Universidad de Granada

Escuela Técnica Superior de Ingeniería Informática y Telecomunicaciones

Sentiment Analysis

Práctica Final

Inteligencia de Negocio

Curso 2020 - 2021

José A. Martín Melguizo - (77561280J)

Correo: josemartin22@correo.ugr.es

Índice general

Capítulo 1

Descripción y análisis del problema

En **análisis de sentimientos** consiste en el uso de técnicas de Procesamiento del Lenguaje Natural, analítica de textos y lingüística computacional para identificar y extraer información subjetiva u opinión expresada en texto.

Se utiliza en sistemas de recomendación, filtrado de mensajes/spam, en aplicaciones de negocios para predecir tendencias en función de las reseñas de los clientes o incluso seguir las opiniones sobre política, elaborar normas en función de la opinión de la gente (inteligencia de gobierno), etc...

El análisis de sentimientos se puede llevar a cabo a 3 niveles distintos en función de la granularidad, profundidad o detalle que se desee conseguir. Estos niveles son:

- **Análisis a nivel de documento:** en este nivel se analiza el sentimiento global del documento, clasificándolo como positivo, negativo o neutro (o usando cualquier otro sistema de calificación). En este nivel se asume que se expresa una valoración sobre un único ente, por lo que no sería aplicable para aquellos casos que hablen sobre varias entidades simultáneamente.
- **Análisis a nivel de oración:** en este nivel se divide el documento en oraciones individuales para extraer posteriormente la opinión que contiene cada una de ellas. La opinión de cada oración puede ser positivia, negativa o neutra (o tomar otro sistema de calificación al igual que el caso anterior).
- **Análisis a nivel de aspecto y entidad:** es el nivel más fino y con mayor detalle posible, una entidad está formada por distintos elementos o aspectos y sobre los que se expresa una opinión cuya polaridad puede ser distinta en cada caso. Este nivel es el que presenta un mayor desafío en la actualidad para los investigadores de dicho ámbito (sentiment analysis).

En nuestro caso nos **centraremos** en el primer nivel, la **clasificación** de la **polaridad** de una frase de opinión (en concreto de tweets), esto es, decidir si en una frase el autor muestra un sentimiento positivo, negativo o neutro.

Se trabajará sobre el conjunto de datos en el dominio de Tweets del conjunto de datos `Sentiment140`, formado por 1.6 millones de tweets en el conjunto de Train y 498 tweets en el conjunto de test etiquetados como polaridad positiva (clase '4') y polaridad negativa (clase '0').

Capítulo 2

Descripción de los algoritmos

2.1. Métodos para la clasificación de documentos

La clasificación de un documento en función a la polaridad del sentimiento que expresa, puede llevarse a cabo de diversas formas. No hay un consenso (como en la mayoría de campos de la inteligencia artificial) sobre qué técnicas o algoritmos se deben usar para obtener mejores resultados en la clasificación de textos en unas situaciones u otras.

Aun así, podemos diferenciar los dos grandes grupos en los se encuentran dichas técnicas:

1. **Métodos supervisados:** Se basa en el empleo de algoritmos de aprendizaje automático de tipo supervisado, ya que se parte de un conjunto de textos previamente etiquetados (se conoce su polaridad) con los que se entrena un modelo que será usado posteriormente para clasificar un nuevo conjunto de textos (que en este ámbito es conocida como colección documental o *corpus*).

   Dichos algoritmos basan su funcionamiento en relaciones matemáticas creadas entre los elementos de entrenamiento. Éstos deben de ser representativos en cuanto a los elementos de test para el óptimo funcionamiento de éste (es el principal problema, no siempre se tiene un conjunto de entrenamiento lo suficientemente grande o bueno, a veces es difícil obtenerlo).

