Práctica laboratorio PLN

Íñigo Gómez Carvajal y Jon Zorrilla Gamboa

Las tareas realizadas han sido las siguientes:

Tareas obligatorias: 1.1, 2.1, 3.1, 4.1, 5.1, 5.2

Tareas opcionales: 1.2, 2.2, 3.2, 3.3, 4.2, 4.3

La implementación de cada una de las tareas, así como sus resultados, se encuentran en el Notebook adjunto.

1 Tarea 1: revisión de los datasets.

1.1 Tarea 1.1:

En primer lugar, cargamos las *reviews* de *yelp_hotels.json* mediante el módulo *json* de Python. Además, imprimimos el número de *reviews*, la primera *review* y el ID asociado a la primera *review*.

1.2 Tarea 1.2:

Ahora, leemos línea a línea los datasets *yelp_beauty_spas.json* y *yelp_restaurants.json*, prestando atención a posibles incidencias en el formato. También imprimimos el número de *reviews*, la primera *review* y el ID asociado a la primera *review*.

2 Tarea 2: Vocabulario de aspectos.

2.1 Tarea 2.1:

Ahora, cargamos e imprimimos el vocabulario de *aspects_hotels.csv*, para poder identificar los términos de las *reviews* que se relacionen con aspectos de hoteles. Para ello, usamos un diccionario, el cual tiene para cada aspecto el conjunto de sustantivos asociados.

```
amenities amenity
amenities amenities
amenities services
atmosphere atmosphere
atmosphere atmospheres
...
```

2.2 Tarea 2.2:

Para tener en cuenta los sinónimos que pueden tener estos sustantivos, hacemos uso de *synsets* de *wordnet*, de tal manera que añadimos al diccionario los sinónimos de los sustantivos que tenemos. Luego, invertimos el diccionario para tener para cada sustantivo, el aspecto asociado. Esto último permitirá hacer una búsqueda de aspectos en siguientes apartados más sencilla.

3 Tarea 3: Léxicos de opinión

3.1 Tarea 3.1:

Cargamos *opinion_lexicon* desde *nltk.corpus* donde podemos ver las palabras con connotación negativa y connotación positiva, a las que asociaremos -1 y $+1$, respectivamente. Así, creamos un diccionario que contiene todas estas palabras como llaves, con su valor $+1$ o -1 asociado.

3.2 Tarea 3.2:

En este caso, buscamos hacer uso de los *modifiers*, los cuales miden la intensidad de las opiniones según los adverbios asociados. Esto nos permitirá evaluar opiniones de forma más precisa. Para ello, nos vamos a generar un diccionario donde las claves serán estos adverbios y el valor será el *modifier* asociado.

3.3 Tarea 3.3:

Para añadir la posibilidad de una inversión en la polaridad de una opinión, tenemos en consideración la posible aparición del adverbio *not* en la frase. Debido a que la dependencia que puede tener es la misma que otros adverbios a la hora de realizar la tarea 4, hemos decidido que una solución sencilla (aunque en cierto modo limitada) es incluir la palabra como un posible modificador más, con valor -1.

4 Tarea 4: Opiniones de aspecto

Hemos decidido no dividir esta sección en subsecciones, puesto que en la función implementada hemos optado por incluir todo lo referente a la tarea, incluidos sus opcionales.

En primer lugar, hemos decidido optar por hacer una extracción de opiniones basada en las dependencias que encontramos en las frases mediante la librería de CoreNLP. El motivo detrás de esto es que consideramos más importante fijarnos en la función que puede tener una palabra con respecto de otra a la hora de interpretar el significado de la misma. Además, creemos que, desarrollado a un nivel más sofisticado, podría ofrecer una extracción más correcta.

Los relaciones que hemos querido modelizar para esta tarea son las siguientes.

- **Relaciones *amod*:** Este tipo de relaciones son las más directas y esenciales, puesto que hablan de una modificación realizada por un adjetivo sobre otra palabra. Para ello, buscaremos tuplas (sustantivo, adjetivo) que tengan esta relación, y buscaremos el posible valor del adjetivo en nuestro lexicon para obtener un valor de polaridad

- **Relaciones *nsubj*:** Con estas relaciones buscamos adjetivos que están en el predicado de la frase, pero que hagan referencia a un sustantivo. Las tuplas que generamos con esta relación son las misma que en el caso *amod*.

- **Relaciones *advmod*:** En este caso buscamos adverbios que supongan una modificación sobre un sustantivo o un adjetivo. Sobre ellos generamos una tupla *(adjetivo/sustantivo, adverbio, modificador)*, siendo *modificador* el valor del diccionario de *modifiers* asociado al adverbio.

