Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Frequency Analysis Interpolation (FAI). Un método de Representación de Textos de Baja Dimensionalidad para Problemas de Author Profiling en Entornos Big Data.

Trabajo Fin de Máster

Máster en Big Data Analytics

Autor: [Òscar Garibo i Orts]

Tutor: [Francisco Manuel Rangel Pardo]

[2016-2017]

Els meus agraiments:

A Juan i Mary, per educar-me en llibertat.

A Pilar, ésto no hubiera sido posible sin tus ánimos y apoyo.

A Guillem, t'estime guapetona.

A Moisès.

Als qui no es van rendir mai, als qui encara no es rendeixen.

Resum

L'objecte d'aquest Treball Final de Màster és l'exploració de nous mètodes de representació de baixa dimensionalitat de textes que permeten abordar problemes de perfilat d'autors (*Author Profiling*) en entorns Big Data, entenent el concepte en les seues quatre vessants: volum, velocitat, varietat i valor. Amb tal finalitat hem proposat el mètode *Frequency Analysis Interpolation* (FAI) i l'hem comparat amb mètodes de l'estat de l'art en problemes d'*Author Profiling*: *Low Dimension Statistical Embeddings* (LDSE) i Word2Vec.

Hem optat per una representació dels textes per la freqüència d'aparició de les paraules, ja que les freqüències són fàcilment concebudes com a probabilitats. A partir d'aquesta representació bàsica hem calculat estadístics tals com la mitja, desviació típica i coeficient d'asimetria o *skewness*.

Una anàlisi del coeficient d'asimetria per a un problema de classificació entre dues classes ens mostra la seua rellevància i valor com a característica. L'estudi dels valors de les probabilitats a priori de pertenència a cadascuna de les classes ens mostra que la distribució es troba desplaçada de la mitja i és per açò que afegim com a característiques el nombre de paraules del vocabulari les probabilitats a priori de pertenència a cadascuna de les classes dels quals pertanya a cadascun dels tres tercils

Definim FAI com un nou mètode de representació de baixa dimensionalitat de text. Realitzem experiments amb sis corpora diferents i comparem els resultats obtinguts amb FAI amb els resultats de LDSE i Word2Vec. Finalment, emprem FAI com a mètode de representació de textes en tres tasques internacionals d'*Author Profiling* i establim que FAI és un mètode de representació competitiu i que aporta avantatges ja que capta les diferències basades en paraules típiques de les diferents classes.

Paraules clau: Author Profiling, FAI, baixa dimensionalitat.

Resumen

El objeto de este Trabajo Final de Máster es la exploración de nuevos métodos de representación de baja dimensionalidad de textos que permitan abordar problemes de perfilado de autores (*Author Profiling*) en entornos Big Data, entendiendo el concepto en sus cuatro vertientes: volumen, velocidad, variedad y valor. Para ello hemos propuesto el método *Frequency Analysis Interpolation* (FAI) y lo hemos comparado con métodos del estado del arte en problemas de *Author Profiling*: *Low Dimensionality Statistical Embeddings* (LDSE) y *Word Embeddings* (Word2Vec).

Hemos optado por una representación de los textos por la frecuencia de aparición de las palabras, ya que las frecuencias son fácilmente concebidas como probabilidades. A partir de esta representación básica hemos calculado estadísticos tales como la media, desviación típica y coeficiente de asimetría o skewness.

Un análisis del coeficiente de asimetría para un problema de clasificación entre dos clases nos muestra su relevancia y valor como característica. El estudio de los valores de las probabilidades a priori de pertenencia a cada una de las clases nos muestra que la distribución resultante se encuentra desplazada de la media. Es por esto que añadimos como características el número de palabras del vocabulario cuyas probabilidades a priori de pertenencia a cada una de las clases pertenezca a cada uno de los tres terciles.

Definimos FAI como un nuevo método de representación de baja dimensionalidad de texto. Realizamos experimentos en seis corpora diferentes y comparamos los resultados obtenidos con FAI con los resultados de LDSE y Word2Vec. Finalmente, utilizamos FAI como método de representación de texto en tres tareas internacionales de *Author Profiling* y establecemos que FAI es un método competitivo y que aporta ventaja en cuanto a que capta particularidades en las palabras típicas de cada una de las clases.

Palabras clave: Author Profiling, FAI, baja dimensionalidad.

Abstract

The goal of this master thesis is the exploration of new low dimensionality representation methods for texts that allow to face Author Profiling tasks in a Big Data environment, understanding the term in its four slopes: volume, velocity, variety and value. To do so we here propose the method Frequency Analysis Interpolation (FAI) and compare it with state of the art methods in Author Profiling: LDSE(*Low Dimension Statistical Embeddings*) and Word2Vec.

We have chosen to represent texts by the words frequency, because frequencies are easily thought of as probabilities. From this basic representation we have computed statistics such as mean, standard deviation and skewness.

An analysis of the skewness in a two classes classification problem show its importance and value as a characteristic. The study of the values of the a priori probabilities of belonging to each of the classes shows us that the resulting distribution is skewed from the mean. So, the number of words in the vocabulary which belong to each of the three tertiles of the a priori probabilities of belonging to each class are added as characteristics.

FAI is defined as a new low dimensional representation method for text. Experiments in six different corpora are performed and FAI's results are compared to the results obtained by using LDSE and Word2Vec. Finally, we took part in three international Author Profiling tasks and it is established that FAI is a competitive method which catches particularities in the typical words of every class.

Keywords : Author Profiling, FAI, low dimensionality.

Tabla de contenidos

1. Introducción....16

2. Métodos de representación en *Author Profiling*....18

2.1. Low Dimension Statistical Embeddings (LDSE)....19

2.2. Word2Vec....20

3. Frequent Analysis Interpolation....20

3.1. Clasificación binaria....20

3.2. Clasificación multiclase....24

3.3. Frequency Analysis Interpolation....26

4. Marco de evaluación....30

4.1. Corpora....30

4.1.1. Corpus CMUQ-ARAP....30

4.1.2. Corpus PAN-AP'17....31

4.1.3. Corpus PAN-AP'18....32

4.1.4. Corpus PAN-AP'14....32

4.1.5. Corpus PAN-AP'13....35

4.1.6. Corpus RUSProfiling....35

4.2. Algoritmos de clasificación....36

4.2.1. Máquinas de Vector Soporte....36

4.2.2. Naïve Bayes....37

4.2.3. Perceptrón multicapa....37

4.2.4. Árboles de decisión....38

4.2.5. Random Forest 39

4.3. Medidas de evaluación....39

4.4. Test de significación estadística....39

4.5. Preprocesamiento....40

5. Resultados experimentales.....42

5.1. Identificación de variedad del lenguaje.....42

5.2. Identificación de género.....43

5.3. Identificación de edad.....45

6. Participación en tareas internacionales.....46

6.1. Author Profiling en PAN@CLEF 2018.....46

6.2. MAPonSMS en FIRE 2018.....47

6.3. HatEval en SemEval 2019 49

7. Conclusiones y trabajo futuro.....52

8. Referencias.....54

Anexo 56

Índice de tablas

Tabla 3.1.1. Resultados de clasificación del corpus de entrenamiento en PAN-AP'18 en base al signo del *skewness*.....24

Tabla 4.1.1. Corpus CMUQ-ARAP.....30

Tabla 4.1.2. Corpus PAN-AP'17.....31

Tabla 4.1.3. Corpus PAN-AP'18.....32

Tabla 4.1.4. Corpus PAN-AP'14.....34

Tabla 4.1.5. Corpus PAN-AP'13.....35

Tabla 5.1. Resultados en variedad del lenguaje para los corpora CMUQ-ARAP y PAN-AP'17.....42

Tabla 5.2. Resultados en identificación de género para los corpora CMUQ-ARAP, PAN-AP'18, PAN-AP'17, PAN-AP'14 tanto en Tweets, como Blogs y Social Media, PAN-AP'13 y RUS Profiling.....44

Tabla 5.3. Resultados en identificación de rango de edad para los corpora CMUQ-ARAP, PAN-AP'14 y PAN-AP'13.....45

Tabla 6.1.1. Corpus PAN-AP'18.....46

Tabla 6.1.2. Resultados en identificación de género para el corpus PAN-AP'18.....47

Tabla 6.1.3. Tiempos de cálculo requeridos por el método.....47

Tabla 6.2.1 Distribución de clases en el corpus de entrenamiento de la tarea MAPonSMS para la identificación de género y rango de edad.....48

Tabla 6.2.2. Resultados obtenidos en MAPonSMS y comparación respecto al *baseline* proporcionado.....48

Tabla 6.3.1. Corpus HatEval.....49

Tabla 6.3.2. Distribución de clases en el corpus HatEval.....50

Tabla 6.3.3. Etiquetado para el aprendizaje automático.....50

Tabla 6.3.4. Resultados obtenidos y comparación respecto a los *Baselines* proporcionados.....51

Índice de figuras

Figura 2.1 Esquema de contrucción del vector wt.....21

Figura 3.1.1. Distribución de las probabilidades a priori de pertenencia a las clases hombre y mujer cuando el texto pertenece a la clase hombre.....23

Figura 3.1.2. Distribución de las probabilidades a priori de pertenencia a las clases hombre y mujer cuando el texto pertenece a la clase mujer.....24

Figura 3.2.1. Distribución de las probabilidades a priori de pertenencia a las clase Palestine y resto de variedades cuando el texto pertenece a la clase Palestine.....25

Figura 3.3.1. Matriz TF de un corpus con n autores y m palabras en el vocabulario.....27

Figura 3.3.2. Vector de frecuencias de términos por autor.....27

Figura 3.3.3. Vectores de probabilidades a priori de pertenencia a cada clase de los textos de cada autor.....27

Figura 3.3.4. Cambio de representación en FAI.....28

Figura 4.2.3.1. Esquema de un Perceptrón Multi-capa.....38

Figura A.1. Distribución de las probabilidades a priori de pertenencia a la clase Argelia y resto de variedades cuando el texto pertenece a la clase Argelia.....56

Figura A.2. Distribución de las probabilidades a priori de pertenencia a la clase Egipto y resto de variedades cuando el texto pertenece a la clase Egipto.....56

Figura A.3. Distribución de las probabilidades a priori de pertenencia a la clase Irak y resto de variedades cuando el texto pertenece a la clase Irak.....57

Figura A.4. Distribución de las probabilidades a priori de pertenencia a la clase Kuwait y resto de variedades cuando el texto pertenece a la clase Kuwait.....57

Figura A.5. Distribución de las probabilidades a priori de pertenencia a la clase Líbano Siria y resto de variedades cuando el texto pertenece a la clase Líbano Siria.....58

Figura A.6. Distribución de las probabilidades a priori de pertenencia a la clase Libia y resto de variedades cuando el texto pertenece a la clase Libia.....58

Figura A.7. Distribución de las probabilidades a priori de pertenencia a la clase Marruecos y resto de variedades cuando el texto pertenece a la clase Marruecos.....59

Figura A.8. Distribución de las probabilidades a priori de pertenencia a la clase Omán y resto de variedades cuando el texto pertenece a la clase Omán.....59

Figura A.9. Distribución de las probabilidades a priori de pertenencia a la clase Qatar y resto de variedades cuando el texto pertenece a la clase Qatar.....60

Figura A.10. Distribución de las probabilidades a priori de pertenencia a la clase Arabia Saudí y resto de variedades cuando el texto pertenece a la clase Arabia Saudí.....60

Figura A.11. Distribución de las probabilidades a priori de pertenencia a la clase Sudán y resto de variedades cuando el texto pertenece a la clase Sudán.....61

Figura A.12. Distribución de las probabilidades a priori de pertenencia a la clase Túnez y resto de variedades cuando el texto pertenece a la clase Túnez.....61

Figura A.13. Distribución de las probabilidades a priori de pertenencia a la clase UAE y resto de variedades cuando el texto pertenece a la clase UAE.....62

Figura A.14. Distribución de las probabilidades a priori de pertenencia a la clase Yemen y resto de variedades cuando el texto pertenece a la clase Yemen.....62

Índice de ecuaciones

Ecuación 2.1. TF-IdF.....19

Ecuación 2.2. Term Frequency.....19

Ecuación 2.3. Inverse Document Frequency.....19

Ecuación 2.4. Puntuación por clase.....20

Ecuación 2.5. Representación de un documento.....20

Ecuación 2.6. Vector correspondiente a una palabra.....21

Ecuación 3.3.1. Vector de frecuencia de términos por clase.....27

Ecuación 3.3.2. Vector de frecuencia de términos para todas las clases.....27

Ecuación 3.3.3. Vectores de probabilidades a priori de pertenencia a cada una de las k clases para cada palabra.....28

Ecuación 4.2.2.1. Naïve Bayes.....37

Ecuación 4.2.3.1. Función de la capa oculta de un PMC de una sola capa.....38

Ecuación 4.3.1. Definición de *accuracy*.....39

Ecuación 4.4.1. *Accuracies* de dos experimentos.....40

Ecuación 4.4.2. Test estadístico.....40

Ecuación 5.2.1. Fórmula utilizada para el ranking en la tarea RUS-Profiling.....44

1. Introducción

La forma en que se utiliza el lenguje nos puede ayudar en la tarea de discriminar entre diferentes clases de autores que utilizan el mismo idioma. La utilización del mismo idioma en diferentes lugares suele producir modismos, vocablos frecuentes, formas verbales y otras características que podemos usar para clasificar los autores conforme a diferentes criterios. *Author Profiling*, o perfiles de autor, es la disciplina que tiene por objeto el estudio del uso del lenguaje para la identificación de ciertos rasgos del autor que lo escribe,

Por ejemplo, en todos los países de habla hispana se comparte un mismo idioma, el español. Pero no todos los hispanoparlantes utilizan los mismos vocablos, ni conjugan igual los verbos. Incluso las mismas palabras pueden tener muy diferentes significados y por ende usos.

La realización de perfiles de autores es una tarea de gran utilidad hoy en día, debido al extendido uso de las redes sociales. Si somos capaces de realizar perfiles de autor para clasificar a los autores obtendremos una información que puede ser muy interesante a empresas, administraciones, gobiernos, cuerpos de seguridad, entre otros.

Una empresa podrá analizar los comentarios de autores acerca de un nuevo producto para segmentarlos por ubicación. De forma que aunque todos los comentarios están escritos en un mismo idioma común para diferentes países o zonas geográficas, se podrá diferenciar entre los comentarios de cada origen. Esta información sería muy útil a la hora de preparar campañas de publicidad, por ejemplo.

Por otro lado, si se detectan amenazas en redes sociales, el hecho de poder segmentar correctamente a los autores podrá acotar la búsqueda y proporciona información que puede ser muy valiosa. Por ejemplo, varón de entre 35 y 45 años de edad y de determinado origen.

De la misma manera, se podría analizar el contenido de chats en los que participen menores para identificar posibles predadores sexuales, ya que su uso del lenguaje diferirá del que hacen los niños o adolescentes, aunque intenten imitarlo.

Todos estos casos tienen en común que parten de la base de analizar cantidades ingentes de información, o Big Data. Por lo que un método que fuera capaz de lidiar con grandes volúmenes de información, con buenas prestaciones tanto en precisión como en tiempo de proceso, sería de gran interés.

1.1. Objetivos.

El objetivo de este trabajo es obtener un método de representación de textos de baja dimensionalidad que permita aproximar de manera eficiente problemas de *Author Profiling* en entornos Big Data, como son las redes sociales.