2. **Métodos no supervisados:** Resuelven el hecho de tener que contar con un conjunto de elementos preetiquetados (salva el problema de la dependencia del dominio). Estos métodos tratan de inferir la polaridad del sentimiento global de un documento a partir de la orientación semántica de las palabras o frases que lo conforman. Estos a su vez se dividen en:

   a) **Basados en diccionarios**: Hacen uso de un listado de palabras o frases junto a la polaridad que expresan, incluso con un grado de intensidad o fuerza de dicha polaridad. Tiene la desventaja de que a veces se pierde precisión ya que las palabras dependendiendo del contexto pueden tener una connotación más positiva o negativa, etc.

   Ese problema se resuelveçreando diccionarios concretos para un ámbito del lenguaje específico. Es decir el conjunto de palabras que lo forman se centra en un

dominio concreto, por ejemplo: medicina. Pero aún dentro de un mismo ámbito podría seguir ocurriendo ésto y habría que hilar más fino en la construcción de los mismos.

b) **Basados en relaciones lingüísticas**: Tratan de buscar ciertos patrones que expresen opiniones y sentimientos similares, con la mayor probabilidad posible, extrayendo las palabras que los formas para ser usadas posteriormente para clasificar textos mediante una función matematica que tenga en cuenta las palabras que lo contengan.

El enfoque no supervisado es elegante pero necesita apoyarse de alguna forma en datos que aporten una cierta orientación y sirvan de base de conocimiento. Por ello nos **centraremos** en métodos de **aprendizaje supervisado**.

2.2. Naive Bayes

La familia de algoritmos Naive Bayes se basan en el conocido *Teorema de Bayes*:

Sea $\{A_1, A_2, \ldots, A_n\}$ *un conjunto de sucesos mútuamente excluyentes y exhaustivos, y tales que la probabilidad de cada uno de ellos es distinta de cero. Sea B un suceso cualquiera del que se conocen las probabilidades condicionales* $P(B|A_i)$.

Entonces la probabilidad de $P(A_i|B)$ *viene dada por la expresión:*

$$P(A_i|B) = \frac{P(B|A_i) * P(A_i)}{P(B)}$$

Para el caso de clasificación de textos, los sucesos son mútuamente excluyentes ya que no podemos considerar (en nuestro caso un tweet) que es de polaridad positiva y negativa al mismo tiempo y además son exhaustivos ya que esa calificación (positivo, negativo y neutro) son los únicos tipos que existen.

Además consideraremos que las palabras de un mismo mensaje no tienen ningún tipo de relación entre sí y por lo tanto es indiferente la posición que tienen dentro del texto al que pertenecen. (Esta condición es conocida en este ámbito como la suposición de bolsa de palabras o *bag of words*).

2.3. Máquinas de vectores de soporte

Las máquinas de vectores de soporte (del inglés, Support Vector Machine o SVM) son un grupo de algoritmos de aprendizaje supervisado en los que se trata de encontrar un hiperplano separador (denominado vector de soporte) que separe con la mayor distancia posible las distintas clases a las que pertenecen cada uno de los elementos.

De esta forma, el vector determina la frontera que sirve para clasificar un nuevo elemento.

Cuenta con una serie de parámetros que permitirán optimizar los resultados durante el proceso de clasificación, uno de ellos es el *kernel* y se utiliza cuando no es posible separar las muestras mediante una línea recta, plano o hiperplano de N dimensiones, permitiendo tal separación mediante otro tipo de funciones matemáticas como polinomios, funciones de base radial, etc.

Otro parámetro conocido es *regularization* o C, que permite una fluctuación de manera que se consideren ciertos errores en la clasificación y se evite el sobreaprendizaje en la manera de lo posible.

Y otro de ellos es *gamma* que determina la distancia máxima a partir de la cual una muestra pierde su influencia en la configuración del vector de soporte y *margin* que es la separación entre el vector y las muestras de cada clase más cercanas al mismo.

