Un método de análisis de lenguaje tipo SMS para el castellano

A SMS-like language analyzer for Spanish

Andrés Alfonso Caurcel Díaz
Sistemas inteligentes para la comunicación y movilidad accesibles
Universidad Politécnica de Madrid
AARcaurcel@gmail.com

José María Gómez Hidalgo
Departamento de I+D
Optenet
jgomez@optenet.com

Yovan Iñiguez del Rio
Sistemas inteligentes para la comunicación y movilidad accesibles
Universidad Politécnica de Madrid
yovan.i.rio@gmail.com

Resumen

Debido a las características propias del lenguaje tipo SMS utilizado en las comunicaciones por medio de Internet y de los teléfonos móviles, no se puede realizar una tokenización o separación de palabras estándar a la hora de dividir en palabras una oración o frase. La cantidad de elementos no alfanuméricos que se pueden insertar en una palabra, los errores tipográficos y el hecho de no utilizar espacios entre palabras son las principales causas de este problema.

En este artículo presentamos un nuevo sistema de separación de palabras para el análisis del lenguaje natural en español en redes sociales y otras comunicaciones electrónicas. El sistema está integrado en una herramientas para la detección de edad en redes sociales enmarcada en el proyecto de investigación y desarrollo WENDY, y se evalúa cuantitativamente tanto de manera directa, como indirectamente en el marco de dicha aplicación, con resultados positivos en ambos casos.

Palabras clave

Lenguaje SMS, lenguaje chat, tokenizador, traductor automático, Procesamiento del Lenguaje Natural, detección de edad

Abstract

The usage of specific language codes and chat and SMS-like messages is a major trend in electronic communications. This fact makes Natrual Language Processing quite hard, even at the simplest step fo text message tokenization, due to the widespread usage of non-alphanumeric symbols, frequent typos and non-standard word separators.

In this work we present a new approach for text message tokenization, specific for the Spanish language as used in Social Networks and in electronic communications. Our system has been integrated in a more general application for age-detection in Social Networks developed in the research and development project WENDY, and it has been quantitatively evaluated both in a direct fashion, and indirectly by its impact on the genearl age-detection application, showing very promising results.

Keywords

SMS language, chat language, tokenizer, automated translation, Natural Language Processing, Age detection

1 Introducción

A la hora de crear aplicaciones informáticas que traten con los nuevo sistemas de comunicación como la mensajería instantánea, chats, foros y redes sociales, es imprescindible abordar la problemática de las características propias del uso del idioma en estos entornos. El uso del idioma en estos ámbitos ha dado lugar a una suerte de dialecto escrito, que con frecuencia se llama lenguaje SMS o lenguaje tipo chat (Forsyth, 2007). Algunas de las características de dicho lenguaje no dependen del idioma del que derivan (por ejemplo español vs. inglés), como el uso de emoticonos, determinadas pautas como la repetición de vocales o la eliminación completa de las mismas, o el uso abusivo de mayúsculas o su ausencia total. Otras características sí que dependen del idioma, proviniendo de siglas de expresiones populares (por ejemplo “LoL” - “Laughing out Loud” en inglés, o “a.p.s.” - “amigos para siempre” en español), o de abreviaciones fonéticas (por ejemplo “cya” - “see you” en inglés, o “xq” - “porque” en español).

Estas características pueden dificultar la lectura a algunos usuarios, y desde luego siempre a cualquier sistema de análisis automático del lenguaje. Por ejemplo, una frase como “felicidadees!! k t lo pases muy bien!! =)Feeeliiciidaaadeeess !! (:Felicidadesss!!pasatelo genialll :DFeliicCiidaDesS! :D Q tte Lo0 paseS bN! ;) (heart)” puede resultar muy complicada tanto para su lectura por parte de un usuario, como para el análisis por parte de un sistema usualmente preparado para el lenguaje normalizado del que deriva, el español estándar. Incluso la simple fragmentación de la frase anterior en constituyentes básicos como las palabras que la componen,

proceso denominado usualmente tokenización[1], se convierte en un proceso no trivial y cuyos errores pueden llevar a importantes pérdidas de efectividad de cualquier sistema de análisis del lenguaje natural.

Asimismo, el uso de símbolos de puntuación y caracteres no alfanuméricos para símbolos y dibujos obliga a que el tokenizador también respete dichas estructuras para su clasificación y su traducción al lenguaje normalizado si esta fuese necesaria. Los sistemas de tokenización estándar utilizan estos símbolos como límites de palabra, por lo que son eliminados o separados del resto de símbolos (dependiendo de las opciones del tokenizador), perdiéndose por tanto la información que acarrean.