Se abordan problemas como detección de género, detección de variedad del idioma o detección de rango de edad del autor. Dichos problemas se abordan en diferentes idiomas, español, árabe, portugués, inglés o ruso. Los mayor parte de los conjuntos de datos se obtienen de *workshops* o tareas en los que hay resultados publicados, y se comparan los resultados obtenidos por los métodos considerados estado del arte y los participantes en las respectivas tareas.

1.2. Estructura del proyecto.

En el apartado 2 presentaremos los modelos de representación más utilizados para representar texto, tanto de alta como de baja dimensionalidad. Seguidamente, en el apartado 3, relataremos el proceso que originó la concepción de FAI, el método de representación de texto propuesto. En el apartado 4 definimos 7 corpora que hemos utilizado para medir los resultados obtenidos con nuestra propuesta y los comparamos con los obtenidos por métodos de representación considerados estado del arte, mostrando que el método propuesto mejora los resultados en algunos de los corpora. En el apartado 5, aplicamos un test estadístico para mostrar cuándo las mejoras son estadísticamente significativas. En el apartado 6 describimos nuestra participación en las tareas internacionales *Author Profiling* en PAN@CLEF 2018, MAPonSMS en FIRE 2018 y HatVal en SemEval 2019, y presentamos los resultados obtenidos. Finalmente, en el apartado 7, presentamos las conclusiones del trabajo y establecemos trabajos futuros que pueden ser de interés.

2. Modelos de representación en Author Profiling

Habitualmente se utilizan técnicas de alta dimensionalidad para afrontar problemas de clasificación textual, no siendo diferente el problema de *Author Profiling* en texto. Algunas de ellas son ampliamente conocidas y suelen utilizarse como *baseline* en tareas de evaluación. Algunas de estas técnicas de representación son:

- *Bolsa de palabras*. El modelo Bolsa de Palabras es un método de representación de texto que ignora el orden de las palabras. Primero se calcula el vocabulario de todos los textos y luego se representa el texto de cada autor por las palabras que contiene, independientemente de su frecuencia u orden. Cada texto asume una dimensionalidad igual al número de palabras en el vocabulario utilizado para entrenar. Es un caso particular de N-gramas de palabras cando N=1.

- *N-gramas de palabras*. Es el mismo concepto, pero uniendo n palabras lo que permite captar relaciones entre palabras. La dimensionalidad es como mínimo igual que la de bolsa de palabras.

- *N-gramas de de caracteres*. De nuevo es el mismo concepto pero utilizando caracteres como unidad de información. También se puede considerar bolsa de n-caracteres.

Estas técnicas se utilizan con diferentes esquemas de ponderación:

- Binario: Los textos se representan mediante un vector de longitud igual al número de palabras en el vocabulario. La información que recoge dicho vector es si cada palabra aparece o no en cada texto.

- *Term Frequency* (TF). Cálculo de frecuencias de aparición de cada palabra en el texto de cada autor. Es similar al anterior, pero en lugar de simplemente indicar si una palabra aparece o no en el texto de un autor, se indica cuántas veces aparece. Existen dos variantes, la feecuencia absoluta y la relativa; ésta última normaliza con la longitud del texto, permitiendo equiparar textos de diferentes longitudes.

- *Term Frequency - Inverse Document Frequency* (Tf-Idf). O cálculo de frecuencias de aparición de cada palabra en el texto de cada autor, pero ajustando mediante la frecuencia de aparición de cada palabra en la totalidad del corpus. Este método permite limitar el impacto de

las palabras altamente frecuentes tales como artículos, preposiciones, conjunciones, determinantes, etc, que tendrán poco peso en los valores que se usarán para clasificar.

Dado un corpus D, un documento t y un término t, se define TF-IdF como:

$$\mathrm{tfidf}(t, d, D) = \mathrm{tf}(t, d) \cdot \mathrm{idf}(t, D)$$

Ecuación 2.1. TF-IdF.

- tf es la frecuencia de aparición de cada palabra normalizada por la longitud del documento.

$$tf = Freq_t / long(d)$$

Ecuación 2.2. Term Frequency.

- IdF es la Frecuencia inversa del Documento, que se calcula como el logaritmo natural del cociente del número de documentos y el número de documentos en los que aparece la palabra.

$$\mathrm{idf}(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

Ecuación 2.3. Inverse Document Frequency.

Todas estas técnicas comparten la alta dimensionalidad.

Entre otras disponemos de dos técnicas que limitan el número de características que se usan para modelar a cada autor:

2.1. LDSE.

Low Dimensional Statistical Embeddings (LDSE) [1] pretende reducir la dimensionalidad a seis características por clase. De forma que si queremos clasificar género, acabaremos representando el corpus con doce características por cada autor, frente a las miles que requieren los métodos anteriormente nombrados.

La idea básica consiste en, para cada término, calcular una puntuación por clase que indique con qué confianza el término pertenece a cada una de las clases. En base a esta confianza calculamos un número reducido de características. Para cada clase c y cada término $t \in V$, calculamos la puntuación S(t/c). Esta puntuación se obtiene sumando los pesos obtenidos con el TF-IDF para el término t en los documentos $d \in D$ pertenecientes a la clase c, y se divide entre la suma de los pesos para ese término en todos los documentos.

$$S(t, c) = P\ d\in D/c=\delta(d)\ wdt\ P\ d\in D\ wdt\ ,\ \forall d \in D,\ c \in C$$

Ecuación 2.4. Puntuación por clase.

A partir de estas puntuaciones, representamos un documento d como sigue:

$$d = \{F(c1), F(c2), \ldots, F(cn)\} \sim \forall c \in C$$

Ecuación 2.5. Representación de un documento.

Cada F(ci) representa las seis características listadas a continuación:

1. Media. Dada por la suma de las puntuaciones de los términos del documento, dividida por el número total de términos que contiene.

2. Desviación típica. Calculada como la raíz cuadrada de la suma de todas las puntuaciones menos la media.

3. Puntuación mínima. La mínima de las puntuaciones que aparece en el documento.

4. Puntuación máxima. La máxima de las puntuaciones que aparece en el documento.

5. Puntuación global. La suma de las puntuaciones dividida entre el número total de términos del documento.

6. Proporción. Proporción entre el número de términos del vocabulario que aparecen en el documento y el número total de términos del documento.

2.2. Word2Vec.

Word2Vec [2] también se basa en reducir la dimensionalidad, en este caso a un número de características seleccionado por el analista. *Word2Vec* toma como entrada enormes corpora de texto y produce vectores que caracterizan a las palabras y su contexto. Estos vectores o *word embeddings* se utilizan para representar a las palabras en el texto y son utilizadas para alimentar sistemas de aprendizaje automático. La idea subyacente es que cuanto más se parezcan dichos vectores más se parecen las palabras representadas por dichos vectores. Es decir, se basan en el concepto de semántica distribuida.

Utilizaremos vectores ya entrenados que proporciona Facebook[1]. Dichos vectores se han contruido utilizando un modelo *Character Bag Of Words* (CBOW) que es una extensión del modelo presentado por Mikolov et al. en [3]. El modelo representa las palabras como

1 https://github.com/facebookresearch/fastText/blob/master/docs/crawl-vectors.md

bolsas de n-gramas con ponderación de pesos en función de la posición del n-grama para capturar mejor la información posicional. El objetivo es predecir una palabra dada w_0 basándonos en las palabras del contexto w_{-n},....., w_{-1}, w_0, w_{+1}, ..., w_n, produciendo un vector h que es la media de los vectores correspondientes a los n-gramas incluidos en la palabra u (uwi), con un factor de ponderación C, multiplicado elemento a elemento, que refleja la lejanía del n-grama:

$$h = \sum_{i=n-1/i\neq 0}^{n} ci * uwi$$

Ecuación 2.6. Vector correspondiente a una palabra.

Definimos 5 como el valor preferido para n, con lo que para calcular el vector de una palabra se han tenido en cuenta las 5 palabras anteriores y las 5 posteriores a la palabra a caracterizar, dotando de esta manera de contexto al vector representativo [4]. Las k palabras [w_{n-5},.....,w_{n-1},w_{n+1},.....,w_{n+5}] se proyectan en una capa común, haciendo la media de sus vectores, de forma que utilizamos las palabras que han aparecido con anterioridad y las que aparecen con posterioridad para construir el vector de la palabra w_n.

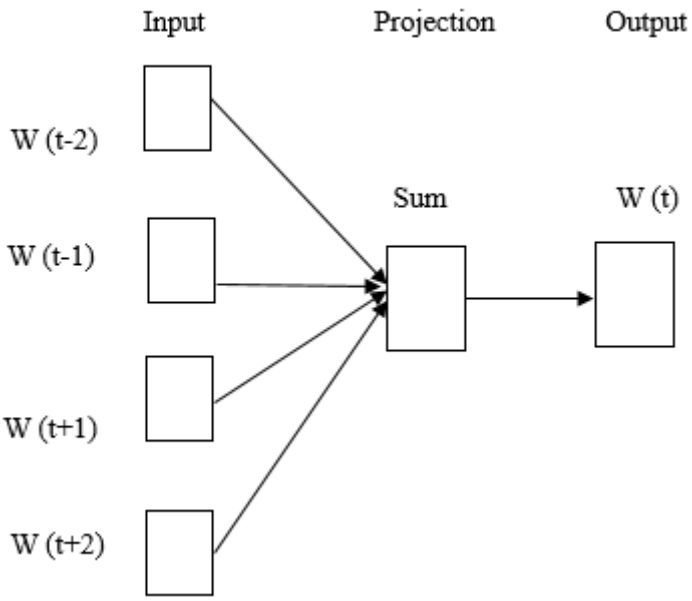

Figura 2.1 Esquema de contrucción del vector w_t.

Utilizaremos los vectores ya entrenados que proporciona Facebook. Utilizaremos una longitud de *embedding* de 300, que es la asumida como estándar por la comunidad. Cada palabra queda representada por un embedding de longitud 300, de forma que para cada autor

calcularemos la media de las i componentes del *embedding*, obteniendo un vector de longitud 300, que caracteriza al autor. Dichos vectores son los que utilizamos para alimentar los sistemas de clasificación.

3. Frequency Analysis Interpolation

Frequency Analysis Interpolation (FAI) es el método de representación de baja dimensionalidad que proponmos en este trabajo, sobre la base de los análisis presentados a continuación.

3.1 Clasificación binaria.

Analizamos alternativas numéricas de representación de textos. Partimos de la idea de LDSE, pero en lugar de calcular Tf-Idf calculamos Term Frequency (TF).

De forma análoga a LDSE, obtenemos un vector por cada clase con las probabilidades a priori de pertenencia de cada palabra del vocabulario a dicha clase. Tras codificar el texto de cada autor calculamos la media, desviación típica y *skewness*. El skewness o coeficiente de asimetría de Fisher, indica cómo de simétrica es una distribución con respecto a su media.

Una inspección visual a los datos obtenidos al representar el corpus de entrenamiento PAN-AP'18 [5] descrito en el siguiente apartado, nos sugiere un patrón en el *skewness*. Observando las figuras 3.1.1 y 3.1.2 comprobamos que el *skewness* de los vectores codificados con la clase *male* es positivo si el autor es hombre en un porcentaje muy alto de los casos, así como negativo en un porcentaje también muy alto de los casos en que es mujer.

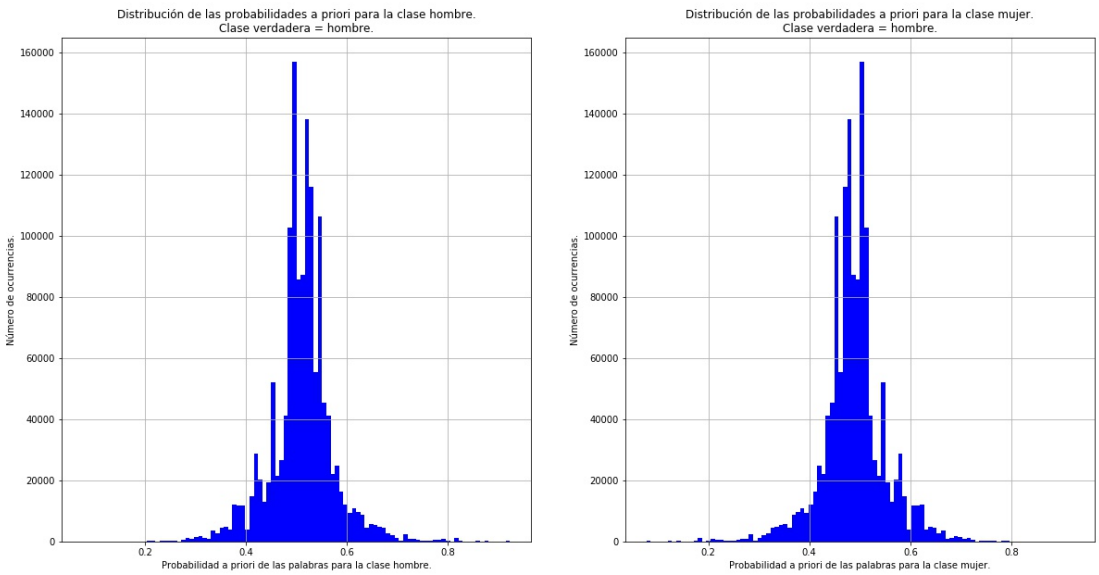

Figura 3.1.1. Distribución de las probabilidades a priori de pertenencia a las clases hombre y mujer cuando el texto pertenece a la clase hombre.

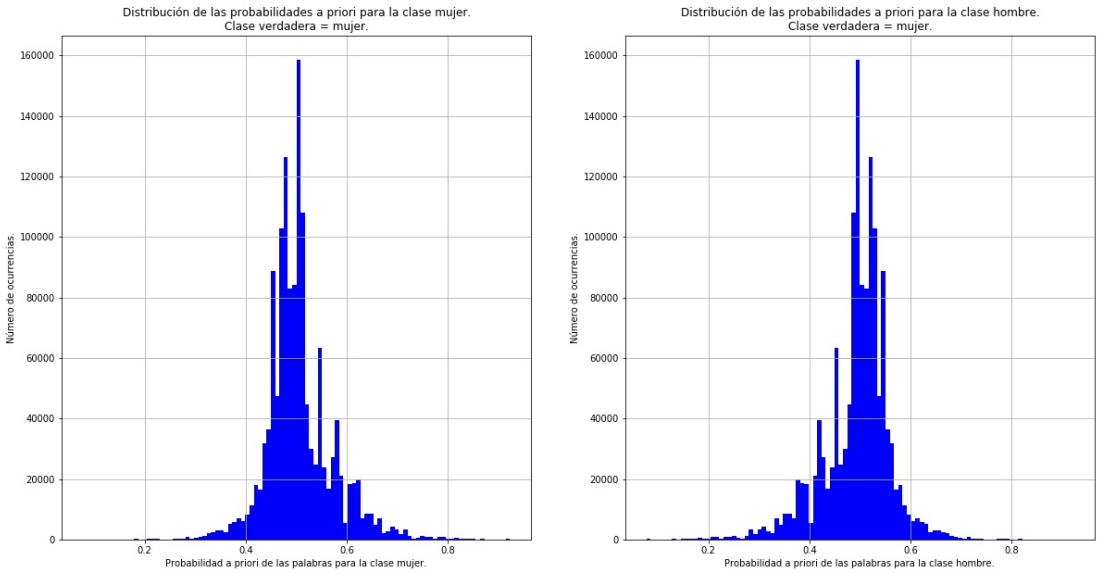

Figura 3.1.2. Distribución de las probabilidades a priori de pertenencia a las clases hombre y mujer cuando el texto pertenece a la clase mujer.