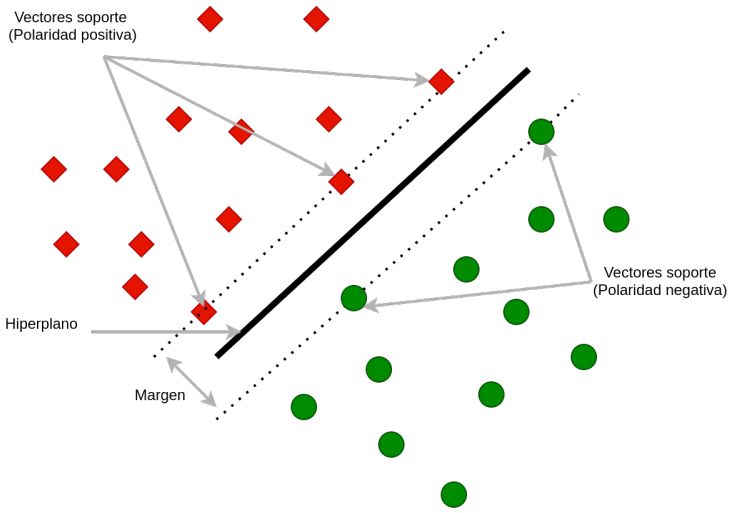

Figura 2.1: Representación de SVM para el problema

2.4. Métricas de evaluación

Para determinar el rendimiento de los resultados obtenidos por los distintos algoritmos, es necesario establecer unas métricas o medidas de evaluación que permitan evaluar de manera objetiva su eficacia a la hora de clasificar los ejemplos que se le proporiconen a éstos.

Para ello, es importante tener en cuenta los cuatro posibles estados de un ejemplo clasificado, por ejemplo con respecto a una clase C:

1. **Verdaderos positivos** (True Positives o TP): son aquellos que han sido marcados de manera correcta como pertenecientes a la clase C.

2. **Falsos positivos** (False Positives o FP): son aquellos marcados como de clase C, pero que en realidad no pertenecen a dicha clase, es decir, han sido clasificados de forma incorrecta.

3. **Verdaderos negativos** (True Negatives o TN): son aquellos que no son de la clase C y han sido clasificados correctamente.

4. **Falsos negativos** (False Negatives o FN): son aquellos clasificados como no pertenecientes a la clase C, pero en realidad sí lo son y por tanto no se han clasificado correctamente.

Teniendo en cuenta los cuatro posibles estados anteriores, podemos definir las siguientes medidas que serán usadas para evaluar el rendimiento de nuestros modelos:

1. **Exactitud o `Accuracy`**: Es la medida de rendimiento más simple e intuitiva. Representa la relación entre las predicciones correctas sobre el total de las predicciones realizadas.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

   Podemos pensar que un modelo con mayor exactitud que otro es mejor. Eso sería así en el caso de que el número de elementos de cada clase es el mismo, es decir, el corpus está balanceado. En caso contrario, es necesario hacer uso de otro tipo de medidas como la precisión o la exhaustividad (valoran el rendimiento sobre clases individuales).

2. **Precisión**: Es la razón entre el número de documentos clasificados correctamente como pertenecientes a la clase C y el número total de documentos que han sido clasificados por el modelo como de clase C. Mide la proporción de identificaciones positivas que son realmente correctas.

$$Precision = \frac{TP}{TP + FP}$$

3. **Exhaustividad o *Recall***: Es la razón entre los documentos clasificados correctamente como pertenecientes a la clase C y la suma de todos los documentos de la clase C. Se puede entender como la capacidad que tiene el modelo de construir de manera correcta las clases.

$$Recall = \frac{TP}{TP + FN}$$

4. **Media armónica ponderada o F1 score**: Puede ser interpretado como un promedio ponderado de la *precisión* y el *recall*, donde el rango de valores oscila entre 0 y 1.

$$F1 = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}} = \frac{2 * (Recall * Precision)}{Recall + Precision}$$

   Se ayudará a penalizar los valores muy bajos en una u otra medida.

Comentar que los algoritmos anteriormente nombrados serán entrenados mediante **validación cruzada** o *cross-validation*, de forma que se reducirá la dependencia entre la partición de los datos usada para la fase de entrenamiento y test.

