Trabajo de Fin de Grado

Recuperación de imágenes basadas en técnicas de etiquetado semántico.

Image retrieval based on semantic tagging techniques.

Miguel Pérez Bello

La Laguna, 15 de agosto de 2015

Dra. **Vanessa Muñoz Cruz**, con N.I.F. 78.698.687-R profesora Ayudante Doctor adscrita al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como tutor

D. **Pedro Antonio Toledo Delgado**, con N.I.F. 45.725.874-B profesor Ayudante adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como cotutor

C E R T I F I C A N

Que la presente memoria titulada:

"Recuperación de imágenes basadas en técnicas de etiquetado semántico"

ha sido realizada bajo su dirección por D. **Miguel Pérez Bello**, con N.I.F. 45.940.754-W.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 15 de agosto de 2015.

Agradecimientos

La elaboración de un Trabajo Final de Grado es una tarea que formaliza la finalización de un ciclo. Un ciclo inicial en la vida que no hubiera sido posible desarrollar sin la ayuda de todas las personas que se han visto implicadas, de una manera u otra. A todos ellos es justo dedicarles unas líneas de agradecimiento.

En primer lugar, agradecer la labor de mis padres al brindarme la oportunidad de optar a los estudios universitarios, pues para ellos supuso un gran esfuerzo económico.

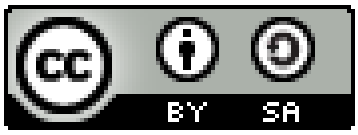
En segundo lugar, a los compañeros y amigos que siempre han estado a mi lado para darme cuenta de cuál es el camino que quiero seguir en el marco profesional, espero poder seguir contando con vuestro apoyo siempre.

A mi directora de TFG, Dra. Vanesa Muñón Cruz y a mi tutor Don Pedro Toledo, por darme las pautas, guías, referencias y herramientas que posibilitaron el desarrollo de este proyecto.

Por último, pero no por ello menos importante, a mi actual pareja, por brindarme la paciencia y el apoyo necesario que me permitió continuar con mi labor.

Gracias.

Licencia



Resumen

Ematic es un software diseñado por la Universidad de la Laguna para ayudar a adquirir de forma interactiva las primeras nociones de conceptos matemáticos, tales como evaluar la pertenencia o no a un conjunto, a través de una amplia variedad de ejercicios, que hacen uso de imágenes.

Pero, a medida que el nivel de dificultad aumenta, se torna difícil la tarea de englobar todas las combinaciones de ejercicios posibles, por lo que surge la necesidad de elaborar un método para la recuperación automática de imágenes con las que elaborar estos ejercicios.

De esta necesidad descrita, surge la idea de elaborar un sistema que, dada una imagen inicial y un amplio banco de imágenes, sea capaz de recuperar conjuntos de imágenes parecidas y distintas a esta imagen inicial seleccionada.

Para lograr dicho objetivo, se elaboran unas técnicas algorítmicas que son capaces de responder queries como 'devolver las n imágenes más cercanas a una imagen inicial', 'devolver las n imágenes más distantes (no semejantes)' o 'devolver ambos conjuntos a la vez'.

La idea de que dos imágenes se parezcan o no, se determina con el análisis y elaboración de una función que calcula la distancia entre dos imágenes cualesquiera.

Para satisfacer la necesidad que se encuentra en Ematic, se implementa una librería que es capaz de responder las queries descritas.

Los algoritmos diseñados y las correspondientes funciones de distancia han sido validados tanto con datos sintéticos, como con una base de datos ad hoc de imágenes reales, elaborada ad hoc.

Finalmente, se ha diseñado una Interfaz Gráfica para realizar las pruebas pertinentes y comprobar, de forma visual, el comportamiento de los algoritmos y las distintas funciones de distancia.

Palabras clave: Recuperación de imágenes, función de distancia, Ematic, librería.

Abstract

Ematic is software developed by the University of La Laguna in order to help to acquire interactively the basic math concepts, such as evaluate the membership in a set, using a wide variety of exercises which make use of images.

But, while increases the level of difficult, the task to enclose all possible combinations of exercises becomes hard. So, there is a needed to develop a method for the automatic retrieval of images that produces these exercises.

From this needed, the idea of develop a system that, given an initial image and a large bank of images, be able to retrieve sets of similar and different images to the initial image selected.

To achieve this goal, it develops some algorithmic techniques that are able to answer queries like 'return the N closest images to an initial image', 'return the N most distant images (not similar)' or 'return both sets'.

To determinate if two images are similar or not is determined by an analysis and design of a function that calculates the distance between any two images.

To meet the need present in Ematic, it implements a software library that is able to answer the described queries.

Designed algorithms and the corresponding distance functions, have been validated with synthetic data and a basic ad hoc image database.

Finally, a graphical user interface was implemented to run tests and check, in a visual way, the behavior of various algorithms and metrics.

Keywords: *Image retrieval, distance function, Ematic, library.*

Índice General

Índice de figuras

Índice de tablas

Introducción

1.1 Contextualización

En Ematic [19][20] se proponen una serie de ejercicios dirigidos al aprendizaje de conceptos matemáticos. Varios de estos ejercicios consisten en hallar la imagen diferente de un conjunto de imágenes propuesto.

El almacenamiento previo de estos ejercicios, de forma estática, limita el nivel de dificultad y la variedad presente en los mismos. Por ello se requiere de una técnica algorítmica que sea capaz de extraer, de forma automática, imágenes según ciertos criterios. Estos criterios describen ítems como el número de imágenes semejantes que se extraerán, la cantidad de imágenes distintas a recuperar y cuán de distintas van a ser estas últimas de las primeras.

En general, el campo de 'image retrieval' o 'recuperación de imágenes' [17][18] trata de resolver esta tarea a través de analizar un conjunto de etiquetas que se asocian a las imágenes, objeto de estudio. Dicho análisis determina el modo de elaborar los conjuntos finales de imágenes similares y diferentes a recuperar.

Para el caso particular de Ematic, se tratará de implementar un modelo de recuperación de imágenes que permita elaborar una mayor variedad de ejercicios. La dificultad de estos ejercicios se determinará de forma dinámica, a través de aumentar o disminuir la distancia de habrá entre el conjunto de imágenes semejantes y el conjunto de imágenes distintas. Cuanto mayor sea esta distancia, más sencillo será captar las diferencias entre estos dos conjuntos de imágenes.

1.2 Antecedentes

En los últimos años, varios estudios en este campo se enfrentan al problema de recuperar imágenes divergiendo en líneas de investigación, que convergen en objetivos, pero emplean diferentes ideas.

Trabajos como los propuestos por [3] Li et Al., describen cómo el entrenamiento de una estructura de datos, denominada taxonomía gramatical, bajo aprendizaje supervisado, determina un modelo apropiado para, a partir de la información asociada a las imágenes en forma de etiquetas, poder clasificar, anotar y organizar de forma jerárquica a las mismas.

Autores que comparten la misma línea de actuación, como [3] Deng et Al. se centran emplear dicha taxonomía gramatical para describir una función que permita medir la similitud entre imágenes, para su posterior recuperación. Los resultados presentes en los trabajos de [3] Salakhutdinov et Al. muestran que el aprendizaje de una jerarquía provee buenas clasificaciones, en clases de incluso pequeños conjuntos de casos de entrenamiento.

Por otro lado, el entrenamiento de un complejo modelo de reconocimiento de imágenes, realizado por [3] Whereas, Zweig et Weinshall, para un conjunto cerrado de categorías de objeto, propició buenos resultados en la clasificación incluso, de un objeto no contemplado en la jerarquía elaborada, dando a este nuevo objeto una clase y un lugar en la jerarquía.

Continuando con la línea de investigación de los autores nombrados, se elabora la hipótesis de utilizar el concepto de Taxonomía Gramatical para la elaboración de una métrica compleja que permita 'hilar fino' en el diseño de una función que permita el cálculo de la distancia entre dos imágenes.

Dicha taxonomía gramatical hace referencia a un tipo de estructura de datos que permite la clasificación jerárquica de conceptos (imágenes), según sus etiquetas asociadas, partiendo desde los conceptos más generales hasta los más específicos, que se encontrarían en los nodos hoja de este árbol semántico.

Esta clasificación jerárquica propicia un modelo que permite contrastar y combinar funciones para calcular la distancia semántica entre dos nodos cualesquiera de la taxonomía. El problema radica en que la construcción de una taxonomía no es trivial e implica un complejo proceso de análisis. En este

documento se detallará el método desarrollado, fruto de la investigación realizada.

A menudo, el concepto de taxonomía gramatical viene acompañado del concepto Ontología Semántica, esto es debido a que la tarea de construir Ontologías de dominio específico comparte gran similitud con la tarea de construir taxonomías ya que se ordenan ítems de acuerdo a materias relevantes, se identifican los tópicos y se organiza según las necesidades; para ello se refleja y estructura el conocimiento en un dominio específico, organizando los conceptos importantes, de forma jerárquica, en una estructura de árbol.

1.3 Objetivos generales

El objetivo principal de esta investigación pretende la elaboración de un sistema que, dado un amplio banco de imágenes y dada una imagen inicial, sea capaz de resolver la tarea de recuperar, según ciertos criterios, un conjunto de imágenes.