Calculando el skewness para todos los autores de entrenamiento observamos que, en los tres idiomas, más del 95% de los hombres presenta un skewness de signo positivo. Utilizando dicha medida como clasificador del género de los autores [6] obtenemos los resultados de la tabla 3.2.

Idioma	Accuracy
Árabe	95.93%
Español	96.47%
Inglés	96.47%

Tabla 3.1.1. Resultados de clasificación dle corpus PAN-AP'18 en base al signo del *skewness*.

3.2 Clasificación multiclase.

Tras la participación en PAN-AP'18, como explicamos en el apartado 6, seguimos avanzando en el análisis de los datos obtenidos. Como queda probado con los resultados en la tarea, el *skewness* es una buena característica para modelar la clasificación del género de los autores. Pero

el análisis simplista de 2 clases es imposible de generalizar para un número mayor de clases.

Hasta el momento hemos considerado el *skewness* como característica para representar el texto. Tal y como se muestra en las figuras 3.1.1. y 3.1.2., en las que hemos representado las distribuciones a priori de pertenencia a cada una de las clases en el corpus de entrenamiento de PAN-AP'18.

Como vemos, las distribuciones presentan una gran acumulación de observaciones en el tramo central [0,33-0,66], que corresponden a las palabras que son utilizadas por hombre y mujeres de forma indistinta. También se observa que la distribución de las colas ([0-0,33] y [0,66-1] es sensiblemente diferente. Vamos a representar el mismo concepto para un problema que presenta un número mayor de clases, que extraemos del corpus CMUQ100-ARAP [6], detallado en el apartado 4, para clasificación de variedad dialectal, que presenta quince clases diferentes. Mostramos una gráfica de las quince posibles, adjuntando las otras catorce en el Anexo I.

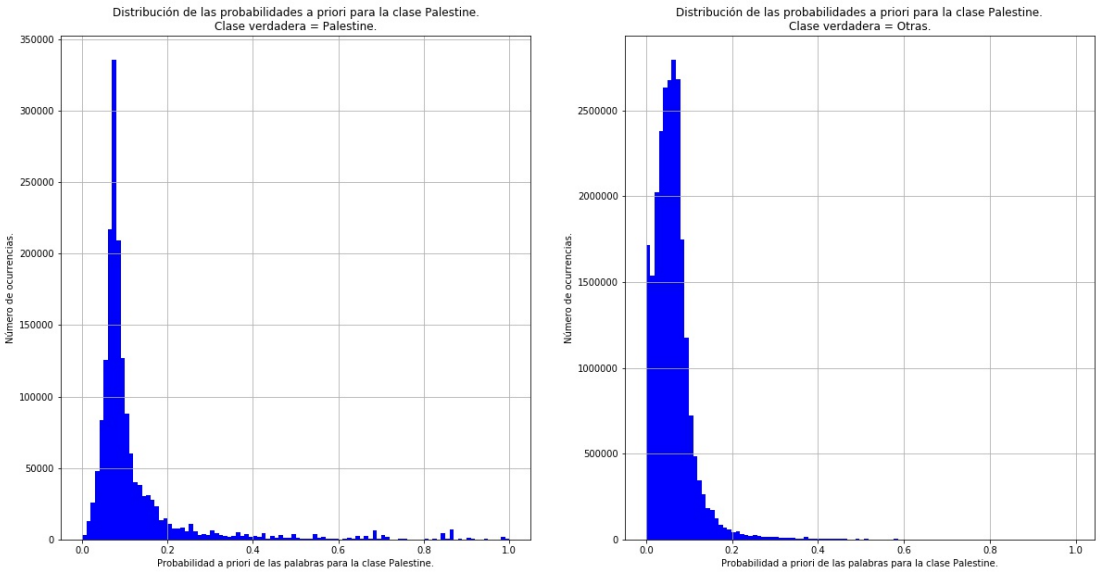

Figura 3.2.1. Distribución de las probabilidades a priori de pertenencia a las clase Palestine y resto de variedades cuando el texto pertenece a la clase Palestine.

Se observa que la distribución de las probabilidades a priori de pertenencia a la clase Palestine difiere en la cola de la derecha de forma significativa respecto a la representación de la distribución para el resto de clases. Esta información nos hace replantearnos los estadísticos que hemos estado utilizando.

Las primeras pruebas que realizamos consistieron en dividir los datos en tres tercios, ya que los tres máximos se encuentran en los intervalos [0,0.33],]0.33, 0.66] y]0.66, 1]. De forma que calculamos media, desviación típica y skewness para los tres intervalos. Tras alimentar los sistemas de clasificación con estos datos vimos que la precisión de las predicciones disminuía.

Obviamente descartamos esta idea, pero seguimos pensando que debemos ser capaces de obtener algún beneficio del conocimiento que hemos adquirido.

A continuación cambiamos la orientación de nuestra aproximación. Partiendo de los mismos tres intervalos, en lugar de calcular sus estadísticos calculamos el número de muestras que caen dentro de dichos intervalos.

Alimentamos los sistemas de clasificación con esos tres parámetros y vemos que los resultados siguen siendo peores que cuando los alimentamos con la media, sdv y skewness globales.

La nueva propuesta consistirá en conservar los estadísticos globales y añadirles el número de muestras en los terciles. Con estas características se observa una mejora en la clasificación.

3.3. Frequency Analysis Interpolation.

Frequency Analysis Interpolation (FAI) es el método de representación alternativa, de baja dimensionalidad, que hemos desarrollado en este trabajo.

Descripción del método:

1. Generación del diccionario de probabilidad a priori de pertenencia a cada clase:

    a) Calculamos la matriz TF de los textos de todos los autores utilizados para entrenar. Obteniendo una matriz de frecuencia de aparición de cada palabra sobre el corpus de entrenamiento para cada autor.

    Sea A el conjunto de Autores y K el conjunto de clases, donde cada autores pertenece a una única clase: $A_k = \{ a / a \in A : k \in K \}$

    Definimos TF como la matriz de frecuencia de términos donde $TF(w_i,a_j)$ es la frecuencia del término i en el autor j.

$$\begin{bmatrix} TF(w_1,a_1) & TF(w_2,a_1) & TF(w_3,a_1) & \dots & TF(w_m,a_1) \\ TF(w_1,a_2) & TF(w_2,a_2) & TF(w_3,a_2) & \dots & TF(w_m,a_2) \\ TF(w_1,a_3) & TF(w_2,a_3) & TF(w_3,a_3) & \dots & TF(w_m,a_3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ TF(w_1,a_{n-1}) & TF(w_2,a_{n-1}) & TF(w_3,a_{n-1}) & \dots & TF(w_m,a_{n-1}) \\ TF(w_1,a_n) & TF(w_2,a_n) & TF(w_3,a_n) & \dots & TF(w_m,a_n) \end{bmatrix}$$

Figura 3.3.1. Matriz TF de un corpus con n autores y m palabras en el vocabulario, donde $TF(w_i,a_j)$ es la frecuencia del término j en el autor i.

b) Para cada clase, recorremos los textos de todos los autores y sumamos la frecuencia de aparición de cada una de las palabras. Así, habremos sumado las veces que aparece cada palabra del vocabulario en cada una de las diferentes clases.

   Denotaremos TF_a el vector de frecuencia de términos del autor a.

$$\left[TF_{(w_1,a)}, TF_{(w_1,a)}, TF_{(w_1,a)}, \dots, TF_{(w_m,a)}\right]$$

Figura 3.3.2. Vector de frecuencias de términos por autor.

   Definimos C_k como el vector de frecuencia de términos para cada clase.

$$C_k = \sum_{a \in A_k} TF_a \forall k \in K$$

Ecuación 3.3.1. Vector de frecuencia de términos por clase.

c) Sumamos, para todas las clases, los vectores de frecuencia de términos de los autores de cada clase, resultando en un vector que contiene el número de veces que aparece cada palabra en todo el corpus de entrenamiento.

   Definimos F como el vector de frecuencia de términos para todas las clases.

$$F = \sum_{a \in A} TF_a$$

Ecuación 3.3.2. Vector de frecuencia de términos para todas las clases.

d) Se dividen los vectores generados para cada clase en el paso b, por el vector generado en el paso c, componente a componente. De esta forma tendremos un vector para cada clase, de longitud igual al número de palabras que hay en el vocabulario del corpus de entrenamiento, que contiene las probabilidades a priori para cada palabra de pertenecer a cada una de las clases.

Definimos P como el vector de frecuencia de términos condicionada a cada clase.

$$P_k = [p_1, p_2, p_3, ..., p_m] / p_i = \frac{C_{ki}}{F_i} \quad \forall i \in W \ : \ \forall k \in K$$

Ecuación 3.3.3. Vectores de probabilidades a priori de pertenencia a cada una de las k clases para cada palabra.

2. Generación de las características que representan el texto de cada autor.

   a) Se codifican los textos de cada autor mediante el uso del diccionario de probabilidades. Se obtiene un vector por cada clase con las probabilidades a priori de cada palabra para esa clase.

$$\begin{bmatrix} Autor_1 & P(C_1(w_1)) & (PC_1(w_2)) & P(C_1(w_3)) & \dots & P(C_1(w_m)) \\ Autor_1 & P(C_2(w_1)) & P(C_2(w_2)) & P(C_2(w_3)) & \dots & P(C_2(w_m)) \\ \dots & \dots & \dots & \ddots & \dots & \\ Autor_1 & P(C_k(w_1)) & P(C_k(w_2)) & P(C_k(w_3)) & \dots & P(C_k(w_m)) \end{bmatrix}$$

$$\begin{bmatrix} Autor_2 & P(C_1(w_1)) & (PC_1(w_2)) & P(C_1(w_3)) & \dots & P(C_1(w_m)) \\ Autor_2 & P(C_2(w_1)) & P(C_2(w_2)) & P(C_2(w_3)) & \dots & P(C_2(w_m)) \\ \dots & \dots & \dots & \ddots & \dots & \\ Autor_2 & P(C_k(w_1)) & P(C_k(w_2)) & P(C_k(w_3)) & \dots & P(C_k(w_m)) \end{bmatrix}$$

$$\begin{bmatrix} Autor_n & P(C_1(w_1)) & (PC_1(w_2)) & P(C_1(w_3)) & \dots & P(C_1(w_m)) \\ Autor_n & P(C_2(w_1)) & P(C_2(w_2)) & P(C_2(w_3)) & \dots & P(C_2(w_m)) \\ \dots & \dots & \dots & \ddots & \dots & \\ Autor_n & P(C_k(w_1)) & P(C_k(w_2)) & P(C_k(w_3)) & \dots & P(C_k(w_m)) \end{bmatrix}$$

Figura 3.3.3. Vectores de probabilidades a priori de pertenencia a cada clase de los textos de cada autor.

   b) Se cambia el espacio de representación. Se representa el texto de cada autor por la media, la desviación típica, el *skewness* y el número de elementos en cada tercil, por cada clase. De forma que cada autor queda representado por seis características por cada clase.

$$\begin{bmatrix} Autor_1 & \overline{P(C_1(texto))} & \delta(PC_1(texto)) & skew(P(C_1(texto))) & len(Q_1) & len(Q_2) & len(Q_3) \\ Autor_2 & \overline{P(C_1(texto))} & \delta(PC_1(texto)) & skew(P(C_1(texto))) & len(Q_1) & len(Q_2) & len(Q_3) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ Autor_n & \overline{P(C_1(texto))} & \delta(PC_1(texto)) & skew(P(C_1(texto))) & len(Q_1) & len(Q_2) & len(Q_3) \end{bmatrix}$$

Figura 3.3.4. Cambio de representación en FAI.

3. Actualización del modelo de representación.

Adicionalmente, los vectores de probabilidad a priori de pertenencia de cada palabra a cada clase se pueden modificar con nuevos textos etiquetados de forma sencilla y con bajo coste. En la primera ejecución hemos guardado en ficheros CSV la información relevante (número de

autores en que aparece cada palabra, así como la matriz TF). El proceso tendría tres pasos:

1.- Calcular TF de los nuevos textos.

2.- Actualizar el diccionario que lleva la cuenta de en cuántos autores aparece cada palabra. Recalcular el vocabulario a usar en base a la regla definida (% de autores).

3.- Recalcular los vectores de probabilidad a priori de pertenencia de cada palabra a cada clase.

4. Marco de evaluación

4.1 Corpora.

En esta sección describimos los diferentes corpora que hemos utilizado para evaluar las prestaciones de nuestro método.

4.1.1 CMUQ-ARAP

El corpus CMUQ100-ARAP [7] consiste en tweets de 1.523 autores diferentes, de los cuales 1.163 se utilizarán para entrenamiento y 360 para test. Se trata de tweets escritos en árabe y de los cuales haremos tres tareas de clasificación: género; edad (la clasificación de edad se realiza en los tres grupos siguientes: menor de 25 años, entre 25 y 34 años, mayor de 34 años); y variedad dialectal, que incopora 15 diferentes variedades dialectales del árabe. Dichas variedades son: palestina y jordana, saudita, libanesa y siria, tunecina, marroquí, argelina, kuwaití, yemení, sudanesa, iraquí, qatarí, libia, egipcia, omaní y de emiratos árabes unidos. El corpus está formado por 100 tweets por cada autor.

		Train	Test
Género	Hombre	585	180
	Mujer	578	180
Edad	menor de 25	390	120
	entre 25 y 34	389	120
	mayor de 34	384	120
Variedad	Argelina	77	24
	Egipcia	78	24
	Iraquí	78	24
	Kuwaití	77	24
	Sirio-libanesa	78	24
	Libia	77	24
	Marroquí	78	24
	Omaní	78	24
	Jordano-palestina	78	24
	Qatarí	77	24
	Saudí	77	24
	Sudanesa	77	24
	Tunecina	77	24
	de los Emiratos	78	24
	Yemení	78	24

Tabla 4.1.1. Corpus CMUQ100-ARAP.

4.1.2 PAN-AP'17.

El corpus PAN-AP'17 ha sido creado para la tarea internacional PAN 2017 [11]. PAN[2] es una serie de eventos científicos y tareas compartidas en análisis forense de texto digital y caracterización de estilo. Se lleva a cabo en el contexto del CLEF (*Conference and Labs of the Evaluation Forum*[3].

En la edición de 2017 la tarea de *Author Profiling* consistió en clasificar género y variedad de idioma para 4 idiomas diferentes (inglés, español, portugués y árabe). El corpus está formado por 100 tweets por cada autor.

			Train	Test
Inglés	Género	Hombre	1.800	1.200
		Mujer	1.800	1.200
	Variedad	Australiana	600	400
		Canadiense	600	400
		Irlandesa	600	400
		Británica	600	400
		Nuevazelandesa	600	400
		Estadounidense	600	400
Español	Género	Hombre	2.100	1.400
		Mujer	2.100	1.400
	Variedad	Argentina	600	400
		Chilena	600	400
		Colombiana	600	400
		Mexicana	600	400
		Peruana	600	400
		Española	600	400
		Venezolana	600	400
Portugués	Género	Hombre	600	400
		Mujer	600	400
	Variedad	Brasileña	600	400
		Portuguesa	600	400
Árabe	Género	Hombre	1.200	800
		Mujer	1.200	800
	Variedad	Egipcia	600	400
		del Golfo	600	400
		Levantina	600	400
		Magreb	600	400

Tabla 4.1.2. Corpus PAN-AP'17.

