Preprocesamiento de un corpus empleando corrección probabilística para precisar el vocabulario

Viridiana Cruz-Gutiérrez, Mario Alberto Posada-Zamora, Maya Carrillo, Luis Enrique Colmenares-Guillén, Abraham Sánchez-López

Benemérita Universidad Autónoma de Puebla, Puebla, México

{viricruz,mariop}@rockruz.net,{cmaya,lecolme,asanchez}@cs.buap.mx

Resumen. La Organización Internacional del Trabajo estimó que en el 2012 existían 20.9 millones de víctimas de explotación laboral y sexual forzada en el mundo. México ocupa el tercer lugar en trata de personas en América Latina y el Caribe. Particularmente, Puebla se encuentra entre los estados donde hay mayor vinculación de desaparición de mujeres y niñas con la trata y feminicidio. Ante esta situación estamos interesados en desarrollar herramientas que puedan ser utilizadas por padres y autoridades para la prevención de delitos ligados a trata de personas por Internet. El presente trabajo representa uno de los primeros pasos en esta dirección, se explora el preprocesamiento de un corpus de chats con contenido sexual empleando corrección probabilística, mediante teorema de Bayes. Para medir los efectos del procesamiento realizado, se trabajó en el agrupamiento de los documentos mencionados, empleando mapas auto-organizados. Los resultados obtenidos muestran que el procesamiento realizado mejora la efectividad del agrupamiento.

Palabras clave: Preprocesamiento de corpus, mapas auto-organizados, teorema de Bayes.

1. Introducción

La tecnología, sin lugar a dudas, ha sido fundamental en la evolución de la sociedad. Sin la tecnología la vida del hombre no sería como la conocemos. Es así como desde las antorchas, con las que el hombre fue capaz de salir a la obscuridad; la agricultura, que volvió al hombre sedentario; la máquina de vapor del siglo XVII; el primer foco inventado por Thomas Alva Edison, hasta la Internet y los *smartphones*, la tecnología ha impulsado un sinfín de cambios en la sociedad.

La tecnología ha permitido que en los últimos años, nos enteremos casi instantáneamente de lo que ocurre en todos los rincones del mundo, pero también ha provocado que la sociedad sea vulnerable a nuevas formas de victimización,

dando como resultado el incremento de delitos, donde los niños son los más propensos a sufrir ataques por parte de cibercriminales.

Hay 20.9 millones de víctimas de explotación laboral y sexual forzada en el mundo, según estimación del 2012 de la Organización Internacional del Trabajo (OIT) [4]. La asociación civil "Infancia Común" expone que la explotación sexual de niños, niñas y adolescentes (ESNNA) en el 2012 ocupaba el segundo lugar en generación de ganancias en México, ubicada en el orden de los 24 mil millones de dólares anuales por encima de la venta de armas y sólo superada por el narcotráfico [5].

Las modalidades de la ESNNA son: la prostitución, la pornografía infantil, el turismo sexual, el abuso sexual, la trata y la venta de niños, niñas y adolescentes para actividades sexuales. Según datos arrojados por la Coalición Contra el Tráfico de Mujeres y Niñas en América Latina y el Caribe (CATWLAC, por sus siglas en inglés) de cada diez personas que son víctimas de trata en el país, dos son menores de 18 años. Con base en la información recabada por la CATWLAC, México ocupa el tercer lugar en trata de personas en América Latina y el Caribe, y es un país de origen, tránsito y destino.

En el estudio "Violencia en el país, aumenta riesgos para que mujeres y niñas sean víctimas de trata", de la Coalición, los estados donde hay mayor vinculación de desaparición de mujeres y niñas con la trata y, luego, con el feminicidio, son Baja California, Puebla, Chihuahua, Oaxaca, Coahuila, Quintana Roo, Chiapas, San Luis Potosí, Durango, Tamaulipas, Estado de México, Tabasco, Guerrero, Tlaxcala, Hidalgo, Veracruz, Jalisco, Zacatecas y Nuevo León. Si bien existe un marco legal que busca eliminar la trata de personas, se necesita un verdadero trabajo de la sociedad y del gobierno para prevenir este delito [6].