Estos objetivos plantean la necesidad de elaborar técnicas algorítmicas que, dada una entrada como la descrita anteriormente, devuelva varios conjuntos de imágenes, según la necesidad planteada.

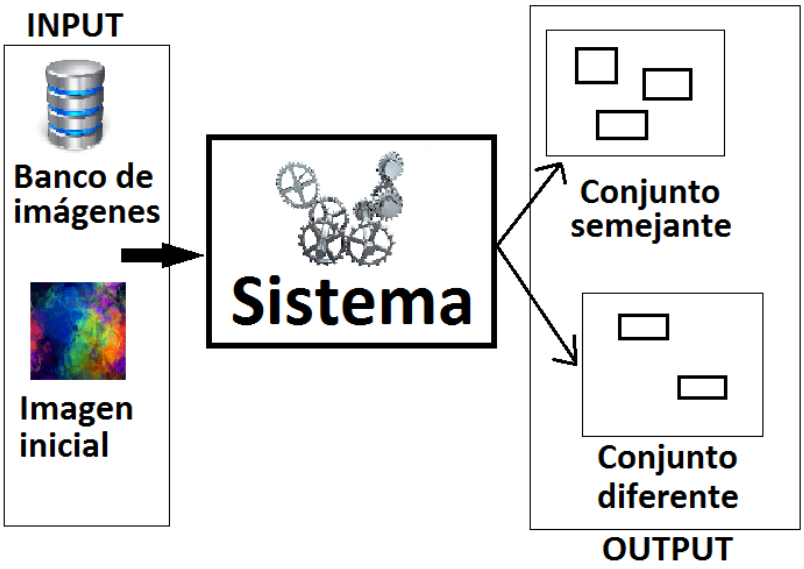


Figura 1.1. Representación conceptual del sistema

La documentación, investigación e implementación experimental de ideas extraídas de trabajos previos también forman parte de los objetivos y facilitará la comprensión de las técnicas englobadas en este ámbito.

1.4 Objetivos específicos

Concretamente, se pretende elaborar un sistema completo que permite la recuperación de imágenes cercanas atendiendo a una serie de criterios establecidos previamente. Dicho sistema debe responder con eficacia a las cuestiones que se plantean y para ello, se han propuestos varios escenarios.

La primera cuestión plantea la necesidad de recuperar de las N imágenes más cercanas a una imagen inicial seleccionada. Para ello se recurre a una estructura algorítmica basada en el algoritmo de los k-vecinos.

Uno de los escenarios elabora un modelo de recuperación de imágenes, denominado **modelo disco**, que permite la recuperación de las N imágenes más cercanas a una imagen inicial dada y de M imágenes que difieran de las primeras.

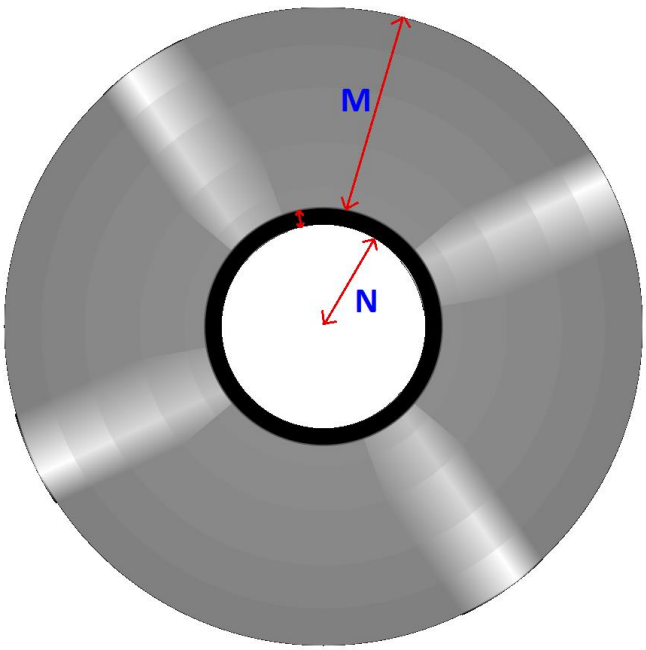


Figura 1.2. Modelo disco

Una última query propone la elaboración de otro escenario basado en un **modelo satélite**, que permite recuperar varios grupos de imágenes. Un primer conjunto de imágenes semejantes a una imagen inicial seleccionada. Y un segundo conjunto de imágenes, semejantes entre sí, pero diferentes al primer grupo según cierto umbral prestablecido.

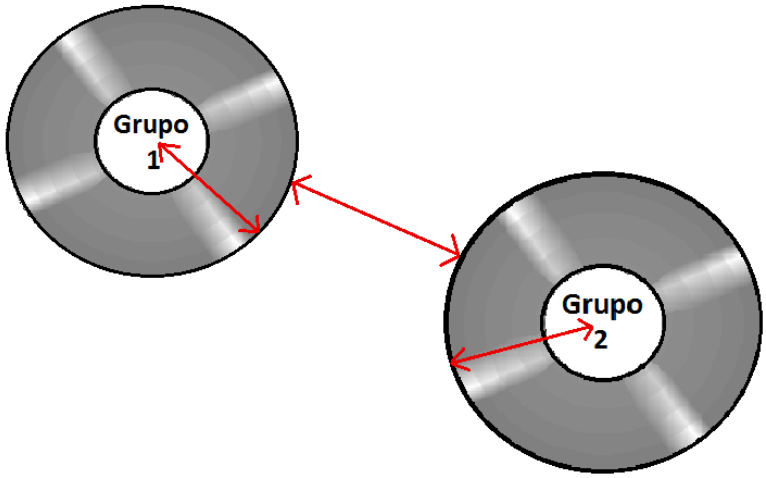


Figura 1.3: Modelo satélite

Para el desarrollo de los dos modelos comentados, se asumirán las siguientes condiciones de partida: se dispone de un banco de imágenes etiquetadas, de una imagen de partida también etiquetada y de ciertas condiciones sobre estas etiquetas. Estas condiciones establecen lo siguiente:

1. Multietiquetas: una imagen puede contener múltiples etiquetas y estas etiquetas a su vez pueden aparecer en varias imágenes.
2. Las etiquetas asociadas a una imagen son suficientes para construir relaciones, es decir, podría describirse el contenido de la imagen partiendo de estas etiquetas y podría relacionarse una imagen con otra por medio de sus etiquetas.
3. Se asume que pueden existir relaciones de jerarquía entre las etiquetas, es decir, puede ocurrir que haya conceptos más generales que engloban otros conceptos específicos, como por ejemplo, ser vivo engloba a animal.

Desarrollo

El desarrollo del sistema de recuperación de imágenes comentado, requiere del análisis y diseño de varios módulos que, en conjunto, lidian con la tarea encomendada. De forma esquemática, podemos separar dichos módulos tal y como se muestra en la siguiente figura:

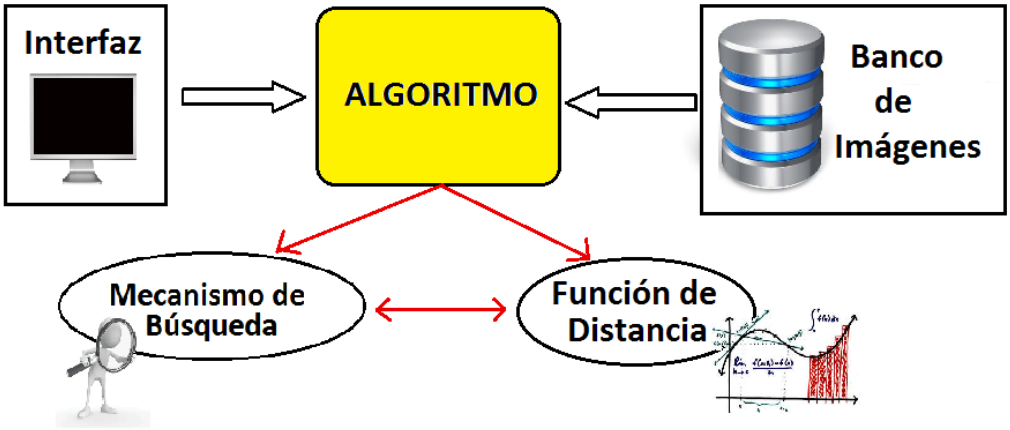


Figura 2.1: esquema del sistema diseñado

Como podemos ver, el sistema diseñado se compone de los siguientes módulos: Una interfaz gráfica, que permite la selección de los diferentes parámetros de entrada para el algoritmo. Y además, constituye un modo intuitivo de experimentar si las agrupaciones de imágenes obtenidas, son válidas aparentemente.

Otro elemento del que se compone el sistema es de un banco de imágenes de entrada, del que se analizan las imágenes y se extraen los grupos deseados. Para ello, se requiere de otra de las partes del sistema, el algoritmo de recuperación de imágenes.

El algoritmo de recuperación de imágenes se relaciona con el mecanismo de búsqueda y con la función que posibilita el cálculo de la distancia entre dos imágenes, pues ambos conforman el método en que se realiza dicha tarea.