2 pan.webis.de

3 www.clef2017.clef-initiative.eu

4.1.3 PAN-AP'18.

El corpus PAN-AP'18 ha sido creado para la tarea internacional PAN 2018 [5]. En la edición de 2018 la tarea de *Author Profiling* consistió en clasificar género para 3 idiomas diferentes (inglés, español y árabe) desde una perspectiva multimodal. Es decir, se proporcionan tweets escritos por los autores más dies imágenes compartidas en sus *timelines*. El corpus está formado por 100 tweets y 10 imágenes por cada autor. En este trabajo, dada su naturaleza, no se han utilizado las imágenes.

Inglés: El corpus consiste en tweets escritos por 4900 autores, de los cuales 3000 se utilizarán para entrenamiento y 1900 para test.

Español: El corpus consiste en tweets escritos por 5200 autores, de los cuales 3000 se utilizarán para entrenamiento y 2200 para test.

Árabe: El corpus consiste en tweets escritos por 2500 autores, de los cuales 1500 se utilizarán para entrenamiento y 1000 para test.

		Train	Test
Inglés	Hombre	1500	950
	Mujer	1500	950
Español	Hombre	1500	1100
	Mujer	1500	1100
Árabe	Hombre	750	500
	Mujer	750	500

Tabla 4.1.3. Corpus PAN-AP'18.

4.1.4 PAN-AP'14.

El corpus PAN-AP'14 ha sido creado para la tarea internacional PAN 2014 [12]. En la edición de 2014 la tarea de *Author Profiling* consistió clasificar género para 2 idiomas diferentes (inglés y español) y clasificar los autores en diferentes rangos de edad (18-24 años, 25-34 años, 35-49 años, 50-64 años, más de 65 años). Disponemos de cuatro corpora diferentes, tweets, social media, blogs y revisiones de hoteles, aunque éste último no ha sido objeto de estudio en este trabajo:

TWEETS.

Inglés: El corpora consiste en tweets escritos por 460 autores, de los cuales 306 se utilizarán para entrenamiento y 154 para test.

Español: El corpus consiste en tweets escritos por 268 autores, de los cuales 178 se utilizarán para entrenamiento y 90 para test.

SOCIAL MEDIA.

Inglés: El corpus consiste en contenido de social media escritos por 11.114 autores, de los cuales 7738 se utilizarán para entrenamiento y 3376 para test.

Español: El corpus consiste en tweets escritos por 1838 autores, de los cuales 1272 se utilizarán para entrenamiento y 566 para test.

BLOGS

Inglés: El corpus consiste en contenido de blogs escritos por 225 autores, de los cuales 147 se utilizarán para entrenamiento y 78 para test.

Español: El corpus consiste en tweets escritos por 144 autores, de los cuales 88 se utilizarán para entrenamiento y 56 para test.

				Train	Test
Tweets	Inglés	Género	Hombre	153	77
			Mujer	153	77
		Edad	18-24	20	12
			25-34	88	56
			35-49	130	58
			50-64	60	26
			65-xx	7	2
	Español	Género	Hombre	89	45
			Mujer	89	45
		Edad	18-24	12	4
			25-34	42	26
			35-49	86	46
			50-64	32	12
			65-xx	6	2
Social Media	Inglés	Género	Hombre	3.871	1.683
			Mujer	3.867	1.693
		Edad	18-24	1.549	701
			25-34	2.096	923
			35-49	2.244	951
			50-64	1.835	793
			65-xx	14	8
	Español	Género	Hombre	636	258
			Mujer	636	308
		Edad	18-24	330	135
			25-34	426	190
			35-49	324	154
			50-64	160	77
			65-xx	65	10
Blogs	Inglés	Género	Hombre	74	42
			Mujer	73	36
		Edad	18-24	6	3
			25-34	60	31
			35-49	54	29
			50-64	23	13
			65-xx	4	2
	Español	Género	Hombre	44	30
			Mujer	44	26
		Edad	18-24	4	4
			25-34	26	13
			35-49	42	27
			50-64	12	9
			65-xx	4	3

Tabla 4.1.4. Corpus PAN-AP'14.

4.1.5 PAN-AP'13.

El corpus PAN-AP'13 ha sido creado para la tarea internacional PAN 2013 [13]. En la edición de 2013 la tarea de author profiling consistió clasificar género para 2 idiomas diferentes (inglés y español) y clasificar los autores en diferentes rangos de edad (13-27: '10s', 23-27: '20s', 33-47: '30s').

Inglés: El corpus consiste en tweets escritos por 342.892 autores, de los cuales 236.596 se utilizarán para entrenamiento y 106.296 para test.

Español: El corpus consiste en tweets escritos por 109.900 autores, de los cuales 75.900 se utilizarán para entrenamiento y 34.000 para test.

			Train	Test
Inglés	Género	Hombre	118.296	53.146
		Mujer	118.300	53.150
	Edad	10s	17.200	7.760
		20s	85.799	38.515
		30s	133.597	60.021
Español	Género	Hombre	37.950	16.979
		Mujer	37.950	17.021
	Edad	10s	2.500	1.093
		20s	42.600	19.094
		30s	30.800	13.823

Tabla 4.1.5. Corpus PAN-AP'13.

4.1.6 RUSProfiling.

El corpus RUSProfiling ha sido creado para la tarea de *Author Profiling* de *RusProfiling Cross-genre Identification (RUSProfiling)* en idioma ruso en el laboratorio PAN en el FIRE en 2017[14]. FIRE se inició en 2008 con el ánimo de organizar un evento en el sur de Asia que fuera el equivalente a los eventos del sector presentes en Europa y América (TREC; CLEF, NTCIR). FIRE aborda temáticas como la detección de plagio, acceso a información legal, detección de idioma nativo, etc.

El corpus presenta características diferentes a las consideradas hasta el momento en este trabajo. Se recuperaron documentos de cinco fuentes diferentes: Ensayos, Facebook, Twitter, Reseñas y textos de Imitación de

Género, es decir textos en los que una persona de un sexo pretende escribir enmascarando su sexo y simulando el contrario.

El corpus consiste en:

- 1.000 textos provenientes de tweets, entre 1 y 200 tweets por autor.
- 228 textos provenientes de Facebook, 1.000 palabras por texto de media.
- 400 textos de ensayos, 150 palabras por texto de media.
- 776 textos de reseñas, 80 palabras por texto de media.
- 94 textos de género imitado o pretendido, entre 80 y 150 palabras por texto.

De estos textos se utilizan 600 provinientes de tweets como entrenamiento y el resto de tweets junto con los textos de las otras categorías como TEST, en total 1.868.

En el corpus de entrenamiento observamos la siguiente partición de autores por clases:

- Género: 300 hombre, 300 mujer.

En el corpus de test observamos la siguiente partición de autores por tipo de texto y clases:

- Ensayos. Género: 185 hombre, 185 mujer.
- Facebook. Género: 114 hombre, 114 mujer.
- Twitter. Género: 200 hombre, 200 mujer.
- Reseñas. Género: 388 hombre, 388 mujer.
- Género imitado. Género: 47 hombre, 47 mujer.

4.2 Algoritmos de clasificación.

En esta sección describimos los distintos clasificadores utilizados para predecir las clases de los autores. Empleamos métodos bayesianos (clasificador Naïve Bayes), métodos basados en árboles de decisión (Arboles de Decisión y Random Forest), Máquinas de Vectores de Soporte o Support Vector Machines (SVM) y técnicas de aprendizaje neuronal (Perceptrón Multicapa). Todas estas herramientas vienen implementados en el toolkit scikit-learn1 para Python.

4.2.1 Máquinas de Vectores de Soporte.

Uno de los clasificadores que utilizamos son las máquinas de vectores soporte [8]. Más concretamente utilizamos las máquinas de vectores soporte con kernel linear implementadas en scikit-learn, que a su vez están implementadas sobre libsvm. Las máquinas de vectores soporte tratan de determinar el hiperplano que separa las muestras de ambas clases,

maximizando la distancia entre las muestras más próximas de cualquier clase y el hiperplano de separación. Las SVM hacen uso de métodos basados en kernels para operar en espacios de alta dimensionalidad sin tener que calcular todos los puntos explícitamente. Se ha utilizado siempre el kernel “linear”, de forma que se buscará un hiperplano que separe las clases linealmente. Se utilizan tanto el método SVM, como el método LinearSVC, que está implementado con la librería liblinear en lugar de libsvm, con lo que tiene más flexibilidad en la elección de las penalties y de la función de pérdida. Además, debería escalar mejor para grandes números de muestras.

4.2.2 Naïve Bayes.

La regla de clasificación de Naïve Bayes es la mostrada en la Ecuación 4.2.2.1. Este clasificador está basado en el teorema de Bayes, asumiendo que todas las características son independientes entre sí.

$$c^{\wedge} = \arg\max c\ P(c)\ Yn\ i=1\ P(xi\ |c)$$

Ecuación 4.2.2.1. Naïve Bayes.

Esta asunción resulta bastante ingénua y parece inadecuada para tratar con texto, pues las frases son construidas en base a las restricciones impuestas por el propio idioma. Por ejemplo, un sustantivo y un adjetivo que lo referencie deben concordar en género y número, por tanto no hay independencia entre las palabras que aparecen (ni en su orden). Sin esta asunción, la Ecuación 4.7 debería considerar también las dependencias condicionales entre las características, lo cual supondría un sobrecoste elevado en tiempo de computación. A pesar de todo, estos clasificadores han servido para resolver exitosamente distintos problemas en Procesamiento del Lenguaje Natural (PLN), como la creación de filtros anti-spam u otros problemas de clasificación de documentos.

4.2.3 Perceptrón Multi Capa (PMC).

El Perceptrón Multicapa [9] puede ser concebido como un clasificador por regresión logística, donde la entrada se transforma primero usando una transformación no linear aprendida Φ. Esta tarnsformación proyecta los datos de entrada a un espacio donde se convierte en linealmente separable. Esta capa intermedia se denomina capa oculta. Una única capa oculta es suficiente para hacer del perceptrón multicapa un aproximador universal.

Un perceptrón multicapa con una sola capa oculta se puede representar gráficamente como sigue:

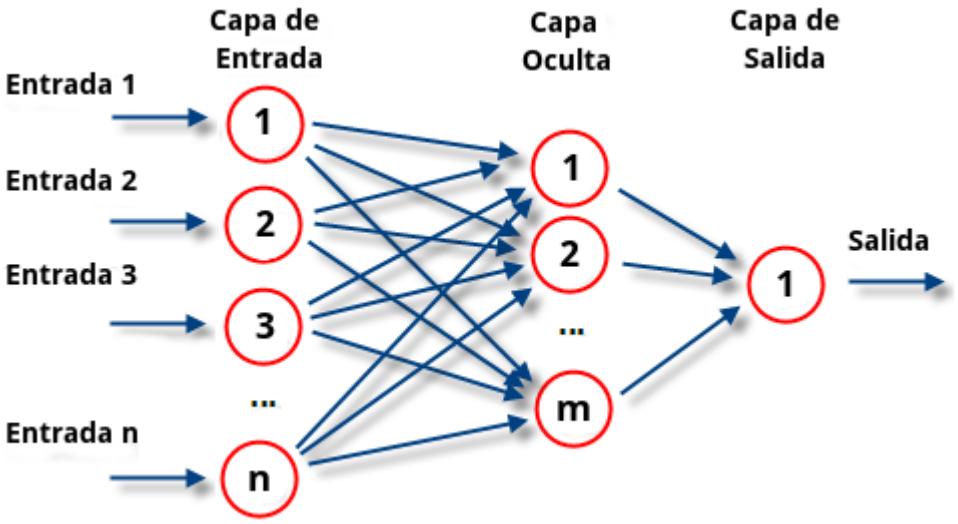

Figura 4.2.3.1. Esquema de un Perceptrón Multi-capa.

Formalmente, un PMC de una capa oculta es una función $f: R^D \rightarrow R^L$, donde D es el tamaño del vector de entrada x y L es el tamaño del vector de salida $f(x)$, de forma que en notación matricial tenemos:

$$f(x) = G(b^{(2)} + W^{(2)} (s(b^{(1)} + W^{(1)}x)))$$

Ecuación 4.2.3.1. Función de la capa oculta de un PMC de una sola capa.

con vectores de bias $b^{(1)}$, $b^{(2)}$; matrices de pesos $W^{(1)}$, $W^{(2)}$ y funciones de activación G y s.

El vector $h(x) = \Phi(x) = s(b^{(1)} + W^{(1)}x)$ constituye la capa oculta. $W^{(1)} \in R^{DxD_h}$ es la matriz de pesos que conecta el vector de entrada con la capa oculta. Cada columna $W_i^{(1)}$ representa los pesos de las entradas unitarias de la i-ésima capa oculta.

Algunas opciones típicas para s son tanh, con $tanh(a) = (e^a - e^{-a})/(e^a + e^{-a})$, o la función logística *sigmoid*, con $sigmoid(a) = 1/(1 + e^{-a})$. Ambas funciones, *tanh* y *sigmoid* son funciones escalares pero con extensión natural a vectores que consiste en aplicarlas elemento a elemento.

El vector de salida se obtiene entonces como: $o(x) = G(b^{(2)} + W^{(2)}h(x))$. Las probabilidades de pertenencia a cada una de las clases se pueden obtener eligiendo G como la función *softmax* (en el caso de clasificación multi-clase).

4.2.4 Árboles de decisión.

Los Árboles de Decisión son un método de aprendizaje supervisado utilizado para tareas de clasificación o regresión. La idea básica consiste en inferir una serie de reglas de decisión simples a partir de los datos de entrenamiento. De forma intuitiva, el proceso de clasificación es el siguiente. Dado un nodo, este se encarga de clasificar la muestra de

entrada utilizando una única característica, previamente seleccionada. En función del valor que toma esta característica se escoge a uno de sus hijos, que evaluará otra característica. Este proceso se repite de forma recursiva, empezando por el nodo raíz y hasta que el algoritmo alcance alguna hoja. Cada hoja contiene una etiqueta, que se le asigna a la muestra. A cada iteración, es deseable que el algoritmo escoja la característica que más información aporte, que sea más útil para la discriminación.

4.2.5 Random Forest.

Random Forest es un meta estimador que ajusta un numero de árboles de decision determinado por el usuario en varias sub-muestras del conjunto de datos y utiliza la ponderación para mejorar la precisión predictiva y controlar el sobreajuste. El tamaño de la sub-muestra es siempre el mismo que el tamaño de la muestra original pero las muestras se descartan con reemplazo si el parámetro bootstrap se ajusta a True, que es la configuración por defecto.

4.3 Medidas de evaluación.

Para evaluar el corpus empleamos el *accuracy* definido como la relación entre el número de muestras bien clasificadas (N_{bien}) frente al total (N) de muestras del conjunto de test.

$$accuracy = N_{bien}/N$$

Ecuación 4.3.1. Definición de *accuracy*.

Decidimos utilizar el accuracy ya que es la medida de evaluación utilizada en las tareas que hemos tomado como referencia, permitiendo así la comparatibilidad de los resultados. Además, todos los corpora de entrenamiento y test están equilibrados entre sus clases, por lo que la elección del accuracy como medida de evaluación es adecuada.

4.4 Test de significación estadística.

Vamos a realizar un test de significación para comprobar si en los casos que FAI obtiene mejores resultados que LDSE y Word2Vec éstos son estadisticamente significativos.

