Centro de Investigación en Matemáticas A.C.
Programación y Algoritmos 1 (2019)

Tarea 10 - Clasificación de textos e Implementación del algoritmo de conjuntos conexos en imágenes pgm

Marco Antonio Esquivel Basaldua

1. Clasificación de textos

En este problema se realiza la clasificación de correos electrónicos como "spam" y "no spam" de acuerdo a un clasificador bayesiano ingenuo. Este tipo de clasificador está fundamentado en el teorema de Bayes y asume independencia entre las variables predictoras.

el modelo de probabilidad para un clasificador es $p(C|F_1,\ldots,F_n)$ sobre una variable dependiente C con un pequeño número de resultados (o clases). Esta variable está condicionada por varias variables independientes desde F_1 a Fn. Usando el teorema de Bayes se tiene:

$$p(C|F_1,\ldots,F_n) = \frac{p(C)p(F_1,\ldots,F_n|C)}{p(F_1,\ldots,F_n)}$$

En la práctica sólo importa el numerador, ya que el denominador no depende de C y los valores de F_i son datos, por lo que el denominador es constante.

El numerador es equivalente a una probabilidad compuesta:

$$p(C,F_1,\ldots,F_n) = p(C)\prod_{i=1}^{n} p(F_i|C)$$

Para la implementación de este algoritmo se utilizaron los datos de kaggle descargados de la pàgina https://www.kaggle.com/uciml/sms-spam-collection-dataset/version/1, de los cuales se usaron el 80% de los correos recuperados (de un total de 5573) para entrenar al algoritmo y el 20% restante para probarlo. A partir de esta prueba se obtienen los siguientes resultados:

Correos spam ingresados: 144
Correos spam detectados: 223
Correos no spam ingresados: 969
Correos no spam detectados: 871
Correos sin clasificar 19

Los correos que no fueron clasificados se debe a que las palabras que fueron leídas no fueron anteriormente clasificadas, por tanto no existe información concluyente para su clasificación.

2. Implementación del algoritmo de conjuntos conexos en imágenes pgm

El algoritmo implementado en este problema localiza la cantidad y tamaño de los conjuntos conexos a partir de una imagen binaria pgm en la que los pixeles en blanco representan elementos de los conjuntos. Se dice que dos pixeles o conjuntos son conexos si a partir de uno de ellos con coordenadas (i, j), se puede localizar al otro en alguna de las coordenadas $(i-1, j-1)$, $(i-1, j)$, $(i-1, j+1)$, $(i, j-1)$, $(i, j+1)$, $(i+1, j-1)$, $(i+1, j)$ o $(i+1, j+1)$.

Recorriendo la imagen dada se aplica el algoritmo BFS (*Breadth First Search*) para encontrar los pixeles conectados.

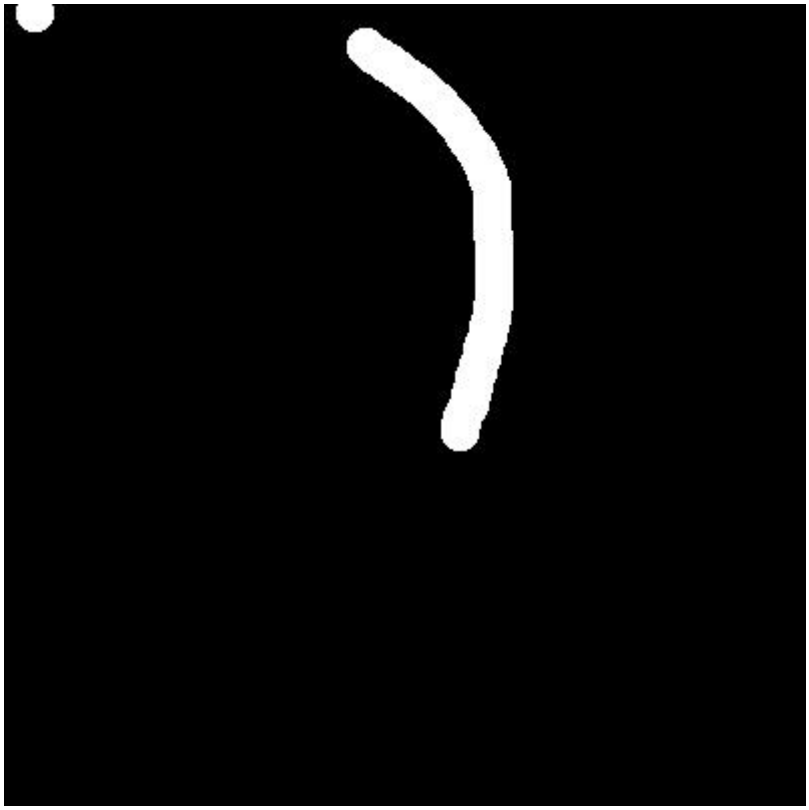
La cantidad de conjuntos conexos son mostrados por la terminal de la consola, mientras que información más detallada conteniendo el número del elemento encontrado y cantidad de componentes que lo componen es registrada en el archivo de texto *Elementos conexos.txt*. Los elementos con mayor y menor número de componentes son conservados y mostrados en la imagen de salida *im_salida.pgm*. Adicional a ello, se contabiliza el tiempo requerido para la ejecución del algoritmo y éste es mostrado en la terminal.