Cada uno de los componentes del sistema serán descritos de forma detallada en este documento.

2.1. Mecanismo de búsqueda. Algoritmos

Como se ha comentado anteriormente, el problema a resolver consiste en:

Dado un conjunto de imágenes I, con etiquetas asociadas y dada una imagen inicial $I_0 \in I$. Determinar dos conjuntos de imágenes, semejantes y diferentes a través del diseño de los dos modelos algorítmicos comentados, el modelo disco y el modelo satélite.

Antes de comentar el funcionamiento de estos modelo, se plantean tres parámetros necesarios para la comprensión de cómo se construyen.

Defínase la distancia 1 ($\mathbf{D_1}$), como la distancia existente entre la imagen de partida y la última imagen que pertenece al grupo de imágenes semejantes a la misma, de ahora en adelante $\mathbf{G_1}$ (grupo 1).

Defínase una distancia 2 ($\mathbf{D_2}$), como la distancia existente entre la primera y la última imagen que pertenecen al grupo de imágenes diferentes a la imagen de partida, de ahora en adelante $\mathbf{G_2}$ (grupo 2).

Por último, defínase la distancia que separa ambos grupos de imágenes como distancia de separación o **GAP**.

1.1.1 Modelo disco

Este modelo se plantea como una extensión del algoritmo de los k-vecinos, aplicado a la tarea planteada inicialmente. La solución que propone, como vemos en la figura 2.1, presenta una estructura en forma de disco, dónde cada uno de estos discos representa un grupo de imágenes recuperado a partir de la imagen inicial.

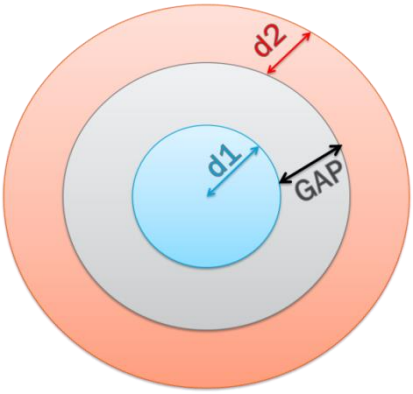


Figura 2.2: Representación del modelo disco

El primer grupo que se forma es un grupo de imágenes semejantes, al igual que en modelo satélite. La peculiaridad la presenta el segundo grupo, pues estas imágenes son diferentes al primer grupo y no necesariamente se parecen entre sí.

Formalmente, la pertenencia de una imagen a uno de estos dos grupos (G_1 y G_2) se define como:

$$\mathbf{G_1 = \{I_i \,/\, d(I_0, I_i) \leq d1\ \}}$$

El conjunto G1 se compone de aquellas imágenes cuya distancia con la imagen de partida sea inferior a d_1.

$$\mathbf{G_2 = \{I_i \,/\, d(I_0, I_i) > d1 + GAP\ \wedge\ d(I_0, I_i) \leq\leq d_1 + GAP + d_2\}}$$

Forman parte del conjunto G2 aquellas imágenes cuya distancia esté comprendida entre (d_1 + GAP) y (d_1 + GAP + d_2).

El proceso que define este modelo se describe en los siguientes pasos:

En primer lugar se comienza por seleccionar la imagen origen (I_0) del primer grupo y se añaden las d_1 siguientes imágenes más cercanas a G_1.

Se continúa la ejecución desechando las imágenes cuya distancia se encuentre entre d_1 y d_1 + GAP.

Finalmente, se seleccionan las d_2 siguientes imágenes más cercanas a la distancia d_1 + GAP + d_2 y se añaden a G_2.

A continuación se muestra un ejemplo que permite ilustrar, de forma simple, el funcionamiento de este modelo algorítmico:

Dado un conjunto de imágenes: 'manzana', 'pera', 'pizza', 'coche', 'pájaro', 'gato'.

Dados un conjunto de parámetros que establecen las condiciones del ranking a elaborar: d1 = 1, d2 = 2 y gap = 1. Y dado un conjunto de etiquetas asociadas a las imágenes de partida, de la siguiente manera:

Imagen	Et1: ser vivo	Et2: comida	Et3: vegetal	EtN: animal
Manzana	0	1	1	0
Pera	0	1	1	0
Pizza	0	1	0	0
Pájaro	1	0	0	1
Gato	1	0	0	1

Tabla 2.1. Ejemplo de asociación imágenes – características.

Como vemos en la tabla 3.1, se caracterizará de forma binaria la relación de pertenencia o no de una etiqueta a una imagen. De este modo, si una imagen posee una característica, se le asocia como valor un 1, en caso contrario 0. Esto permite discretizar las condiciones del problema.

Por último, dada una imagen inicial I_0 = 'Manzana' y asumida la utilización de una métrica simple, como la Distancia Euclídea (descrita posteriormente). Se determina de forma simple el siguiente ranking:

Grupo 1: {Pera}

d_E (Manzana, Pera) = 0

Grupo 2: {Gato, Pájaro}

d_E (Manzana, Gato) = d_E (Manzana, Pájaro) = $\sqrt{4} = 2$

Como vemos, una métrica tan simple como la distancia Euclídea se comporta de forma válida para problemas cuya dimensión está muy acotada. El problema lo encontramos cuando el número de objetos y el número de características crece de forma exponencial. En este caso se deben emplear métricas más complejas.

1.1.2 Modelo satélite

El modelo denominado 'satélite', describe un espacio en el que se diferencian claramente dos grupos que, entre sí comparten similitudes, pero difieren uno de otro.

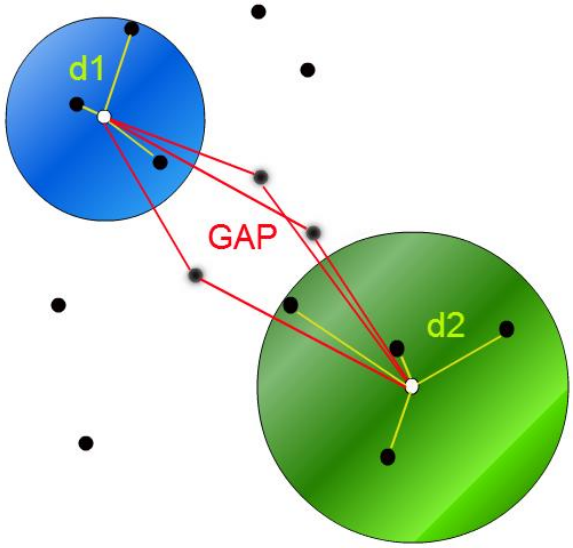


Figura 2.3. Modelo satélite.

Formalmente, la construcción de cada uno de estos grupos atiende a las siguientes expresiones:

$$\mathbf{G_1 = \{I_i \,/\, d(I_0, I_i) \leq d1\ \}}$$

El grupo G1 se compone de las imágenes con distancia inferior a d_1, al igual que ocurre en el modelo disco.

$$\begin{aligned}\mathbf{G_2 = \{I_i \,/\, d(I_0, I_i) \geq d1 + GAP \wedge}\\ \mathbf{d(I_j, I_i) \leq\leq d_2 \wedge}\\ \mathbf{\forall\, I_k, |d(I_0, I_k) - d_1 + GAP + d_2| \geqslant}\\ \mathbf{|d(I_j, I_0) - d_1 + GAP + d_2|\}}\end{aligned}$$

El segundo grupo se compone de las imágenes más cercanas a la segunda imagen de referencia y que además no superen la distancia d_2, ni pertenezcan al grupo G_1. Esta segunda imagen de referencia será la imagen más cercana a la distancia $d_1 + GAP + d_2$.

Como vemos, la diferencia con el primer modelo radica en la construcción del segundo grupo, para ello: primero se elabora el ranking de imágenes semejantes a la imagen de partida, se continúa localizando la imagen más cercana a la distancia d1 + gap + d2 y se finaliza agrupando las imágenes más cercanas a esta última, que no superen la distancia d_2 y cuya distancia sea superior a la distancia d1 + gap, es decir que no pertenezcan al grupo G_1.

El proceso que describe el modelo satélite podría describirse como sigue:

En primer lugar, se seleccionan las d1 imágenes más cercanas a la imagen origen (I_0) y se añaden al conjunto G_1.

Se busca la imagen más cercana a la distancia $d_1 + GAP + d_2$. Esta imagen será considerada como la segunda imagen de referencia.

Finalmente se agrupan las d_2 imágenes más cercanas a esta segunda imagen de referencia. Atendiendo a las restricciones impuestas en la definición formal ($G2 = \{Ii \ / \ d(I0, Ii) \geq d1 + GAP \ \hat{} \ d(Ij, Ii) \leqq d2 \ \hat{} \ \ V \ Ik, \ |d(I0, Ik) - d1 + GAP + d2| \geqq |d(Ij, I0) - d1 + GAP + d2|\}$).

De este modo se construyen los dos grupos y se desecha el resto de imágenes.

Mecanismo de búsqueda. Métricas

3.1. Análisis de Métricas simples

Con el fin de elaborar una función que permita el cálculo de distancias entre imágenes, se describen éstas en una serie de características binarias que representan la presencia / ausencia de una etiqueta en una imagen.