Se dividirá el conjunto de documentos o *corpus* en 5 particiones de iugal tamaño, de forma que que 4 serán para entrenar el modelo y el sobrante para la evaluación. Esto se repetirá 5 veces de forma que al final se promediará el valor de las métricas para tener una medida más robusta.

Capítulo 3

Estudio experimental

3.1. Empleo de métodos de análisis de sentimientos preentrenados (SAMs)

3.1.1. VaderSentiment

3.1.2. TextBlob

3.2. Creación de un propio método de análisis de sentimientos

La construcción de un clasificador de documentos de texto basado en un sistema de aprendizaje automático consta de varias etapas. En primer lugar es necesario preparar los datos del *corpus* antes de pasárselos a los algoritmos. Se deben limpiar y normalizar de forma que se reduzca o eliminen datos innecesarios para el aprendizaje o que puedan influir de manera negativa en éste.

A continuación cada uno de los tweets se someterá a un proceso de **tokenización**, el cual dividiremos el texto en unidades de significado conocidas como *tokens* (suelen ser palabras).

A partir de ellos se extraerán las características que representen a los mensajes y se aplicarán métodos para reducir su número, como podría ser la aunación de términos morfológicamente similares para reducir su número. Finalmente se ponderarán las características en función de la importancia que se les quiera dar y con ellas se entrenarán los clasificadores.