La tokenización de textos electrónicos provenientes de comunicaciones informales en Internet (chats, mensajes cortos, comentarios en foros, etc.) es un tema que ha despertado bastante interés en la comunidad del Procesamiento del Lenguaje Natural, pero los estudios que se han realizado hasta el momento son fundamentalmente para el idioma inglés – por ejemplo (Forsyth, 2007), ya que éste es el más utilizado por los usuarios de la red. No existe ningún estudio comparativo para lenguas romances, aunque existen trabajos para japonés, chino y turco (Ptaszynski, 2010) (Yin, 2009) (Pendar, 2007).

En este trabajo presentamos un nuevo sistema para la tokenización del español en el contexto del lenguaje informal escrito en redes sociales, chats, SMS y mensajería instantánea, orientado a mejorar el reconocimiento de los constituyentes de la oración. Se trata de un sistema que trabaja en dos fases, extrayendo primero los candidatos a constituyentes por medio de una tokenización simple, para luego analizarlos y descomponerlos sucesivamente de acuerdo con una serie de patrones de uso del lenguaje tipo SMS y con la ayuda de recursos lingüísticos. El sistema se dirige exclusivamente hacia el idioma español, y no se ha evaluado su posible validez sobre otros idiomas.

Este tokenizador forma parte de un sistema general orientado a la detección de la edad de los usuarios de redes sociales de acuerdo con patrones de biometría del comportamiento, enmarcado en en el proyecto de investigación de protección al menor WENDY (WEb-access coNfidence for chilDren and Young[2]). Este marco ha permitido evaluar el método de tokenización tanto de manera directa (en función de sus resultados explícitos), como de manera indirecta (en función de como afecta a la efectividad del sistema de detección de edad). En ambos casos se han obtenido resultados positivos.

El resto de este artículo está organizado de la siguiente manera. En primer lugar presentamos el marco general de trabajo, que es el proyecto WENDY, y que permite definir los requisitos particulares del procesamiento de texto en el entorno de las redes sociales. A continuación describimos el sistema de tokenización desarrollado, y seguidamente el entorno de evaluación y los resultados obtenidos. Finalizamos el artículo con las conclusiones obtenidas y las propuestas de trabajo futuro.

2 Marco general: el proyecto WENDY

La investigación propuesta en este trabajo se engloba dentro del proyecto de investigación WENDY, que plantea el desarrollo de un sistema de clasificación para detectar la edad de usuarios de redes sociales en castellano relativamente similar al presentado en (Tam, 2009). Los rangos propuestos en el sistema son menores de catorce años (-14) mayores de catorce años pero menores de edad (+14) y mayores de edad (+18). Estos rangos están motivados por la existencia de dos comportamientos significativos a detectar en el marco de la protección del menor en redes sociales:

- Los menores de 14 años no pueden estar dados de alta en una red social sin consentimiento paterno de acuerdo con la legislación española, y con mucha frecuencia mienten con respecto a su edad. Por ejemplo, un estudio reciente revela que se puede estimar en unos 5,6 millones el número de menores de 13 años dados de alta en Facebook[3] en EE.UU.
- Los mayores de 18 años que simulan ser menores de edad pueden estarlo haciendo para establecer contacto con menores, y potencialmente convertirlos en víctimas de acoso sexual ("*grooming*"). Es frecuente encontrarse con casos de este tipo en los medios de comunicación[4].

El sistema desarrollado en WENDY se basa en el análisis del comportamiento de los usuarios dentro de la popular red social Tuenti[5]. En esta red social, los usuarios pueden describir sus gustos (cine, libros, etc.), escribir mensajes de estado, comentarios a los mensajes de sus contactos,

1 La expresión "tokenización" se utiliza con frecuencia en los artículos científicos en español dedicados al análisis del lenguaje natural.

2 http://wendy.optenet.com.

3 http://www.europapress.es/portaltic/socialmedia/noticia-hay-mas-millones-ninos-facebook-si-no-quieren-20120920112441.html.

4 http://www.antena3.com/noticias/sociedad/detenido-entrenador-futbol-infantil-acosar-menores-internet_2012102000066.html.

5 http://www.tuenti.com.

mantener conversaciones por chat y subir o ver fotografías y vídeos. El prototipo desarrollado en WENDY analiza todos los elementos textuales y las fotografías con el fin de determinar la edad del usuario, y avisa al administrador de la red social cuando identifica a un usuario cuyo comportamiento no se corresponde con el habitual de su franja de edad, o simplemente cuando induce que es menor de 14 años o mayor de 18 y no refleja su edad en su perfil.

