

Aproximando a los Sistemas Recomendadores desde los Algoritmos Genéticos

Oswaldo Vélez Langs* Carlos Santos†

Resumen

El presente trabajo abarca un enfoque alternativo, desde los algoritmos evolutivos, a la manera tradicional en que se abordan los sistemas recomendadores (SR de aquí en adelante). Se examinan las posibilidades de los algoritmos genéticos para brindar características adaptativas a estos sistemas. Nuestro objetivo, además de proporcionar una panorámica informativa general sobre las posibilidades y potencialidades de los SR, es proveer mecanismos para que los SR sean capaces de aprender características personales desde los usuarios, con miras a mejorar la efectividad a la hora de encontrar recomendaciones y sugerencias apropiadas para un individuo en particular.

Palabras claves: *Filtrado Colaborativo de la Información, Aprendizaje Automático, Algoritmos Evolutivos, Interfaces de Usuario Adaptivas*

Abstract

This work presents an alternative approach (Evolutionary Algorithms approach) to traditional treatment of Recommender Systems (RSs). The work examines genetic algorithms possibilities to offer adaptive characteristics to this systems trough learning. The main goal, in addition to give a general view about RSs capabilities and possibilities, it is to provide an example mechanism for to extend RSs learning capabilities (from users's personal chracteristics), with the purpose to improve the effectiveness in the moment of to find recommendations and appropriate suggestions for particular individuals.

Keywords: *Collaborative Information Filtering, Machine Learning, Evolutionary Algorithms, Adaptive User Interfaces.*

1 Introducción

El crecimiento de Internet ha sido increíblemente acelerado, hay cantidades ingentes de datos casi sobre cualquier tópico, lo que hace de Internet la fuente más completa de información para casi todo. Sin embargo, hay más cantidad de artículos, libros, películas, programas de televisión, entre otras cosas, de los que las personas podemos manejar efectivamente, con lo cual hay una sobrecarga de información hacia los usuarios finales.

* Universidad del Sinú, Carrera 1w Calle 38, Montería, Colombia, oswaldo.velez.langs@urjc.es.

† Universidad Rey Juan Carlos, Departamento de Arquitectura, Tecnología de Computadores, Ciencias de la Computación e Int. Artificial, Madrid, España.

[Shell]

8 Command=2

Oswaldo Vélez Langs / Carlos Santos

IconFile=explorer.exe,3

El `Taskbar` se necesita para hacer una búsqueda sobre el conjunto total de tópicos es demando `Command=ToggleDesktop` búsquedas tradicionales pueden ser infructíferos la mayoría de las veces o con resultados poco exitosos.

Los SR tratan de ser un paso adelante en el contexto de la recuperación de información tradicional, que se da por palabra claves del tópico que se desea encontrar a través de los bien conocidos motores de búsqueda (Google, Lycos y Altavista entre otros). Como su nombre nos lo indica los SR, se encargan de recomendar o sugerir a los usuarios ítems o productos concretos basándose en sus preferencias, son usados por sitios web de comercio electrónico como herramientas de mercadeo para incrementar ventas al presentar al usuario aquellos productos que desea (o desearía) comprar, es así como se construye una base de entendimiento de necesidades concretas respecto a lo que gusta o no a los clientes, lo cual puede verse como reflejado en un incremento de la lealtad de los clientes. La gran mayoría de los SR trabajan con enfoques de filtrado colaborativo, en términos simples, por *filtrado colaborativo* se entiende al método de hacer predicciones automáticas (filtrado) acerca de los intereses de un usuario, colectando información sobre los gustos de varios usuarios (colaboración). Por ejemplo, un filtrado colaborativo o sistema de recomendaciones para gustos musicales puede hacer predicciones acerca de qué música debería desear un usuario, dada una lista parcial de gustos de otros usuarios.

El enfoque tratado en este artículo utiliza a los algoritmos genéticos como afinadores en el proceso de concordancia de perfiles dentro de un SR, adecuando finalmente estos a las preferencias individuales, lo que redundará en una mayor precisión del SR a la hora de predecir que gusta o no a un usuario específico. Este trabajo se enfoca primordialmente en la capacidad de predicción, obtenida mediante el aprendizaje de los pesos, que se logra con el Algoritmo Genético (AG)

El artículo se organiza de la siguiente forma: la sección 2 muestra una panorámica general del área del filtrado colaborativo de información y los sistemas recomendadores haciendo al final especial mención de las aplicaciones de los Algoritmos Evolutivos, la sección 3 presenta el experimento realizado y finalmente la sección 4 presenta la discusión final y algunas observaciones acerca del trabajo futuro.