- **Relaciones *conj*:** Aquí buscamos modelizar adjetivos adicionales que vayan coordinados en una frase con otros adjetivos con relacion *amod* o *nsubj*, puesto que hacen referencia al mismo sustantivo y aportan significado adicional a la opinión. Para ello, guardamos el adjetivo que tiene relación directa con el sustantivo con el adjetivo coordinado, para luego generar una tupla de relación entre el sustantivo y el coordinado.

- **Relaciones *compound*:** Para este último caso prestamos atención a sujetos que estén formados por más de un sustantivo (por ejemplo: *boutique shops*). Esto lo tenemos en cuenta de cara a localizar en los sujetos palabras relacionadas con aspectos que se nos puedan escapar, sobre los que haremos una nueva tupla que sea idéntica a la parte del sujeto que CoreNLP relaciona con un adjetivo. Esto no genera que las opiniones se sumen, puesto que luego solo nos quedamos con la palabra que haga referencia a algún término de nuestro vocabulario de aspectos.

Sobre esto, una vez captadas todas las tuplas, tratamos las que se relacionan con palabras relevantes a aspectos que buscamos, lo que hacemos es aplicar los modificadores adverbiales relacionados con los adjetivos a la polaridad obtenida, con especial atención a las posibles inversiones de polaridad, porque indicamos como parte de la tupla final si la frase tiene la polaridad invertida o no.

Vamos a adjuntar a continuación algunos de los resultados que hemos obtenido con unas frases de prueba.

En primer lugar, podemos ver una frase que muestra como el modelo capta modificadores y relaciona los términos con el aspecto correspondiente.

```
review_aspect_extraction("Great bar. Incredibly bad bedrooms", polarities, modifiers,
tokens_dict)
```

```
[('bar', 'bar', 'Great', 'Non negated', [], 1),
('bedrooms', 'bedrooms', 'bad', 'Non negated', ['Incredibly'], -2.0)]
```

Ahora, probamos con una oración que tiene en cuenta el uso de múltiples adjetivos para un mismo término, cada uno con su modificador asociado.

```
review_aspect_extraction("Slightly dirty and incredibly bad bedrooms",
                         polarities, modifiers, tokens_dict)
```

```
[('bedrooms', 'bedrooms', 'dirty', 'Non negated', ['Slightly'], -0.5),
('bedrooms', 'bedrooms', 'bad', 'Non negated', ['incredibly'], -2.0)]
```

Por último, probamos con una oración que incluye el adjetivo en el predicado, además de una inversión de polaridad.

```
review_aspect_extraction("The food was not bad at all",
                         polarities, modifiers, tokens_dict)
```

```
[('cuisine', 'food', 'bad', 'Negated', ['not'], 1)]
```

5 Tarea 5: Síntesis de opiniones

5.1 Tarea 5.1:

Para la realización de esta tarea, hemos optado por agrupar todas las tuplas de opinion en forma de DataFrame, en el que se incluye la misma información que se muestra en las tuplas de la tarea 4. Este enfoque permite hacer una agrupación rápida de la polaridad de las opiniones y facilitarnos la tarea.

Para la realización de esta tarea, hemos optado por desarrollar dos funciones. La primera, llamada *group_aspects*, donde simplemente hacemos una agrupación por aspectos y la polaridad de las opiniones se suma, y la segunda, llamada *plot_aspect_opinion*, donde hacemos un gráfico de barras donde se muestra la valoración final de la review por aspectos. Con ello obtenemos este tipo de resultados.

Texto de la review: *This is my favorite spa in the world! Hopefully, I'll make it to others...but if I could only get to the Camelback Inn, I'd be in heaven. Here's the rundown of how I enjoy the place: Early morning appointment for massage - Rose does fantastic hot stone - I like her the best. Then I shower and get into the eucalyptus steam room. Next I head out to the pool and savor the view of Camelback Mountain and the amazing desert landscaping they've procured. I drink some water and as I lounge there, I order a small, healthy and very tasty meal (usually breakfast). I swim a little, lounge a lot - regardless of the time of year. If I have planned it, I then head back into the showers and clean up for my next treatment. I spend hours there and if I feel like I might be pressed for time, I reschedule.*

