INTELLIFILTER
SISTEMA PARA EL FILTRADO SOPORTADO POR INTELIGENCIA COMPUTACIONAL

HECTOR FABIO CADAVID RENGIFO
JORGE HUMBERTO CELY HIGUERA
JUAN PABLO GARCIA SEGURA

Escuela Colombiana De Ingeniería
Decanatura De Ingeniería de Sistemas

***Abstract*- Intellifilter es un sistema para el filtrado de páginas web soportado por inteligencia artificial, el cual para hacer la clasificación de la página se basa en el texto y las imágenes que contenga la pagina en ese momento. El proceso de filtrado inicia cuando un usuario solicita el acceso a una página web dese cualquier browser. Al momento de recibir la información de la pagina solicitada, un proxy instalado en el navegador, en el cual esta nuestro software, intercepta la información (texto e imágenes) contenida en la página y dependiendo de un criterio de clasificación, encontrado luego de un proceso de entrenamiento con las técnicas de inteligencia computacional, determina si una página es nociva o no dependiendo del contenido que la pagina tenga en el momento de la intercepción. Si determina que la página tiene contenido nocivo, el proxy envía una página de advertencia ocultando el contenido original de la página, en caso contrario, envía el contenido original de la página y permite la navegación normal por el sitio web.**

***Index Terms*- Filtro web, contenido nocivo, contenido no nocivo, inteligencia computacional, arboles de decisión, redes neuronales, *Naive* Bayes, clasificación, proxy, browser.**

INTRODUCCION

En nuestra sociedad actual, cada vez más influenciada por los nuevos paradigmas de comunicación e interacción social generados por Internet, es ampliamente conocida la problemática con la creación y publicación para libre acceso de contenidos nocivos para menores de edad, como los sitios pornográficos, los sitios que incitan a la violencia, e incluso sitios que ponen en riesgo la integridad física de los niños al ponerlos en contacto con explotadores de menores.

A causa del principio de libertad de expresión, ha sido imposible que alguien tome el control de estos contenidos, por lo cual, la responsabilidad de proteger a los menores de esta exposición recae ahora en los padres y en las instituciones educativas. Es por esto, que desde hace varios años, la pregunta de cómo crear una herramienta capaz de detectar con suficiente precisión si un sitio es nocivo o no, se ha vuelto una cuestión de investigación abierta de mucha importancia.

PROBLEMÁTICA Y ESTADO DEL ARTE

Como un esfuerzo para proveer a padres y educadores de herramientas que permitan convertir las aulas de informática y los computadores del hogar en recursos seguros para los menores, desde la industria y desde la academia se han creado filtros de contenidos, que en su mayoría se basan en el esquema de coincidencia de palabras clave (se determina si un sitio es nocivo si contiene determinadas palabras), el cual ha tenido siempre el problema de los 'falsos positivos' con páginas de contenidos no nocivos (el ejemplo tradicional, es el constante bloqueo de sitios relacionados con anatomía humana, que se confunden con sitios pornográficos).
También se han creado sistemas basados en 'listas negras', donde simplemente se tiene una base de datos de los sitios reportados como nocivos. Esta solución sin embargo, ha resultado poco efectiva, dado el hecho de que es inviable mantener actualizadas estas listas con los miles de sitios Web emergentes día a día.

Por otro lado, las técnicas de inteligencia computacional resultan ser muy apropiadas para los problemas de clasificación de información desconocida a partir de ejemplos conocidos -lo cual constituye el problema en mención-, ya que permiten, a partir de un número grande de muestras, determinar características no evidentes para una clasificación más precisa. Por esto, se propone hacer uso de dichas técnicas como aproximación a la solución del problema de clasificación y filtrado de contenidos Web.

DESARROLLO DE LA PROPUESTA

Para el desarrollo de la propuesta Se hizo uso de la metodología ágil XP (Programación Extrema), haciendo énfasis en los ciclos de desarrollo cortos, y en el desarrollo orientado por pruebas.