2 Panorámica Sobre Filtrado Colaborativo y Sistemas Recomendadores

Para elegir una de las alternativas que conforman una decisión se necesita tener conocimiento acerca de cada una de ellas, normalmente la obtención del conocimiento requiere una búsqueda y análisis de la información necesaria, la recomendación se presenta entonces como una alternativa que permite obtener conocimiento sin necesidad de buscar y analizar información sobre todas las alternativas posibles.

Los SR proporcionan sugerencias personalizadas acerca de ítems o productos que un usuario

puede llegar a encontrar interesantes, el sistema debe estar en capacidad de hacer predicciones sobre la preferencia (o no) de un usuario sobre cierto ítem. El proceso básico es hacer un proceso de concordancia entre la información que se tiene acerca del perfil del usuario actual y los perfiles de los otros usuarios que se encuentran ya almacenados y de cuyas preferencias se tiene conocimiento, a menudo esto se conoce como “*filtrado colaborativo de vecindad más cercana*” (ver Fig. 1), los SR son un intento por automatizar las conocidas recomendaciones tipo “boca a boca” en que las personas se recomiendan productos unos a otros. En estos sistemas los usuarios son individuos y a su vez son miembros de un grupo (que es conocido como vecindad)

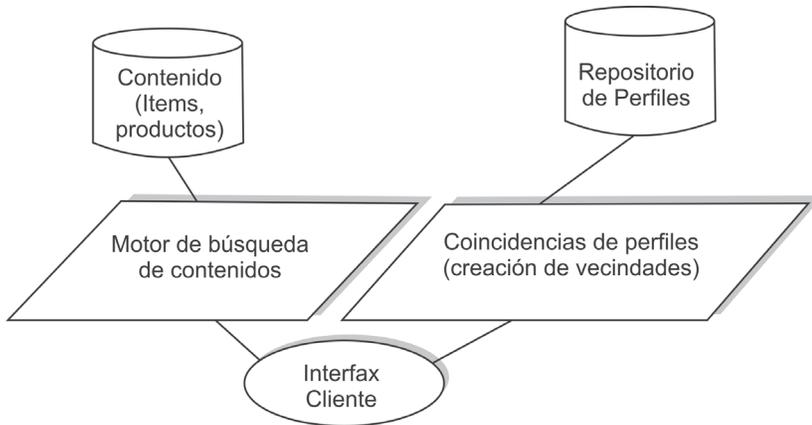


Fig. 1. Esquema de funcionamiento básico de un Sistema Recomendador

2.1 Clasificaciones

Hay diversas maneras de crear una tipología para los SR, principalmente y entre otras formas tenemos:

- De acuerdo al método usado ([3])
 - *Basados en modelos*: Estos algoritmos antes de desarrollar una recomendación, primero desarrollan un modelo de las calificaciones de los usuarios, el proceso de producción de los modelos que usan los algoritmos en esta categoría se desarrolla con distintos enfoques de aprendizaje automático como Redes Bayesianas, encasillamiento o enfoques basados en reglas. *Basados en memoria*: utilizan la BD completa de usuarios-ítems para generar las predicciones, el SR usa una técnica estadística para hallar un conjunto de usuarios que tengan una historia de concordancia con el usuario objetivo (vecinos), una vez es formada una vecindad se utiliza un algoritmo que combine preferencias de los vecinos para producir una predicción y un *ranking* de los “N principales” para el usuario objeto.

- De acuerdo a la forma de recopilar el conocimiento del usuario
 - *Explícito*: El usuario le “dice” al sistema lo que piensa respecto a la información de los ítems
 - *Implícito*: Se registran las acciones de los usuarios y se concluyen las calificaciones sobre estas acciones
- De acuerdo a las fuentes de conocimiento del usuario
 - *Usuario único*
 - *Muchos usuarios (FC)*
- De acuerdo a los tipos de acción
 - *Búsqueda*
 - *Filtrado de Información*
 - *Recomendación de sitios*
 - *Tutoría*
 - *Recomendación de personas y grupos*

2.2 Sistemas de Filtrado

Hay distintas formas de realizar el filtrado de la información, por ejemplo dependiendo en sí del algoritmo de aprendizaje usado, el cual se encarga de hacer un mapeo desde las características de los ítems a un número que indica la utilidad de dicho ítem para un usuario, basándose en la información previa que se tiene de ese usuario respecto a otros ítems. La otra alternativa es usar las calificaciones previas que han hecho usuarios similares (vecinos) sobre ciertos ítems, que el usuario activo aún no ha calificado, para predecir cuál sería la preferencia del usuario sobre dichos ítems.

