180,
    "listed_count": 2,
    "favourites_count": 319,
    "statuses_count": 10221,
    "created_at": "Thu Aug 05 20:03:16 +0000 2010",
    "utc_offset": -36000,
    "time_zone": "Hawaii",
    "geo_enabled": false,
    "lang": "en",
    "contributors_enabled": false,

```

**Figura 9.1.** Formato JSON.

*Fuente:* elaboración propia.

Aquellas palabras o conjunto de palabras que aparecen en un tweet, y no tengan un “sentimiento” asociado en el archivo, se contabilizarán como 0.

Para procesar el archivo hay que tener en cuenta que cada palabra o conjunto de palabras están separadas de su valor mediante el carácter “\t”. El archivo podría almacenarse en un diccionario usando el siguiente trozo de código (figura 9.2.).

```

sentimientos = open("Sentimientos.txt")
valores = {}
for linea in sentimientos:
    termino, valor = linea.split("\t")
    valores[termino] = int(valor)
print (valores.items() )

```

```

dict_items([('abandon', -2), ('abandoned', -2), ('abandons', -2), ('abducted', -2), ('abduction', -2), ('abductions', -2), ('abhor', -3), ('abhorred', -3), ('abhorrent', -3), ('abhors', -3), ('abilities', 2), ('ability', 2), ('aboard', 1), ('absentee', -1), ('absentees', -1), ('absolve', 2), ('absolved', 2), ('absolves', 2), ('absolving', 2), ('absorbed', 1), ('abuse', -3), ('abused', -3), ('abuses', -3), ('abusive', -3), ('accept', 1), ('accepted', 1), ('accepting', 1), ('accepts', 1), ('accident', -2), ('accidental', -2), ('accidentally', -2), ('accidents', -2), ('accomplish', 2), ('accomplished', 2), ('accomplishes', 2), ('accusation', -2), ('accusations', -2), ('accuse', -2), ('accused', -2), ('accuses', -2), ('accusing', -2), ('ache', -2), ('achievable', 1), ('aching', -2), ('acquit', 2), ('acquits', 2), ('acquitted', 2), ('acquitting', 2), ('acrimonious', -3), ('active', 1), ('adequate', 1), ('admire', 3), ('admired', 3), ('admires', 3), ('admiring', 3), ('admit', -1), ('admits', -1), ('admitted', -1), ('admonish', -2), ('admonished', -2), ('adopt', 1), ('adopts', 1), ('adorable', 3), ('adore', 3), ('adored', 3), ('adores', 3), ('advanced', 1), ('advantage', 2), ('advantages', 2), ('adventure', 2), ('adventures', 2), ('adventurous', 2), ('affected', -1), ('affection', 3), ('affectionate', 3), ('afflicted', -1), ('affronted', -1), ('afraid', -2), ('aggravate', -2), ('aggravated', -2), ('aggravates', -2), ('aggravating', -2), ('aggression', -2), ('aggressions', -2), ('aggressive', -2), ('aghost', -2), ('agog', 2), ('agonize', -3), ('agonised', -3), ('agonises', -3), ('agonising', -3), ('agonize', -3), ('agonized', -3), ('agonizes', -3), ('agonizing', -3), ('agree', 1), ('agreeable', 2), ('agreed', 1), ('agreement', 1), ('agrees', 1), ('alarm', -2), ('alarmed', -2), ('alarmist', -2), ('alarmists', -2), ('alas', -1), ('alert', -1), ('alienation', -2), ('alive', 1), ('allergic', -2), ('allow', 1), ('alone', -2), ('amaze', 2), ('amazed', 2), ('amazes', 2), ('amazing', 4), ('ambitious', 2), ('ambivalent', -1), ('amuse', 3), ('amused', 3), ('amusement', 3), ('amusements', 3), ('anger', -3), ('angers', -3), ('angry', -3), ('anguish', -3), ('anguished', -3), ('animosity', -2), ('annoy', -2), ('annoyance', -2), ('annoyed', -2), ('annoying', -2), ('annoys', -2), ('antagonistic', -2), ('anti', -1), ('anticipation', 1), ('anxiety', -2),

```

**Figura 9.2.** Procesamiento del archivo Sentimientos.txt.

*Fuente:* elaboración propia.



La ejecución del código se muestra en el cuaderno Caso\_práctico\_final\_ejercicio\_1\_1.ipynb, el cual se puede descargar en este [enlace](#).

Como resultado, se debe mostrar por pantalla un valor numérico en cada línea que represente el “sentimiento” de un tweet.

No todos los tweets que se van a considerar tienen contenido, por lo que hay que filtrar aquellos que tienen de los que no tienen.

El programa tendrá 2 parámetros de entrada: el archivo que contiene los sentimientos de los términos y el archivo que contiene los tweets. Estos parámetros se pueden leer como input desde el teclado, directamente en el código o como parámetros con sys.args.



Si se toma como referencia el siguiente tweet:

- I will go on vacation to the beach.

Supongamos que las siguientes palabras aparecen en el archivo de sentimientos:

- vacation: 2
- beach: 1

Ya que el resto de palabras no se encuentran en el archivo de sentimientos, tendríamos un sentimiento del tweet de  $(2+1) = 3$  y, por lo tanto, tendríamos la siguiente salida del programa:

- EL SIGUIENTE TWEET: 'I will go on vacation to the beach' TIENE UN SENTIMIENTO ASOCIADO DE: 3

## Ejercicio 2

Dado un archivo que contiene en cada línea una palabra o conjunto de palabras seguido de un valor numérico denominado “sentimiento” y un conjunto de tweets, se pide calcular el sentimiento de aquellas palabras o conjunto de palabras que no tienen un valor asociado en el archivo de “sentimientos”. Se pueden seguir distintas estrategias para asignar un valor. Por ejemplo, se podría asignar como valor el valor del “sentimiento” del tweet en que se encuentra la palabra o conjunto de palabras sin valor, o el valor medio del “sentimiento” del tweet.



Obsérvese que:

- Para procesar el archivo hay que tener en cuenta que cada palabra o conjunto de palabras están separadas de su valor mediante el carácter “t”. El archivo podría almacenarse en un diccionario usando el siguiente trozo de código (figura 9.3.).
- Cuaderno *Caso\_práctico\_final\_ejercicio\_1\_1.ipynb* que se ha facilitado anteriormente.

