Instituto Tecnológico de Costa Rica
Unidad de Computación

Inteligencia Artificial
Tarea Programada 1°

Kevin Walsh Muñoz
Jonathan Rojas Vargas

Sede San Carlos
20/09/2016

Contenido

Análisis del Problema

Requerimientos Identificados y Organización del Equipo de Trabajo

Requerimiento	Responsable
Indagar sobre la programación en R	Kevin Walsh Muñoz
Indagar sobre cómo obtener los datos desde correo, documentos HTML y Twitter	Jonathan Rojas Vargas
Realizar la base de datos	Jonathan Rojas Vargas
Realizar la interfaz con C#	Jonathan Rojas Vargas
Obtener datos correo	Kevin Walsh Muñoz
Obtener datos HTML	Jonathan Rojas Vargas
Obtener datos Twiter	Kevin Walsh Muñoz
Identificar idioma en R	Kevin Walsh / Jonathan Rojas
Bayes Naive en R	Kevin Walsh / Jonathan Rojas
Mecanismo de aprendizaje	Kevin Walsh / Jonathan Rojas

Solución planteada

En este apartado se detalla la solución del problema, desde la indagación sobre el lenguaje de programación “R” y hasta su implementación, también se investigó sobre las fórmulas bayesianas para obtener la probabilidad de que un texto sea de una categoría, determinando esta probabilidad se le indica a la aplicación que memorice aquellas palabras que no se encuentren en el análisis de un texto, más adelante se detalla sobre la solución planteada.

Programación en R

Para vincular R con C# se utiliza el framework “R.net” el cual permite la interoperabilidad entre C# y R, este framework se encuentra disponible en NuGet.org. Los pasos para programar en R y mostrar los resultados en c# son:

- Crear un script de R con su respectivo código.
- Ejecutar el siguiente comando en c# con la dirección del script de R

```
engine.Evaluate("source('c:/src/path/to/myscript.r')");
```

Lenguaje R

Entre las principales funciones o instrucciones que se indagaron para utilizar en los diferentes algoritmos son:

Conexión de R con postgresql.

```
con=dbConnect(PostgreSQL(), user="postgres", password="12345", dbname="prograIA")
```

Obtener la media y la varianza de una columna con valores.

```
media <- apply(palabrasDocumentos,2,median)
varianza <- apply(palabrasDocumentos,2,var)
```

Obtener una distribución normal usando media, varianza y cantidad de repeticiones

```
probAux <- dnorm(cantPalabra, mean=media, sd=varianza)
```

Obtener el contenido de una matriz en una posición específica.

```
cantPalabra <- palaAc[i,"num_repeticiones"]
```

Obtener correos, html y tweets

Para realizar esta implementación se indago sobre librerías que facilitaran la obtención de estas desde el lenguaje “R”, sin embargo no se encontró alguna que funcionara con el proyecto.

