Clasificación de Perfiles Políticos
Aplicaciones de la Lingüística Computacional

Raül Fabra Boluda

1 Introducción

El problema planteado en este proyecto ha consistido en la clasificación de usuarios de *Twitter* según su perfil político. Por simplificación, se asume que los usuarios pueden ser sólo de izquierdas o de derechas (más concretamente, PSOE o PP). Para ello se ha construido un clasificador que usa como información únicamente el contenido de los *tweets* escritos por el usuario.

Como el clasificador se basa únicamente en las características lingüísticas de los *tweets*, se pueden aplicar las técnicas empleadas para *Author Profiling* o *Sentiment Analysis*. Esto incluye la limpieza del texto y la representación de los *tweets* como bolsas de n-gramas. Como método de clasificación se ha optado por combinar PCA para reducir la dimensionalidad con clasificación por el vecino más próximo, basado en la distancia euclidiana.

En las siguientes secciones se describe el corpus utilizado, el la experimentación llevada a cabo y un análisis de los resultados obtenidos.

2 Experimentación

2.1 Preproceso

2.1.1 Tokenización y Limpieza del Corpus

Antes de crear los modelos de n-gramas, resulta conveniente realizar un proceso de tokenización y limpieza, ambos llevados a cabo con NLTK.

El proceso de limpieza ha consistido en la eliminación de las *stopwords* y en *lemmatizar* el texto, con tal de agrupar las diferentes formas de inflexión de las palabras en un único ítem. No se ha prestado ningún trato especial a los *hashtags* (#*etiqueta*) ni a las menciones (@*usuario*).

2.1.2 Representación de los datos: Modelos de n-gramas

Para representar los datos se ha decidido emplear la técnica de los n-gramas, más concretamente se han usado tri-gramas.

Para todos los *tweets* de todos los usuarios del conjunto de *training* se obtiene un modelo global con todos los tri-gramas posibles y su frecuencia de aparición. Esto da lugar a un modelo que contiene unos tres millones de tri-gramas.

Como la inmensa mayoría de estos tri-gramas aparecen muy pocas veces no resultan tan relevantes. Por tanto, parece buena idea realizar un filtrado de este modelo conservando sólo aquellos tri-gramas que aparezcan con algo más de frecuencia.

Para determinar la mejor frecuencia se hace un barrido, conservando aquellos tri-gramas que aparezcan con una frecuencia $> f$. Este modelo global es importante porque los tri-gramas que sobrevivan serán los que se tengan en cuenta a la hora de codificar como tri-gramas los *tweets* de los usuarios.

Para los experimentos posteriores se ha filtrado el modelo global como se indica en la Tabla 1.

Frecuencia f	Número de tri-gramas
5	43497
10	15075
15	8467
20	5656

Table 1: *Número de tri-gramas resultantes al aplicar un filtrado, conservando aquellos tri-gramas que aparecen con una frecuencia* $> f$.

2.2 Reducción de la Dimensionalidad y Clasificación

Además del filtrado por frecuencia previamente explicado, también se puede reducir la dimensionalidad mediante PCA. Con ello se proyectan los datos sobre 10, 20, 30,...,150 dimensiones y se prueba a clasificar.

Para la clasificación se ha decidido emplear la técnica del vecino más cercano usando distancia euclidiana.

La clasificación se ha llevado a cabo de dos formas distintas:

Por frecuencia. Cada dimensión representa el número de veces que el usuario ha escrito en tri-grama correspondiente.

Por aparición. Cada dimensión puede tomar valor 1 o 0 dependiendo de si el tri-grama ha sido escrito o no por el usuario, al menos una vez.

3 Corpus

El corpus proporcionado está compuesto por usuarios de *Twitter* que son afines al PP o al PSOE. En este corpus hay 507 usuarios del PP y 848 del PSOE, un total de 1355 usuarios. De cada usuario se tiene hasta mil *tweets*, en total 990864 *tweets*.

Para desarrollar este trabajo el corpus proporcionado ha sido dividido en tres partes: *training*, *development* y *test*.