A partir de esta discretización binaria de las características de las imágenes, se emplean un conjunto de métricas simples para comprobar su funcionamiento a la hora de ser empleadas en los dos modelos algorítmicos planteados, el modelo disco y el modelo satélite.

3.1.1. Distancia Euclídea

La primera de las métricas empleadas es la distancia Euclídea [6]. Esta métrica representa la distancia que mediríamos con una regla entre dos puntos, y que se deduce a partir del teorema de Pitágoras.

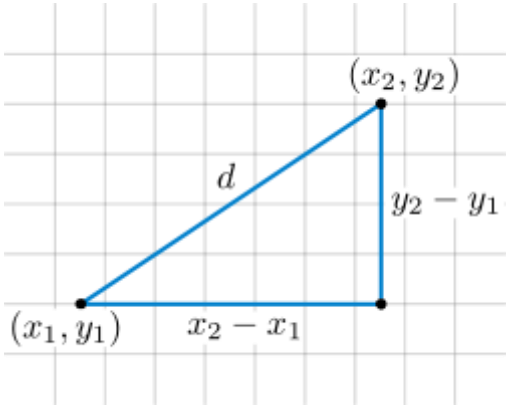


Figura 3.1. Representación distancia Euclídea.

Se formula de la siguiente manera:

$$d_E(P_1, P_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Figura 3.2. Distancia Euclídea para espacio bidimensional.

La fórmula descrita atiende a un espacio descrito por dos características (x e y). Dado que las imágenes con las que trataremos se componen de tantas etiquetas como sea necesario, necesitamos extrapolar dicha fórmula al caso n-dimensional. En este caso la fórmula se transforma de la siguiente manera:

$$d_E(P, Q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}.$$

Figura 3.3. Distancia Euclídea para espacio N-dimensional.

Siendo P y Q dos vectores P = (p_1, p_2, …, p_n) y Q = (q_1, q_2, …, q_n) de características binarias, que representan la presencia o ausencia de una etiqueta en una imagen.

Un ejemplo de uso de esta función de distancia podría ser el siguiente:

1. Definamos las siguientes características: C = {‘Ser vivo’, ‘Comida’, ‘Vegetal’, ‘Animal’}.

2. Definamos la imagen 'Manzana', por el siguiente conjunto P de características como sigue: P = (0, 1, 1, 0). En P se indica que la imagen Manzana contiene las etiquetas 'Comida' y 'Vegetal'.
3. Definamos la imagen 'Gato' con el conjunto Q de características siguiente: Q = (1, 0, 0, 1).

Obtenemos por tanto, que la distancia entre las imágenes 'Manzana' y 'Gato', representadas ambas por los vectores P y Q respectivamente, se define como:

$$\mathbf{D_E\ (`Manzana',\ `Gato')} = \sqrt{(\mathbf{0-1})^2 + (\mathbf{1-0})^2 + (\mathbf{1-0})^2 + (\mathbf{0-1})^2} = \sqrt{\mathbf{1+1+1+1}} = \sqrt{\mathbf{4}} = \mathbf{2}$$

Por lo que obtenemos, que la distancia resultante es de 2 unidades.

3.1.2. Distancia de Manhattan

Otra función empleada en el cálculo de la distancia entre dos imágenes es la denominada distancia de Manhattan [7], esta métrica representa la suma de las longitudes de las proyecciones del segmento de línea entre los puntos de interés, sobre el sistema de coordenadas particular.

Se formula de la siguiente forma:

$$d_1(\mathbf{p}, \mathbf{q}) = \|\mathbf{p} - \mathbf{q}\|_1 = \sum_{i=1}^{n} |p_i - q_i|,$$

Figura 3.4. Distancia Manhattan.

Donde $\mathbf{p} = (p_1, p_2, \ldots, p_n)$ y $\mathbf{q} = (q_1, q_2, \ldots, q_n)$ representan vectores; en este caso particular son las características binarias asociadas a cada imagen.

En este caso particular, un ejemplo de cálculo de la distancia entre dos imágenes, empleando la distancia de Manhattan, quedaría como sigue:

1. Definamos las siguientes características: C = {'Ser vivo', 'Comida', 'Vegetal', 'Animal'}.

2. Definamos la imagen 'Manzana', por el siguiente conjunto P de características como sigue: P = (0, 1, 1, 0). En P se indica que la imagen Manzana contiene las etiquetas 'Comida' y 'Vegetal'.
3. Definamos la imagen 'Gato' con el conjunto Q de características siguiente: Q = (1, 0, 0, 1).

La distancia de Manhattan nos daría el siguiente cálculo:

$$\mathbf{D_M = |0\text{-}1| + |1\text{-}0| + |1\text{-}0| + |0\text{-}1| = 1 + 1 + 1 + 1 = 4}$$

Por lo que obtenemos que la distancia resultante es de 4 unidades.

3.1.3. Distancia por penalización

La última de las métricas simples a comentar es una función de distancia diseñada ad hoc bajo el nombre de métrica por penalización. El fundamento de esta función se basa en querer penalizar el valor final a medida que encontramos características diferentes entre las imágenes a evaluar, o en premiar dicho valor en caso de que se encuentren características comunes.

Actúa de la siguiente manera:

a) En caso de que ninguna de las imágenes contenga la característica a evaluar, el valor final no se verá afectado:

$$\mathbf{Si\ A_i{=}0 \wedge B_i{=}0 \rightarrow D(A_i,B_i) = \sim 0 = 0}$$

b) En caso de que una de las imágenes contenga la característica a evaluar, pero la otra imagen no la contenga, se penalizará el valor de distancia aumentándola en una unidad:

$$\mathbf{Si\ A_i{=}0 \wedge B_i{=}1 \rightarrow D(A_i,B_i) = \sim(\text{-}1) = 1}$$

$$\mathbf{Si\ A_i{=}1 \wedge B_i{=}0 \rightarrow D(A_i,B_i) = \sim(\text{-}1) = 1}$$

Si por el contrario, ambas imágenes poseen la característica en conjunto, se premiará el valor de distancia disminuyéndolo en una unidad:

$$\textbf{Si } A_i{=}1 \wedge B_i{=}1 \rightarrow D(A_i,B_i) = {\sim}1 = -1$$

Nótese que A_i y B_i representan dos características binarias y '~' representa una operación de complementación que actúa de la siguiente manera:

$$\sim 1 = -1$$

$$\sim 0 = 0$$

$$\sim (-1) = 1$$

De este modo, si dos imágenes comparten una característica en común (las dos contienen la etiqueta), la distancia entre las mismas disminuye una unidad. Si una de las imágenes contiene la característica y la otra no, esta distancia se penaliza en una unidad (se suma 1). Si por el contrario, ninguna de las imágenes contiene la característica existente en el conjunto total, la distancia permanece invariable (sumamos 0).

Formalmente, la forma de representar matemáticamente el comportamiento anterior se describe de la siguiente manera:

$$D(A,B)=\sum_{i=0,\ldots,N} {\sim}\max(A_i, B_i) * -1^{(A_i + B_i)}$$

Dónde:

- A y B son dos imágenes descritas por características A_i y B_i,
- A_i y B_i representan la característica i-ésima de las imágenes A y B respectivamente. A_i y B_i {0, 1},
- '~' representa la operación de complementar el resultado, de este modo se obtiene el comportamiento deseado en la fórmula.

Siguiendo con el ejemplo planteado en las métricas Euclídea y de Manhattan, la distancia entre las imágenes 'Manzana' y 'Gato' quedaría como sigue:

Manzana: P = (0, 1, 1, 0)
Gato: Q = (1, 0, 0, 1)

$$D_P = (\sim\max(0, 1) * -1^{(0+1)}) + (\sim\max(1, 0) * -1^{(1+0)}) +$$
$$(\sim\max(1, 0) * -1^{(1+0)}) + (\sim\max(0, 1) * -1^{(0+1)}) =$$
$$(\sim(1 * -1^1)) + (\sim(1 * -1^1)) + (\sim(1 * -1^1)) + (\sim(1 * -1^1)) =$$
$$\sim(-1) + \sim(-1) + \sim(-1) + \sim(-1) = 1 + 1 + 1 + 1 = 4$$

Por lo que determinamos, que la distancia en este caso particular, es de 4 unidades.

3.2. Métrica compleja.

3.2.1. Fundamentos

Las métricas simples comentadas presentan resultados que a simple vista parecen lógicos, pero se tornan insuficientes a medida que aumenta el tamaño del banco de imágenes a clasificar y/o el número de características (etiquetas) de las imágenes.

Para solventar el problema presente en el uso de las métricas simples, se ha diseñado una métrica compleja, en la que se refleja el concepto de taxonomía gramatical nombrado en este documento.

La idea de esta métrica radica en intentar solventar posibles errores a la hora de asignar etiquetas a las imágenes. Estos errores se producen ya que los humanos tendemos a categorizar objetos atendiendo a características específicas, olvidándonos a veces de lo general.