El sistema integra una serie de módulos especializados por cada tipo de información: gustos del perfil, comentarios, fotografías, etc. En este artículo nos centramos específicamente en dos de los elementos textuales, que son el perfil y los comentarios (sus actualizaciones de estado y las respuestas a otros comentarios).

Para el análisis de estos tipos de información, se ha diseñado un sistema basado en aprendizaje que analiza los elementos textuales y entrena una serie de clasificadores orientados a las franjas de edad descritas anteriormente. Para ello se ha confeccionado una colección de datos obtenidos de perfiles reales de Tuenti compuesta por más de 120.000 perfiles distribuidos de la siguiente manera: Menores de 14 años: 372; de 14 a 17 años: 11.530; más de 18 años: 7.432; sin información sobre la edad: 100.670. Se puede observar que la clase de los menores de 14 años está escasamente representada, lo que dificulta enormemente el entrenamiento de clasificadores efectivos.

Una de las hipótesis de trabajo fundamentales en el proyecto es que el uso del lenguaje tipo SMS puede ser un elemento discriminador en la clasificación por edades. Por ello, se han desarrollado una serie de sistemas orientados al procesamiento de este tipo de lenguaje, que hacen uso del tokenizador descrito en este artículo. A continuación describimos los clasificadores de texto utilizados, así como una serie de sistemas de análisis y normalización del lenguaje tipo chat empleados sobre los textos obtenidos en dicha red social.

2.1 Clasificador textual

El clasificador de los textos por edades es un clasificador de texto tradicional – véase (Sebastiani, 2002), en el que los textos:

- Se separan en unidades lingüísticas o términos utilizando el tokenizador que describimos más adelante.
- Los términos obtenidos se traducen y normalizan usando el traductor de lenguaje SMS y el sistema Deflogger descritos en las próximas secciones.
- Se representa cada texto por medio de vectores de pesos de términos en base al modelo del espacio vectorial con pesos de tipo TF.IDF.
- Se seleccionan aquellos términos más predictivos de acuerdo con el criterio de que su Ganancia de Información respecto a las clases constituidas por las franjas de edad sea mayor que cero.
- Se aplica el algoritmo de aprendizaje Bayes Ingenuo, que predice la probabilidad de pertenencia de un texto a una determinada franja de edad en función de las probabilidades condicionadas de pertenen-cia de cada término a dichas franjas.

Se ha utilizado la implementación de Bayes Ingenuo (Naive Bayes) incluida en el paquete de aprendizaje WEKA[6] (Hall et al. 2009), con sus opciones por defecto.

2.2 Traductor de lenguaje SMS

Se ha desarrollado un sistema de traducción de los elementos textuales utilizando una serie de recursos lingüísticos preexistentes, a saber: un diccionario de lenguaje tipo SMS para el castellano[7], y un diccionario del idioma castellano[8] (Padró et al., 2010).

El sistema traductor funciona de la siguiente manera:

1. Dada una palabra objetivo (posiblemente una expresión en SMS), se busca la misma en el diccionario de castellano.
2. Si la palabra aparece en el diccionario, se finaliza el proceso. Si la palabra no aparece en el diccionario, entonces se busca en el diccionario de lenguaje SMS.
3. Si la palabra aparece en el diccionario SMS, se selecciona el significado (o traducción) más frecuente o popular. Si la palabra no aparece en el diccionario SMS, se deja como está.

Es preciso resaltar que el diccionario SMS no sólo incluye expresiones coloquiales como “xq” (por “porque”), sino que también contiene una cantidad significativa de emoticonos (por ejemplo “:-)”, etc.). De ahí que se precise que la tokenización sea capaz de mantener y reconocer estos símbolos.

El proceso de traducción se aplica a los textos antes de entrenar un clasificador sobre los mismos.

6 http://www.cs.waikato.ac.nz/ml/weka/.
7 http://www.diccionariosms.com/contenidos/.
8 El incluido en el sistema de análisis lingüístico Freeling: http://nlp.lsi.upc.edu/freeling/.

2.3 Analizador tipo Deflogger

La informalidad del lenguaje escrito en foros y redes sociales ha motivado el desarrollo de analizadores específicos capaces de hacer frente a fenómenos como los siguientes:

- Contaminación con abreviaturas típicas del lenguaje SMS (por ejemplo "tkm" por "te quiero mucho", "bss" por "besos", "dsp" por "después",etc.).
- Alternancia de mayúsculas y minúsculas (por ejemplo "AlGuNa LeTrA dE uNa CaNcIoN").
- Repetición de letras (por ejemplo "hoooolllaaaaaa!!!!").
- Omisión de la letra "u" en la las sílabas "que" o "qui", o bien, reemplazo por "k" (por ejemplo "qiero", "kiero").
- Reemplazo de la sílaba "ca" o de la letra "c" por la letra "k" (reemplazo fonético, por ejemplo "kompre kfe").
- Faltas ortográficas intencionales como "soi", "voi", "i" en lugar de "y", etc.