Ante la situación planteada, como sociedad e investigadores estamos interesados en desarrollar herramientas que puedan ser utilizadas por padres y autoridades para la prevención de delitos ligados a trata de personas por Internet. El presente trabajo representa uno de los primeros pasos en esta dirección y si bien no pretende impactar en el estado del arte, si contribuir a probar técnicas de inteligencia artificial y procesamiento de lenguaje natural que permitan el desarrollo de dichas herramientas.

Estamos especialmente interesados en crear corpus de chats en español con conversaciones establecidas entre los posibles predadores sexuales y sus víctimas. Dichas conversaciones en español tendrán que ser ubicadas y recabadas a lo largo del proyecto. Con el objetivo de definir el método adecuado para construir dichos corpus, decidimos explorar el tipo de procesamiento previo que deberá hacerse sobre los chats que se recaben.

Dada la carencia de recursos en español, se utilizaron chats en inglés sobre los cuales se realizó un procesamiento normal: eliminación de palabras vacías, eliminación de palabras de acuerdo a su longitud, pero además se efectuó una corrección probabilística, empleando distancia de edición. Para medir los efectos del procesamiento previo realizado, se trabajó en la tarea de agrupamiento pues los documentos que se obtengan de Internet deberán ser agrupados en documentos útiles para la detección de predadores y los inútiles. El agrupamiento

de documentos se realizó con un mapa auto-organizado (SOM, por sus siglas en inglés). Los resultados obtenidos muestran que el procesamiento previo realizado sobre los documentos mejoró la efectividad del agrupamiento realizado por el SOM.

El resto de este documento está organizado de la siguiente manera, en la Sección 2 se describe brevemente lo que es un mapa auto-organizado, en la Sección 3 se presentan los pasos seguidos para la construcción del corpus de experimentación, así como el procesamiento previo y corrección del mismo. En la Sección 4 se explican los experimentos realizados y resultados obtenidos, finalmente en la Sección 5 se muestran las conclusiones y trabajo futuro.

2. Mapas auto-organizados

Se ha observado que en la corteza cerebral de los animales superiores aparecen zonas donde las neuronas detectoras de rasgos están topológicamente ordenadas; de manera que las información captada del entorno mediante los órganos sensoriales, se representan internamente en forma de mapas bidimensionales.

Aunque en gran medida esta organización neuronal está predeterminada genéticamente, es probable que parte de ella se origine mediante el aprendizaje. Esto sugiere, por tanto, que el cerebro podría poseer la capacidad inherente de formar mapas topológicos a partir de las informaciones recibidas del exterior.

También se ha observado que la influencia de una neurona sobre las demás, está en función de la distancia entre ellas, siendo pequeña cuando están alejadas. A partir de estas ideas, el académico finlandés Teuvo Kohonen presentó en 1982[7] un sistema con un comportamiento semejante. Se trataba de un modelo de red neuronal con capacidad para formar mapas de características de manera similar a como ocurre en el cerebro.

Los mapas auto-organizados de Kohonen constituyen un método de proyección no lineal que mapea un espacio de datos multidimensional en un mapa usualmente bidimensional en forma ordenada. Los nodos del mapa son asociados a los llamados vectores de referencia que actúan como modelos locales de los datos más cercanos y, más ampliamente, de las regiones vecinas del mapa. Gracias a las propiedades del algoritmo de los SOM, posiciones cercanas del mapa contienen datos similares, permitiendo una visualización intuitiva del espacio de datos. Más aún, los vectores de referencia dividen los datos en subconjuntos de datos similares, efectuando una categorización de los datos.

El proceso de aprendizaje del SOM es ilustrado a continuación:

1. Inicialice los pesos con valores al azar:

$$\forall i \in M : w_i(0) = random() \tag{1}$$

   Donde M es el número de neuronas
2. Elegir al azar un patrón $x(t)$ del conjunto de entrenamiento, para la iteración t.