```
sentimientos = open("Sentimientos.txt")
valores = {}
for linea in sentimientos:
    termino, valor = linea.split("\t")
    valores[termino] = int(valor)
print (valores.items() )
```

**Figura 9.3.** Procesamiento del archivo Sentimientos.txt.

*Fuente:* elaboración propia.

Como resultado se debe mostrar por pantalla en cada línea, una palabra o conjunto de palabras y a continuación un valor numérico que represente el “sentimiento” de dicha palabra o conjunto de palabras. Solo se mostrarán la palabra o conjunto de palabras que no aparecían con un valor en el archivo original.

No todos los tweets que se van a considerar tienen contenido, por lo que hay que filtrar aquellos que tienen de los que no tienen.

El programa tendrá 2 parámetros de entrada: el archivo que contiene los sentimientos de los términos y el archivo que contiene los tweets.



Si se toma como referencia el tweet anterior:

- I will go on vacation to the beach.

Sabemos que las palabras 'vacation' y 'beach' ya tienen un valor asociado en el archivo de sentimientos; por lo tanto, tenemos que actualizar el sentimiento del resto de palabras. Podríamos hacerlo de dos formas:

**Versión 1, asignando el valor del sentimiento del tweet (3) a las palabras:**

- 'I': 3
- 'will': 3
- 'go': 3
- 'on': 3
- 'vacation': No mostramos esta palabra, dado que tiene un sentimiento ya asociado anteriormente.
- 'to': 3
- 'the': 3
- 'beach': No mostramos esta palabra, dado que tiene un sentimiento ya asociado anteriormente.

**Versión 2, asignando la media del valor del tweet, siendo esta media el número de palabras del tweet dividido entre el valor de sentimiento del tweet; el tweet de ejemplo tiene 8 palabras, por lo tanto,  $8 \div 3 = 2.66$ .**

- 'I': 2.66
- 'will': 2.66
- 'go': 2.66
- 'on': 2.66
- 'vacation': No mostramos esta palabra, dado que tiene un sentimiento ya asociado anteriormente.
- 'to': 2.66
- 'the': 2.66
- 'beach': No mostramos esta palabra, dado que tiene un sentimiento ya asociado anteriormente.