Al aplicar este programa a las imágenes: *CC1.pgm*, *CC2.pgm*, *CC3.pgm*, *CC4.pgm* y *CC5.pgm* se obtienen los siguientes resultados:

CC1.pgm

Cantidad de elementos en la imagen: 16
Los componentes conectados en cada uno de ellos son:
Elemento 1 tiene 230 componentes.
Elemento 2 tiene 293 componentes.
Elemento 3 tiene 517 componentes.
Elemento 4 tiene 823 componentes.
Elemento 5 tiene 874 componentes.
Elemento 6 tiene 881 componentes.
Elemento 7 tiene 1321 componentes.
Elemento 8 tiene 1582 componentes.
Elemento 9 tiene 1586 componentes.
Elemento 10 tiene 1587 componentes.
Elemento 11 tiene 1862 componentes.
Elemento 12 tiene 2221 componentes.
Elemento 13 tiene 2929 componentes.
Elemento 14 tiene 3518 componentes.
Elemento 15 tiene 4322 componentes.
Elemento 16 tiene 4587 componentes.

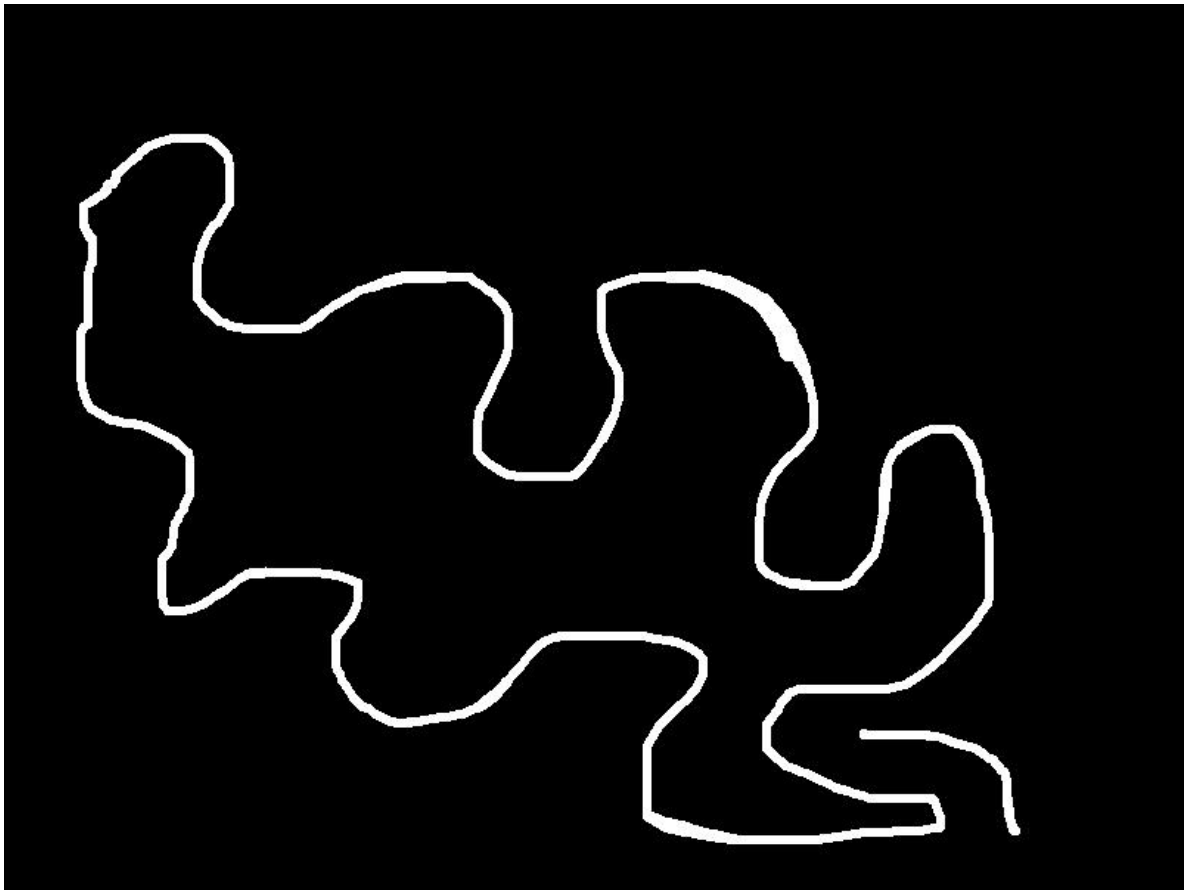
Tiempo tomado en la ejecucion del algoritmo
0.975355 seconds