Según dice D. Nichols en [14], en [10] son descritas tres formas de realizar el filtrado de información: social (colaborativo o FC de aquí en adelante), de contenido (cognitivo) y económico, el filtrado social actual se basa en grandes cantidades de calificaciones explícitas por parte de los usuarios, el filtrado de contenido es más cercano al enfoque tradicional de recuperación de información teniendo en cuenta palabras claves, el filtrado económico se basa en ciertos factores de costos asociados como pueden ser el costo/beneficio del uso de la información, el ancho de banda disponible y el tamaño mismo de los objetos, el uso de este filtrado es muy reducido en los sistemas actuales.

2.2.1 Filtrado basado en el usuario

Esta tecnología es la más frecuentemente usada dentro de los SR comerciales. Hay un problema de escalabilidad subyacente en este tipo de método ya que el costo computacional crece con el número de clientes, como alternativa se tiene el enfoque de filtrado basado en los ítems (descrito más adelante).

- *Filtrado Colaborativo*: Como ya se ha mencionado, en este enfoque se compara lo que gusta o no a otra gente para predecir las preferencias de un usuario dado.
- *Filtrado Social*: A menudo se usa como sinónimo del anterior ya que en un nivel básico ofrece lo mismo que el FC, aunque frecuentemente se compara también con el filtrado basado en ítems. Aquí las recomendaciones se basan más en la calidad en sí de los productos que propiedades concretas de estos mismos.
- *Filtrado Adaptativo*: Es una combinación de los filtrados basados en usuario con los filtrados basados en ítems. La idea es que el sistema va “aprendiendo” a medida que pide al usuario ir valorando las cosas y controlando los click del mismo, observando así lo que hace.
- *Otros*: Podemos hacer mención adicional del filtrado sico-gráfico y del diagnóstico de personalidad. El primero es similar al FC, excepto que la predicción se basa en un perfil sico-gráfico que se deriva de un cuestionario, en el diagnóstico de personalidad una vez se tienen preferencias de un usuario para cierto ítem, se calcula la probabilidad que dicho usuario tenga el mismo tipo de personalidad que otros.

2.2.2 Filtrado basado en los ítems

En esta otra perspectiva se usan técnicas que analizan una matriz de relaciones usuario-ítem, con lo que se identifican relaciones entre los distintos ítems y se usan estas para computar una lista de recomendaciones, inicialmente se buscan similitudes (o diferencias) entre los ítems, dicha evaluación se conjunta con los usuarios o grupos de estos

- *Filtrado basado en características*: Se basa en la idea de que es posible capturar que características que le gustan (o no) a un usuario con respecto a cierto ítem y entonces es posible proceder a realimentar al usuario con información de diversos ítems.
- *Filtrado basado en el contenido*: En esta técnica, también denominada filtrado cognitivo, se recomiendan ítems basados en la correlación entre el contenido de los ítems y las preferencias de los usuarios, p ej. El sistema puede correlacionar la presencia de ciertas palabras claves en un artículo, con las preferencias del usuario, una versión más simple de este método es el Filtrado basado en palabras claves.
- *Filtrado basado en perfiles*: Es el enfoque más directo ya que el usuario describe su interés escogiendo en una lista de palabras claves y el sistema rechaza cualquier ítem que no coincide con estas.

2.3 Métodos en el Filtrado Colaborativo

En esta sección se detallan brevemente las distintas formas algorítmicas de implementar un FC junto a las características básicas de cada una.

2.3.1 Encasillamiento

Es una técnica diseñada para realizar una clasificación asignando patrones a grupos de tal forma que cada grupo sea homogéneo y distinto de los demás. Los algoritmos de encasillamiento usualmente producen menos recomendaciones personales que otros métodos y en algunos casos las casillas formadas tienen peor precisión que los algoritmos de vecindad cercana, sin embargo una vez la casilla está bien conformada el desempeño puede ser bueno si el número de grupos a analizar es pequeño. En Ungar y Foster [20] se presenta un trabajo relacionado.

2.3.2 Horting

Es una técnica basada en grafos en la cual los nodos son los usuarios y las aristas entre nodos son indicadores de los grados de similitud entre dos usuarios. Las predicciones se producen al recorrer el grafo entre nodos cercanos y combinando las opiniones entre usuario cercanos ([1]). Esta técnica difiere de los algoritmos de vecindad más cercana, en la forma como el grafo puede ser recorrido por otros usuarios que no han valorado los ítems, luego esta técnica explora las relaciones transitivas que los algoritmos de vecindad más cercana no tienen en cuenta.