3. Por cada neurona i en el mapa de características Φ, calcular la similitud entre el conjunto de pesos w_i y el patrón $x(t)$. Para esto puede usarse la distancia Euclidiana:

$$\forall i \in M : d^2(w_i, x) = \sum_{k=1}^{N} (w_{ik} - x_k)^2 \tag{2}$$

4. Encontrar un neurona ganadora i^* correspondiente a la que obtuvo la mínima distancia (máxima similitud).
5. Modificar los pesos de la neurona ganadora i^* y los de sus vecinas:

$$\forall j \in \Lambda_{i^*}(t) : w_j(t+1) = w_j(t) + \alpha(t)(x(t) - w_j(t)) \tag{3}$$

Λ_{i^*} corresponde a una función de vecindad centrada en la neurona ganadora i^* y $\alpha(t)$ es una función de proporción de aprendizaje, definida como:

$$\alpha(t) = \frac{1}{t} \tag{4}$$

o también se define de la siguiente manera:

$$\alpha(t) = \alpha_1 \left(1 - \frac{1}{t}\right) \tag{5}$$

6. Regresar al paso dos, hasta que no existan más cambios en el mapa de características Φ o hasta que el número máximo de iteraciones se alcance.

Los SOM han sido utilizados exitosamente en diversas tareas, se pueden destacar aplicaciones relacionadas con el reconocimiento de patrones (voz, texto, imágenes, señales, etc), codificación de datos, comprensión de imágenes, resolución de problemas de optimización, análisis de imágenes y monitoreo de procesos [8,9,10,11,12].

3. Desarrollo

En esta sección se muestran los pasos que se siguieron para la construcción de un corpus con chats de predadores sexuales con sus víctimas, complementado con documentos de tecnología Linux, y la experimentación con dicho corpus.

3.1. Construcción del corpus

Se construyó un corpus balanceado que permitiera comprobar si preprocesar el texto empleando técnicas de corrección probabilistica, podría contibuir a mejorar el rendimiento de la tarea de agrupamiento de textos. Como se ha mencionado previamente, para tal efecto se empleó una red neuronal de tipo mapa auto-organizado (SOM).

El corpus contiene dos grupos de documentos de chats, el primero con contenido sexual y el segundo con contenido referente a tecnologías Linux. Ambos grupos contienen únicamente texto en inglés.

Para la construcción del primer grupo de documentos del corpus, se eligieron los chats y mensajes de texto de 593 predadores sexuales, obtenidos de la página Perverted Justice[1]. Inicialmente se descargaron las páginas de la Web en texto plano. Para conformar el segundo grupo de documentos, se descargó un compendio de chats en texto plano de la página Ubuntu Chats Corpus[2].

Debido a que los chats de los predadores sexuales eran muy extensos, se separaron por fechas y posteriormente, se eliminaron fechas, nombres de usuario e información de sistema de los chats, ya que son datos irrelevantes para el agrupamiento.

Se contabilizó el número de palabras de cada chat de los predadores sexuales y se determinó que en promedio contenían entre 1,000 y 1,300 palabras, por lo que se eligió un chat de cada predador sexual con dicha extensión, obteniendo una muestra de 500 documentos. Para tener corpus balanceados, se eligió el mismo número de chats de Ubuntu que cumplieran con los criterios anteriormente mencionados.

Finalmente se eliminaron de los 1,000 documentos, caracteres especiales y palabras que no cumplieran con la codificación Unicode.

Todo el proceso de construcción del corpus, se realizó con un programa desarrollado en lenguaje Java.

3.2. Preprocesamiento

El preprocesamiento para preparar el corpus de este experimento se realizó empleado el lenguaje de programación Python 3.5 y la librería de procesamiento de lenguaje natural NLTK. Se desarrollaron tres esquemas para hacer más conciso el vocabulario generado:

1. Remoción de palabras 'vacías', signos de puntuación y números.
2. Selección de palabras cuya longitud comprende un rango específico.
3. Uso de un algoritmo de corrección de palabras.