CC2.pgm

Cantidad de elementos en la imagen:9
Los componenetes conectados en cada uno de ellos son:
Elemento 1 tiene 939 componentes.
Elemento 2 tiene 945 componentes.
Elemento 3 tiene 1039 componentes.
Elemento 4 tiene 1347 componentes.
Elemento 5 tiene 2105 componentes.
Elemento 6 tiene 3163 componentes.
Elemento 7 tiene 4146 componentes.
Elemento 8 tiene 4275 componentes.
Elemento 9 tiene 18800 componentes.

Tiempo tomado en la ejecucion del algoritmo
3.15293 seconds



CC3.pgm

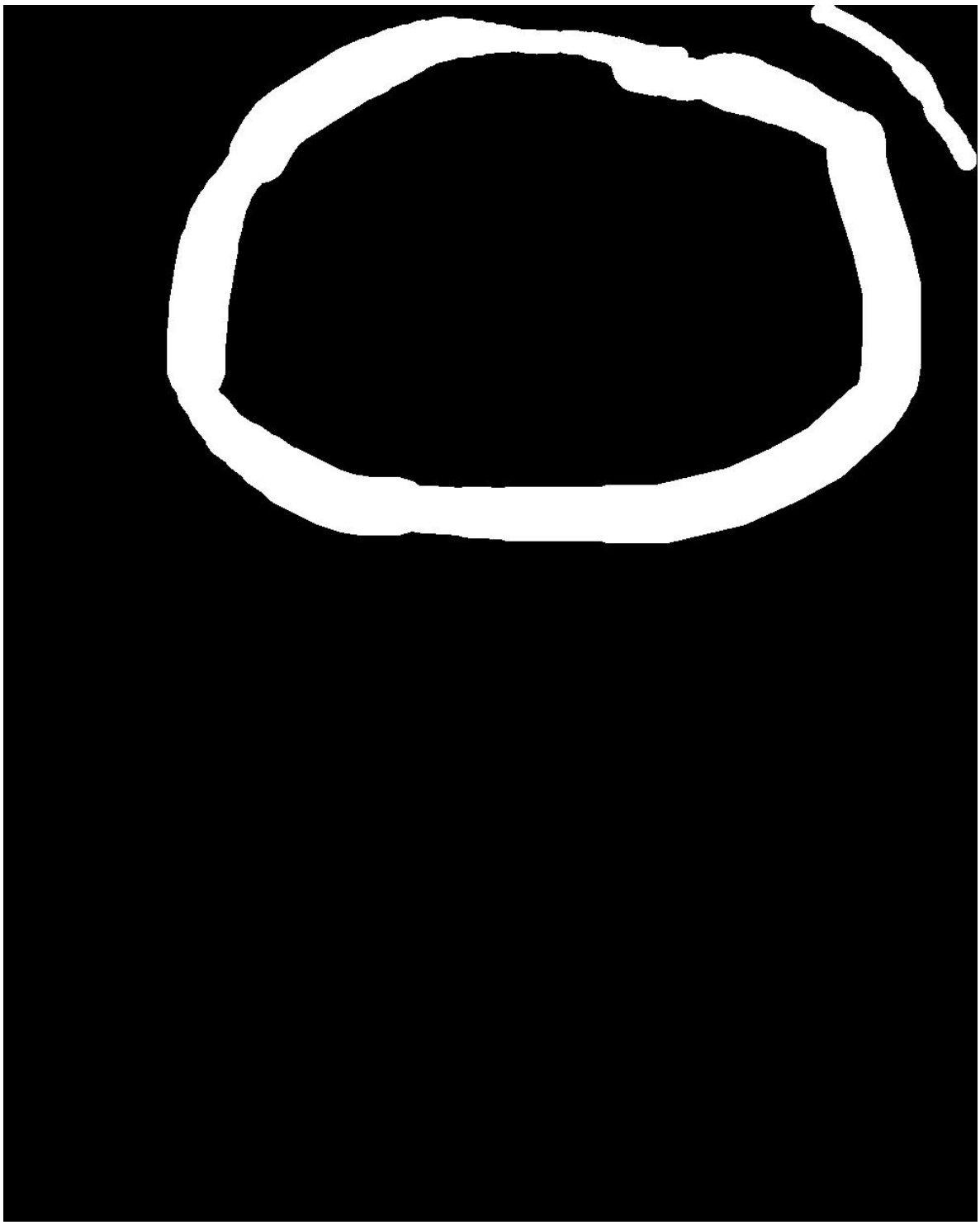
Cantidad de elementos en la imagen:12
Los componenetes conectados en cada uno de ellos son:
Elemento 1 tiene 988 componentes.
Elemento 2 tiene 1545 componentes.
Elemento 3 tiene 1626 componentes.
Elemento 4 tiene 1879 componentes.
Elemento 5 tiene 2659 componentes.
Elemento 6 tiene 2723 componentes.
Elemento 7 tiene 2954 componentes.
Elemento 8 tiene 3079 componentes.
Elemento 9 tiene 3936 componentes.
Elemento 10 tiene 4021 componentes.
Elemento 11 tiene 4649 componentes.