2.3.3 Redes Bayesianas de Creencia

Las Redes Bayesianas de Creencias (RBC) también se conocen como Redes de Creencias, Redes Probabilísticas Causales, Redes probabilísticas Gráficas entre otras. Una RBC es una red gráfica que representa relaciones probabilísticas entre variables. Las RBCs permiten razonar bajo incertidumbre y combinar las ventajas de una representación visual intuitiva con una base matemática en la probabilidad bayesiana:

$$P(A/B) = P(A,B)/P(B) \quad (1)$$

El trabajo de Heckerman et al. [7] nos muestra una aproximación de este tipo de redes al filtrado colaborativo.

2.3.4 Redes Neuronales

Las Redes Neuronales (RN) proporcionan una forma muy conveniente de representación del conocimiento en tareas de Recuperación de Información (RI), donde los nodos representan objetos del proceso de RI como palabras claves, autores y citas y los enlaces representan la asociación ponderada de estos (relevancia). Las propiedades de aprendizaje de las RN tipo Retropropagación del error y las propiedades de búsqueda en paralelo de las Redes tipo Hopfield proporcionan mecanismos efectivos para identificar información relevante de ítems en bases de datos. Variantes del aprendizaje tipo Retropropagación del error en RI pueden hallarse en [2] y [9].

Las Redes Neuronales (RN) aplicadas al filtrado colaborativo son de reciente uso, en [13] hallamos una aplicación en la cual un perceptrón multicapa es usado para predecir URLs que se dan como recomendación a los usuarios de acuerdo a su nivel de interés. [4] también presenta el uso de perceptrones multicapa, en donde se combinan el filtrado basado en contenido y el colaborativo, el sistema es usado en el ámbito de recomendación de películas y utiliza la base de datos de *MovieLens* (usada también para nuestro experimento).

2.3.5 Máquinas de Soporte Vectorial

En este enfoque cada ítem es mapeado en un espacio vectorial, pero la meta aquí es hallar la mejor superficie que separe los ejemplos positivos de los negativos, “mejor” significa que la superficie separe los positivos de los negativos por el margen más amplio posible.

En teoría y por definición es posible aplicar el anterior enfoque sólo cuando los ítems son linealmente separables, pero también es posible cuando la separación lineal no se presenta. Una de las soluciones consiste en aplicar una función que duplica la dimensión del espacio vectorial. Es posible demostrar que algunas funciones garantizan una separación lineal. Algunas de estas funciones son denominadas “kernels”, también tiene como característica un bajo costo a nivel computacional. Para mayores detalles y referencias relacionadas puede verse el trabajo de Sebastiani en [17].

2.3.6 Similitud Basada en el Coseno

Esta similitud da una buena medida del “parecido” de dos vectores en un espacio multidimensional, el espacio puede describir características de usuarios o de ítems, tales como palabras claves [16]. La similitud entre ítems es medida computando el coseno entre el Angulo entre estos dos (ver Fig. 2).

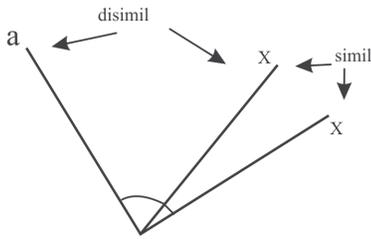
$$sim(i, j) = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{\|\vec{i}\|_2 * \|\vec{j}\|_2}$$


Fig. 2. Similitud basada en el Coseno

Para otras medidas de similitud puede verse [15]

2.3.7 Correlación de Pearson

En Herlocker et al [8] se mostró que una de las mejores medidas de similitud es el denominado Coeficiente de Correlación de Pearson (CCP), esta es una métrica típica de similitud entre funciones de preferencias de usuarios o (menos frecuentemente) distancias de vectores o productos puntos, hay diversas fórmulas, en la Ecuación 2 se presenta una de uso común. La formula da una aproximación de que tan bien los vectores comparados (perfiles, ítems) coinciden en la escala desde cero (no similares) a uno (total coincidencia) o -1 (total diferencia)

$$r = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{(\sum X^2 - \frac{(\sum X)^2}{N}) (\sum Y^2 - \frac{(\sum Y)^2}{N})}} \quad (2)$$

2.4 Trabajo Relacionado de aplicaciones de Algoritmos Evolutivos

A continuación se describirán algunas aplicaciones relacionadas de los algoritmos evolutivos, aunque en la literatura es mayor la presencia de referencias relacionadas con la recuperación de información, aquí se presentan aquellas (cuya presencia en la literatura es menor) que se enfocan a sistemas de filtrado de información, la descripción se encuentra organizada de acuerdo al tipo de algoritmo que es usado.