Por ello, lo que pretendemos es tratar de construir una jerarquía implícita de conceptos, a partir del conjunto de etiquetas. Para ello se recurre a la minería de reglas de asociación. Estas técnicas permiten extraer las relaciones

existentes entre las etiquetas de las imágenes, en forma de regla. Cada una de estas reglas de asociación puede entenderse como una descripción parcial de una relación jerárquica entre etiquetas.

Por ejemplo, analizando de la regla 'Perro => Animal', vemos que disponemos de la información de que la etiqueta 'Animal' expresa un concepto más general que engloba a la etiqueta 'Perro'.

Mediante un ejemplo, podemos ver como en una imagen en la que se muestra un perro, una asociación normal de etiquetas sería 'Animal', 'Canino', 'Marrón' olvidándonos por ejemplo de asociarle las etiquetas como 'Ser vivo' o 'Mamífero'.

La idea que se persigue consiste en determinar una función de distancia que utilice esta jerarquía implícita, construida a partir de reglas de asociación, para mejorar la estimación del cálculo de la distancia entre dos imágenes.

3.2.2. Diseño

Una vez se extraen las reglas existentes entre las características de las imágenes del banco de imágenes inicial, se estructuran las relaciones entre etiquetas a través de la construcción de la taxonomía gramatical. Para ello, el proceso que se realiza es el siguiente:

Se asignará un peso a cada etiqueta presente en una imagen. Este peso permitirá ponderar cuán de fuerte o débil es la relación de pertenencia de esa etiqueta a la imagen.

Después, se toma cada una de las etiquetas inicialmente asociadas a la imagen y se ponderan con el peso máximo de 1, debido a que la etiqueta ya la contenía la imagen.

El proceso continúa buscando entre los antecedentes de las reglas conjuntos de etiquetas presentes en las imágenes.

Cuando se localiza en el antecedente de una regla una etiqueta, o un conjunto de etiquetas existente en una de las imágenes, se añade las etiquetas presentes en el consecuente de la regla hallada al conjunto de etiquetas de la imagen, asignando un valor ponderado en menor medida.

El proceso de disminuir el valor que define la nueva pertenencia de una etiqueta a una imagen, permite distinguir y ponderar entre etiquetas asignadas inicialmente a una imagen y etiquetas deducidas en el proceso.

De este modo, podemos esquematizar este proceso con el siguiente pseudocódigo:

Mientras (Hay variaciones es **TRUE**) **{**

Si (alguna etiqueta o conjunto de etiquetas aparece en el antecedente de una regla) **{**

Añadimos a la imagen las etiquetas presentes en el consecuente de la regla.

Ponderamos el valor de esta etiqueta.

Hay variaciones toma valor **TRUE**.

}

}

Cuando una nueva etiqueta se asigna, es añadida con un peso que se calcula de la siguiente manera: de todas las etiquetas de la imagen en cuestión, que se encuentren en el antecedente de una regla, se toma aquella etiqueta de menor peso y se multiplica dicho peso por la confianza de la regla en cuyo antecedente se localizó.

Este proceso permite ir asignando pesos, cada vez inferiores, a medida que la etiqueta es deducida de una regla. De este modo, la relación de asociación entre la etiqueta y la imagen se ve significativamente afectada.

Esta métrica, denominada métrica por taxonomía, permite crear un modelo que estructura o cataloga las etiquetas, pero no calcula un valor de distancia semántica entre dos imágenes. Para ello se combina este modelo, generado de forma anticipativa, con alguna de las métricas simples, como la distancia por penalización descrita en este documento. De este modo, la fórmula continúa siendo la misma, a excepción de que el valor que poseen las etiquetas de las imágenes está ponderado por un peso asociado a cada una de estas etiquetas.

3.2.3. Empleo de la Minería de Reglas de Asociación

En el desarrollo de la métrica por taxonomía gramatical, se ha hecho uso de un algoritmo de minería de reglas de asociación, cuya implementación se ha extraído del software WEKA, desarrollado por la Universidad de Waikato[26]. Este algoritmo ha facilitado las reglas de asociación que posibilitaron estructurar en una taxonomía implícita los conjuntos de etiquetas.

Este algoritmo utilizado, denominado algoritmo A priori, deduce implicaciones del tipo 'la mayoría de las veces que aparece la etiqueta X, aparece con ella la etiqueta Y', por lo que deducimos que están relacionadas.

Su funcionamiento sigue el siguiente fundamento que detallaremos a continuación. Analizando por ejemplo el siguiente conjunto:

Imagen	Et1: ser vivo	Et2: comida	Et3: fruta	EtN: animal
Manzana	0	1	1	0
Pera	0	1	1	0
Pizza	0	1	0	0
Pájaro	1	0	0	1
Gato	1	0	0	1

Podemos deducir reglas como:

1. Cada vez que encontramos 'animal', encontramos 'ser vivo', por lo que todo animal es un ser vivo.
2. Cada vez que encontramos la etiqueta 'fruta', encontramos 'comida', por lo que deducimos que toda fruta es comestible.

Extrayendo todas las reglas posibles, con un nivel de confianza considerable (frecuencia de aparición de las etiquetas, de forma conjunta, en imágenes diferentes), tendremos un modelo que sirve de respaldo para la tarea de completar las etiquetas que deberían estar asociadas a cada imagen.

3.3. Comparativa entre tipos de métricas

A la hora de seleccionar una función para el cálculo de la distancia entre dos imágenes, se comienza analizando el comportamiento de varias métricas simples debido al ahorro computacional que éstas aportan.

Existen varias métricas simples diseñadas que, en primera instancia, parecen de gran ayuda, por ejemplo el uso de la distancia de edición; el problema radica en que este tipo de métricas no atienden a una diferenciación basada en las relaciones existentes entre las características de los ítems a evaluar, sino indagan en una regla matemática para la devolución de un valor discreto.

Por ejemplo, usando la distancia de edición entre dos cadenas de caracteres (número de posiciones de letras de diferencia entre ambas cadenas), las palabras 'maceta' y 'maleta' tendrían menor distancia entre sí que 'maceta' y 'planta', que están semánticamente más relacionadas. Por ello es que surge la necesidad de elaborar métricas más complejas.

Cuando se analiza en una métrica compleja, como la métrica por taxonomía desarrollada, el esfuerzo computacional va in crescendo, pero aportando mejores resultados. El desafío se encuentra entonces en encontrar el equilibrio justo entre la ganancia que brinda la métrica y el coste computacional que la misma requiere.

Esta razón justifica que la taxonomía gramatical no se implemente de forma explícita con una estructura de datos arbórea, sino que en su lugar ésta sea deducida del proceso de asignación de nuevas características (catalogación). Este paso aumenta el nivel de abstracción requerido para el entendimiento de la métrica, aunque disminuye significativamente el costo computacional del algoritmo.

Capítulo 4.

Validación estadística de los algoritmos

A la hora de desarrollar una aproximación algorítmica, una de las fases más importantes la constituye la fase de validación.

Para la validación de un algoritmo, aparte de realizar las pruebas oportunas en la implementación del mismo, se requiere hacer validaciones en cuanto al significado matemático de los cálculos que se están desarrollando. Pues bien, la línea base de investigación parte de unos resultados que se deben ir mejorando a medida que avanzan las pruebas y se incluyen nuevos elementos de cómputo en el algoritmo.

Para la validación del algoritmo, se ha implementado una clase con métodos estadísticos que permiten valoras cuan de cerca o lejos están los resultado experimentales, de la hipótesis inicial elaborada en la línea base de investigación.

Como parte de esta justificación, comentar que esta clase provee una generación aleatoria de instancias en un espacio N-dimensional a partir de una distribución normal multivariante [8]. Por tanto, podemos conocer la forma en que se distribuyen en el espacio n-dimensional las características (discretizadas) de las imágenes a clasificar.

En un espacio bidimensional es simple representar la forma que toma esta distribución, como vemos en la figura 4.1, pero a medida que aumentan las dimensiones de las características (el número de características que definen a una imagen), se torna imposible imaginar e incluso representar dicha distribución.

La distribución normal multivariante nos permite simular puntos n-dimensionales de forma aleatoria y que podemos usar como instancias que

toman la forma del resultado deseado por el algoritmo, de este modo podemos contrastar cuán de bien está funcionando éste.

4.1. Experimento de validación

Emplearemos un conjunto de medias y de una matriz de covarianzas, debida y cuidadosamente elaborada, para simular unas instancias de prueba que permitirán comprobar el funcionamiento de nuestro algoritmo.

Por ejemplo, para el caso bidimensional, podemos simular puntos entorno a dos medias, tomando como input una matriz de covarianzas que indica la desviación que habrá entre los puntos generados y las medias establecidas.