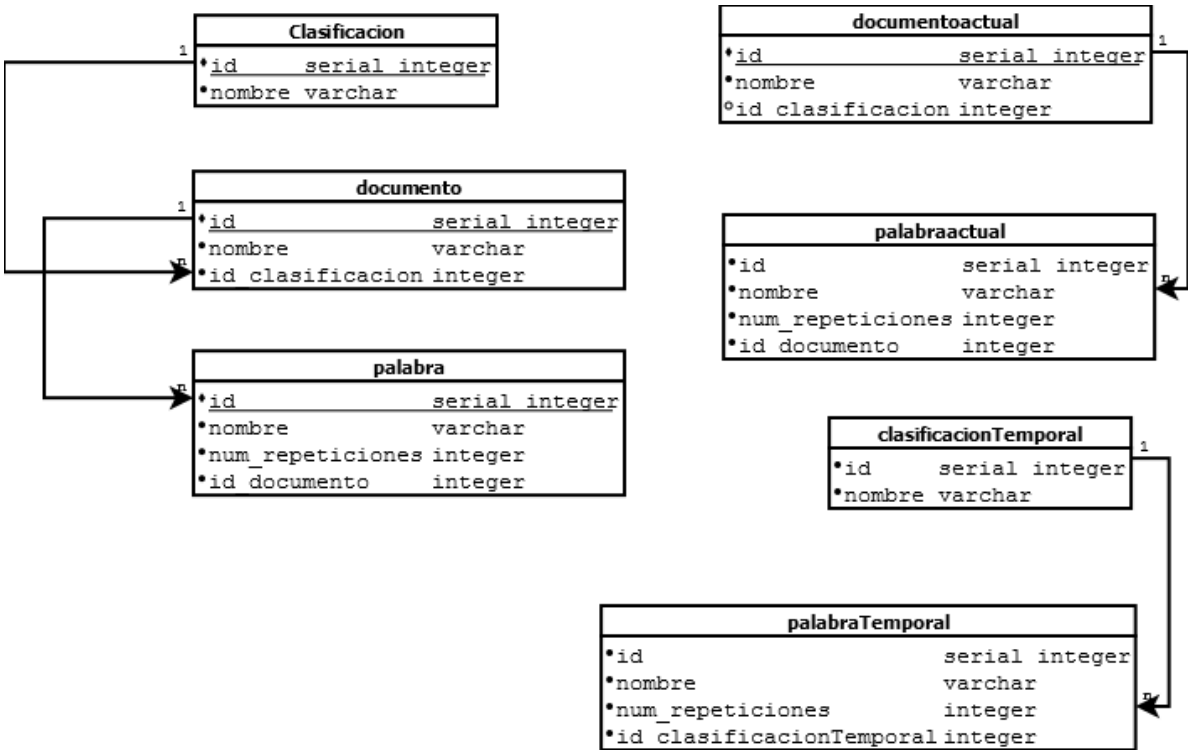
Sin embargo se implementó una librería llamada “gmailr” pero no funcionó, debido a la poca información que existe sobre ella, lo cual se realizó desde el lenguaje C# con la librería “OpenPop”, para obtener los correos de una cuenta de gmail es necesario el usuario y la contraseña de una persona, además autorizar desde el correo el acceso a la librería OpenPop para descargar los correos y almacenarlos en la base de datos para ser analizados.

Para obtener los textos de los html sucedió lo mismo, debido a esto se implementó desde C# con la librería “WebClient” de System.net.

Con los tweets fue diferente, ya que se encontró una librería desde el lenguaje de programación “R” llamada “twitteR”, que obtiene los tweets con el nombre de usuario de una persona y los almacena en la base de datos para ser analizados.

Base de datos

La base de datos se implementó en PostgreSQL con las siguientes tablas y atributos:



La funcionalidad de cada tabla se presenta a continuación:

Tabla	**Funcionamiento**
Clasificación	Esta tabla contiene las clasificaciones posibles que la aplicación pueda tener para analizar los textos.
Documento	Esta tabla posee los documentos de una clasificación.
Palabra	Esta tabla contiene las palabras de los documentos que se encuentran en una clasificación. Con las cuales usamos para clasificar un texto.
Documento Actual	Almacena los documentos para ser analizados para clasificarlos.
Palabra Actual	Almacena las palabras de los documentos actuales para

	analizarlas con las palabras almacenadas, que se encuentran dentro de una categoría.
Clasificación Temporal	Una vez analizados los textos, las palabras nuevas se agregan a la clasificación temporal con la clasificación que se obtuvo en el análisis.
Palabras Temporales	Son las palabras nuevas que están en un texto que ha sido analizado y clasificado. Estas se almacenan en palabras temporales y una vez que tengamos una serie de repeticiones sobre cada una, las agregamos en la tabla de palabras, asociadas a un documento y a la clasificación que corresponde.

Bayesiano

Para aplicar este método en el análisis de los textos se realizó una indagación sobre el teorema de bayes, el teorema de bayes se dice que expresa una probabilidad condicional de un evento aleatorio 'A' dado en 'B' en términos de la distribución de probabilidad condicional del evento 'B' dado 'A'.