Un enfoque de filtrado semi-automático que combina AGs y el denominado “aprendizaje desde la re-alimentación” se presenta en el trabajo de Sheth y Maes [18], dicho enfoque se adapta de forma dinámica a los cambios de interés del usuario. Se muestra como una pequeña población de agentes de filtrado va evolucionando para hacer una selección personalizada de noticias de USENET. Los resultados ilustran que el componente evolutivo es responsable de la mejora en el ratio de recuerdo mientras el otro componente mejora el ratio de precisión.

En Min Tjoa et al [11], se presenta un sistema denominado CIFs (Cognitive Information Filtering System) el cual aplica un modelo evolutivo (sistema clasificador de aprendizaje) a la par que aprende de la realimentación del usuario. CIFS filtra correos electrónicos en el basado en el ranqueo que hace el usuario a los correos y también con base a la monitorización del comportamiento del mismo. CIFS utiliza, a diferencia del sistema citado en el párrafo anterior, un esquema de recuperación booleana además del vectorial y adicionalmente no solo se adapta a los perfiles de usuario sino que hace un proceso de generación de los mismos.

Amalthea (Moukas y Maes [12]) es un ecosistema multi-agente para filtrado personalizado, el cual descubre y monitorea sitios de información. El dominio de aplicación de Amalthea es la Web y su propósito principal es asistir a sus usuarios para encontrar información interesante. Se presentan dos categorías distintas de agentes: agentes de filtrado, que modelan y monitorean los intereses del usuario y agentes descubridores que modelan los recursos de información. Se desarrolla un ambiente similar a un Mercado en donde los agentes evolucionan, compiten, y colaboran, aquellos agentes que son útiles bien para el usuario o bien para otros agentes se reproducen, mientras que los de bajo desempeño son destruidos. Este trabajo presenta diversos experimentos con distintas configuraciones y varios ratios de interés de usuario.

myVU (Geyer-Schulz et al. [6]), se presenta como una nueva generación de sistemas recomendadores, su enfoque se basa en el comportamiento de consumo observado en el usuario y la computación evolutiva interactiva, este sistema implementa un manejo de las relaciones de los clientes y un mercadeo uno-a-uno en un sistema de corretaje educativo y científico de una universidad virtual. myVU proporciona una interfaz web adaptiva para los distintos miembros de una universidad virtual y presenta rutinas de recomendación para los sitios frecuentemente utilizados.

3 Experimento

Nuestro experimento se basa esencialmente en el trabajo de S. Uyin y P. Bentley [19], dicho trabajo usa un AG para aprender características personales de cada usuario y entonces proporcionar sugerencias a la medida de este, los resultados obtenidos son congruentes con el trabajo inicial hacia donde se enfocaba principalmente el experimento, aunque nuestro experimento es nuevo, en el sentido que se enfoca más en la medición de desempeño del mismo desde un punto de vista que lo abarca como proceso de aprendizaje, es decir contrastar que el algoritmo realmente predice lo que el usuario ha votado. El esquema general usado se ilustra en la Fig. 3 y se detalla a continuación.

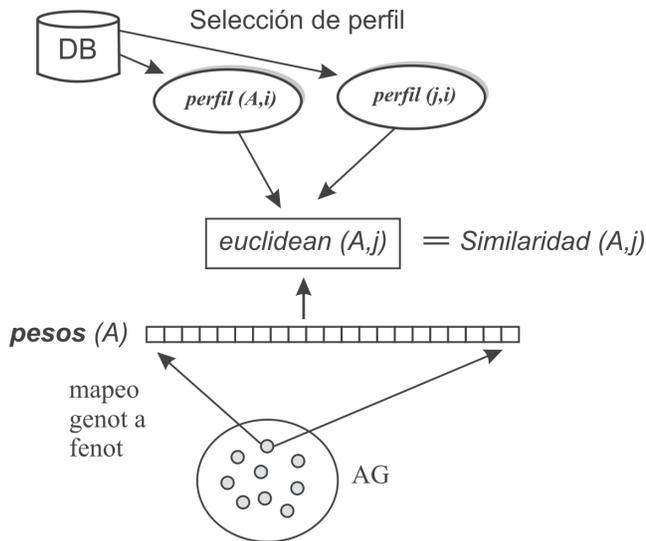


Fig. 3. Esquema General del uso del Algoritmo Genético en el proceso de filtrado

3.1 Generación de Perfiles

En este ejercicio se hace uso de conjunto de datos *MovieLens* (<http://www.movielens.umn.edu>), el sistema creado usa las 22 características de dichos datos: puntuación de película, edad, sexo, ocupación y 18 géneros de películas: acción, aventura, animación, infantil, comedia, crimen, documental, drama, fantasía, cine negro, horror, musical, misterio, romance, ficción, thriller, guerra y oeste.