En cada aplicación de las técnicas mencionadas anteriormente, se generó un archivo que contiene el vocabulario perteneciente al corpus resultante, el cual se puede observar como un conjunto sin repeticiones de las palabras utilizadas en todos los documentos.

La técnica de la remoción de palabras vacías se basa en el uso de un conjunto de palabras P_v, las cuales tienen poco o nulo significado y relevancia para la descripción de los documentos.

Para cada documento d del corpus C, al vocabulario generado V_d se le sustrae el conjunto de palabras vacías P_v, con el fin de obtener un nuevo vocabulario V_i' más reducido y con mayor relevancia. La ecuación 6, muestra cómo se eliminan las palabras vacías del documento.

$$\forall d \in C : V'_d = V_d - P_v \tag{6}$$

El conjunto P_v utilizado, está conformado por el conjunto de las palabras que podrían no tener significado para nuestro experimento, tales como pronombres personales, algunos verbos, auxiliares, saludos y números escritos con letra.

Para la técnica de selección de palabras de cierta longitud, por observación se dedujo que las palabras que podrían ser relevantes para representar un documento contenían generalmente de 3 a 13 caracteres, sobre todo en el ámbito de representación de un documento sexual. Algunas muestras de palabras en inglés son: *sex*, *masturbation*, *underage*, por citar algunos ejemplos.

Es entonces que el vocabulario reducido V' para esta técnica se representa de acuerdo con la ecuación 7.

$$\forall d \in C : V'_d = \{x : |x| > 2 \wedge |x| < 14\} \tag{7}$$

Otra justificación para utilizar esta medida es, particularmente para el corpus empleado, que los documentos contienen palabras que, antes de pasar por la eliminación de símbolos y caracteres especiales, eran direcciones http, lo que ocasionaba nuevas palabras en el vocabulario total, cuya longitud generalmente rebasaba los 14 caracteres continuos.

En la técnica de corrección, se empleó un corrector desarrollado en Python basado en el teorema de Bayes[3], para establecer qué tan probable es para una palabra estar mal escrita, y la probabilidad de que la palabra a la que se quiere hacer referencia sea otra. Este programa realizado por Peter Norving, director de investigación de Google, fue desarrollado con la finalidad de crear un corrector ortográfico "de juguete"(comparado con el desarrollado a nivel industrial), que tuviera una considerable precisión y velocidad de procesamiento de al menos 10 palabras por segundo.

Estamos buscando una corrección c para una palabra w de tal manera que se maximice la probabilidad de que sea la palabra buscada, la expresión 8 ilustra esta técnica.

$$argmax_c P(c|w) \tag{8}$$

La ecuación 9 hace referencia a la expresión 8 aplicando el teorema de Bayes.

$$argmax_c P(c|w) = argmax_c P(w|c)P(c)/P(w) \tag{9}$$

Reduciendo el término derecho de 9, se obtiene la expresión 10.

$$argmax_c P(w|c)P(c) \tag{10}$$

Donde $P(c)$ es la probabilidad de que c aparezca en documentos escritos bajo un lenguaje. $P(w|c)$ se refiere a la probabilidad de que la palabra w haya sido escrita queriendo decir realmente c.

Por último, $argmax_c$ es una función que selecciona de la lista todas las posibles formas aceptables de c, la mejor c cuyo puntaje de probabilidad sea el más alto.

Para realizar la corrección de palabras se utilizó la distancia de edición o distancia de Levenshtein, considerando las siguientes operaciones:

- Eliminación de un caracter.
- Trasposición de caracteres.
- Reemplazos de caracteres.
- Inserción de caracteres.

Como resultado obtenemos una palabra corregida que es bastante útil en el contexto de corpus conformados con conversaciones en línea, en las cuales existe una gran cantidad de errores.