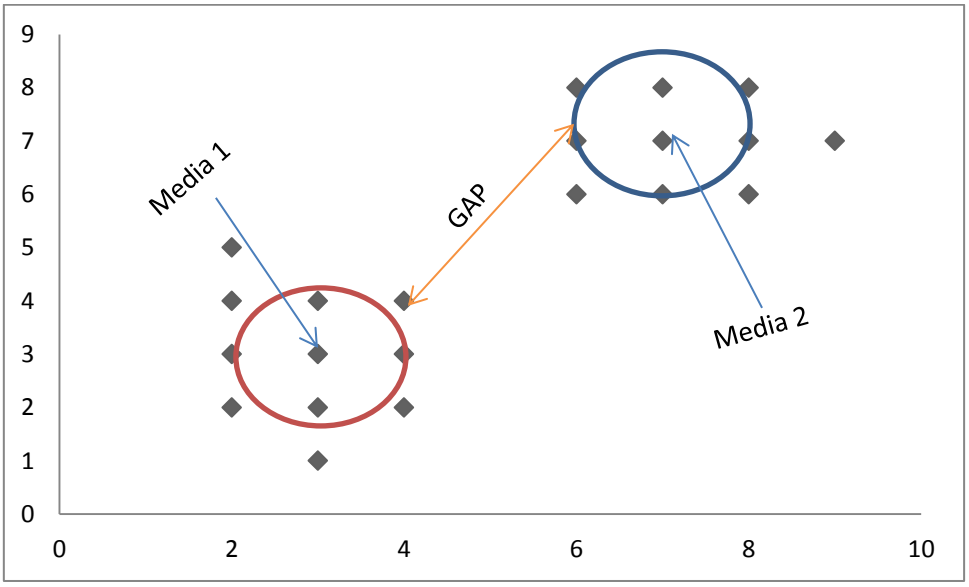


Figura 4.1. Diagrama Distribución Normal bidimensional.

Para obtener una distribución en torno a la que se representa en la figura 4.1, elaboraremos una matriz de covarianzas con los siguientes valores:

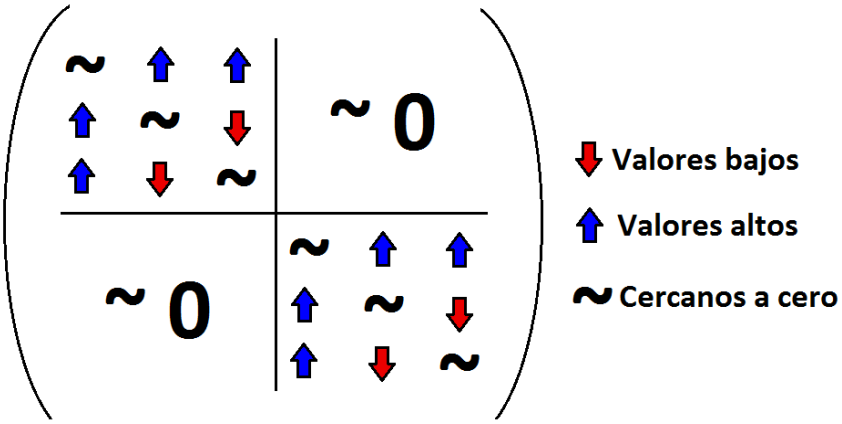


Figura 4.2. Ejemplo de matriz de covarianza

La matriz de covarianza definida, provee dos distribuciones en torno a dos medias que establezcamos, estas dos distribuciones sirven de base para establecer un conjunto de testeo de los algoritmos, esto es a lo que se denomina ground truth.

Ejecutando los algoritmos sobre los datos generados, podemos comprobar si el comportamiento que presentan es correcto o no, tomando varias instancias pertenecientes a los grupos elaborados y contrastando su clasificación final con el resultado que se visualiza a priori en el ground truth.

Aparte de validar a través de comparar resultados usando la generación normal multivariante, podemos utilizar métodos de contrastación que calculan un factor de error para el agrupamiento. Estos métodos van, desde el más simple y directo que es promediar la suma de las distancias entre todas las imágenes de un grupo y la imagen origen de dicho grupo; hasta métodos algo más elaborados como el cálculo del error cuadrático medio o ECM.

El error cuadrático medio es un estimador que mide el promedio de los errores al cuadrado, es decir, calcula la diferencia entre el estimador y lo que se estima. Esta métrica de error se expresa de la siguiente manera:

$$\mathrm{ECM} = \frac{1}{n}\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2.$$

Figura 4.3. Fórmula del ECM

Capítulo 5.
Interfaz Gráfica

Para ilustrar los resultados obtenidos con el desarrollo del sistema implementado, de una forma amigable, se ha desarrollado una interfaz simple de usuario.

Dicha interfaz, además de permitir la visualización de los conjuntos de imágenes recuperados por el algoritmo, permite la selección de los criterios que hacen que varía el número de imágenes a recuperar en cada conjunto y cuan de fuerte deben ser las relaciones de semejanza entre los mismos.

Por ello, contiene la opción de seleccionar el tipo de modelo algorítmico a ejecutar (modelo disco o satélite), los valores para las distancias 1 y 2 (propias del primer y segundo conjunto a recuperar respectivamente) y la distancia o gap de separación entre ambos conjuntos finales. Cabe añadir, que también permite seleccionar el tipo de métrica que se empleará en los algoritmos a ejecutar, para poder contrastar los resultados,

Estas funcionalidades descritas se han implementado a través de una web simple que contiene el acceso a un banco de imágenes y a un formulario que permite establecer los parámetros de entrada del algoritmo y visualizar la salida del mismo.

Veamos un ejemplo: utilizando los parámetros que vemos a continuación.

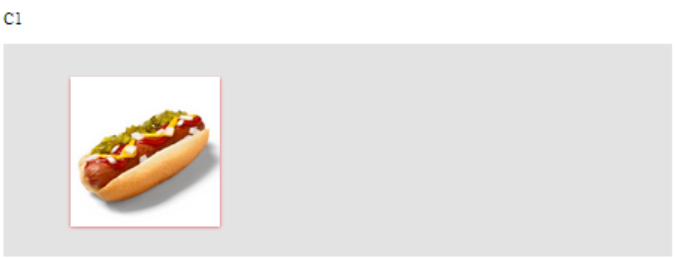
Figura 5.1: Formulario de ejecución del algoritmo.

Como vemos en la figura 5.1, se pretende recuperar una imagen similar a la seleccionada (16.jpg, que corresponde a una hamburguesa) y, tras 2 de separación GAP, recuperar otras dos imágenes que difieran un poco de esta imagen inicial.

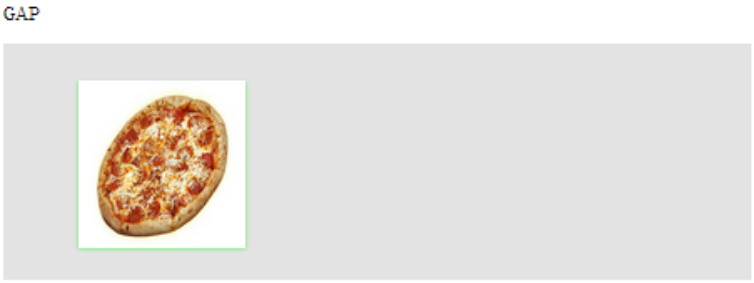
Se empleará el modelo algorítmico en disco y la métrica compleja, denominada métrica por taxonomía. El resultado obtenido es el siguiente:

a) Objeto referencia:

b) Grupo 1:



c) Grupo desechado por la distancia GAP:



d) Grupo 2:

Capítulo 6.
Conclusiones y líneas futuras

El campo objeto de estudio, a pesar de disponer de una gran variedad de trabajos asociados, se encuentra aún en desarrollo. Actualmente existen varios desafíos encaminados a encontrar técnicas algorítmicas que cataloguen automáticamente grandes conjuntos de imágenes y permitan la recuperación eficiente de las mismas. Las ideas expuestas en este trabajo constituyen un pequeño avance en el campo de investigación propuesto pues, la inclusión de la idea de Taxonomía Gramatical implícita a través del uso de la minería de reglas de asociación supone un componente innovador en este ámbito.

Se ha tratado de indagar en un sistema que permita solventar la recuperación de imágenes para la elaboración automática de ejercicios que mejoren la experiencia de aprendizaje propuesta en Ematic.

Para solventar dicha necesidad, se han propuesto dos escenarios en los que se requería de una recuperación de imágenes, el modelo algorítmico en disco y el modelo satélite.

Se elaboró una herramienta que, apoyada con el diseño de una interfaz gráfica simple y de los algoritmos implementados, permite la extracción eficiente de imágenes atendiendo a varios criterios, descritos en este documento.

Los algoritmos implementados se han apoyado en una variedad de métricas, que conforman diferentes variantes a los modelos propuestos. Estos algoritmos han sido validados utilizando un test estadístico sobre un conjunto de datos generados sintéticamente y sobre una base de imágenes diseñada ad hoc. Los resultados experimentales han sido satisfactorios y se han encapsulado finalmente en una librería, implementada en Java, que conforma una herramienta Open Source y que está disponible para toda la comunidad científica.

Capítulo 7.
Summary and Conclusions

The field under study, has a wide range of partners works but, is still in development. There are currently several challeges focused on finding algorithmic techniques that automatically retrieve images from a large set of images.

The ideas expressed in this paper are a small step in the proposed research field for the inclusion of the implicit grammatical taxonomy extracted by mining association rules, that is an innovate component on this area.

We have tried to delve into a system that solves the automatic image retrieval for automatic generation of exercises that improve the learning experience that Ematic propose.