Ya que el *accuracy* es, en este caso, la proporción de muestras correctamente clasificadas, podemos aplicar el test de hipótesis aplicado a un sistema de dos proporciones [10].

Sean $\widehat{p}_1$y $\widehat{p}_2$ los *accuracies* obtenidas por los clasificadores 1 y 2 respectivamente, y n el número de muestras. El número de muestras correctamente clasificadas por los clasificadores 1 y 2 son x_1 y x_2 respectivamente.

$$\widehat{p_1} = x_1/n \qquad \widehat{p_2} = x_2/n$$

Ecuación 4.4.1. *Accuracies* de dos experimentos.

El test estadístico viene dado por:

$$Z = \frac{\widehat{p_1} - \widehat{p_2}}{\sqrt{2\,\hat{p}(1-\hat{p})/n}} \text{ donde } \hat{p} = (x_1 + x_2)/2n$$

Ecuación 4.4.2. Test estadístico.

Nuestra intención es proba que el accuracy global del clasificador 2, esto es p_2, es mejor que el del clasificador 1, que es p_1. Esto define nuestras hipótesis como:

- H_0: $p_1 = p_2$ (hipótesis nula, ambos accuracies son iguales)
- H_α: $p_1 < p_2$ (hipótesis alternativa, el nuevo clasificador es mejor que el existente)

La zona de rechazo viene dada por :

$Z < -z_\alpha$ (si es verdadero rechazamos H_0 y aceptamos H_α)

donde z_α se obtiene de una distribución normal estándar que pertenece a un nivel de significación, α. Por ejemplo, $z_{0.5} = 1.645$ para un nivel de significación del 5%. Esto significa que si la relación $Z < -1{,}645$ es verdadera, podremos decir con un nivel de confianza del 95% (1-α) que el clasificador 2 tiene mejor accuracy que el clasificador 1.

Este test de significación tiene algunas restricciones, ya que para tamaños de corpus (n) pequeños o resultados cercanos al 0% o 100% puede dar resultados equivocados. Es por ello que se estipulan como restricciones que $n*p_1$, $n*q_1$, $n*p_2$ y $n*q_2$ deben ser siempre mayores o igual que 5.

4.5 Preprocesamiento.

Con el objetivo de reducir el tiempo de proceso, se eliminan del vocabulario las palabras que aparecen en pocos autores, ya que pueden no aportar información al modelo. En cada caso se han computado diferentes límites para la eliminación de palabras, oscilando entre:

1. Dejar todas las palabras en el vocabulario.
2. Eliminar las palabras que aparecen en menos del 1% de los autores.
3. Eliminar las palabras que aparecen en menos del 2% de los autores.
4. Eliminar las palabras que aparecen en menos del 3% de los autores.

El número de autores diferentes totales que tenemos para entrenar es influyente en el límite que establecemos, a menor número de autores, mayor % podremos eliminar.

Este proceso permite reducir el vocabulario a utilizar para modelar los textos a un 20-30% del vocabulario total. Esto viene a mostrar que hay muchas palabras que son usadas por muy pocos autores. Y éstas son las palabras que descartaremos de nuestro modelo.

5. Resultados experimentales

A continuación exponemos los resultados de los experimentos realizados, agrupándolos por tipo de tarea. En cada caso se mostrarán los resultados obtenidos por los métodos descritos: LDSE, Word2Vec y FAI.

5.1 Identificación de variedad del lenguaje.

Se ha realizado experimentación con los corpora CMUQ-ARAP y PAN-AP'17. El corpus CMUQ-ARAP consiste en *tweets* escritos en lengua árabe por 1.523 autores diferentes donde las clases de variedad del lenguaje están equilibradas. El corpus PAN-AP'17 consiste en *tweets* escritos en cuatro idiomas diferentes: inglés (4.900 autores), español (5.200 autores), portugués (2.000 autores) y árabe (4.000 autores) y en todos los casos las clases están equilibradas. En la tabla 5.1 se muestran los resultados obtenidos:

Corpus	Idioma	LDSE	Word2Vec	FAI
CMUQ-ARAP	Árabe	88.89%	85.00%	92.50%
PAN-AP'17	Inglés	86.96%	78.71%	81.63%
	Español	96.25%	86.14%	92.00%
	Portugués	98.75%	98.00%	98.50%
	Árabe	82.50%	76.00%	81.44%

Tabla 5.1. Resultados en variedad del lenguaje para los corpora CMUQ-ARAP y PAN-AP'17.

Word2Vec utiliza vectores generados a partir de contenidos escritos en un idioma, sin diferenciar variedades de un mismo idioma. Es por ello que en este tipo de tarea, pese a obtener un buen resultado, tiene prestaciones inferiores a LDSE y FAI. Esto es debido a que tanto FAI como LDSE construyen vectores independientes para cada clase, y recogen mejor las diferencias entre variedades diferentes de cada idioma. Por ejemplo, tanto LDSE como FAI recogerán las diferencias entre el uso de elevator (inglés de USA) y lift (inglés de UK). La diferencia entre LDSE y FAI puede ser debida a la diferente caracterización de los stop-words, ya que LDSE utiliza Tf-Idf y FAI TF. Las preposiciones, artículos y demás, tendrán diferente ponderación en cada uno de los métodos.

En el caso del portugués observamos que los tres métodos obtienen unos resultados similares. Esto puede ser debido a que los vectores de Word2Vec se hayan construido con textos escritos predominantemente con una de las dos variedades del idioma portugués, facilitando de este modo la diferenciación entre ambos.

En la tarea de clasificación de variedades dialectales del idioma portugués nuestro método obtendría un 1º lugar sobre 19 participantes en el ránking de PAN-AP'17, mientras que en el idioma árabe obtendría un 5º lugar sobre 21 participantes.

Los mismos razonamientos se aplican al caso de la lengua Árabe. Por ejemplo, en la lengua árabe existe un único artículo determinado, y se escribe junto a la palabra que acompaña, conformando un solo vocablo. Esto podría explicar la menos diferencia en rendimiento entre LDSE y FAI.

5.2 Identificación de sexo.

Encontramos este problema en los corpora CMUQ, PAN-AP'18, PAN-AP'17, PAN-AP'14 tanto en Tweets, como Blogs y Social Media, PAN-AP'13 y RUSProfiling. El corpus CMUQ-ARAP consiste en *tweets* escritos en lengua árabe por 1.523 autores diferentes donde las clases de variedad del lenguaje están equilibradas. El corpus PAN-AP'18 consiste en *tweets* escritos en 3 idiomas diferentes: inglés (4.900 autores), español (5.200 autores) y árabe (2.500 autores) y en todos los casos las clases están equilibradas. El corpus PAN-AP'17 consiste en *tweets* escritos en 4 idiomas diferentes: inglés (4.900 autores), español (5.200 autores), portugués (2.000 autores) y árabe (4.000 autores) y en todos los casos las clases están equilibradas. El corpus PAN-AP'14 consiste en tweets en 2 idiomas: inglés(460 autores) y español (268 autores), textos de *social media* en 2 idiomas: inglés (11.114 autores) y español (1.838 autores), y entradas en blogs en 2 idiomas, inglés (225 autores) y español (144 autores). En todos los casos las clases están equilibradas. El corpus PAN-AP'13 consiste en *tweets* escritos en 2 idiomas diferentes: inglés (342.892 autores) y español (109.900 autores). En ambos casos las clases están equilibradas. El corpus RUSProfiling consiste en textos escritos en ruso que provienen de 5 fuentes diferentes: *tweets* (1.000 autores), *posts* en Facebook (228 autores), ensayos (400 autores), reseñas (776 autores) y textos en los que se ha pretendido escribir como el género opuesto al del autor (94 autores).

En la tarea de identificación de género podríamos esperar que Word2Vec pudiera verse perjudicado por el hecho de que los vectores hayan sido generados con textos que pueden provenir mayoritariamente o no equitativamente de todas las variaciones existentes. Sin embargo, aunque los tres métodos presentan unas prestaciones similares, nuestra representación presenta una mejora significativa en las prestaciones en el idioma árabe en las tareas CMUQ-ARAP y PAN-AP'18 y en la tarea PAN-AP'17 en el idioma portugués. Esta diferencia es estadísticamente significativa con un nivel de confianza del 95%, según el método descrito en 4.4.

Corpus	Idioma	LDSE	Word2Vec	FAI
CMUQ-ARAP	Árabe	75.28%	77.22%	**86.11%**
PAN-AP'18	Inglés	-	77.89%	75.16%
	Español	-	75.05%	71.23%
	Árabe	-	72.50%	**76.40%**
PAN-AP'17	Inglés	72.20%	78.42%	75.92%
	Español	78.63%	76.32%	71.79%
	Portugués	71.71%	73.13%	**81.50%**
	Árabe	70.44%	71.50%	73.44%
PAN-AP'14 Tweets	Inglés	-	51.11%	58.44%
	Español	-	57.78%	62.22%
PAN-AP'14 Blogs	Inglés	-	62.82%	62.82%
	Español	-	51.79%	57.14%
PAN-AP'14 Social Media	Inglés	-	54.74%	54.68%
	Español	-	63.60%	65.19%
PAN-AP'13	Inglés	-	62.31%	60.20%
RUS Profiling	Ensayos	81.40%	77.57%	78.65%
	Facebook	85.96%	84.21%	84.65%
	Twitter	67.59%	69.09%	65.58%
	Reseñas	65.81%	67.10%	60.85%
	Imitación	55.32%	54.26%	55.32%

Tabla 5.2. Resultados en identificación de género para los corpora CMUQ-ARAP, PAN-AP'18, PAN-AP'17, PAN-AP'14 tanto en Tweets, como Blogs y Social Media, PAN-AP'13 y RUS Profiling. En negrita, significación estadística al 95%.

En la tarea PAN14, nuestra representación hubiera obtenido un 4º lugar sobre 9 participantes en inglés y 3º sobre 8 participantes en español con el corpus de tweets, un 5º lugar sobre 10 participantes en inglés y un 2º sobre 9 participantes en español con el corpus de blogs y un 1º lugar sobre 10 participantes en inglés y un 2º sobre 9 participantes en español con el corpus de social media.

En la tarea RUSProfiling, nuestra representación hubiera obtenido un 1º lugar en clasificación de textos provenientes de ensayos, un 7º en clasificación de textos provenientes de Facebook, un 3º en clasificación de textos provenientes de Twitter, un 2º en clasificación de textos provenientes de Reseñas y un 3º en clasificación de textos provenientes de Imitación de género. La clasificación final se hizo mediante la media de la precisión del modelo ponderada con el número de muestras de cada tipo.

$$\text{global acc} = \text{P ds accuracy(ds)} * \text{size(ds))} / \text{P ds size(ds)}$$

Ecuación 5.2.1. Fórmula utilizada para el ranking en la tarea RUS-Profiling.

En este caso nuestro método obtiene una puntuación de 68,02%, casi 3,5 puntos porcentuales mejor que el ganador de la tarea.

Como hecho destacable cabe mencionar que FAI, en el caso de las reseñas, obtiene el peor de los resultados. Las reseñas son de un sitio web de valoración de servicios de terceros, donde entendemos que los stop-words son frecuentes, ya que mayritariamente se tratará de descripciones y opiniones donde abundarán los astículos, conjunciones y determinantes Pensamos que estas cuestiones pueden penalizar el rendimiento del método, ya que las dos clases estarán muy superpuestas. No esperamos que las descripciones escuetas difieran mucho en el uso de las palabras en ambos sexos.

5.3. Identificación de edad.

Encontramos este problema en los corpora CMUQ, PAN-AP'14 y PAN-AP'13. Cabría observar que, en mayor o menor medida, Word2Vec obtendría el peor rendimiento. Los vectores utilizados para entrenar se han obtenido de Wikipedia y no podemos asumir que los autores de los mismos sean representativos de todas las clases que se pretenden clasificar. Es un problema parecido al que veíamos con la clasificación de variedad de idiomas. Mientras que LDSE y FAI adquirirán conocimiento durante el entrenamiento del uso de las palabras en cada una de las clases. No disponemos de resultados de LDSE para los corpora de PAN-AP'14.

Corpus	Idioma	LDSE	Word2Vec	FAI
CMUQ-ARAP	Árabe	59.17%	51.39%	57.78%
PAN-AP'14	Inglés	-	37.23%	38.96%
Tweets	Español	-	46.84%	50.00%
PAN-AP'14	Inglés	-	41.03%	44.87%
Blogs	Español	-	44.64%	48.21%
PAN-AP'14	Inglés	-	34.77%	35.16%
Social Media	Español	-	41.70%	45.05%
PAN-AP'13	Español	-	51.79%	57.14%

Tabla 5.3. Resultados en identificación de rango de edad para los corpora CMUQ-ARAP, PAN-AP'14 y PAN-AP'13.

En la tarea PAN-AP'14 nuestro método hubiera ontenido un 5º lugar en ambos idiomas con el corpus de Tweets, un 5º en inglés y un 3º en español con el corpus de Social Media y un 2º en inglés y un 1º en español con el corpus de Blogs, quedando 8º sobre 20 participantes en la tarea PAN-AP'13.

6. Participación en tareas de evaluación internacionales

En el transcurso del desarrollo del método de representación que se presenta en este trabajo hemos participado en diversas tareas internacionales. A continuación describimos el método utilizado y sus resultados.

6.1. Author Profiling en PAN@CLEF 2018.

En la edición de 2018 la tarea de *Author Profiling* consistió clasificar género para tres idiomas diferentes (inglés, español y árabe). En todos los casos se dispone de 100 tweets por autor.

Inglés: El corpus consiste en tweets escritos por 4900 autores, de los cuales 3000 se utilizarán para entrenamiento y 1900 para test.

Español: El corpus consiste en tweets escritos por 5200 autores, de los cuales 3000 se utilizarán para entrenamiento y 2200 para test.

Árabe: El corpus consiste en tweets escritos por 2500 autores, de los cuales 1500 se utilizarán para entrenamiento y 1000 para test.

		Train	Test
Inglés	Hombre	1500	950
	Mujer	1500	950
Español	Hombre	1500	1100
	Mujer	1500	1100
Árabe	Hombre	750	500
	Mujer	750	500

Tabla 6.1.1. Corpus PAN-AP'18.

En esta edición de la tarea de *Author Profiling* hubo un total de 23 equipos que participaron de manera total o parcial en la tarea, enviando predicciones para alguno o todos los idiomas. Para participar en la tarea utilizamos únicamente los tweets proporcionados, ya que el tratamiento de imágenes no es objeto de este trabajo. Calculamos los vectores de probabilidad de pertenencia a cada una de los dos clases y con ellos codificamos los textos de los autores. A continuación calculamos el skewness de la distribución. Con el signo del skewness como única característica, a diferencia del método FAI descrito en el apartado 3.3, predecimos el género de los autores del corpus de test y obtenemos los siguientes resultados:

Idioma	Accuracy
Árabe	95.93%
Español	96.47%
Inglés	96.47%

Tabla 6.1.2. Resultados en identificación de género para el corpus PAN-AP'18.

En el contexto de la tarea estos resultados no son buenos, obteniendo el puesto 19 sobre 23. Hemos de considerar que esta aproximación incluye una única característica y que el método de predicción es una condición sobre el signo del skewness. Una representación más rica en características, como FAI, es de esperar que mejore estos resultados, como se muestra en la tabla 5.2. No obstante, nuestro método presenta unos tiempos de predicción altamente interesantes.