En base a los tres esquemas enunciados al inicio de esta sección, se produjeron tres vocabularios, que se muestran en la Tabla 1.

Tabla 1. Descripción de los vocabularios y su tiempo de generación

	Vocabulario 1	**Vocabulario 2**	**Vocabulario 3**
Técnicas utilizadas	Remoción de palabras vacías	Remoción de palabras vacías y longitud acotada	Remoción de palabras vacías, longitud acotada y algoritmo de corrección de palabras
Longitud de vocabulario (número de palabras)	40,043	36,656	19,681
Tiempo de generación de vocabulario	25 seg	40 seg	2.25 horas

3.3. Representación vectorial

Una vez generado el vocabulario con las técnicas descritas en la Tabla 1, se empleó el modelo vectorial para representar los documentos, de acuerdo con el esquema de pesado tf_idf, cuyas formulas se muestran en las ecuaciones 11, 12 y 13.

$$tf = \frac{n_{i,j}}{|d_j|} \tag{11}$$

Siendo $n_{i,j}$ el número de ocurrencias de la palabra i para el documento j y $|d_j|$ la longitud del documento.

$$idf(t, D) = log\left(\frac{|D|}{|\{d \in D : t \in d\}|}\right) \tag{12}$$

La ecuación 12 hace referencia al logaritmo del número de documentos del corpus, entre el número de documentos que contienen el término t.

$$tf\_idf(t, D) = tf(t, d) * idf(t, D) \tag{13}$$

Con la ecuación 13 se da importancia a algunos términos por encima de otros, sobre todo cuando la rareza de un término en el corpus es considerable. Además con la frecuencias de las palabras en un documento en particular, se da menos prioridad a palabras que aparecen poco en dicho documento, en contraste con las que aparecen con frecuencia considerable.

4. Experimentos y resultados

Para comprobar si el preprocesamiento del corpus mejora la tarea de agrupamiento, se utilizó un SOM y se realizaron experimentos con lo tres tipos de vocabulario explicados en el apartado de Preprocesamiento. A continuación se describen las condiciones de los experimetos realizados y se presentan los resultados obtenidos.

4.1. SOM con vocabulario sin palabras vacías únicamente

En el primer experimento, los documentos se subministraron al SOM de manera ordenada: los primeros 500 eran referentes a los predadores sexuales y los últimos 500 sobre temas de Ubuntu. El resultado del SOM muestra un agrupamiento de 670 documentos en la primera clase y 330 en la segunda para una primera ejecución (Figura 1.a), mientras que para un segundo experimento, se obtuvo un agrupamiento de 672 para la primera clase y 328 para la segunda clase, como puede observarse en la Figura 1.b.

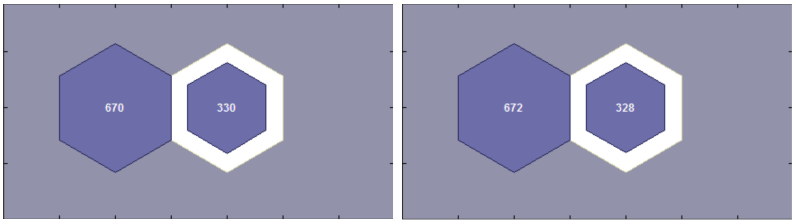

Fig. 1. Resultados de la ejecución del SOM con un corpus ordenado con el Vocabulario 1, a) Agrupamiento en primera corrida, b) Agrupamiento en segunda corrida. En ambas imagenes, el hexágono de la izquierda corresponde al agrupamiento referente a predadores sexuales y el hexágono de la derecha, a documentos de tecnología Linux.

Con este mismo vocabulario los 1000 documentos fueron proporcionados al SOM de manera aleatoria, el resultado fue muy similar a los obtenidos anterior-

mente, siendo el agrupamiento de 671 y 329 elementos para la primera y segunda clase respectivamente, el resultado se muestra en la Figura 2.