Se poseen muestras: $\{A_1, A_2, \ldots, A_i, \ldots, A_n\}$

Conjunto de sucesos o eventos donde la probabilidad es mayor a cero. Sea 'B' una palabra de un documento de entrenamiento que se encuentra en una categoría "Deportes", la cual se conoce la probabilidad condicional $P(B|A_i)$. Entonces, la probabilidad $P(A_i|B)$ viene dada por lo siguiente:

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{P(B)} \qquad P(B|A_i) = P(x = v|c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e^{-\frac{(v-\mu_c)^2}{2\sigma_c^2}}$$

Donde $P(A_i|B)$ son las probabilidades posteriori.

- $P(B|A_i)$ es la probabilidad de que ‘A’ una palabra pertenece a ‘B’ una categoría
- $P(A_i)$ son las probabilidades a priori, la probabilidad de ‘A’ una palabra pertenezca a una categoría..
- ***P(B)*** es la evidencia, suma de las probabilidades de las muestras (palabras diferentes).

Resultados obtenidos

Tarea	Estado	Comentarios
Obtener datos correo	Completo	
Obtener datos HTML	Incompleto	No se pudo implementar
Obtener datos Twiter	Completo	
Clasificación	Incompleto	Se realiza con C#
Bayes Naive en R	Completo	
Mecanismo de aprendizaje	Completo	

Análisis del Mecanismos de aprendizaje implementado

El mecanismo de aprendizaje sigue una serie de pasos:

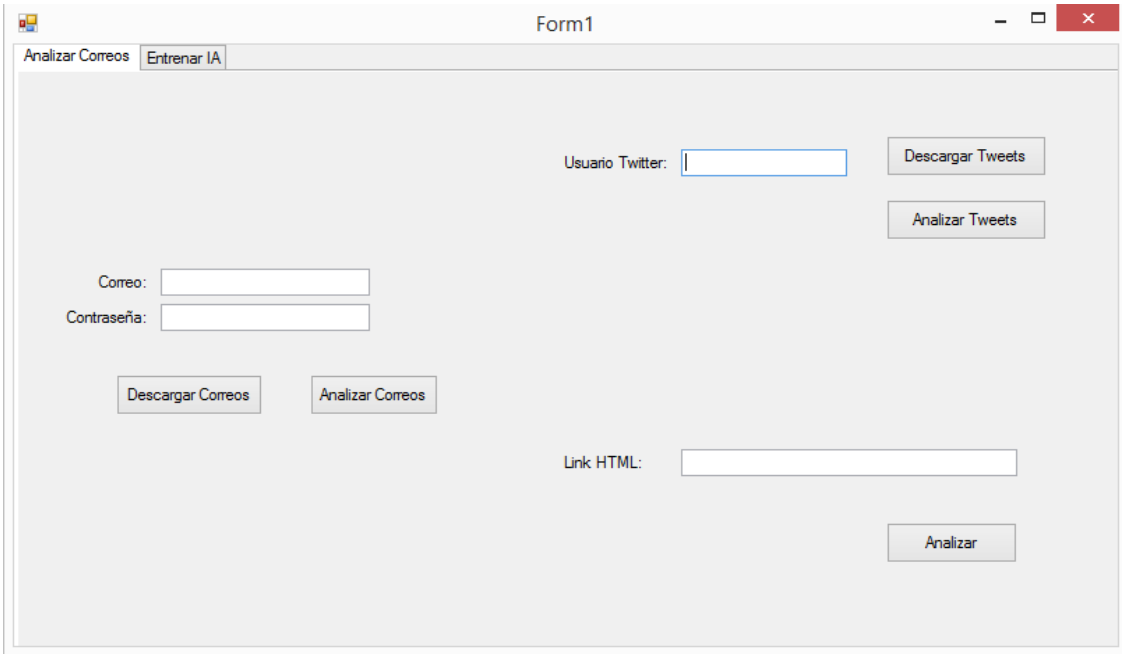
- Cada vez que un texto es analizado, las palabra que no se utilizaron para calcular la probabilidad en el algoritmo de Bayes debido a que estas no eran iguales a las palabras de entrenamiento son almacenadas en la base de datos junto a su cantidad de repeticiones y categoría.
- Luego de que 5 textos de la misma categoría son analizados se realiza una evaluación de las palabras que fueron almacenadas anteriormente con el objetivo de obtener las mejores.
- Como parámetro para obtener las mejores palabras se combinan las palabras iguales que se encontraron en los diferentes documentos y ordenarlas en orden de que las que más se repitan estén de primeras y de este modo seleccionar las 25 mejores y guardarlas como textos de entrenamiento.