Estos fenómenos y otros se encuentran reconocidos y analizados en la herramienta Deflogger[9], que es un sistema desarrollado para normalizar el lenguaje utilizado en la red social Fotolog[10] y aproximarlo al castellano normal, y que hemos integrado en nuestro sistema. Hemos aplicado el sistema Deflogger a los textos disponibles después de su traducción con el traductor de lenguaje tipo SMS, y antes de entrenar el clasificador.

3 Descripción del tokenizador

El sistema de separación de términos debe partir de un texto en lenguaje natural (un comentario de texto, un gusto musical, una lista de películas, etc.) y devolver un conjunto de tokens o unidades textuales básicas compuesto por cadenas de caracteres.

El sistema WENDY ha sido implementado en el lenguaje Java, y se ha desarrollado aprovechando el paquete de aprendizaje automático WEKA, que además del algoritmo de aprendizaje, incluye también métodos de selección de atributos por Ganancia de Información, e incluso un tokenizador simple que separa los textos en palabras de acuerdo a una serie de caracteres configurables por el usuario: espacios, tabuladores, saltos de línea, símbolos de puntuación, etc.

El separador de palabras desarrollado en WENDY parte de la implementación de este tokenizador básico, y aplica dos fases de separación. La primera fase consiste en separar la frase inicialmente por medio de espacios en blanco. Cada parte de frase separada por espacios ofrece tres posibilidades:

- Que sea una palabra propiamente dicha (secuencia de caracteres alfanuméricos).
- Que sea una estructura de signos de puntuación y caracteres no alfanuméricos (posiblemente un emoticono).
- Que sea una mezcla de ambas.

En los dos primeros casos, se considera que las secuencias de caracteres son ya palabras de por sí. Esto implica que las secuencias de caracteres no alfanuméricos, que en particular incluyen a los símbolos de puntuación aislados, se consideran tokens relevantes.

En el tercer caso, se procede a una segunda tokenización de la sub-frase, reconociendo como tokens todas las sub-secuencias de caracteres del mismo tipo – alfanuméricos vs. no alfanuméricos. Por ejemplo, dada la secuencia "Hola:-)ketal", ésta se separaría en esta segunda fase en los tokens siguientes: "Hola", ":-)", y "ketal".

Adicionalmente, se ha tenido en cuenta el sistema de emoticonos propio del chat de Tuenti, en el que las letras en mayúsculas entre paréntesis se representan automáticamente con un símbolo. Por ejemplo, "(L)" se representa gráficamente como un corazón, o "(M)" como una nota musical.

En un ejemplo real, si aplicamos el proceso completo a la frase siguiente:

> *"Felicidades LauraHey, felicidades! ^^felicidiadeees;DFelicidades!Un beso! FELIZIDADESS LAURIIIIIIIIIIII (LL)felicidadeeeeeees! :D jajaja mira mi tablonme meo jajajajajjajate quiero(:,"*

Se obtiene como salida un vector cuyos componentes se muestran en la tabla 1:

Felicidades	*LauraHey*	*,*
felicidades	*!*	*^^*
felicidiadeees	*;*	*DFelicidades*
!	*Un*	*beso*
!	*FELIZIDADESS*	*LAURIIIIIIIIIIII*
(LL)	*felicidadeeeeeees*	*!*
:D	*jajaja*	*mira*
mi	*tablonme*	*meo*
jajajajajjajate	*quiero*	*(:,*

Tabla 1: Vector de salida de ejemplo usando el nuevo sistema de separación.

9 http://code.google.com/p/deflog/.

10 http://www.fotolog.com.

Como se puede observar, la tokenización no es perfecta. Por ejemplo, en la expresión "felicidiadeees;DFelicidades" no se reconoce individualmente como emoticono ";D", debido a que la letra "D" se considera anexa a la siguiente palabra. Igualmente, no se separan las palabras "jajajajajjaja" y "te", ya que se trata de una única secuencia alfanumérica.

La implementación del tokenizador ha sido realizada en Java, partiendo del tokenizador previamente existente `NgramTokenizer`[11] ya existente en el paquete de aprendizaje WEKA. Este sistema de separación genera secuencias de tokens de las longitudes deseadas (ngramas de tokens), y acepta los siguientes parámetros:

- Secuencia de separadores, que es una cadena que incluye los elementos que se usan como separadores entre tokens. La cadena de separadores por defecto es: " \r\n\t.,;:'"()?!".
- Número mínimo de tokens en secuencia, por defecto 1.
- Número máximo de tokens en secuencia, por defecto 3.