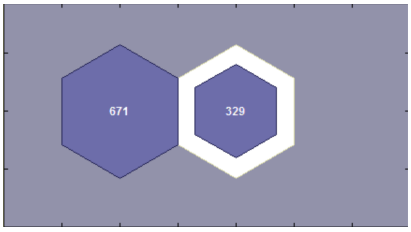

Fig. 2. Resultados de la ejecución del SOM con un corpus en orden aleatorio con el Vocabulario 1. El hexágono de la izquierda corresponde al agrupamiento de documentos sobre predadores sexuales y el hexágono de la derecha, a documentos de tecnología Linux.

En promedio el tiempo de procesamiento que le tomó al SOM para el agrupamiento fue de 125.62 segundos.

4.2. SOM con palabras de longitud acotada y sin palabras vacías

El segundo experimento se realizó con una combinación de dos técnicas, resultando para una corrida con datos ordenados de 500 documentos sobre abuso sexual seguidos de 500 documentos de chats de Ubuntu.

El agrupamiento arrojado en este experimento fue un resultado de 266 y 734 documentos para la primera y segunda clase respectivamente. Como se observa en la Figura 3.

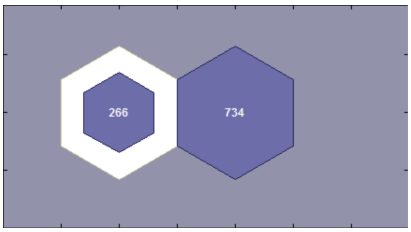

Fig. 3. Resultados de la ejecución del SOM con un corpus ordenado tomando el Vocabulario 2. Hexágono izquierdo pertenece a documentos con contenido sexual y el hexágono derecho a documentos sobre tecnología Linux.

Para ser un corpus balanceado, los resultados obtenidos fueron menos satisfactorios de lo esperado, pues estuvieron por debajo de los resultados al eliminar únicamente las palabras vacías. Analizando el vocabulario utilizado, se encontró

que la eliminación de palabras pertenecientes a direccones web provocó que los resultados no fueran favorables.

El tiempo de procesamiento que le tomó al SOM para el agrupamiento fue de 108.93 segundos.

4.3. SOM con vocabulario sin palabras mal escritas, longitud acotada y sin palabras vacías

La primera prueba del experimento con este vocabulario más reducido se realizó nuevamente con los documentos ordenados en un primer grupo de 500 y un segundo grupo de 500, para los chats de depredadores sexuales y de Ubuntu, respectivamente. El resultado del SOM mostró un agrupamiento de 500 para la primera clase y 500 para la segunda, por lo que se obtuvo el 100 % de agrupamiento correcto. Figura 4.

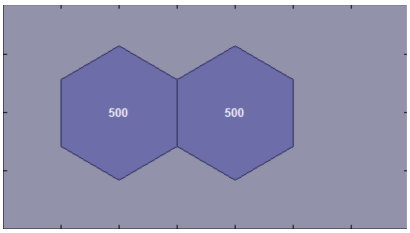

Fig. 4. Resultados de la ejecución del SOM con un corpus ordenado tomando el Vocabulario 3. Hexágono izquierdo referente a documentos con contenido sexual y hexágono derecho sobre documentos de tecnología Linux.

La siguiente prueba fue colocar únicamente 500 documentos de contenido sexual y 205 de Ubuntu, obteniendo nuevamente los resultados esperados, un agrupamiento de 500 en la primera clase y 205 en la segunda. Se decidió hacer una prueba con un corpus desbalanceado ya que los resultados obtenidos en la prueba de la Figura 4 fueron favorables, y queríamos corroborar que se podían producir resultados similares de esta manera.

La última prueba con este vocabulario fue con orden aleatorio de los 1,000 documentos. El resultado fue nuevamente exitoso, logrando agrupar el 100 % de los documentos, esto se observa en la Figura 6.

El tiempo de procesamiento en el SOM durante el agrupamiento fue de 66.24 segundos.