Idioma	Tiempo	№ documentos	docs/segundo	mls/documento
Árabe	2 segundos	1000	500	2.00
Español	3 segundos	2200	733	1.36
Inglés	3 segundos	1900	633	1.58

Tabla 6.1.3. Tiempos de cálculo requeridos por el método.

Hemos conseguido una solución que puede satisfacer los requerimientos de quasi tiempo real de un entorno Big Data con miles de tweets llegando por segundo. A pesar de que la precisión no es la mejor posible, el tiempo de procesado se ajusta a un gran volumen de tweets. Estos tiempos se han conseguido con una máquina virtual con 1 core y 4 Gb de RAM.

6.2. MAPonSMS en FIRE 2018.

Multilingual Author Profiling on SMS (MAPonSMS)[4] es una tarea que estudia la identificación del género y la edad de un autor utilizando mensajes de texto (SMS) en los que los autores utilizan indistintamente dos lenguas, en este caso Roman Urdu e inglés.

El corpus de entrenamiento que se proporciona está compuesto por una colección de SMS escritos por 350 autores diferentes, siendo el número de SMS por autor variable. Las clases no están equilibradas en el corpus. En lo que a género respecta el 60% de los autores son hombres y el 40% mujeres. Mientras que en cuanto a rango de edad, el 30,86% de los autores está comprendido en el rango 15-19, el 50,29% en el rango 20-24 y el 18,85% restante corresponde al rango 25-xx o 25 en adelante.

4 https://lahore.comsats.edu.pk/cs/MAPonSMS/index.html

		% de autores
Género	Hombre	60.00%
	Mujer	40.00%
Edad	15-19	30.86%
	20-24	50.29%
	25-xx	18.85%

Tabla 6.2.1 Distribución de clases en el corpus de entrenamiento de la tarea MAPonSMS para la identificación de género y rango de edad.

En este evento utilizamos FAI con la configuración que quedó establecida en el apartado 3.3, utilizando como características la media, distribución típica, skewness y longitud de los terciles de la distribución de probabilidades de pertenencia a cada clase [15]. Se eliminaron del vocabulario las palabras que aparecían en menos de 5 autores.[5]

En la tabla 6.2.2. presentamos nuestros resultados versus el *baseline* proporcionado por los organizadores[6]. El Baseline se estableció con la clase de control o dominante. Nuestros resultados mejoraron el *baseline* de género en casi un 30% y el de rango de edad en casi un 12%.

Género

Participante	Accuracy
Sharmila et al.	87.00%
Thenmozhi et al.	85.00%
Ali	83.00%
Garibo	77.00%
Deepanshu	75.00%
Kosmajac at al.	74.00%
Ramsha et al.	73.00%
Asmara et al.	69.00%
Baseline	60.00%
Abdul et al.	55.00%

Edad

Participante	Accuracy
Sharmila et al.	65.00%
Deepanshu	64.00%
Thenmozhi et al.	63.00%
Ali	60.00%
Kosmajac at al.	59.00%
Garibo	57.00%
Ramsha et al.	53.00%
Asmara et al.	53.00%
Baseline	51.00%
Abdul et al.	37.00%

Tabla 6.2.2. Resultados obtenidos y comparación respecto al *baseline* proporcionado.

5 Posteriores pruebas han determinado que para el caso de rango de edad, los resultados óptimos se consiguen sin eliminar palabras del vocabulario.

6 https://lahore.comsats.edu.pk/cs/MAPonSMS/de.html#results

6.3. HatEval en SemEval 2019.

SemEval[7] es un Taller Internacional de Evaluación Semántica en el que se llevan a cabo diferentes tareas de evaluación, tales como extracción de información y resolución de preguntas, procesamiento del lenguaje natural para aplicaciones ciéntificas, detección de opinión y emoción en el lenguaje, etc. Dentro de la categoría de detección de opinión, emoción y lenguaje abusivo se enmarca la tarea HatEval: *Multilingual Detection of Hate Speech Against Immigrants and Women in Twitter*[8]. El objetivo de la tarea es desarrollar un método automatizado de lenguaje homófobo o xenófobo en la red social Twitter en dos idiomas, inglés y español. Como parte de este trabajo hemos participado en dicha tarea utilizando la representación FAI de los textos.

La tarea HateEval consistió en dos sub-tareas:

- Tarea A: Detección de odio en el discurso contra inmigrantes y mujeres. Consiste en una clasificación binaria donde de debe predecir si un tweet implica o no odio contra mujeres o inmigrantes.
- Tarea B: Comportamiento agresivo y clasificación del objetivo. Los sistemas deben clasificar los tweets entre los que implican odio y los que no. Si un tweets implica odio se debe clasificar como agresivo o no agresivo y con objetivo individual o colectivo (contra una mujer o contra todas, por ejemplo).

	Idioma	Train	Evaluación
Número de tweets	Inglés	9.000	1.000
	Español	4.500	500

Tabla 6.3.1. Corpus HatEval.

Hemos participado en esta tarea en la sub-tarea B. En dicha sub-tarea hubo un total de veinticuatro equipos participantes con el corpus en español y cuarenta y dos en inglés. La medida de evaluación consiste en la proporción de tweets correctamente clasificados teniendo en cuenta las tres posibles etiquetas (odio, agresividad, número). Referimos dicha medida como EM (exact match).

En ambos idiomas el corpus está desequilibrado entre sus clases. Siendo mayoritarios los tweets que no implican odio. Entre ambos idiomas hay diferencias importantes en la distribución de las clases, como puede observarse en la tablaX(poner número).

7 alt.qcri.org/semeval2019/

8 https://competitions.codalab.org/competitions/19935

Idioma	Clase	Train	Evaluación
Inglés	Neutro	5.217	887
	Solo HS	1.350	113
	HS y AG	1.092	95
	HS y TR	874	110
	HS, AG y TR	467	109
Español	Neutro	2.643	278
	Solo HS	279	36
	HS y AG	449	49
	HS y TR	76	10
	HS, AG y TR	1.053	127

Tabla 6.3.2. Distribución de clases en el corpus HatEval.

Nuestra aproximación consiste en concatenar las diferentes etiquetas y abordar la tarea de clasificación en un solo paso. Es decir, tendremos 5 clases diferentes, con diferentes combinaciones de las posibles clases:

- HS: con valor 1 si denota odio en el lenguaje.
- AG: con valor 1 si denota agresividad en el lenguaje.
- TG: con valor 0 si el objetivo es individual y 1 si es colectivo.

HS	AG	TR	Etiqueta
0	0	0	000
1	0	0	100
1	0	1	101
1	1	0	110
1	1	1	111

Tabla 6.3.3. Etiquetado para el aprendizaje automático.

En esta tarea utilizamos FAI con la configuración que quedó establecida en el apartado 3.3, utilizando como características la media, distribución típica, skewness y longitud de los terciles de la distribución de probabilidades de pertenencia a cada clase. Se eliminaron del vocabulario las palabras que aparecían en menos de 5 autores.

En la tabla XXX presentamos los resultados obtenidos para la sub-tarea B de HatEval para ambos idiomas. Para el caso del idioma español nuestro método consigue quedar entre los 5 primeros clasificados, lo que permitirá que el método sea citado y descrito por los organizadores.

Inglés

Posición	Participante	EM
	MFC Baseline	0.58
1	ninab	0.57
2	iqaameer133	0.568
3	scmhl5	0.483
4	garain	0.482
5	gertner	0.399
6	amontejo	0.384
7	alonzorz	0.382
8	saagie	0.374
9	OscarGaribo	0.373
⋮	⋮	⋮
	SVC Baseline	0.308
⋮	⋮	⋮
42	abaruah	0.159

Español

Posición	Participante	EM
1	hammad.fahim57	0.705
2	iqaameer133	0.675
3	gertner	0.671
4	francolq2	0.657
5	OscarGaribo	0.6449
6	kwinter	0.638
⋮	⋮	⋮
12	choal	0.616
	SVC Baseline	0.605
⋮	⋮	⋮
16	Taha	0.593
	MFC Baseline	0.588
⋮	⋮	⋮
24	guzimanis	0.428

Tabla 6.3.4. Resultados obtenidos y comparación respecto a los *Baselines* proporcionados.

7. Conclusiones y trabajo futuro

En este trabajo hemos presentado un nueva representación de baja dimensionalidad que obtiene resultados similares a los obtenidos por los métodos considerados como estado del arte. Tanto LDSE como *Word2Vec* son métodos ampliamente utilizados tanto en la empresa como en el entorno académico, sobre todo como *baselines* para competiciones en las que se pretende mejorar sus prestaciones.

Un apunte importante es que todos los procesos se han ejecutado en un portátil con un procesador de 4 cores i3 y con 16 Gb de memoria RAM[9].

Con un volumen bajo de datos (todos los corpora excepto PAN13) los tiempos de proceso han sido similares tanto en LDSE, como Word2Vec, como en FAI. Obviamente, los vectores utilizados para ejecutar Word2Vec estaban pre-procesados con un corpus enorme (Wikipedia) en cada uno de los idiomas. Este tiempo, en caso de Word2Vec nos lo hemos ahorrado, no así la carga en RAM de dichos vectores.

En el caso de los datasets correspondientes al PAN13 es donde mejor hemos visto la ventaja de utilizar FAI. Mientras los algoritmos de clasificación convergían en el orden de minutos, en el caso de Word2Vec han tardado días. En castellano, en el caso de clasificación de género 2 días y en el caso de edad 5 días. Esto se debe a que tanto LDSE como FAI reducen la dimensionalidad a 6 multiplicado por el número de clases, mientras que el tamaño de embedding en Word2Vec se ha establecido en el que es asumido como estándar de 300.

FAI ha demostrado funcionar significativamente mejor que el resto en algunos casos. FAI es un método que es independiente del lenguaje analizado. Además, no requiere ningún conocimiento de la lengua analizada. De hecho, el autor de este trabajo no tiene ningún conocimiento de árabe y FAI ha demostrado tener un excelente comportamiento con textos escritos en dicha lengua.

En definitiva, FAI supone un nuevo método, independiente del lenguaje, que no requiere conocimientos del mismo, que reduce la dimensionalidad permitiendo abordar problemas con gran cantidad de datos en equipos de prestaciones medias. En el campo de la empresa, la productividad es una máxima necesaria. Generar los vectores de Word2Vec con todo el corpus de Wikipedia requerirá de un cluster con altas capacidades de procesamiento y mucha, muchísima RAM. Los vectores de probabilidad a priori de pertenencia de cada palabra a cada clase, que son el corazón de

9 Los procesos se han ejecutado en un solo hilo, de forma que solo uno de los cores se ha utilizado.

FAI, se pueden construir en un equipo doméstico. Así mismo, la clasificación de grandes volúmenes de texto se puede hacer de forma eficiente, rápida y en equipos de bajo coste.

FAI se podría utilizar conjuntamente con LDSE y/o Word2Vec para intentar mejorar las prestaciones del sistema clasificador. También podemos evaluar el comportamiento de los 3 y seleccionar el que mejor funcione dados nuestros datos. También podríamos evaluar la posibilidad de aplicar el método FAI para el caso de Word2Vec. En este trabajo hemos utilizado el método de general uso que consiste en asignar a cada palabra su vector de 300 características y calcular la media por columna de todas las palabras de un texto. Un ejercicio interesante consistiría en calcular también la desviación típica y el skewness, de forma que cada texto quedara caracterizado por 900 características en lugar de 300. Somos conscientes de que tal escenario ralentizaría el proceso de clasificación para conjuntos de datos grandes, pero valdría la pena estudiar su impacto en la precisión del modelo.

8. Referencias

[1] Rangel F., Franco-Salvador M., Rosso P. A Low Dimensionality Representation for Language Variety Identification. In: Postproc. 17th Int. Conf. on Comput. Linguistics and Intelligent Text Processing, CICLing-2016, Springer-Verlag, Revised Selected Papers, Part II, LNCS(9624), pp. 156-169 (arXiv:1705.10754)

[2] Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems 26. pp. 3111–3119 (2013)

[3] Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., and Dean, J. Distributed representations of words and phrases and their compositionality. In: Adv. NIPS.

[4] Grave, E., Bojanowski, P., Gupta, P., Joulin, A., Mikolov, T. Learning Word Vectors for 157 languages. In: Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018).

[5] Rangel, F., Rosso, P., Montes-y-Gómez, M., Potthast, M., Stein, B. Overview of the 6th Author Profiling Task at PAN 2018: Multimodal Gender Identification in Twitter. CLEF 2018 Labs and Workshops. Notebook Papers. CEUR Workshop Proceedings. CEUR-WS.org.

[6] Garibo, O. A Big Data Approach to Gender Classification in Twitter. CLEF 2018 Labs and Workshops. Notebook Papers. CEUR Workshop Proceedings. CEUR-WS.org/Vol-2125/paper_204.pdf.

[7] Zaghouani, W. and Charfi, A. ArapTweet: A Large Multidialect Twitter Corpus for Gender, Age and Language Variety Identification. In Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC), Mizayaki, Japan (2018).

[8] Joachims T. Text categorization with support vector machines: Learning with many relevant features. InEuropean conference on machine learning 1998 Apr 21 (pp. 137-142). Springer, Berlin, Heidelberg.

[9] Rumelhart DE, Hinton GE, Williams RJ. Learning internal representations by error propagation. California Univ San Diego La Jolla Inst for Cognitive Science; 1985 Sep.

[10] Johnson, R., Freund, J. Miller and Freund's Probability and Statistics for Engineers, 8th Ed. Prentice Hall International, 2011.

[11] Rangel, F., Rosso, P., Stein, B. Overview of the 5th Author Profiling Task at PAN 2017: Gender and Language Variety Identification in Twitter. In: Capellato, L., Ferro, N., Goeuriot, L., Mandl, T. (Eds) CLEF 2017 Labs

and Workshops. Norebook Papers. CEUR Workshop Proceedings. CEUR-WS.org, vol. 1866.

[12] Rangel, F., Rosso, P., Ghugur, I., Trenkmann, M., Stein, B, Verhoeven, B., Daelemans, W. Overview of the 2nd Author Profiling Task at PAN 2014. In: Capellato, L., Ferro, N., Halvey, M., Kraaij, W. (Eds) CLEF 2014 Labs and Workshops. Norebook Papers. CEUR-WS.org vol. 1180. pp. 898-827.

[13] Rangel, F., Rosso, P., Koppel, M., Stamatatos, E., Inches, G. Overview of the Author Profiling Task at PAN 2013. In: Forner P., Navigli R., Tufis D. (Eds.)Notebook Papers of CLEF 2013 LABs and Workshops. CEUR-WS.org, vol. 1179

[14] Litvinova, T., Rangel, F., Rosso, P., Seredin, P., Litvinova, O. Overview of the RusProfiling PAN at FIRE Track on Cross-genre Gender Identification in Russian. In: Notebook Papers of Fire 2017, Bangalore, India, December 8-11, CEUR Workshop Proceedings. CEUR-WS.org, vol. 2036, pp. 1-7.

[15] Garibo, O., Rangel, F. A Statistical Approach to Gender and Age Range Classification in Multilingual Corpus. In: Working Note of Fire 2018 – Forum for Information Retrieval Evaluation. CEUR-WS.org, vol.2266, pp. 277-281.

9. Anexo

Gráficas de la distribucion de probabilidad a priori de catroce de las clases de variedad del lenguaje en el corpus CMUQ100-ARAP [6].