In order to solve the Ematic needs, two situations which require an image recovery, the algorithmic model on disc, and the satellite model.

A tool with a simple graphical user interface was developed. This tool has been implemented with both algorithmic models in order to allow the efficient image retrieval based on several criteria, described on this paper.

Implemented algorithms have relied on a variety of metrics, which make different variants of proposed models. These algorithms have been validated using a statistical test on a synthetically generated data set an on a simple ban of images designed ad hoc. The experimental results have been successfully and have finally encapsulated in a library, implemented in Java that makes up an Open source that is available to the entire scientific community.

Apéndice A.

Minería de Reglas de asociación. Algoritmo A priori

Indagando un poco más en la extracción de reglas que posibilitan la clasificación por taxonomía, cabe comentar la idea que subyace tras este proceso.

Dentro del ámbito de la minería de datos y del aprendizaje automático, las reglas de asociación se emplean para detectar hechos que suceden en común dentro de un conjunto de datos. El ejemplo más común en el que se emplean ese tipo de reglas es en el análisis de la cesta de la compra.

Imaginemos un banco de datos extraídos de transacciones comerciales realizadas en un supermercado. Cada fila de este conjunto de datos contiene la información relativa a los artículos en cada transacción. Utilizando un algoritmo de minería de reglas de asociación podríamos detectar productos que frecuentemente se compran en conjunto, es decir, podríamos responder a preguntas como ¿qué tipo de colocación de los productos dentro del establecimiento comercial se debe establecer si deseamos que el cliente visite el mayor número de productos relacionados posibles?, ¿Qué tipo de colocación emplear si se desea una mejor satisfacción de compra en el cliente?

Para ilustrar mejor esta idea, véase el siguiente ejemplo. Sea el siguiente conjunto de transacciones:

ID	Leche	Pan	Mantequilla	Cerveza
1	1	1	0	0
2	0	1	1	0
3	0	0	0	1
4	1	1	1	0
5	0	1	0	0

Tabla A.1. Ejemplo de transacciones comerciales.

En la tabla se indica con un '1' que el producto en cuestión fue comprado en la transacción y con un '0' lo contrario.

El conjunto de ítems es I = {'Leche', 'Pan', 'Mantequilla', 'Cerveza'}.

Un ejemplo de regla que se extrae del conjunto de datos iniciales podría ser {'Leche', 'Pan'} => {'Mantequilla'}, esto quiere decir que es muy frecuente que cuando se compra leche y pan, el cliente se lleve también mantequilla. Este tipo de conclusiones aporta información de gran valor añadido al establecimiento comercial, pues situando la leche y el pan algo separados, dejando por medio la mantequilla, obligaría de algún modo al cliente a plantearse si necesita también comprar mantequilla. O por el contrario, situando los tres productos en un mismo emplazamiento, conseguiríamos mejorar la experiencia de compra, pues el cliente conseguiría los productos que busca más rápidamente.

Para hallar reglas más precisas, se requiere de dos medidas de "significancia" e "interés", las más conocidas son los umbrales mínimos de **soporte** y **confianza**.

El soporte de un conjunto de ítems representa la frecuencia en la que dicho conjunto aparece en transacciones del conjunto de datos.

$$\text{sop}(X) = \frac{|X|}{|D|}$$

Figura A.1. Soporte de un conjunto.

En el ejemplo anterior, el conjunto {'Leche', 'Pan'} posee el soporte:

Sop (X) = 2 / 5 = 0.4, es decir el soporte es de un 40%, 2 de cada 5 transacciones contienen el conjunto.

Por otro lado, la confianza de una regla se define como sigue:

$$\text{conf}(X \Rightarrow Y) = \frac{\text{sop}(X \cup Y)}{\text{sop}(X)} = \frac{|X \cup Y|}{|X|}$$

Figura A.2. Confianza de una regla.

Para la regla {'Leche', 'Pan'} => {'Mantequilla'}, la confianza sería:

$$\text{conf}(\{\text{Leche}, \text{Pan}\} \Rightarrow \{\text{Mantequilla}\}) = \frac{\text{sop}(\{\text{Leche}, \text{Pan}\} \cup \{\text{Mantequilla}\})}{\text{sop}(\{\text{Leche}, \text{Pan}\})} = \frac{0.2}{0.4} = 0.5$$

Esto significa que la mitad de las reglas extraídas de la base de datos que contienen los ítems 'Leche' y 'Pan' en el antecedente, también poseen 'Mantequilla' en el consecuente. Por lo que se deduce que la regla es cierta en un 50% de los casos. Este indicador podría interpretarse como la probabilidad de encontrar el consecuente, condicionado a que se encuentre también el antecedente.

Nota: este ejemplo ha sido extraído de la definición de reglas de asociación expuesta en el portal web 'Wikipedia' [12].

La tarea de extraer reglas a partir del conjunto inicial de datos es simple, pero un proceso complejo computacionalmente, tanto que se torna imposible extraer todas las reglas para un conjunto de datos con muchísimos productos.

Imaginemos que tenemos un conjunto de datos con solo 10 productos, haciendo combinaciones de los mismos podemos extraer hasta un total de 1024 reglas posibles, si tuviéramos 20 productos, habría 1.048.576 reglas posibles, y va in crescendo de forma exponencial.

El algoritmo A Priori utilizado emplea estas ideas descritas para extraer estas reglas a partir del conjunto inicial de datos, pero emplea una regla "anti monótona" que permite realizar una poda. Esta poda emplea la idea de que si un conjunto es infrecuente, todos los conjuntos en los que se encuentre éste último también serán infrecuentes, por lo que serán ignorados.

Por ejemplo, si determinamos que el conjunto {A, B} es infrecuente, también lo serán {A, B, C}, {A, B, E} y {A, D, B, E} por contener los ítems A y B conjuntamente en ellos.

Para ilustrar mejor esta idea de poda, véase las siguientes imágenes:

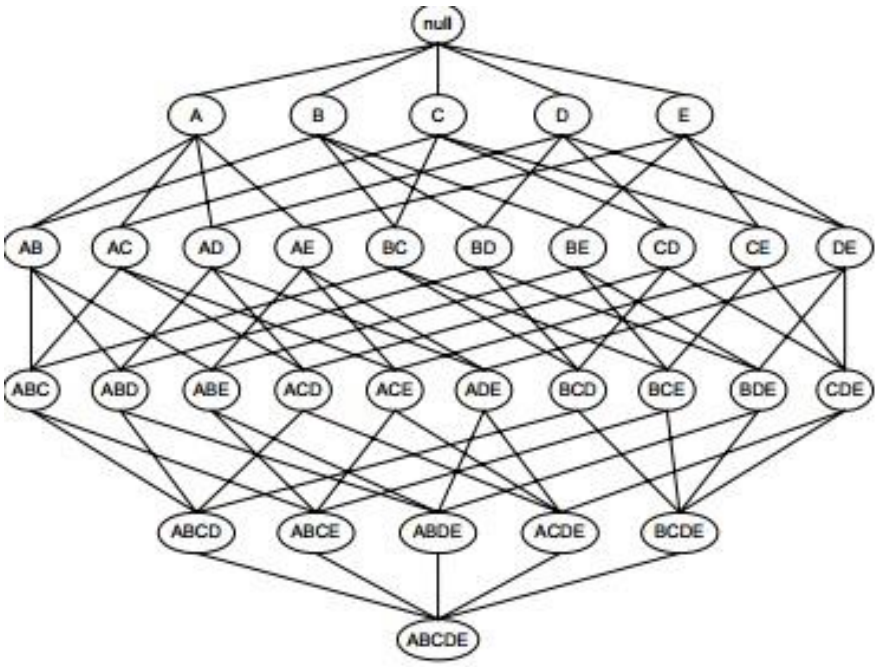


Figura A.3: Conjuntos derivados a partir de los ítems A, B, C, D y E

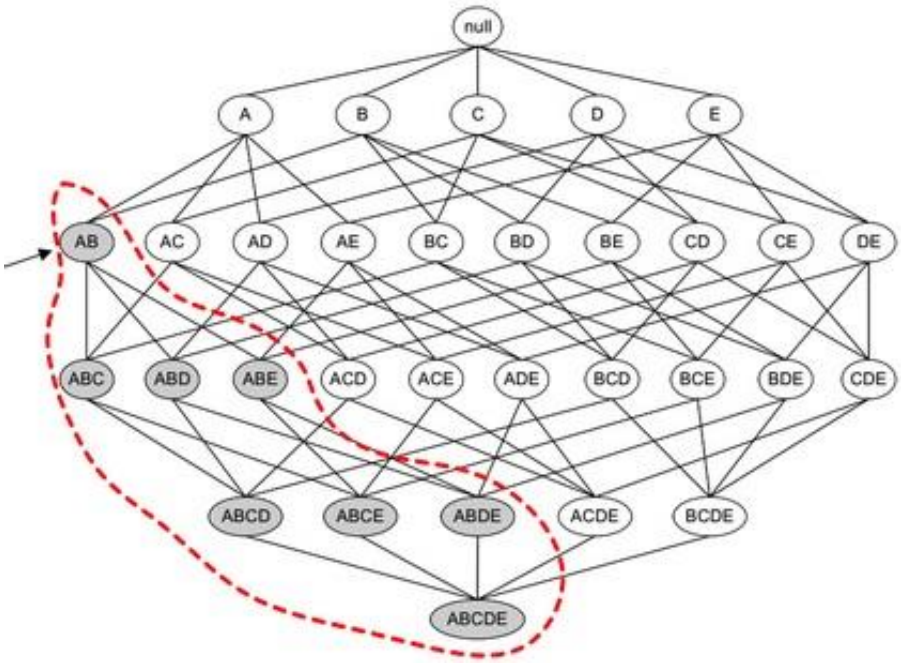


Figura A.4: Poda de conjuntos infrecuentes.