El producto desarrollado puede describirse en las dos etapas del proceso de puesta en funcionamiento del filtro: el entrenamiento y la clasificación. En esta primera versión, se incorporan los siguientes modelos de clasificación:

- Redes neuronales
- Naive Bayes
- Árboles de decisión

ENTRENAMIENTO DE TEXTO:

Proceso de entrenamiento

Para el proceso de entrenamiento, se parte de dos conjuntos de URL. Uno, de sitios previamente catalogados como nocivos (particularmente páginas con contenido pornográfico), y otro de sitios que tengan reconocido ser no nocivos. A partir de esta información, se realiza el siguiente proceso (descrito en la figura 1):

1. Se extrae el texto contenido en cada URL, removiendo palabras de parada (pronombres, conectores y otras palabras auxiliares) de acuerdo al idioma identificado para dicho contenido, en caso de que dicho URL siga en línea. Inicialmente el producto reconoce dos idiomas, inglés y español.
2. Para el contenido extraído del conjunto completo en cada categoría y para cada idioma, se registra en una base de datos la información estadística de las palabras contenidas en él, específicamente el número de ocurrencias de cada palabra.
3. A partir de la información estadística obtenida de las páginas, se aplica una función que determina qué palabras harán parte del vector de características de las páginas. Un vector de características será un arreglo de bits, donde cada posición corresponde a una palabra, y su valor, para un URL determinado, será 1 ó 0 dependiendo de si existe o no dicha palabra en su contenido, teniendo en cuenta que una palabra se considera existente dentro de un texto, si cumple con una frecuencia mínima establecida.
4. Una vez se determine cuál será el vector de características, se calculan los vectores de cada una de las contenidos extraídos inicialmente.
5. Los contenidos de cada uno de los URL de los conjuntos iniciales, transformados a su forma de vector binario, son aplicados a un proceso de entrenamiento para cada una de las técnicas que vaya a utilizar el filtro integrado al proxy.
6. Para poder obtener indicadores de precisión y exhaustividad, se entrenan diferentes modelos con cada una de las técnicas escogidas para la clasificación, en donde como resultado a estas pruebas se identifica cual es la técnica con más altos índices de exactitud a la hora de hacer una clasificación. Estos experimentos se realizan entrenando los modelos variando el número de características y el número de las muestras, tanto paginas nocivas como no nocivas.

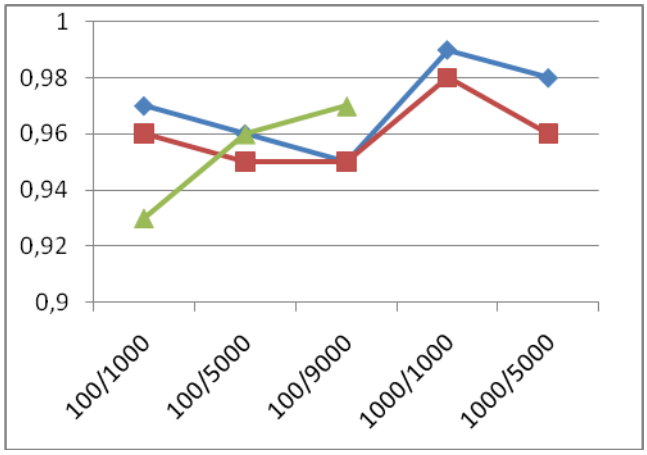


Figura1: indicadores de precisión

De esta fase de entrenamiento preliminar se obtiene que las técnicas de clasificación alcanzan a tener una precisión superior al 90% y en general una exhaustividad superior al 85%.