Esta forma de aprendizaje lo que busca es obtener nuevas palabras que no se usan a la hora de realizar las probabilidades, y cómo combina las palabras entre varios documentos aumenta la efectividad de que sean palabras significativas para una clase, y como solo se eligen las que tengan más apariciones se evita insertar palabras poco significativas.

Manual de usuario

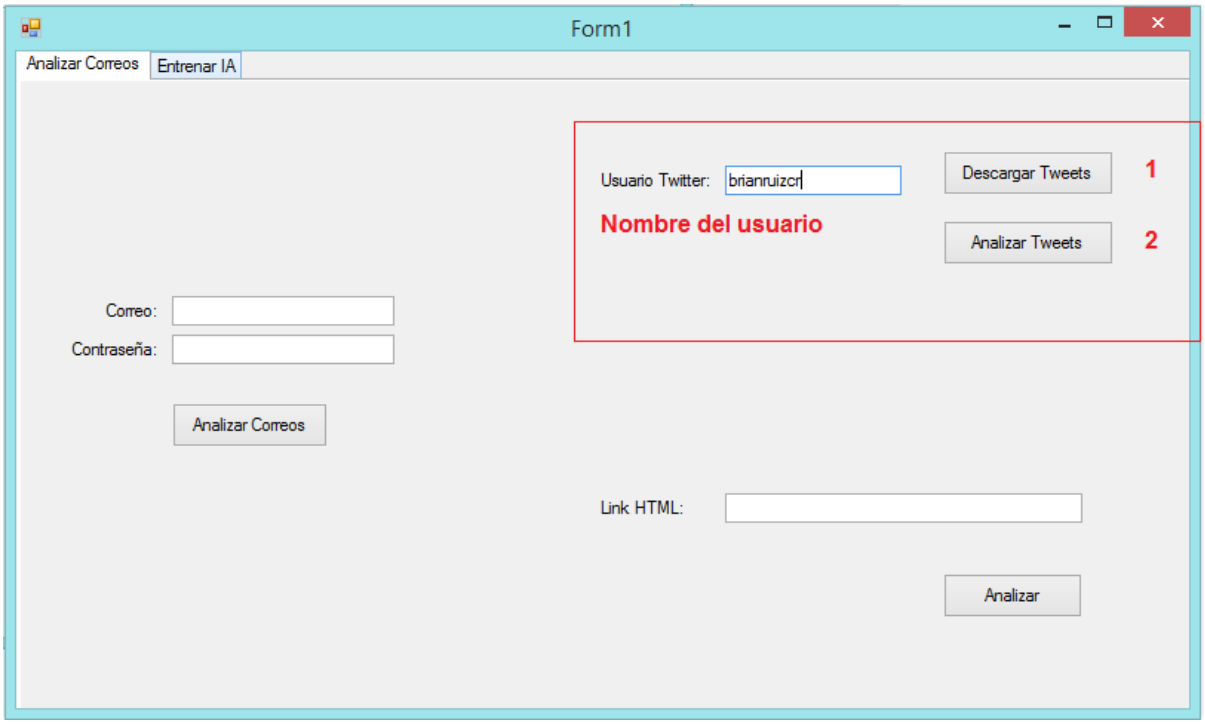
A continuación se muestran las principales partes del proyecto

Vista inicial



Analizar correos

Analizar tweets



Bibliografía

Leon, H. d. (03 de 2015). HDELEON.NET. Obtenido de http://hdeleon.net/realizar-una-conexion-el-correo-para-ver-los-mails-con-c-net-gmail-pop3-openpop-net/

Perez, A. (06 de 2013). AJPD Soft. Obtenido de http://www.ajpdsoft.com/modules.php?name=Content&pa=showpage&pid=264

R.Net. (14 de 06 de 2014). Obtenido de https://rdotnet.codeplex.com/documentation

Twitter Analytics Using R Part 1: Extract Tweets. (05 de 2014). Obtenido de https://www.credera.com/blog/business-intelligence/twitter-analytics-using-r-part-1-extract-tweets/