Antes de que la recomendación sea hecha, el conjunto se separa en perfiles separados, uno para cada persona que define las preferencias de películas para dicha persona, se define $perfil(j,i)$ como el perfil del usuario j para la película i (Fig. 4)

1 Puntaje	2 Edad	3 Género	4 Ocupación	..22 18 Frecuencias de género
5	23	0	45	000000100010000000

Fig. 4. $perfil(j,i)$, perfil para el usuario j con puntaje sobre la película i , si i tiene un puntaje de 5

El perfil(j) entonces es entonces una colección de los $perfil(j,i)$ para todas las películas que j haya visto, una vez se construyen los perfiles entonces puede iniciar el proceso de recomendación, dado un usuario activo A , un conjunto vecindad de perfiles similares a A debe ser hallado.

3.2 Selección de Vecindad

En el caso ideal la BD completa debería usarse para seleccionar los mejores perfiles posibles, sin embargo no es la opción más factible dado si hay una cantidad de datos y no hay disponibilidad de recursos es por ello que la gran mayoría de sistemas optan por hacer un muestreo aleatorio.

En esta parte se computa la distancia o similitud entre los perfiles seleccionados anteriormente y el usuario actual, usando una función de distancia, la gran mayoría de sistemas recomendadores usan algoritmos estándar que consideran información de votos como la característica sobre la cual se hace la comparación de perfiles, sin embargo en la vida real la manera en la que se dice que dos personas son similares, no se basa en si asignan puntos o no, a una película de la misma forma, sino más bien en aspectos personales y de su accionar pasado, cada usuario usa una distinta prioridad para cada característica (Fig. 5), el enfoque que se pretende utilizar en este experimento es el de un AG que puede evolucionar los pesos que cada usuario asigna a sus prioridades, una potencial solución se ilustra a continuación.

w_1	w_2	w_3	...	w_{22}
-------	-------	-------	-----	----------

Fig. 5. Fenotipo de un individuo en la población

Donde w_f es el peso asociado con la característica f , cada individuo contiene 22 genes que deben evolucionarse por un AG elitista (descrito más adelante).

La comparación entre dos usuarios puede ser hecha entonces por una distancia euclidiana modificada, la cual toma en cuenta múltiples características, $Euclidean(A,j)$ es la distancia entre el usuario activo A y el usuario j :

$$euclidean(A, j) = \frac{1}{z} \sum_{i=\lambda_1}^{\lambda_z} \sqrt{\sum_{f=1}^2 w_f * diff_{i, f}(A, j)^2} \quad (3)$$

Donde:

A es el usuario activo

j es un usuario provisto por el proceso de selección de perfil además $j \neq A$

Los items comunes que A y j han puntuado se definen como el conjunto $\lambda_1 \dots \lambda_z$.

z es el numero de películas en común.

w_f es el peso del usuario activo para la característica f

i es un ítem común de película, donde $profile(A,i)$ y $profile(j,i)$ existen.

$diff_{i,j}(A,j)$ es la diferencia en el valor de perfil para la característica f entre los usuarios A y j sobre la película i .

Los valores de perfiles deben normalizarse antes de hacer dichos cálculos, cuando el peso para cualquier característica es 0 dicha característica es ignorada, de esta forma es posible permitir que la selección de característica sea adaptativa a cada preferencia de usuario, la diferencia en los perfiles para la ocupación es 0, si la ocupación es la misma, o 1 si es distinta.

3.3 Algoritmo Genético Empleado

Para esta tarea se ha elegido desarrollar un AG elitista en donde el 25% superior de los individuos son mantenidos para la próxima generación, para cada nueva generación se seleccionan individuos de forma aleatoria del 40% superior, dos hijos se producen de cada dos padres con cruce de un punto con $Pc=1$, la mutación se aplica en cada posición con $Pm=0.01$, se usa una codificación binaria simple, usando 8 bits para cada uno de los 22 genes, el mapeo genotipo-fenotipo (pesos) se hace, al convertir de binario a decimal, los pesos pueden luego calcularse de estos valores, la importancia de los 18 géneros es reducida mediante un factor de reducción esto se hace para dar igual oportunidad a las otras características (edad, sexo), el valor total del fenotipo se halla sumando los valores de las 22 características, el valor del peso se halla dividiendo el valor real por el valor total, se usará la siguiente función de predicción (la aptitud finalmente):

$$predict_vote(A, i) = mean_A + k \sum_{j=1}^n euclidean(A, j)(vote(j, i) - mean_j) \quad (4)$$