Elemento 12 tiene 4798 componentes.

Tiempo tomado en la ejecucion del algoritmo
0.505547 seconds

CC4.pgm

Cantidad de elementos en la imagen:7
Los componenetes conectados en cada uno de ellos son:
Elemento 1 tiene 5226 componentes.
Elemento 2 tiene 5992 componentes.
Elemento 3 tiene 7195 componentes.
Elemento 4 tiene 23206 componentes.
Elemento 5 tiene 32180 componentes.
Elemento 6 tiene 33170 componentes.
Elemento 7 tiene 108262 componentes.

Tiempo tomado en la ejecucion del algoritmo
16.4819 seconds



CC5.pgm

Cantidad de elementos en la imagen:10
Los componenetes conectados en cada uno de ellos son:
Elemento 1 tiene 1843 componentes.
Elemento 2 tiene 2260 componentes.
Elemento 3 tiene 4223 componentes.
Elemento 4 tiene 6819 componentes.
Elemento 5 tiene 8657 componentes.
Elemento 6 tiene 13543 componentes.
Elemento 7 tiene 14978 componentes.
Elemento 8 tiene 16286 componentes.

Elemento 9 tiene 19601 componentes.
Elemento 10 tiene 22848 componentes.

Tiempo tomado en la ejecucion del algoritmo
3.97052 seconds