Nota: ejemplo extraído del blog de IA, por Fermín Pito [13].

El proceso que describe el algoritmo A priori para la extracción de las reglas de asociación es el siguiente:

1. Generación de combinaciones frecuentes: en este paso se localizan conjuntos que se den frecuentemente en la base de datos. Para determinar la frecuencia de las reglas, se emplea el soporte. Este umbral se utiliza para descartar conjuntos infrecuentes.
2. Generación de reglas: a partir de los conjuntos frecuentes, se generan las reglas a partir del nivel de confianza que las mismas posean.
3. El proceso continúa hasta conseguir extraer todas las reglas que cumplan con las restricciones planteadas.

- Pseudocódigo:

$\mathbf{C_k}$: Conjunto de ítems candidato de tamaño k

$\mathbf{L_k}$: Conjunto de ítems frecuentes de tamaño k

$\mathbf{L_1}$= {elementos frecuentes};

Para (**k**= 1; $\mathbf{L_k}$!= **∅**; **k**++) hacer **{**

$\mathbf{C_{k+1}}$= candidates generated desde $\mathbf{L_k}$;

Para cada transacción **t** en **database** hacer **{**

Incrementar la cuenta de candidatos en $\mathbf{C_{k+1}}$ que están en **t**

$\mathbf{L_{k+1}}$= candidatos en $\mathbf{C_{k+1}}$ con **soporte** > **min_soporte**

}

}

return $\cup_k$ $\mathbf{L_k}$;

Apéndice B. Acerca de la librería implementada.

B.1. Características técnicas

La librería desarrollada requiere de un entorno en el que se encuentre instalada la máquina virtual de Java, disponible en la mayor parte de los sistemas tecnológicos actuales.

Se trata de un fichero (librería) con formato de compresión JAR, formato de compresión utilizado para ejecutables JAVA. Dicho archivo contiene encapsulado el comportamiento del algoritmo y funciona de la siguiente manera:

El algoritmo requiere de varios inputs para su funcionamiento: un fichero de datos y una serie de parámetros de configuración del algoritmo.

Estos parámetros determinan:

si la generación de instancias es aleatoria, o se obtiene desde el tratamiento de un fichero (CSV ó XML).

El tipo de algoritmo a ejecutar (Línea Base o Alternativo).

El tipo de métrica a emplear (Simple: Euclídea, Manhattan,... Compleja: por taxonomía).

Los valores de distancia máxima para los grupos y la separación existente entre ambos (d1, d2 y GAP).

La salida del algoritmo sigue un formato concreto determinado por el nombre del grupo que generó y los valores de las imágenes que pertenecen a dicho grupo.

B.2. Diseño modular

La implementación de la librería atiende a un diseño particular analizado y diseñado para resolver el problema utilizando las estructuras de datos y relaciones pertinentes entre las mismas, de la forma más eficiente posible.

La ejecución de un algoritmo u otro, incluso la instanciación de un tipo de métrica atienden a un modelo basado en el patrón de diseño Strategy o patrón estrategia [16]:

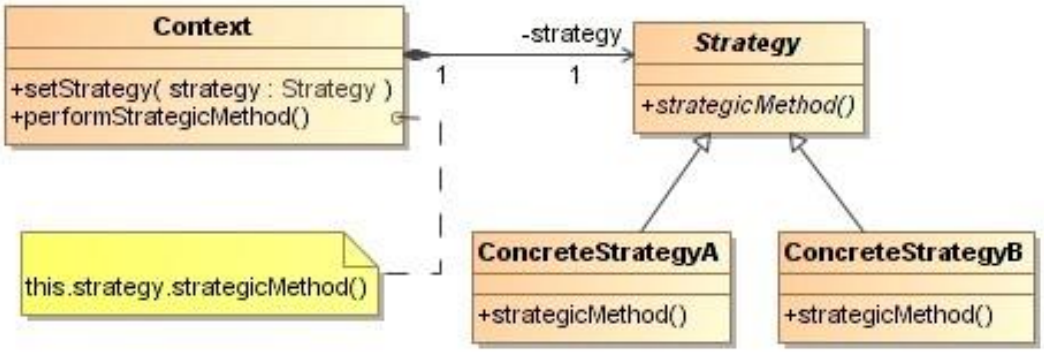


Figura B.1. Patrón de diseño estrategia.

Como vemos, en el contexto del problema tenemos los parámetros que determinan que estrategia concreta ejecutar. En nuestro caso, los algoritmos base y alternativo heredan de la clase base Algoritmo. El contexto lo posee el objeto 'Clasificador', quien según conveniencia del usuario, instancia una estrategia algorítmica u otra; esta situación también ocurre con la métrica, pues se instancia una métrica u otra a conveniencia del usuario.

Todo el sistema ha sido diseñado de forma modular, de modo que si deseáramos incluir, suprimir o modificar ciertos métodos del algoritmo o incluso alguna métrica, podría realizarse sin complicación alguna.

B.3. Compatibilidades e integración

Una de las principales ventajas de la librería es, que al estar implementada en Java, el código es totalmente portable a casi cualquier otro tipo de sistema. Esta característica hace que no estemos limitados a proyectos descritos en el mismo lenguaje.

Un ejemplo de esta portabilidad lo encontramos en el proyecto en el que se pretende integrar el algoritmo, fruto de esta investigación. Dicho proyecto está desarrollado en Django (bajo el lenguaje Python), y la integración es totalmente viable.

Además del ejemplo anterior, el desarrollo de la interfaz simple se ha realizado bajo el uso de tecnologías web, con integración de estas vía el lenguaje de servidor PHP.

Bibliografía

[1] S. Taásan. Image Retrieval: Ideas, Influences, and Trends of the New Age. PhD thesis, The Pennsylvania State University, Pennsylvania, 2008.

[2] Hui Yang et Jamie Callan.Learning the Distance Metric in a Personal Ontology, 2008.

[3] Nakul Verma, Dhruv Mahajan, Sundararajan Sellmanickam et Vinod Nair. Learning Hierarchical Similarity Metrics, 2010.

[4] Pengcheng Wu, Steven C.H. Hoi, Peilin Zhao, Ying He. Mining Social Images with Distance Metric Learning for Automated Image Tagging.

[5] Feng Tian and Xukun Shen. Annoting Web Images by Combining Label Set Relevance with Correlation.

[6] Distancia Euclídea. https://es.wikipedia.org/wiki/Distancia_euclidiana.

[7] Distancia Manhattan. https://es.wikipedia.org/wiki/Geometría_del_taxista.

[8] Librería Distribución Normal Multivariable. http://commons.apache.org/proper/commons-math/apidocs/org/apache/commons/math3/distribution/MultivariateNormalDistribution.html.

[9] Error cuadrático Medio. https://es.wikipedia.org/wiki/Error_cuadrático_medio.

[10] Librería WEKA. http://www.cs.waikato.ac.nz/ml/weka/.

[11] Algoritmo A priori, librería WEKA. http://weka.sourceforge.net/doc.dev/weka/associations/Apriori.html.

[12] Algoritmo A priori. Soporte y confianza. http://es.slideshare.net/jorgeklz1/apriori-algoritmo-reglas-de-asociacion-datamining-mineria-datos-soporte-confianza.

[13] Algoritmo A priori. Procedimiento. http://ferminpitol.blogspot.com.es/2014/05/reglas-de-asociacion-algoritmo-apriori.html.

[14] Algoritmo A priori. Procedimiento. http://bsolano.com/ecci/claroline/backends/download.php/UHJlc2VudGFjaW9uZXMvNy5fVGFyZWFzX2RlX2xhX21pbmVy7WFfZGVfZGF0b3MsX3JlZ2xhc19kZV9hc29jaWFjafNuLnBkZg%3D%3D?cidReset=true&cidReq=CI2352.

[15] Reglas de asociación. https://es.wikipedia.org/wiki/Reglas_de_asociación.

[16] Patrón Strategy. https://es.wikipedia.org/wiki/Strategy_(patrón_de_diseño).

[17] Image retieval. https://en.wikipedia.org/wiki/Image_retrieval.

[18] Ritendra datta, Dhiraj Joshi, Jia Li, and James Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. http://infolab.stanford.edu/~wangz/project/imsearch/review/JOUR/data.pdf

[19] EMATIC. http://encelado.isaatc.ull.es/.

[20] EMATIC. Enseñanza de las matemáticas a través de las TICS. http://riull.ull.es/xmlui/handle/915/281.