Donde:

- $mean_j$ es la media de voto para el usuario j
- k es un factor de normalización tal que la suma de las distancias euclidianas es igual a 1
- $vote(j,i)$ es el actual voto del usuario j sobre el ítem de película i
- n es el tamaño de vecindad

Todas las películas vistas por el usuario activo se particionan de forma aleatoria en dos conjuntos: entrenamiento (1/3) y prueba (2/3)

4 Conclusiones y Resultados

El trabajo que hemos presentado (además de servir como base general del trabajo en filtrado colaborativo de información), ha reformulado la tarea del filtrado de información, como una tarea de aprendizaje en donde se realiza un afinamiento de parámetros (que ponderan características de preferencias de usuario) que luego son confrontados a través de la función de predicción de voto. Los resultados que se ilustran abajo, muestran una adecuada tendencia en la predicción por parte de la técnica de los AGs.

A continuación (Fig. 6, Fig. 7,

Fig. 8) se ilustran algunos resultados de predicción obtenidos (a través de la Ecuación 4 y luego de que el AG ha hallado la configuración de pesos *más apropiada*), con miras a una presentación más concisa, se ilustra el análisis para un número predeterminado (fijo) de las primeras n películas. El conjunto de prueba es en este caso 15, sobre el conjunto total de películas que anteriormente ha votado un usuario dado, esto es se ve la capacidad de predicción del algoritmo en base a la vecindad elegida. La cantidad que refleja el eje Y es el valor de puntaje (votado y predicho para una película dada), a nivel del ejemplo que se ilustra el tamaño de la vecindad (n) es dada por la suma del número de aquellos usuarios que tienen una distancia menor a 0.25 del usuario elegido.

La métrica de evaluación usada, para cada uno de los resultados obtenidos, es la MAE (Acrónimo en inglés del error absoluto medio), la tabla ilustra los valores para cada uno de los tres resultados que se ilustran.

Tabla 1. Resultados de la métrica MAE para los resultados obtenidos

Usuario	MAE
21	0,473
22	0,408
202	0,442

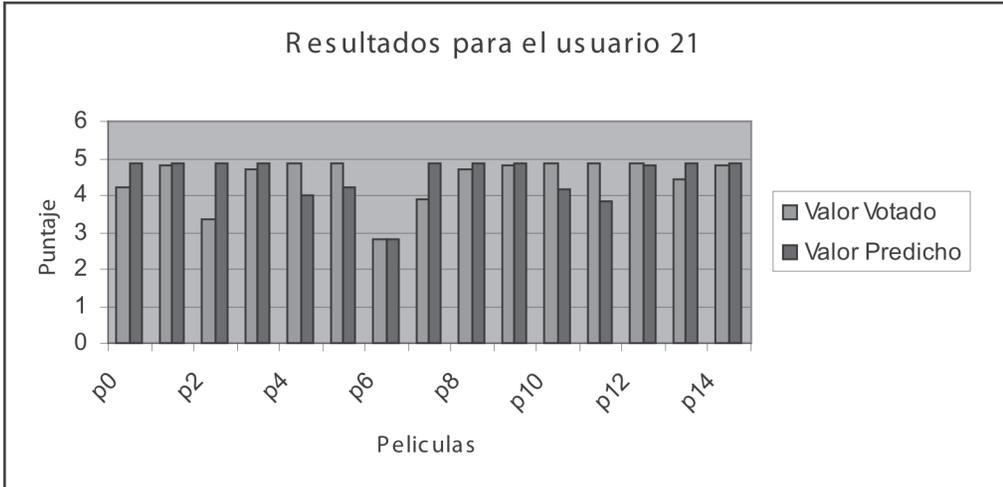


Fig. 6. Resultados votados versus predichos en la primeras 15 películas para el usuario 21