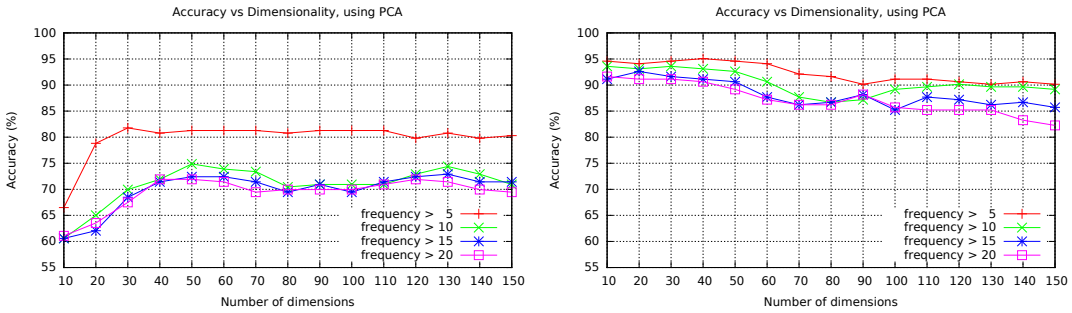
Para el conjunto de *training* se han empleado el 60% de las muestras, para el de *development* el 15% y para *test* el 25% restante. Para la construcción de estas particiones se han escogido al azar usuarios de ambos partidos políticos en esas mismas proporciones. Por ejemplo, el conjunto de *training* contiene el 60% de las muestras del PP y el 60% de las muestras del PSOE.

La primera tanda de experimentos utiliza solamente las particiones de *training* y *development*, usada para determinar como se comporta el clasificador en función de los parámetros para los que hay que hacer un barrido.

Una vez conocido el comportamiento, se repite la experimentación entrenando el sistema con los conjuntos de *training* y *development* y probando a clasificar con el conjunto de *test*.

4 Resultados

Las Figuras 1 y 2 muestran los resultados obtenidos.



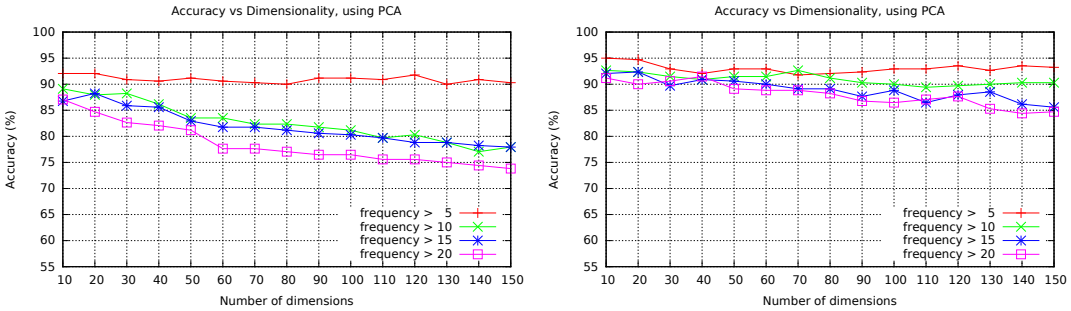
(a) *Clasificación según frecuencia.* (b) *Clasificación según aparición.*

Figure 1: *Resultados de clasificación obtenidos entrenando con el conjunto de training y testeando con el de development.*

En el caso de la Figura 1 se observa que considerando sólo la aparición de los tri-gramas en lugar de la frecuencia el clasificador funciona bastante mejor, llegando a obtener un *Accuracy* cercano al 95%. Además, en este caso el clasificador funciona bastante mejor con dimensionalidades bajas, entre 10 y 40.

En el caso de la Figura 1a se observa que el resultado es más sensible a la frecuencia de filtrado de los tri-gramas que en el caso de la 1b. Los resultados muestran que, cuanto más baja es la frecuencia de filtrado (mayor número de tri-gramas en el modelo) el sistema funciona mejor. Esto ocurre porque a medida que el filtrado es más restrictivo, tienden a conservarse siempre los mismos tri-gramas para ambos partidos, lo que dificulta la discriminación. Los tri-gramas que más tienden a sobrevivir son los *hashtags* y las menciones.

El hecho de que funcione mejor con aparición que con frecuencia puede deberse a que el hecho de manifestar una idea es más relevante para la discriminación que el número de veces que se repita esa misma idea. Esto tiene sentido, ya que si un alguien dice por ejemplo "hay que prohibir el aborto" inmediatamente somos capaces de identificar su tendencia política, mientras que el número de veces que lo diga no es tan relevante.



(a) *Clasificación según frecuencia.* (b) *Clasificación según aparición.*

Figure 2: *Resultados de clasificación obtenidos entrenando con los conjuntos de training y development y testeando con el de test.*

En la Figura 2 se observa un comportamiento análogo al mostrado en la Figura 1. La principal diferencia es que en global los resultados son mejores (aunque el máximo es el mismo), probablemente debido a un mayor número de muestras de entrenamiento.