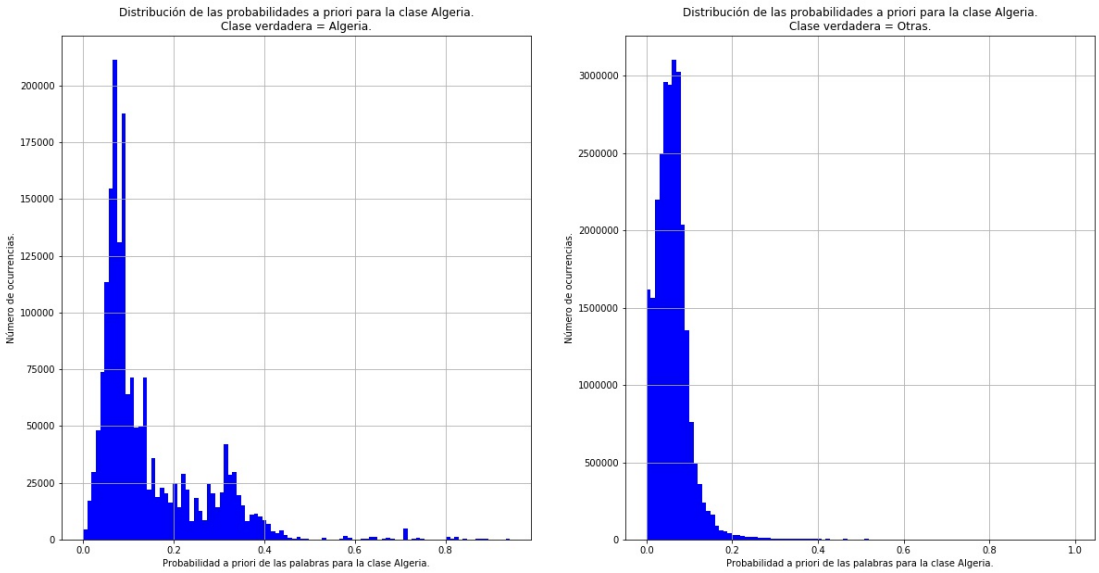

Figura A.1. Distribución de las probabilidades a priori de pertenencia a la clase Argelia y resto de variedades cuando el texto pertenece a la clase Argelia.

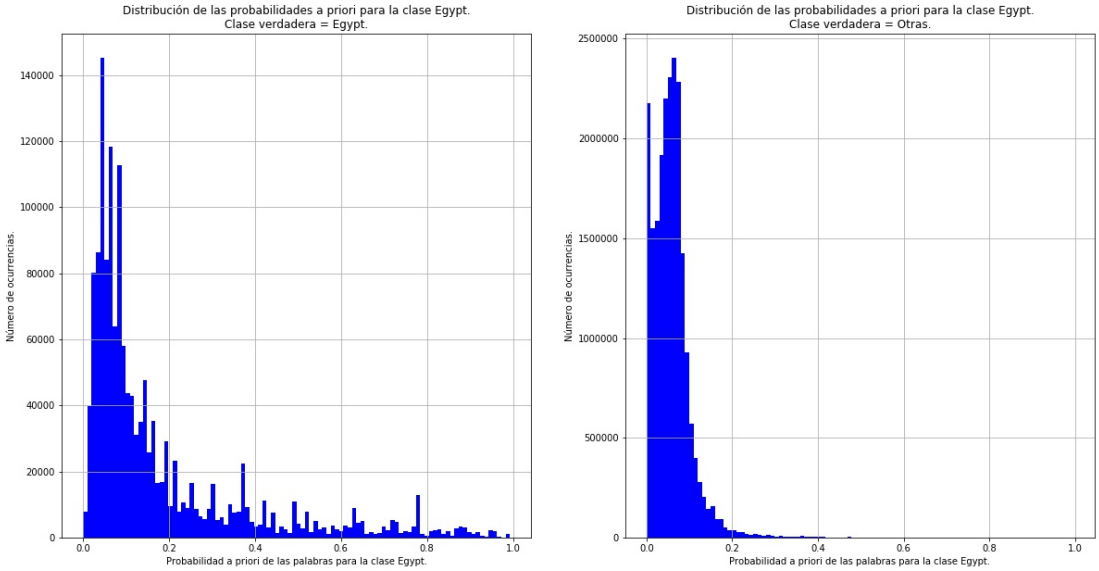

Figura A.2. Distribución de las probabilidades a priori de pertenencia a la clase Egipto y resto de variedades cuando el texto pertenece a la clase Egipto.

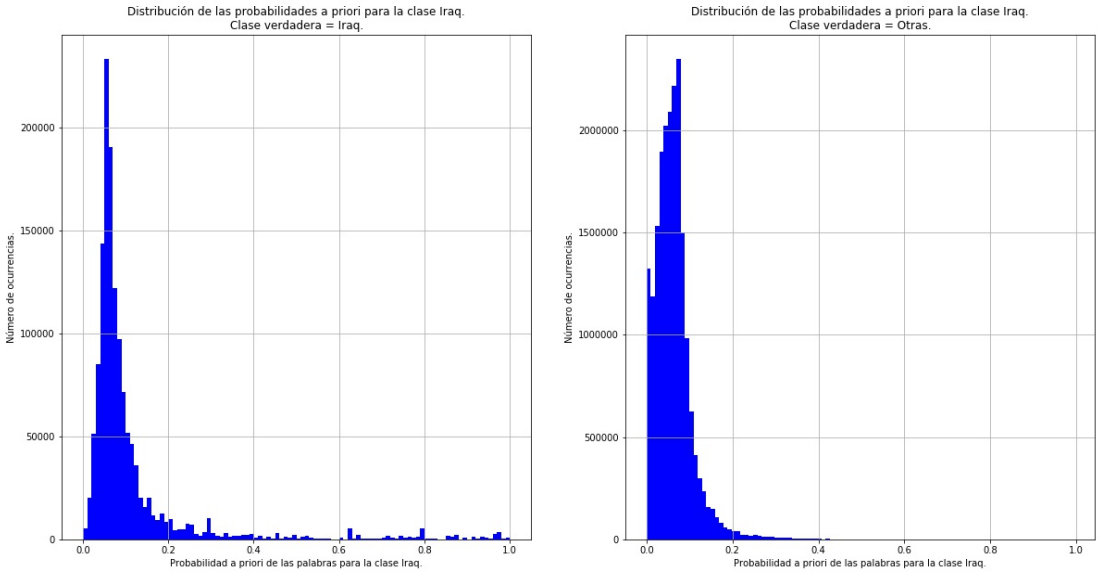

Figura A.3. Distribución de las probabilidades a priori de pertenencia a la clase Irak y resto de variedades cuando el texto pertenece a la clase Irak.

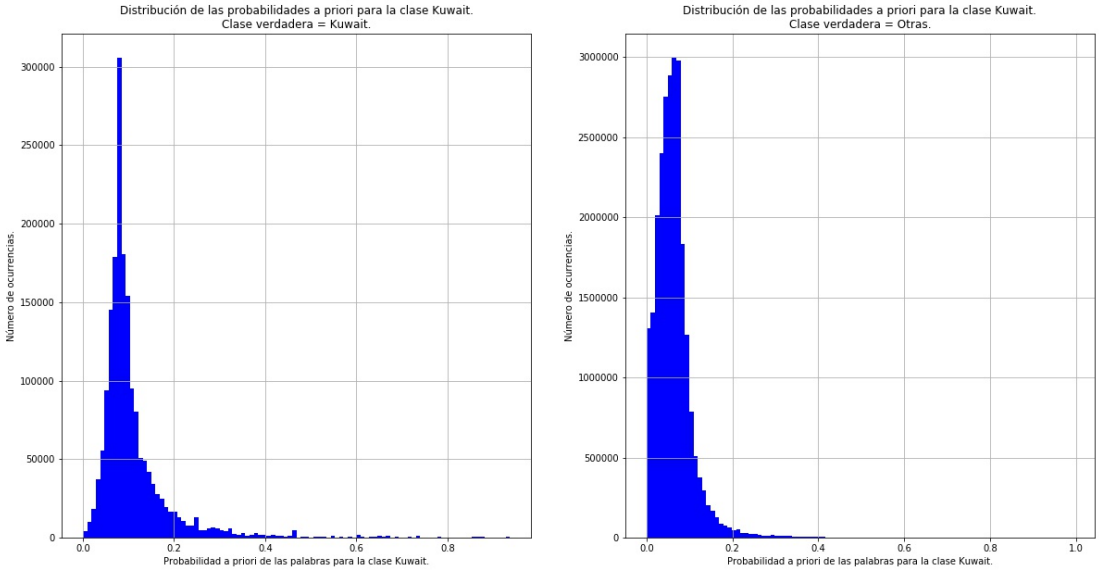

Figura A.4. Distribución de las probabilidades a priori de pertenencia a la clase Kuwait y resto de variedades cuando el texto pertenece a la clase Kuwait.

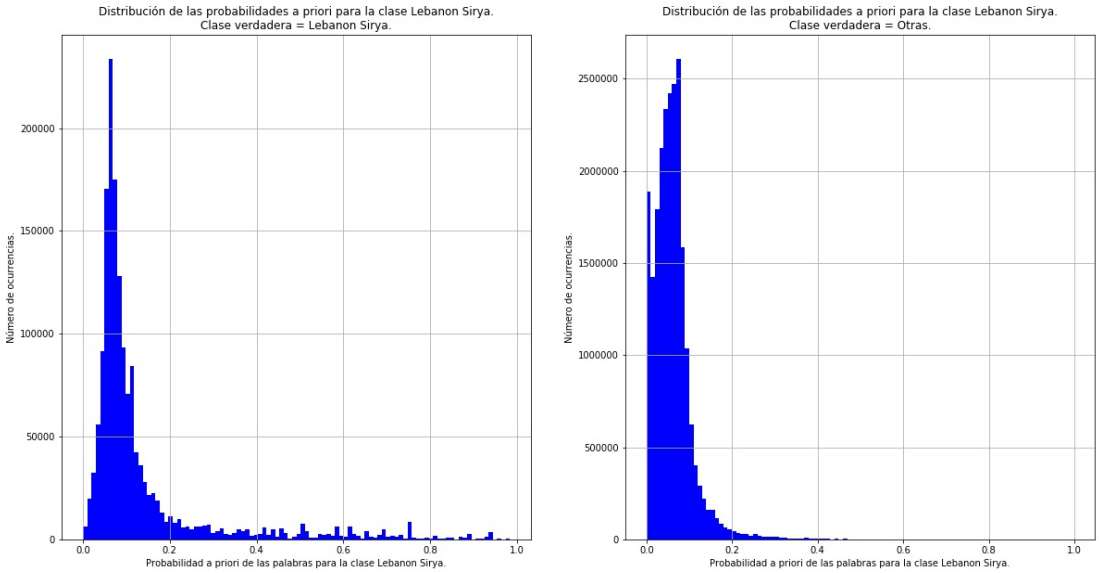

Figura A.5. Distribución de las probabilidades a priori de pertenencia a la clase Líbano Siria y resto de variedades cuando el texto pertenece a la clase Líbano Siria.

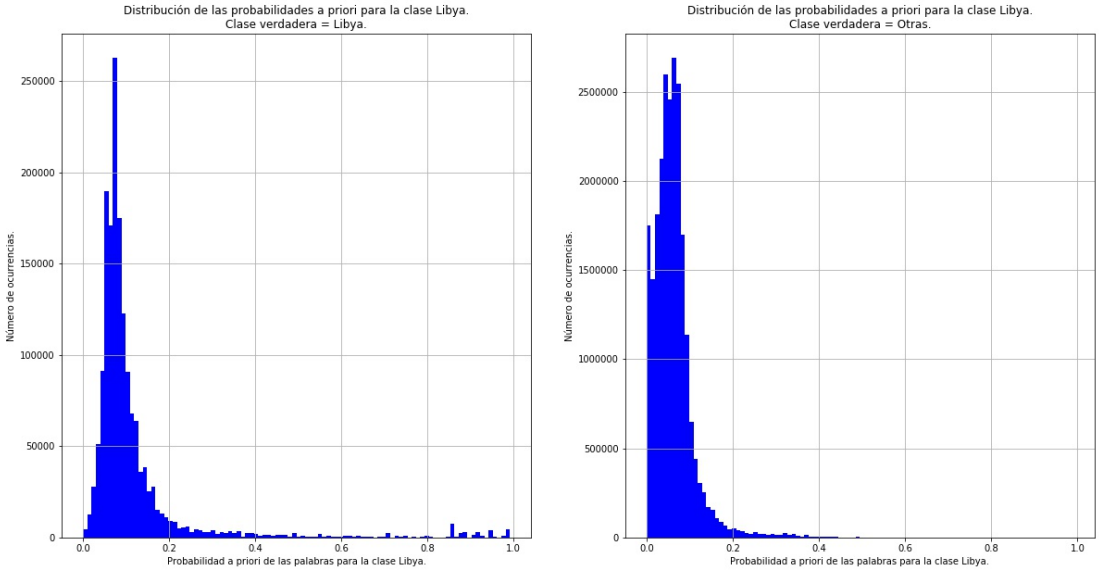

Figura A.6. Distribución de las probabilidades a priori de pertenencia a la clase Libia y resto de variedades cuando el texto pertenece a la clase Libia.

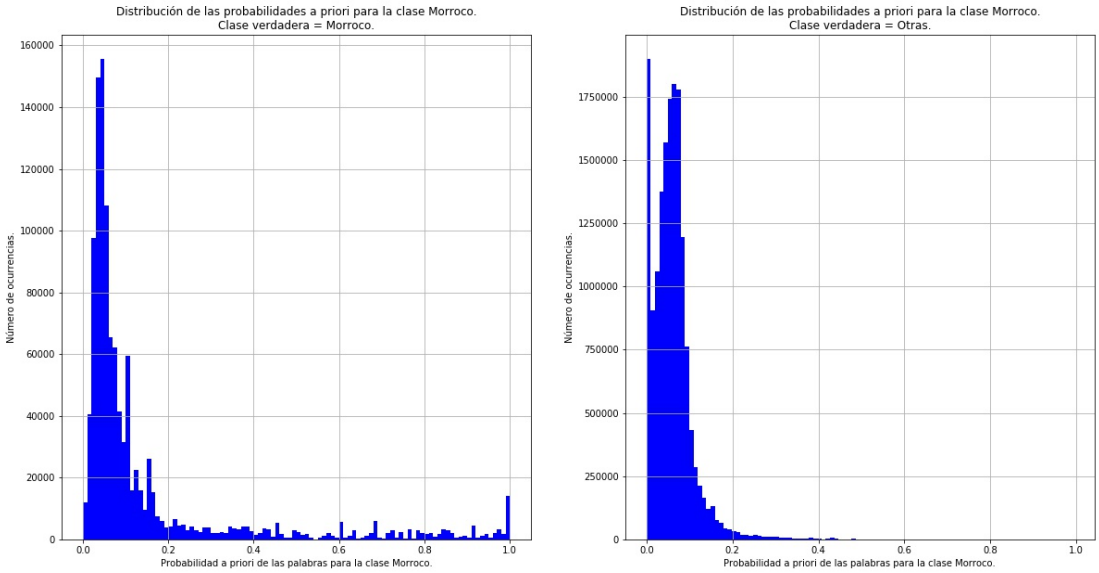

Figura A.7. Distribución de las probabilidades a priori de pertenencia a la clase Marruecos y resto de variedades cuando el texto pertenece a la clase Marruecos.