5. Conclusiones y trabajo futuro

Se comprobó que contar con un vocabulario preciso y de dimensión adecuada permite mejorar el desempeño de la tarea de agrupamiento, en particular en nuestro caso se logró agrupar de manera correcta el 100 % de los documentos.

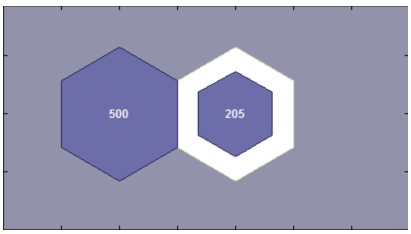

Fig. 5. Resultados de la ejecución del SOM con un corpus desbalanceado ordenado tomando el Vocabulario 3. Hexágono izquierdo referente a documentos con contenido sexual y hexágono derecho sobre documentos de tecnología Linux.

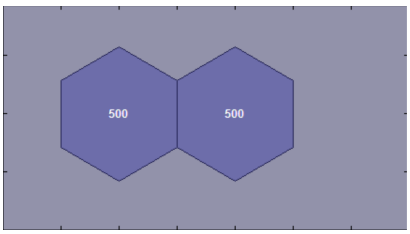

Fig. 6. Resultados de la ejecución del SOM con un corpus en orden aleatorio tomando el Vocabulario 3. Hexágono izquierdo referente a documentos con contenido sexual y hexágono derecho sobre documentos de tecnología Linux.

Aunado a esto, debe mencionarse que el tiempo de procesamiento disminuyó en un 47.27 %, ya que en cada paso del preprocesamiento se redujo la dimensión del vocabulario.

Se realizaron tres pasos de pre-procesamiento, los dos primeros fueron los comunmente usados en el procesamiento de cualquier corpus, con los cuales no obtuvimos resultados extraordinarios. Sin embargo, el tercer paso, la corrección de palabras mediante el teorema de Bayes, permitió obtener mejores resultados, gracias a que se rescataron palabras mal escritas, que son importantes dentro del contexto.

Como trabajo futuro, se pretende experimentar con otros corpus de mayor dimensión y con chats en idioma español, así como probar la eficiencia de los vocabularios generados, en algoritmos genéticos de agrupamiento.

Referencias

1. Perverted Justice, `http://www.perverted-justice.com`
2. Ubuntu Chats Corpus, `http://daviduthus.org`
3. How to write a spelling corrector, `http://norvig.com`
4. Estimación mundial sobre el trabajo forzoso (resumen ejecutivo), `http://www.ilo.org/wcmsp5/groups/public/---ed_norm/---declaration/documents/publication/wcms_182010.pdf`

5. Contralínea, http://contralinea.info/archivo-revista/index.php/2010/09/05/mexico-pasividad-ante-explotacion-sexual-infantil/
6. Terreno ideal para la trata de personas, http://eleconomista.com.mx/sociedad/2013/11/10/terreno-ideal-trata-personas
7. Kohonen, T.: The Self-Organizing Map. Proceedings of the IEEE, Vol. 78, pp. 1464–1480 (1990)
8. Honkela, T.: Emerging categories and adaptive prototypes: Selforganizing maps for cognitive linguistics. Extended abstract, accepted to be presented at the International Cognitive Linguistics Conference (1997)
9. Kohonen, T., Kaski, S., Lagus, K., Honkela, T.: Very large two-level SOM for the browsing of newsgroups. In: Proceedings of ICANN'96, International Conference on Artificial Neural Networks (1996)
10. Kurimo, M.: Fast latent semantic indexing of spoken documents by using self-organizing maps. In: IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'00), vol. 6, pp 2425–2428, IEEE, Istanbul (2000)
11. Deboeck, G., Kohonen, T.: Visual Explorations in Finance with Self-Organizing Maps. Springer, New York (1998)
12. López, H., Machón, I.: Self-organizing map and clustering for wastewater treatment monitoring. Engineering Applications of Artificial Intelligence 17(3), 215–225 (2004)