Los últimos dos parámetros determinan el tamaño N de los ngramas. Este tokenizador se utiliza de manera integrada dentro del filtro `StringToWordVector`[12] incluido en el paquete WEKA, que acepta un tokenizador o separador como parámetro entre otros, y que transforma una colección de ejemplares textuales en una representación de bolsa de palabras o de vectores de pesos de términos según el Modelo del Espacio Vectorial (Sebastiani, 2002).

Dado que el nuevo separador desarrollado, que hemos llamado `SMSTokenizer`, desciende de la clase `Tokenizer` del paquete WEKA, puede ser usado como cualquier otro separador dentro de este paquete de aprendizaje.

4 Evaluación del tokenizador

Para evaluar la utilidad de este nuevo método de separación, nos hemos planteado dos preguntas importantes:

- ¿Se identifican mejor los tokens del lenguaje?
- ¿El nuevo tokenizador mejora los resultados de clasificadores en textos SMS?

La primera pregunta se corresponde con una evaluación directa o de tipo caja de cristal, donde se analizan los resultados concretos del proceso de tokenizado. La segunda pregunta se corresponde con una evaluación indirecta o de tipo caja negra, y es también la más importante por cuanto el objetivo del sistema global de WENDY es reconocer con la máxima eficacia posible la edad de los usuarios en función de sus interacciones textuales.

En ambos casos se ha partido del conjunto de frases de comentarios descargados de Tuenti con edades identificadas en sus perfiles, obteniéndose un total de 84.956 frases. Los textos tienen un alto contenido de lenguaje SMS. En dichos textos, han sido traducidas las marcas de lenguaje HTML que contenían y se han sustituido las marcas propias de la red social por medio de identificadores de las mismas (e.g: hiperenlaces, frases automáticas del sistema). La distribución de comentarios por rango de edad se muestra en la Tabla 2.

Clase	Número	Porcentaje
-14	1.810	2,13%
+14	59.778	70,36%
+18	23.368	27,51%
Total	84.956	100,00%

Tabla 2: Distribución de comentarios por clase.

Como se puede observar, y en línea con las estadísticas generales de la colección de datos descritas en la Sección 2, la clase más sensible (los menores de 14 años, denotada por "-14") es la menos representada, y la siguiente clase más sensible (los mayores de 18 años, denotados por "+18") es bastante minoritaria. Dada esta distribución de clases, podemos afirmar que la detección de la edad vinculada con estos comentarios es un problema bastante difícil, ya que los algoritmos de aprendizaje tienden a intentar optimizar la eficacia global y no la eficacia sobre determinadas clases. Por ejemplo, en este caso un sistema trivial que clasificase todo comentario como perteneciente a un menor entre 14 y 17 años (clase denotada por "+14"), tendría ya una eficacia del 70%.

Como sistema de tokenización de referencia o línea base, hemos utilizado el NgramTokenizer original, centrándonos en los ngramas de longitud uno (es decir, se invoca con número mínimo y máximo de tokens 1, y con los separadores por defecto).

4.1 Evaluación directa

Para evaluar de manera directa la calidad del nuevo sistema, debemos tener en cuenta dos aspectos:

- En primer lugar, el objetivo es reconocer mejor los elementos individuales del lenguaje o palabras aisladas, ya sean palabras del castellano estándar o palabras incluidas en el

11 http://weka.sourceforge.net/doc.dev/weka/core/tokenizers/NGramTokenizer.html.

12 http://weka.sourceforge.net/doc.dev/weka/filters/unsupervised/attribute/StringToWordVector.html.

diccionario de lenguaje SMS usado en el proyecto WENDY.

- En segundo lugar, el procesamiento de los textos no es un proceso aislado, sino que está enmarcado dentro del proceso general de detección de edad, por lo que a los textos originales se le debe aplicar el procedimiento antes mencionado de traducción de lenguaje tipo SMS.

Recordando que el proceso de traducción de lenguaje tipo SMS busca secuencialmente las palabras o tokens aislados en los diccionarios del castellano estándar y en el diccionario SMS, se ha realizado el proceso de traducción y se han computado el número de palabras detectadas sucesivamente en cada uno de los pasos de la traducción. A continuación, hemos comparado las palabras encontradas en cada diccionario usando el separador por defecto y el nuevo método de separación.

En las tablas 3, 4 y 5 se muestra la diferencia entre el número de tokens de cada comentario encontrado en el diccionario de castellano estándar (tabla 3), en el diccionario de lenguaje SMS (tabla 4) y no encontradas (tabla 5). En la primera columna se muestra el número medio de palabras de diferencia entre usar el tokenizador original y el nuevo tokenizador, mientras que en la segunda columna se muestra la desviación típica de esta diferencia, y en la tercera y cuarta columnas se muestran las diferencias mínimas y máximas.