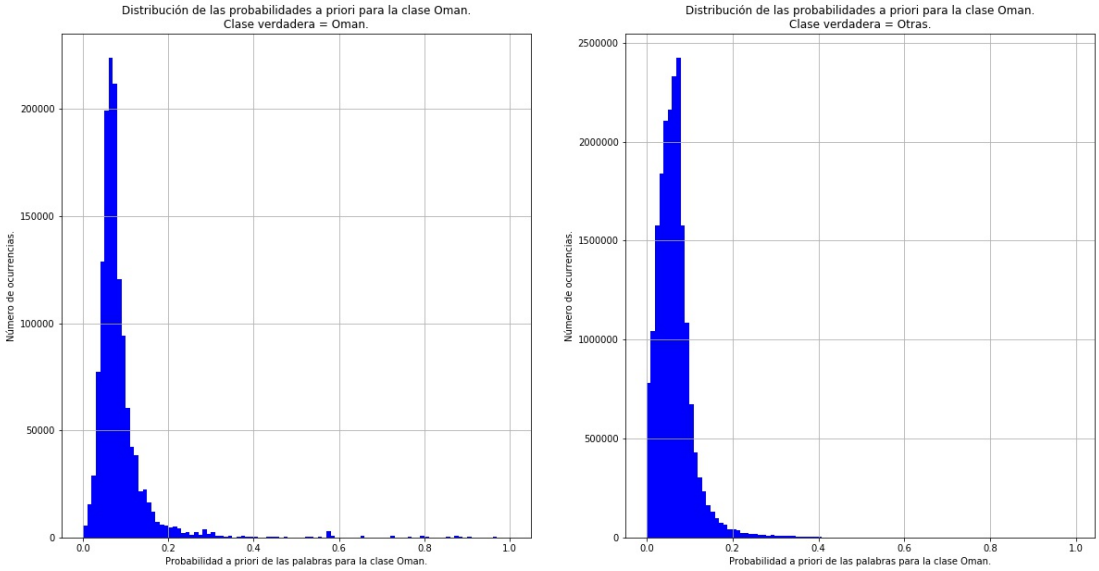

Figura A.8. Distribución de las probabilidades a priori de pertenencia a la clase Omán y resto de variedades cuando el texto pertenece a la clase Omán.

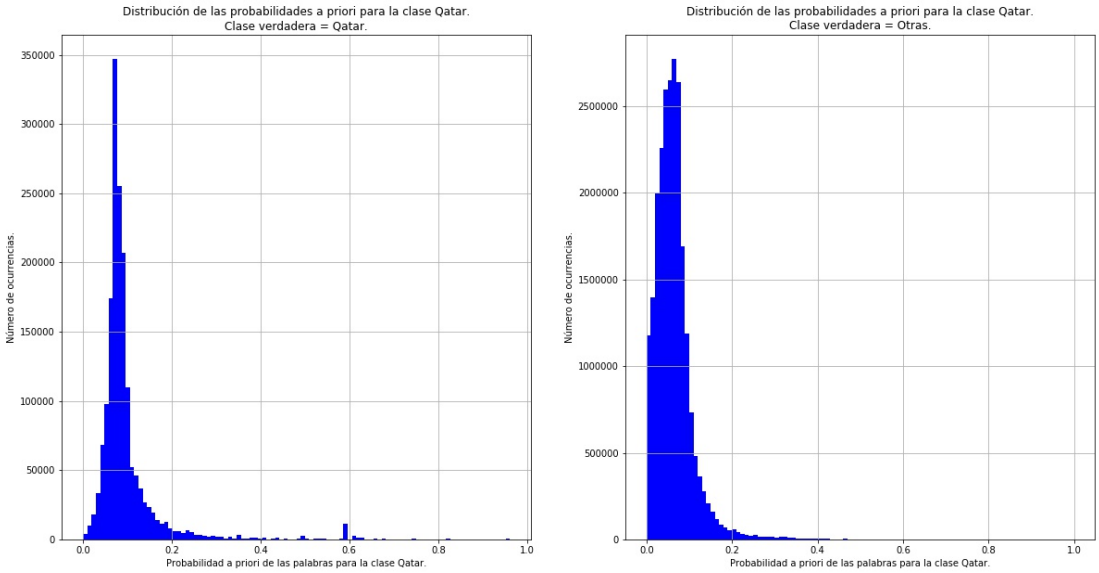

Figura A.9. Distribución de las probabilidades a priori de pertenencia a la clase Qatar y resto de variedades cuando el texto pertenece a la clase Qatar.

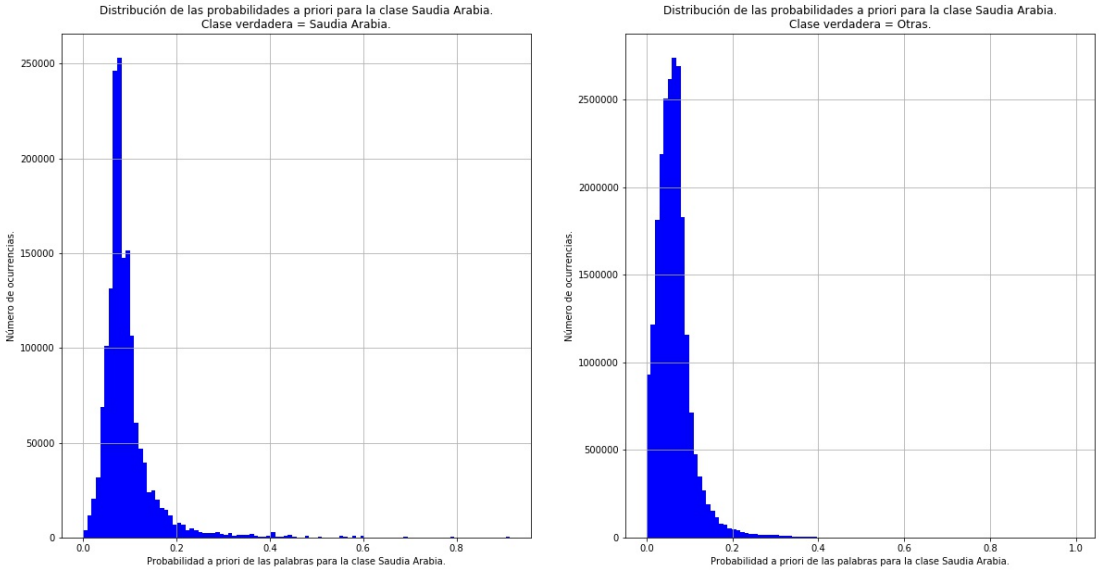

Figura A.10. Distribución de las probabilidades a priori de pertenencia a la clase Arabia Saudí y resto de variedades cuando el texto pertenece a la clase Arabia Saudí.

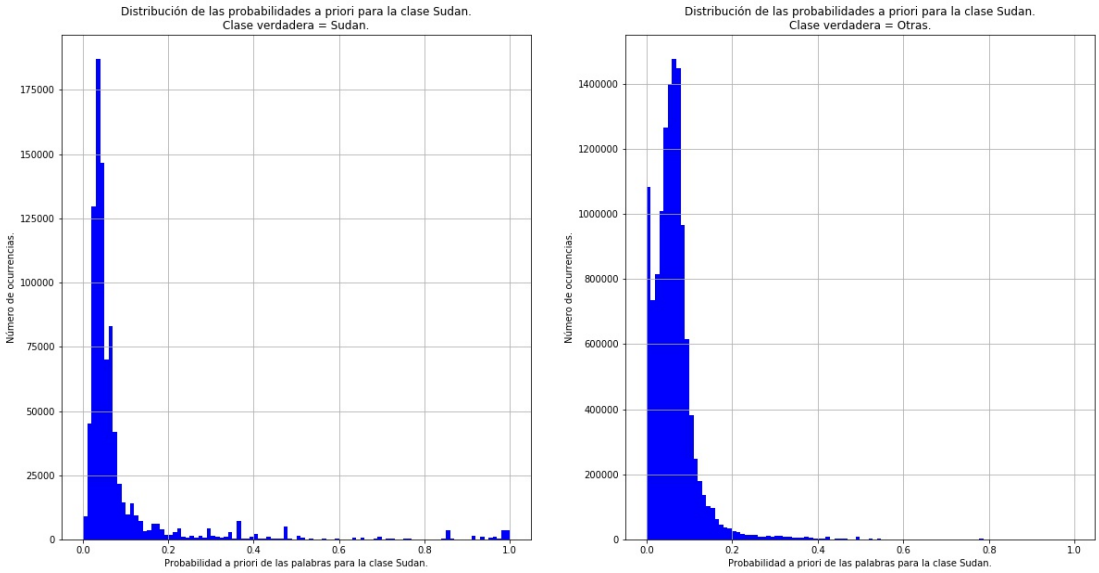

Figura A.11. Distribución de las probabilidades a priori de pertenencia a la clase Sudán y resto de variedades cuando el texto pertenece a la clase Sudán.

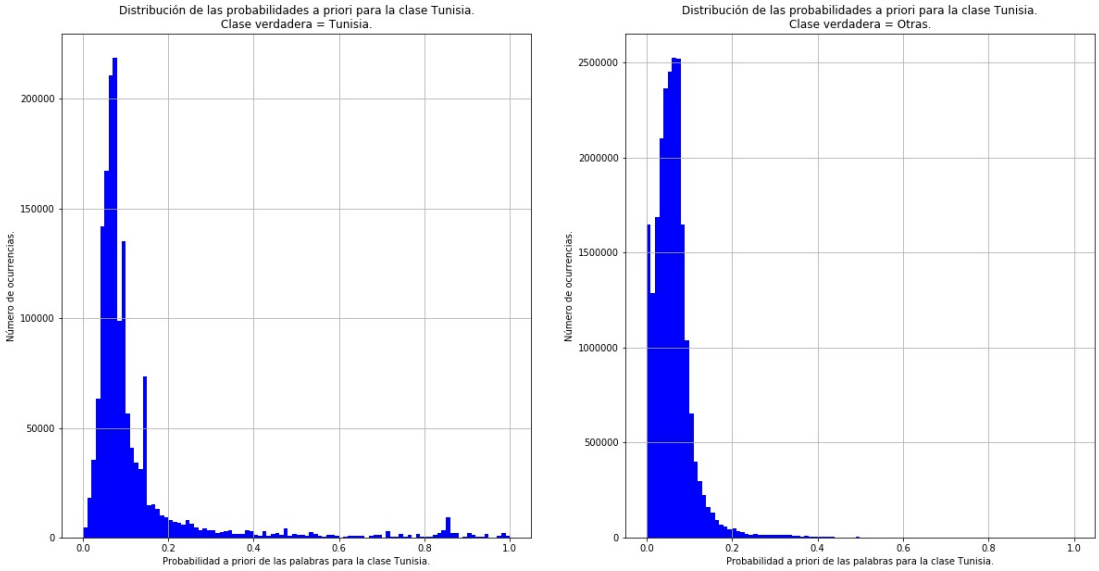

Figura A.12. Distribución de las probabilidades a priori de pertenencia a la clase Túnez y resto de variedades cuando el texto pertenece a la clase Túnez.

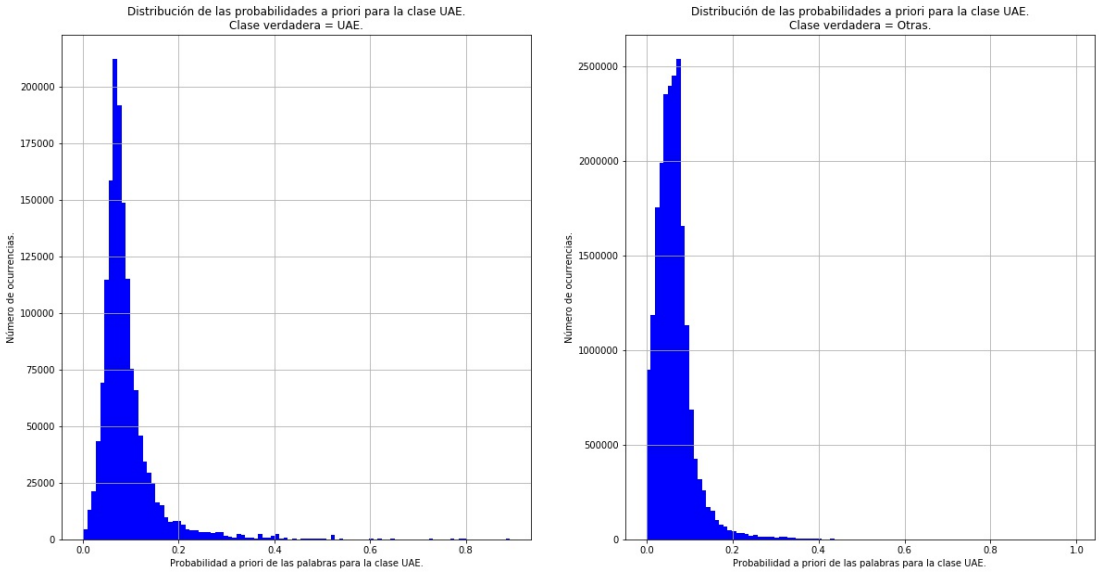

Figura A.13. Distribución de las probabilidades a priori de pertenencia a la clase UAE y resto de variedades cuando el texto pertenece a la clase UAE.

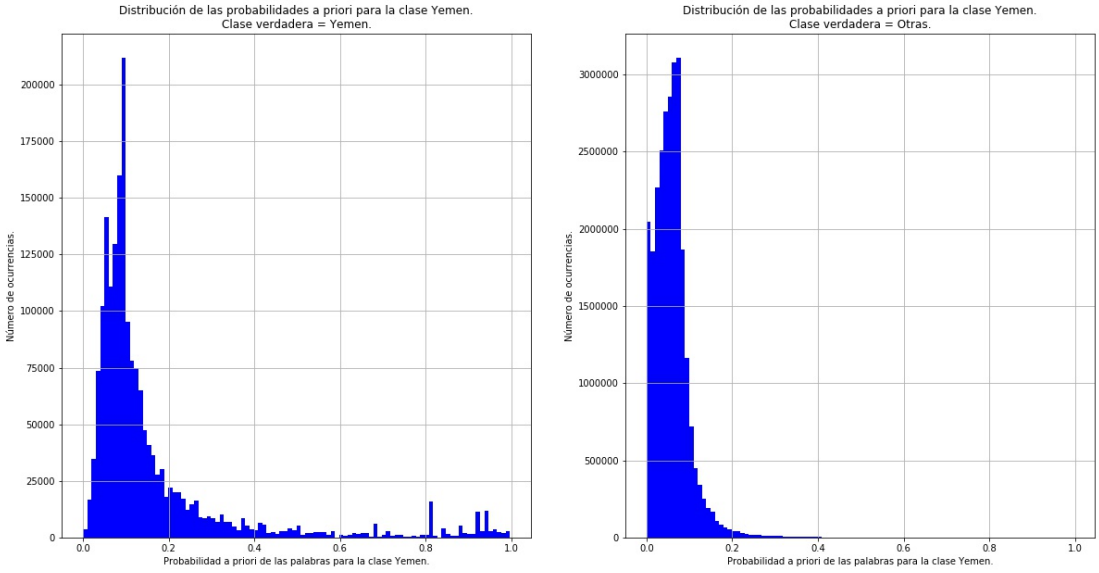

Figura A.14. Distribución de las probabilidades a priori de pertenencia a la clase Yemen y resto de variedades cuando el texto pertenece a la clase Yemen.