

# Asignatura Text Mining II

## Máster/Diploma Big Data Analytics

Pedro Henrique Mano Figueiredo Fernandes  
pedromorfeu@gmail.com

### Abstract

La aproximación al problema de *Text Mining* para *Author Profiling* ha tenido como base la técnica conocida por *bag of words*. Esta técnica consiste en el aprendizaje del vocabulario de un conjunto de documentos, a través del que se construye una representación de los datos en forma de matriz numérica, adecuada para aplicar *machine learning*. A la matriz de vocabulario se han añadido otras características, como contadores de polaridad de sentimientos y determinadas estadísticas del texto de los documentos. El *metadata* de los tweets se ha usado también para reforzar las características de la matriz. El modelo usado para el aprendizaje del vocabulario se basa en contadores de palabras, calculando un coeficiente de tipo TF-IDF. Se ha configurado el modelo con un máximo de 2000 características, que se traduce en el cálculo de las 2000 palabras ms significativas en el corpus de entrenamiento. Para la división de los documentos en trozos (*tokens*) se han aplicado *tokenizers* especializados en texto de Twitter. Una vez construida la matriz con todas las características, se ha elegido un clasificador de tipo *Random Forest* para entrenar un modelo matemático. Este clasificador es de tipo *ensemble*, aplicando varias iteraciones de predicción sobre conjuntos aleatorios de los datos, lo que garantiza una mejor robustez y generalización del modelo.

## 1 Introducción

La exploración de datos de language natural permite descubrir características de los au-

tores basadas en sus patrones de escrita. El objetivo de este ejercicio es explorar la información de un dataset constituido por *tweets* de varios usuarios de distintos países de habla hispanica, con el intuito de crear un modelo clasificador de perfiles de sexo y país.

Determinados patrones de escrita pueden mostrar indicios de perfiles. Algunos estudios indican que los hombres tienen tendencia a usar más determinantes y adjetivos que las mujeres; y las mujeres suelen usar más pronombres y la negación. En general, las mujeres tienen tendencia a demostrar más carga emocional en sus frases que los hombres. Métricas como el tamaño de las frases pueden eventualmente ser significativas también. Además, hay que tener en cuenta los usuarios corporativos, cuya escrita será diferente de la de otros perfiles, probablemente más formal.

De la misma forma, los perfiles de países obedecen a patrones de escrita, con elementos significativos como el uso de modismos y otras variaciones lingüísticas.

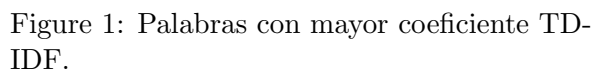
El corpus de texto se usa para aprender el vocabulario significativo y luego aplicar técnicas de *machine learning*, que se encargarán de encontrar patrones y correlaciones para generar un modelo matemático. Los datos sirven como materia prima, a la que se aplican herramientas para crear nuevo conocimiento.

## 2 Dataset

El dataset de este ejercicio está constituido por cientos de *tweets* de usuarios de 7 países de habla hispanica. Cada país tiene *tweets* de 650 usuarios y cada usuario tiene entre 600 y 1000 *tweets*.

El conjunto de entrenamiento cuenta con datos de 2730 usuarios previamente clasificados en sexo y país. En total hay 2.616.338

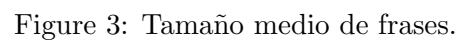
Se ha realizado un breve análisis estadístico sobre la matriz de *bag of words* con R, cuyos resultados se presentan a continuación. El gráfico siguiente representa las palabras y símbolos con mayor coeficiente TF-IDF. Se verifica que las palabras más significativas son también las más usuales - este dato será relevante en el desarrollo del estudio propuesto.



A boxplot titled 'data\$positive + data\$negative' on the y-axis and 'data\$sex' on the x-axis. The y-axis has major ticks at 0, 1000, 2000, and 3000. The x-axis has three categories: 'female', 'male', and 'UNKNOWN'. For 'female', the median is approximately 800, with the box spanning from 600 to 950. For 'male', the median is approximately 750, with the box spanning from 550 to 950. For 'UNKNOWN', the median is approximately 750, with the box spanning from 600 to 900. Outliers are present for all categories, with 'male' having the most outliers, including one near 2900.