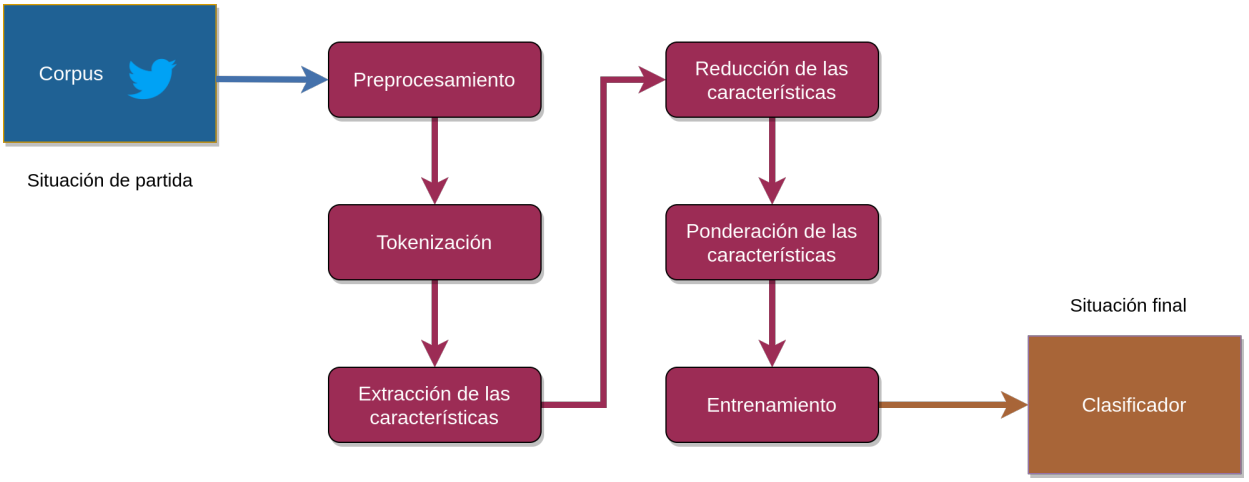

Figura 3.1: Fases para la creación del clasificador

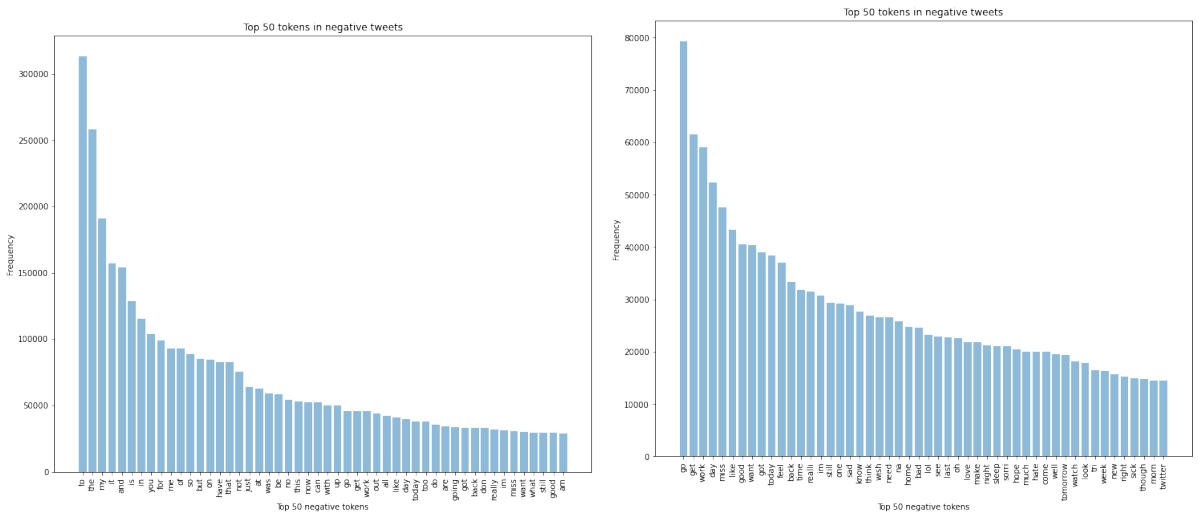

Figura 3.2: Top 50 términos negativos antes y después de preprocesar

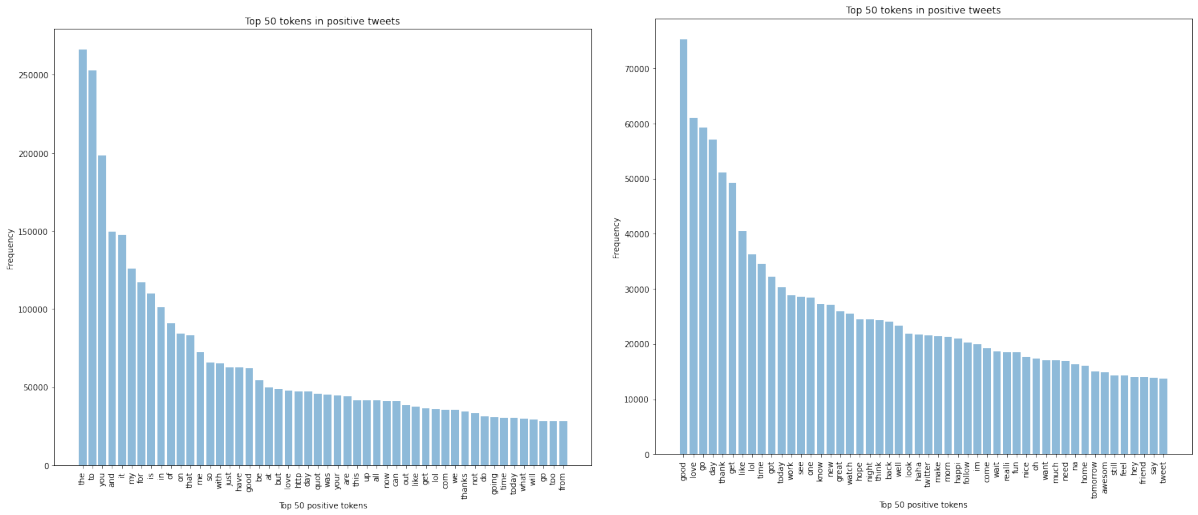

Figura 3.3: Top 50 términos positivos antes y después de preprocesar

3.2.1. Preprocesamiento

3.2.2. Tokenización

3.2.3. Extracción de las características

3.2.4. Reducción de las características

3.2.5. Ponderación de las características

3.3. Comparación y análisis de los modelos empleados

3.4. Modelos más avanzados

Capítulo 4

Trabajo Futuro