Por ejemplo, examinando la tabla 3, se puede observar que el nuevo tokenizador reconoce 9,5 palabras más que el tokenizador básico por comentario, en media. Es decir, que si al realizar el proceso de traducción se encuentran X palabras de un comentario en el diccionario de castellano estándar usando el separador original, usando el nuevo separador se encuentran X + 9,5, lo que indica que el nuevo tokenizador reconoce mejor las palabras del castellano estándar incluidas dentro de los comentarios.

En la tabla 4 también se puede observar una cierta ganancia (en este caso, sólo de 1,13 palabras de media reconocidas por comentario), mientras que en la tercera tabla, el comportamiento es coherente, ya que al mostrar un número negativo (que es igual a la suma de los correspondientes a las tablas 3 y 4), indica que quedan menos palabras o tokens por reconocer por cada comentario. Cabe señalar que es razonable esperar que se localicen menos términos en el diccionario SMS, ya que sólo se buscan en él los términos que no han sido localizados en el diccionario de castellano, y que su tamaño es de dos órdenes de magnitud menor que el del diccionario del castellano.

Media	Desviación	Mínimo	Máximo
9,5	14,63	-58,93	100

Tabla 3: Diferencia de aciertos en diccionario castellano.

Media	Desviación	Mínimo	Máximo
1,13	4,09	-41,67	50

Tabla 4: Diferencia de aciertos en elementos SMS.

Media	Desviación	Mínimo	Máximo
-10,62	15,56	-100	58,93

Tabla 5: Diferencias de tokens no encontrados.

Dado que existe una desviación típica tan grande en las tabla 3 y 4, hemos tomado la decisión de realizar un análisis más detallado teniendo en cuenta el tamaño de los distintos comentarios.

Rango	Tokens totales	Aciertos castellano	Aciertos SMS	Fallos
1 – 6	19.345	4.625	298	14.422
7 – 20	169.246	82.966	5.379	88.967
21 – 50	403.366	210.916	13.469	178.981
51 – 99	519.784	270.146	18.982	230.656
100 – 250	1,247.541	656.809	44.935	545.797

Tabla 6: Frecuencias por franjas con el tokenizador básico.

Rango	Tokens totales	Aciertos castellano	Aciertos SMS	Fallos
1 – 6	29.322	12.793	1.065	15.464
7 – 20	258.109	142.806	11.033	104.270
21 – 50	601.453	344.710	25.361	231.382
51 – 99	802.203	448.135	35.095	318.973
100 – 250	1,939.077	1,081.811	83.002	774.264

Tabla 7: Frecuencias por franjas con el nuevo tokenizador.

Presentamos este análisis en las tablas 6 y 7, mostrando en la primera de ellas el comportamiento del tokenizador básico, mientras que en la segunda mostramos el comportamiento del nuevo sistema de tokenización presentado en este artículo. En cada tabla mostramos por fila los resultados para los comentarios con una longitud en número de palabras dentro de una serie de rangos (de 1 a 16 palabras, de 7 a 20, etc.). En cada columna mostramos el número de palabras totales localizadas, cuántas de ellas se encuentran en el diccionario en castellano, cuántas en el diccionario SMS y cuántas

no se encuentran en ningún diccionario (fallos de búsqueda).

La primera observación a realizar al comparar ambas tablas es que las celdas de la segunda están pobladas con cantidades mucho mayores que la primera – por ejemplo, observando la primera columna en ambas tablas, se concluye que el número de tokens detectado en la segunda (por el nuevo tokenizador) es superior aproximadamente en un 50% al obtenido por el tokenizador básico. Con las demás columnas se puede hacer un análisis similar, siendo los incrementos próximos al 60% en ocasiones. Sin embargo, este hecho no es factor relevante desde el punto de vista del análisis de la calidad en la separación de palabras, ya que debemos tener en cuenta que el nuevo tokenizador admite como tokens las secuencias de caracteres no alfanuméricos (potenciales emoticonos), mientras que el primero considera la gran mayoría de signos de puntuación como separadores y por tanto no como tokens.

El análisis de los resultados de las tablas 6 y 7 debe ser relativo, es decir: ¿aumenta el número de tokens encontrados en los diccionarios en términos relativos (sobre el número de tokens encontrados) cuando se compara ambos tokenizadores? De ser así, la separación de palabras realizada por el nuevo tokenizador sería de mayor calidad.

Para que sea más sencillo analizar los resultados, mostramos también las 8 y 9, en las que se muestran los porcentajes de elementos correspondientes a cada una de las columnas anteriores.