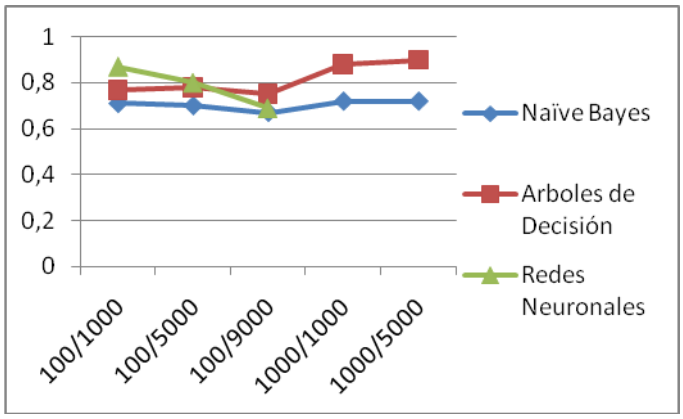


Figura2: indicadores de exhaustividad

7. Como fase final del proceso de entrenamiento se depuran las técnicas de clasificación y las características seleccionadas con las cuales se construye el vector de características el cual es la base para realizar los entrenamientos. Esto con el fin de conseguir niveles de precisión y exhaustividad más altos.
8. Se selecciona el algoritmo AdaBoost el cual con base en modelos entrenados bajo diferentes técnicas, en este caso Naive Bayes, Redes Neuronales y Arboles de Decisión, esto nos permite realizar un consenso para llegar a una nuevo modelo el cual se genera partiendo de los fallos o desaciertos de las técnicas anteriormente escogidas.

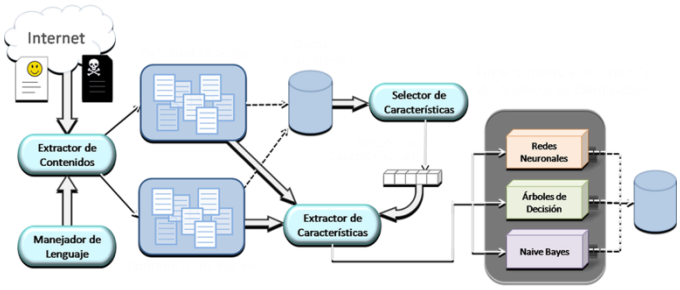


Figura1: proceso de Entrenamiento de Contenidos

ENTRENAMIENTO DE IMAGENES:

Para la clasificación de imágenes se hace un proceso similar al seguido en el entrenamiento de contenidos. Se parte de un conjunto de imágenes que se clasifican como nocivas y otro conjunto de imágenes clasificadas como no nocivas..

Para el manejo de imágenes, lo primero que realizamos es escalar las imágenes a un tamaño apropiado para poder manipularlas de una forma rápida. En esté momento ya podemos manipular las imagen la cual deseamos clasificar. Seguido a esto, hacemos la extracción de siluetas en una imagen blanco y negro, según una semilla con un color determinado ingresado extraemos de la imagen original todos aquellos pixeles que estén en un rango de distancia con el color semilla ingresado. Luego de tener la silueta de la imagen procedemos a convertirla en una imagen cuadrada de 10x10 para tener un estándar de todas las siluetas extraídas durante todo el proceso. Se escogió el tamaño de 10x10 ya que esta silueta es la que le ingresamos a una red neuronal para el entrenamiento y posterior clasificación y, al igual que con el entrenamiento de texto, una red neuronal se bloquea al momento de ingresarle una cantidad grande de entradas. Teniendo la imagen de 10x10 con la silueta extraída, procedemos a convertir esta silueta en un vector de tamaño 100 (10x10), en la cual cada posición del vector representa un pixel de la imagen con la silueta, de esta manera donde encuentre que haya un pixel blanco en el vector coloca un 0 y cuando encuentre un pixel negro coloca un 1 en el vector. Después de tener el vector que representa la silueta de una imagen, se procede a ingresar este vector a una Red Neuronal con un bit adicional que dice si es nociva o no la imagen, de esta manera se empieza a entrenar él modelo. Se hace este proceso con el conjunto inicial de imágenes para tener las siluetas de la mayor cantidad de imágenes que se pueda que se sepa que son nocivas y no nocivas.

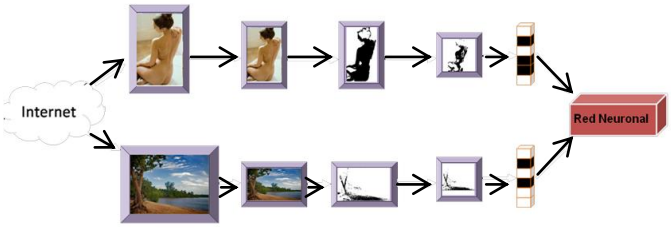
En las siguientes imágenes se ve una imagen de prueba la cual se sabe que es nociva, se ve la imagen original seguida de su silueta original y la imagen de 10x10 reducida de la silueta.