Figure 2: Sentimiento.

sante que el tamaño medio de las frases de mujeres y desconocido (*UNKNOWN*) es parecido:



The plot displays the distribution of 'profile\_sidebar\_fill\_color' for three categories of 'data\$sex': female, male, and UNKNOWN. The y-axis represents the 'profile\_sidebar\_fill\_color' values, ranging from 0.0e+00 to 1.5e+07. A legend on the right indicates the color scale for 'profile\_sidebar\_fill\_color', ranging from 0.0e+00 (light gray) to 1.6e+07 (dark gray). The distributions for 'female' and 'male' are similar, with most values concentrated between 0.0e+00 and 5.0e+06, and a tail extending up to 1.5e+07. The 'UNKNOWN' category shows a similar distribution but with a slightly higher density of values in the 5.0e+06 to 1.0e+07 range.

Figure 4: Colores del perfil de Twitter.

### 3 Propuesta del alumno

La técnica *bag of words* ha dado un *baseline* para acercar el problema. Esta técnica se basa sencillamente en contadores de palabras, en su concepto. Sin embargo, se puede

customizar la forma de contar (acumulador, cuoficiente, binario), la forma de separar los componentes del texto, el número máximo de palabras del vocabulario o conjuntos de varias palabras. Las decisiones de configuración se describen a continuación.

Se ha usado un *tokenizer* especializado en texto de *tweets* (`TweetTokenizer`, del paquete *NLTK*), que ofrece opciones de ignorar referencias a otros usuarios y reducir las repeticiones de texto. Se ha usado también la opción de tratamiento *case-sensitive*, por poder ser significativo en la distinción de género.

La transformación de los datos en características numéricas adecuadas para *machine learning* es un área conocido por *feature extraction*; en el caso de *bag of words* el proceso es de vectorización, generando un vector de características. Para esta tarea se ha usado la clase `TfidfVectorizer` de *scikit-learn*, que implementa la separación en *tokens* y el cálculo de frecuencias, así como la normalización y atribución de pesos de importancia a las palabras. Se ha configurado el modelo para generar una matriz de características de cuoficientes TF-IDF.

El aprendizaje del vocabulario es una tarea intensiva para CPU y memoria, directamente proporcional al número de características/palabras del vector generado. Se ha configurado el modelo para un máximo de 2000 palabras, que presentaba rendimiento y resultados aceptables. Sin embargo, se han realizado pruebas y se ha verificado que los resultados de predicción son mejores con mayor número de características, a coste de mayor tiempo de ejecución. Otra configuración importante es el rango de *ngrams*, que son conjuntos de palabras consecutivas. Se ha aplicado el rango de 1 a 2 palabras, con lo que se han mejorado los resultados.

Determinadas palabras son muy comunes y aparentemente aportan poco significado, como es el caso de artículos y preposiciones, como “de”, “la”, “para”. Estas palabras se llaman *stop words*. La clase `TfidfVectorizer` ofrece la posibilidad de usar una lista de *stop words* y el paquete *NLTK* tiene listas pre-determinadas para varias lenguas. Se ha probado con esa configuración y no se han obtenido mejoras, probablemente porque son palabras significa-

tivas en la escrita de hombres y mujeres - por ejemplo, las mujeres usan más artículos que los hombres. Así que se ha decidido no usar *stop words*.

Con el vector numérico generado y los datos de training etiquetados para sexo y país, ya se pueden aplicar algoritmos de aprendizaje supervisado. Para esta tarea, se ha usado un clasificador de tipo *Random Forest*, en particular `RandomForestClassifier` de *scikit-learn*. Este clasificador es una técnica de *ensemble*, que entrena varios árboles de decisión con sub-conjuntos del *dataset* para así obtener un mejor acierto y prevenir el sobre-ajuste.

## 4 Resultados experimentales

Resultados obtenidos con un vocabulario de 2000 características y *Random Forest* de 100 árboles:

	Sexo	País
Acierto	52.69%	93.46%

Table 1: Resultados de predicción.