Rango	% castellano	% SMS	% fallos
1 – 6	23,91%	1,54%	74,55%
7 – 20	49,02%	3,18%	52,57%
21 – 50	52,29%	3,34%	44,37%
51 – 99	51,97%	3,65%	44,38%
100 – 250	52,65%	3,60%	43,75%
Media	45,97%	3,06%	51,92%

Tabla 8: Porcentajes correspondientes a las frecuencias obtenidas con el tokenizador básico.

Rango	% castellano	% SMS	% fallos
1 – 6	43,63%	3,63%	52,74%
7 – 20	55,33%	4,27%	40,40%
21 – 50	57,31%	4,22%	38,47%
51 – 99	55,86%	4,37%	39,76%
100 – 250	55,79%	4,28%	39,93%
Media	53,58%	4,16%	42,26%

Tabla 9: Porcentajes correspondientes a las frecuencias obtenidas con el nuevo tokenizador.

Con carácter global (en media), con el nuevo tokenizador se reconocen un 53,58% de los tokens en el diccionario de castellano mientras que con el tokenizador básico, el porcentaje es del 45,97%. La mejora en el reconocimiento dentro del diccionario SMS es menor pero relevante, ya que pasamos de un 3,06% a un 4,16%, y en términos de tokens no reconocidos, éstos descienden casi en un 10%.

El comportamiento es similar en ambas tablas, ya que el porcentaje de elementos localizados en los diccionarios tanto cuando se utiliza el tokenizador básico como cuando cuando se utiliza el nuevo tokenizador, es mayor en los rangos centrales, mientras que es menor en los rangos extremos. Existe una diferencia mayor particularmente entre el rango 1 a 6 y los demás rangos, y es precisamente en ese rango donde se produce una mayor mejora en los porcentajes de tokens reconocidos en los diccionarios, y en consecuencia, en los tokens no reconocidos o fallos, que se reducen drásticamente de un 74,55% a un 52,74%.

Por contraste, en los comentarios más largos la mejora es la menor, lo cual se debe probablemente a que, al tratarse de textos más largos, los usuarios ponen una mayor atención en su redacción – en otras palabras, si se toman su tiempo en escribir un comentario de 250 palabras, también se molestan en escribirlo en un castellano más correcto.

4.2 Evaluación indirecta

La evaluación indirecta del nuevo tokenizador consiste en examinar su impacto en la tarea objetivo, que es el reconocimiento de edad según se ha descrito anteriormente. Para ello, se construyen clasificadores siguiendo los pasos descritos en la sección 2.1, usando el tokenizador básico y el nuevo tokenizador, para comparar sus resultados en términos de eficacia.

El proceso consiste en la tokenización de los 84.956 comentarios, la traducción SMS y la normalización con Deflogger de los términos obtenidos, la representación de los comentarios como vectores de pesos de términos, el filtrado de los términos por Ganancia de Información, y el entrenamiento de un clasificador Bayes Ingenuo (Naive Bayes[13]) incluido en WEKA con sus opciones por defecto.

Cabe resaltar que el tokenizador estándar es capaz de localizar 227.781 términos distintos, mientras que el nuevo tokenizador detecta 272,769 términos diferentes (incluyendo por ejemplo las secuencias de símbolos de puntuación, interpretadas como emoticonos). Sin embargo, una vez se aplica el filtro de términos por su valor de Ganancia de

13 http://weka.sourceforge.net/doc/weka/classifiers/bayes/NaiveBayes.html.

Información, en el clasificador basado en el tokenizador básico quedan 38.564 términos, mientras que en el basado en el nuevo tokenizador quedan 26.233, es decir, muchos menos. Nosotros interpretamos esta circunstancia como una mejora, ya que el nuevo tokenizador aumenta la calidad de los términos usados en la representación al concentrar sus estadísticas – en otras palabras, términos que anteriormente eran poco frecuentes y aportaban información positiva pero marginal, pasan a estar concentrados en términos más frecuentes que aportan más información al tener frecuencias mayores.

Para evaluar los clasificadores obtenidos, se ha aplicado un proceso de validación cruzada en cuatro carpetas (es decir, se han ejecutado cuatro experimentos usando en cada uno de ellos, un 75% de la colección de datos para entrenamiento, y un 25% para evaluación). Como métricas de evaluación, y teniendo en cuenta que es preciso observar el detalle de las clases más sensibles (menores de 14 años y mayores de 18 años), se ha calculado la tabla de confusión en cada caso (agregada sobre los 4 experimentos) y la precisión y la cobertura por cada clase.