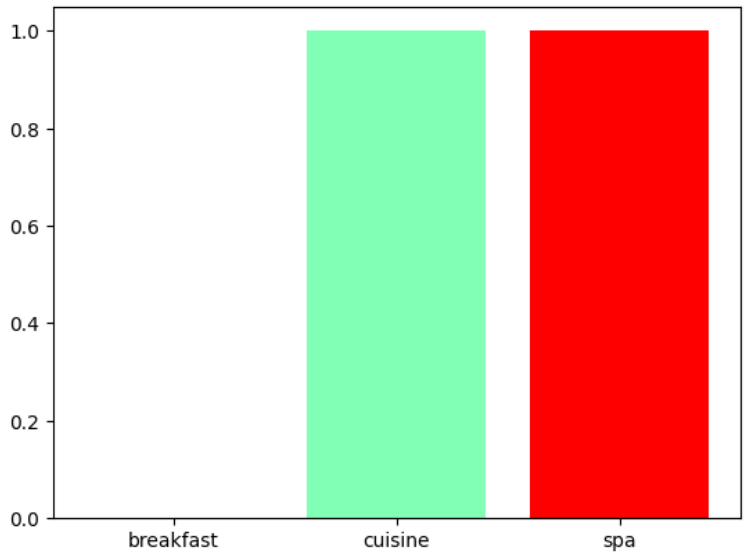

Figura 1: Resultado de la tarea 5.1 dada una review

5.2 Tarea 5.2:

Para la primera parte de esta tarea, debido al planteamiento de estructura por DataFrames que hemos realizado en la tarea 5.1, la síntesis de opiniones por hotel es una extensión natural del problema. Seleccionando el identificador de un hotel dada una review, decidimos extraer todas las reviews relacionadas con ese hotel, obtenemos las tuplas de opinión de todas las reviews y agrupamos la polaridad por aspecto, del mismo modo que hemos hecho la anterior vez. En este caso, mostramos las reviews relacionadas con el hotel de identificador *EcHuaHD9IcoPEWNsU8vDTw*.