Los tiempos de ejecución en una máquina con 4GB de RAM y procesador i3 han sido:

Operación	Tiempo
Leer <i>tweets</i> de training	5 min.
Cálculo TF-IDF de training	10 min.
Características extra de training	4 min.
Leer <i>tweets</i> de test	4 min.
Cálculo TF-IDF de test	4 min.
Características extra de test	4 min.
<i>Random Forest</i>	1 min.
<b>Total</b>	<b>32 min.</b>

Table 2: Tiempos de ejecución.

Los resultados obtenidos para la predicción de género de los usuarios han sido peores que para la predicción del país. Se verifica así que las variaciones de la lengua castellana en los distintos países son más significativas que las variaciones entre hombre y mujeres. La presencia de la etiqueta *UNKNOWN* (instituciones, por ejemplo) puede dificultar esta tarea.

El vocabulario con *stop words* (palabras comunes) mejora un poco los resultados - como descrito anteriormente, puede ser significativo

para distinguir la escrita de hombres y mujeres.

La inclusión de información extra (sentimiento y metadata) en el vector de características ha tenido efectos considerables. En la predicción de sexo, por ejemplo, la mejora fue de la orden del 10%.

El método de *Random Forest* ha demostrado mejores resultados respecto a otros clasificadores, como gaussianas. El hecho de ser una técnica de *ensemble* contribuye bastante, al entrenar varios árboles con sub-conjuntos de datos. Sin embargo, esta técnica exige más tiempo de ejecución.

## 5 Conclusiones y trabajo futuro

*Text Mining* para *Author Profiling* es un área con muchas aplicaciones prácticas, que pueden ir desde la determinación del género a la detección de fraude. Es interesante verificar como las técnicas de *machine learning* permiten descubrir patrones en la forma de escribir, que pueden dar mucha información. De esa forma se consigue nuevo conocimiento, al que se puede aplicar inteligencia de decisión.

El procesamiento de lenguaje natural puede ser una tarea compleja. Hay variaciones en la escrita de la misma persona, errores ortográficos o palabras con repeticiones para dar énfasis (por ejemplo, “noooooo”); los sentimientos manifestados pueden ser ambíguos; la ironía es difícil de detectar.

La predicción de sexo no tiene tan buenos resultados como la de país. Probablemente el análisis de sentimientos podría ayudar en ese sentido, por lo que necesitaría una explotación más profunda. Igualmente, otros elementos de metadata podrían haber sido estudiados y utilizados.

Respecto al metadata, los campos `name` y `description` se han incluido en el texto que se pasa al *vectorizer*, esperando que este encontrase alguna correlación con el género del usuario. Sin embargo, esta implementación no es correcta, una vez que esos campos serán tratados como texto y probablemente no tienen significado suficiente para entrar en la matriz. Se ha verificado que la mejora no ha sido significativa. En una implementación futura, estos campos se podrían usar como características extra, para asegurar que se usan

en la clasificación.

Quedan algunas dudas por resolver respecto a la técnica de acumulación de todo el texto de los *tweets* de cada usuario en un documento único. Potencialmente, la cualidad del vocabulario y de los coeficientes TF-IDF están comprometidos con esta solución, una vez que se está considerando que el texto es de un solo documento.

El uso de mayor número de características de vocabulario podría ser una mejora futura, para lo que haría falta mayor capacidad de computo.

Hay otros clasificadores de *supervised learning* que podrían aportar mejoras en el ajuste a este tipo de datos. Los métodos de *Support Vector Machines* serían buenos candidatos. Estos métodos usan hiperplanos para separar los datos, son adecuados para espacios dimensionales grandes y versátiles a través de sus *kernels* de ajuste a los datos.

## References

- Kaggle. *Part 1: For Beginners - Bag of Words* <https://www.kaggle.com/c/word2vec-nlp-tutorial/details/part-1-for-beginners-bag-of-words>
- Scikit-learn. *1. Supervised learning* [http://scikit-learn.org/stable/supervised\\_learning.html#supervised-learning](http://scikit-learn.org/stable/supervised_learning.html#supervised-learning)
- NLTK 3.0 documentation. *nlk.tokenize package* <http://www.nltk.org/api/nltk.tokenize.html>