En las tablas 10 y 11 se muestran los resultados obtenidos por los clasificadores entrenados usando el tokenizador básico y el nuevo tokenizador, respectivamente. En cada fila se muestran las poblaciones de las clases, mientras que por columnas se muestran las decisiones de clasificación. Por tanto, las cuatro primeras filas y columnas constituyen la tabla de confusión. Los aciertos aparecen en la diagonal principal de cada tabla. En las dos últimas columnas se muestran la precisión y la cobertura por cada clase.

	-14	14	18	Precisión	Cobertura
-14	530	1232	48	0,61	0,29
14	262	57298	2218	0,77	0,95
18	73	15850	7445	0,76	0,31

Tabla 10: Tabla de confusión y valores de efectividad con el tokenizador por defecto.

	-14	14	18	Precisión	Cobertura
-14	597	1146	67	0,63	0,32
14	290	57336	2152	0,77	0,96
18	69	15752	7546	0,77	0,32

Tabla 11: Tabla de confusión y valores de efectividad con el nuevo tokenizador propuesto.

Como se puede observar, el proceso de clasificación arroja mejores resultados con el nuevo proceso de tokenización. En números globales, el nuevo tokenizador conlleva un incremento de efectividad del 0,768 al 0,770, que se corresponde con 206 nuevos aciertos. Aunque en términos absolutos la mejora puede no parecer muy significativa, esta mejora se concentra sobre todo en las clases menos representadas y más significativas, que son las de menores de 14 años y las de mayores de 18, por lo que a efectos prácticos desde el punto de vista de la eficacia de la detección de edad, consideramos que la mejora es importante.

5 Conclusiones y trabajo futuro

En este artículo hemos presentado un nuevo sistema de tokenización específico para el texto informal que utilizan los usuarios de redes sociales, enmarcado en un sistema de detección de edad en este contexto. El sistema de tokenización ha sido evaluado con resultados positivos, tanto desde el punto de vista de la calidad explícita de los términos detectados, como desde el del impacto que tiene en la efectividad del clasificador de edad.

En futuros trabajos planeamos evaluar el sistema de tokenización no sólo con el algoritmo de aprendizaje propuesto, sino también con otros algoritmos de aprendizaje, integrandolo en el sistema multimodal de reconocimiento de edad que utiliza otros elementos de información (perfiles, fotografías, etc.).

Agradecimientos

Esta investigación ha sido financiada por Optenet S.A. y por el Ministerio de Economía y Competitividad y el Centro para el Desarrollo Tecnológico Industrial (CDTI), en el marco del proyecto de investigación industrial "*WENDY: WEb-access coNfidence for chilDren and Young*" (TSI-020100-2010-452).

Referencias

Forsyth Eric N. and Craig H. Martell, *Lexical and Discourse Analysis of Online Chat Dialog,* Proceedings of the First IEEE International Conference on Semantic Computing (ICSC 2007), pp. 19-26, September 2007.

Hall Mark, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann and Ian H. Witten (2009); *The WEKA Data Mining Software: An Update*; SIGKDD Explorations, Volume 11, Issue 1.

Padró, Lluís, Miquel Collado, Samuel Reese, Marina Lloberes and Irene Castellón. *FreeLing 2.1: Five Years of Open-Source Language Processing Tools.* Proceedings of 7th Language

Resources and Evaluation Conference (LREC 2010), ELRA.
La Valletta, Malta. May, 2010.

Pendar, Nick Nick Pendar, *Toward Spotting the Pedophile Telling victim from predator in text chats*, 2012 IEEE Sixth International Conference on Semantic Computing, pp. 235-241, International Conference on Semantic Computing (ICSC 2007), 2007

Ptaszynski, Michal, Pawel Dybala, Tatsuaki Matsuba, Fumito Masui, Rafal Rzepka and Kenji Araki, *Machine Learning and Affect Analysis Against Cyber-Bullying*, In Proceedings of The Thirty Sixth Annual Convention of the Society for the Study of Artificial Intelligence and Simulation of Behaviour (AISB'10), 29th March – 1st April 2010, De Montfort University, Leicester, UK, pp. 7-16, 2010.

Sebastiani, F. 2002. *Machine learning in automated text categorization*. ACM Computing Surveys, 34(1):1–47.

Tam, J., and C. H. Martell. 2009. *Age detection in chat.* Paper presented at Semantic Computing, 2009. ICSC '09. IEEE International Conference on, .

Yin D., Z. Xue, L. Hong, B. D. Davison, A. Kontostathis, and L. Edwards, *Detection of Harassment on Web 2.0* in CAW 2.0 '09: Proceedings of the 1st Content Analysis in Web 2.0 Workshop, Madrid, Spain, 2009.