Imagen Original

Silueta Extraída de la figura original

Silueta reducida 10x10

El diagrama correspondiente al proceso de entrenamiento de imágenes es el siguiente:



ETAPA DE CLASIFICACIÓN

La herramienta de filtrado, como se mencionó anteriormente, se basa en el uso de un Proxy de Internet, que intercepta el tráfico del navegador hacia y desde Internet. Éste, como se muestra en la figura 2, una vez captura el contenido de un sitio de Internet, extrae su contenido textual, y lo transforma al vector de características descrito anteriormente. Éste vector de características es dado al clasificador previamente entrenado,

y dependiendo de la clasificación dada por este último (no nocivo o nocivo), se le envía al usuario, bien sea el contenido original, o un contenido alterno sano, actualmente, el clasificador utiliza una técnica Boosting. La cual hace uso de dos o más técnicas para la clasificación de la pagina haciendo un consenso entre los clasificadores y de ese consenso decide si una página es nociva o no.

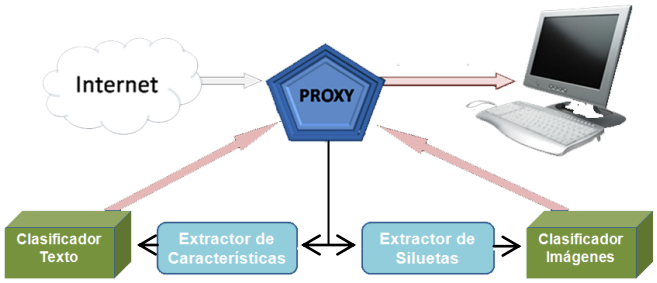


Figura2: Proceso de Interceptación y clasificación.

REFINAMIENTO DE LOS DATOS DE ENTRADA:

Para la fase 2 del proyecto Intellifilter, se seleccionaron un conjunto de técnicas para lograr un refinamiento de los datos y hacer un nuevo entrenamiento de los modelos usando las mismas técnicas ya tensionadas.

Básicamente estos datos de entrada son el vector de características el cual identificamos en el paso numero 3 del proceso de entrenamiento. Anteriormente estas características las tomamos con con base en la frecuencia con la que estas ocurrían en las paginas trabajadas para los entrenamientos.

Para la selección de estas características, se usó un componente de software llamado Best First de la herramienta para minería de datos Weka, el cual implementa un algoritmo para la búsqueda de características llamado Beam Search.

Para realizar esta selección de características es necesario partir de una muestra de un vector de características inicial, y un conjunto de muestras clasificadas.

Una ves identificadas las nuevas características se procede a hacer el nuevo entrenamiento de los modelos, esta vez con un vector de características mas pequeño pero con las mismas muestras ya trabajadas.

La segunda técnica utilizada para el proceso de refinamiento de datos es una técnica llamada AdaBoost la cual es aplicada a alas técnicas de clasificación como un meta algoritmo de clasificación, el cual ayuda a los clasificadores a mejorar sus resultados.
Esta técnica fue aplicada a la técnica de Naive Bayes, pues tes ultima fue la que nos presento una mayor precisión y exhaustividad a la hora de hacer una clasificación en un ambiente de pruebas.

CONCLUSIONES

El proyecto Intellifilter al ser desarrollado gracias al uso de diferentes herramientas libres es la muestra clara de cómo mediante la unión de esfuerzos, una buena razón de ser y la unificación de conocimiento se puede obtener un producto con fundamentos teóricos fuertes que contribuya al desarrollo de una sociedad, dando una solución a un problema actual, el cual es el contenido malicioso que circula a lo largo del internet, la red más grande de computadores.

REFERENCIAS

[1] Weka, Software para la minería de datos, http://www.cs.waikato.ac.nz/ml/weka/
[2] Paw Project, Software para la implementación de filtros por medio mediante un proxy.
[3] http://www.machinelearning.org/proceedings/icml2007/papers/168.pdf