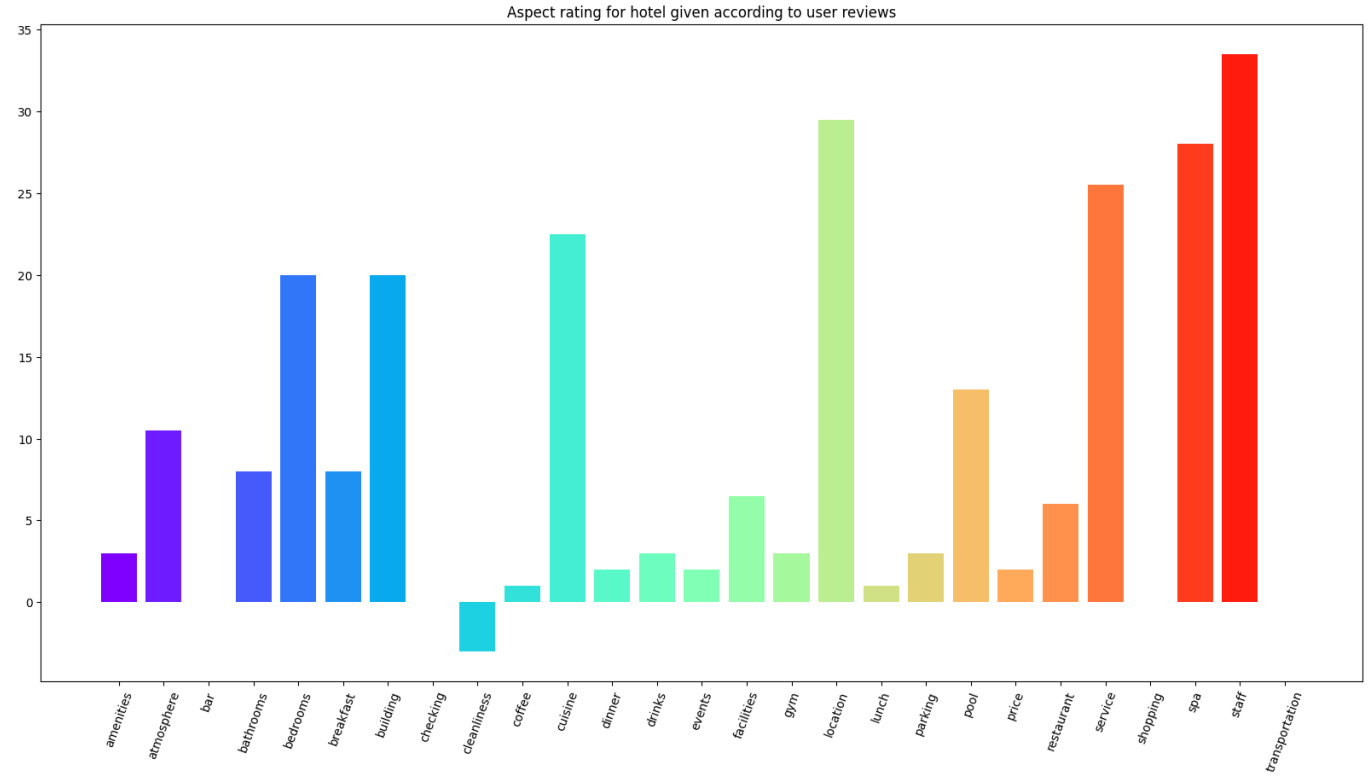

***Figura 2*:** Resultado de la tarea 5.2 para el hotel indicado

De cara a la recolección del número de opiniones negativas y positivas sobre los aspectos de un hotel, lo que hemos realizado es implementar dos funciones. La primera, *positive_negative_counter*, nos genera dos columnas adicionales al DataFrame original, una conteniendo un marcador a 1 o 0 si la opinión es positiva o no, y la otra columna haciendo lo mismo en caso de que la opinión sea negativa. Volvemos a agrupar por aspecto, sumando todos los valores de estas dos columnas para obtener el DataFrame pertinente. La segunda función, *plot_positive_negative_opinion* nos imprime por pantalla un gráfico de barras conteniendo el número de opiniones positivas y negativas de las reviews. El resultado con el mismo hotel es el siguiente.

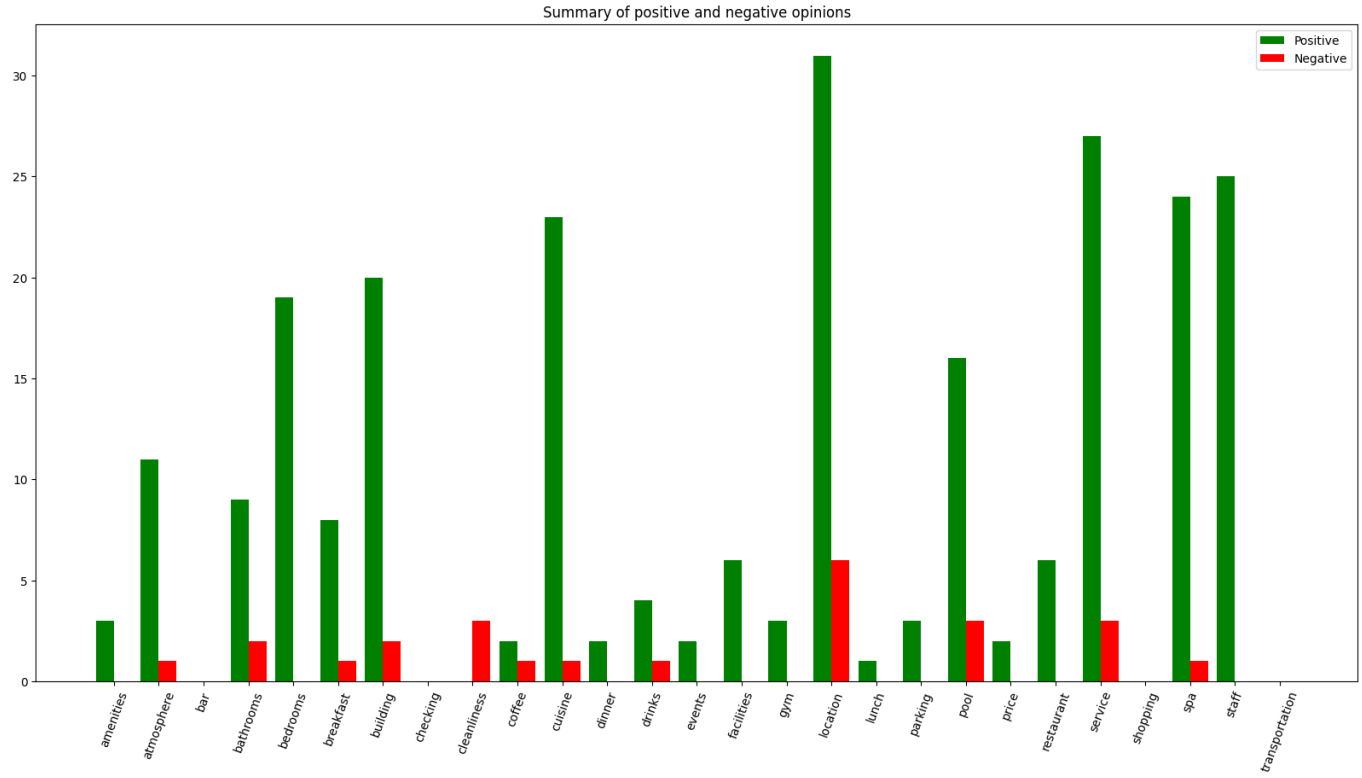

***Figura 3*:** Contador de opinoines positivas y negativas para el hotel indicado