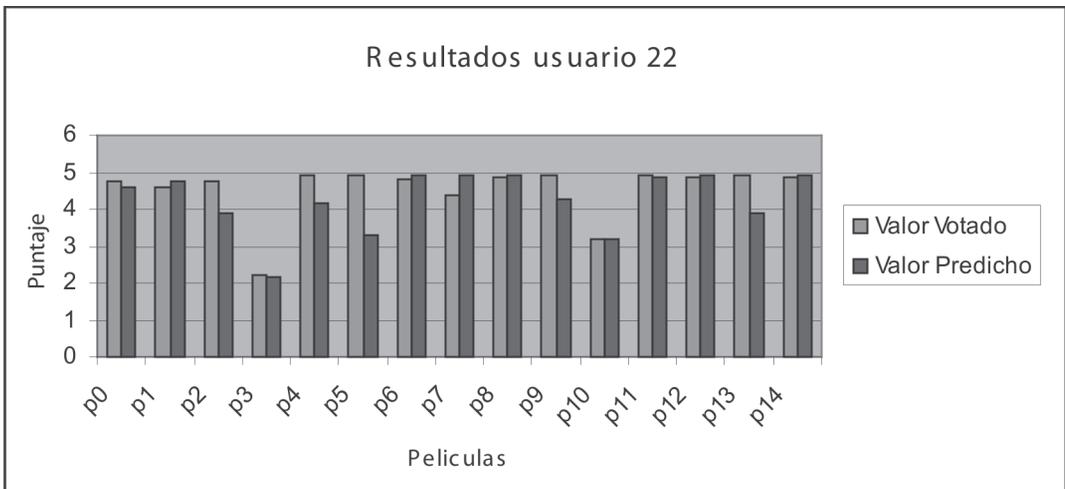


Fig. 7. Resultados votados versus predichos en la primeras 15 películas para el usuario 22

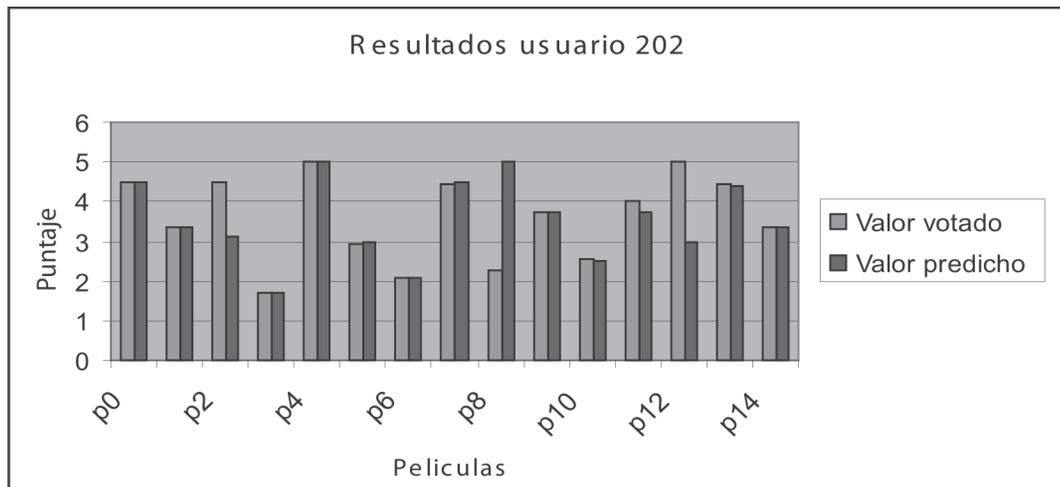


Fig. 8. Resultados votados versus predichos en la primeras 15 películas para el usuario 202

5 Trabajo Futuro

Dado que nuestra propuesta trata de enfocarse en el aprendizaje (o evolución) de pesos (hallar la configuración más apropiada de estos) no hay (al estado actual del trabajo) un desarrollo que también involucre dicha configuración de pesos dentro de métricas distintas a la usada (como la similitud del coseno o la de Pearson las cuales así como se plantean, hayan distancias entre ítems y no entre usuarios como la métrica aquí usada), con lo cual es difícil comparar directamente el uso de dichas similitudes, un posible experimento futuro contempla dicha comparación desde un enfoque adaptivo y uno no-adaptivo (usando dichas métricas).

En una versión más actualizada del experimento realizado, pretendemos en segunda instancia realizar un análisis empírico con mayor profundidad que constante la efectividad de la técnica del AG versus otras técnicas de filtrado (como las expuestas en [3]), y sobre otros dominios (bromas en *Jester DataSet* (<http://www.ieor.berkeley.edu/~goldberg/jester-data/>) y libros en *BookCrossing* (<http://www.informatik.uni-freiburg.de/~ctiegl/BX/>)), se propone también una evaluación complementaria en donde se tendrán en cuenta aspectos como los propuestos por Cleverdon et al [5], donde se identifican cinco métricas a tener en cuenta:

- Retardo: Intervalo de tiempo transcurrido desde que se hace la demanda hasta que se da la respuesta.
- Presentación: El formato físico de la salida del sistema.

- Esfuerzo del usuario: El esfuerzo, intelectual o físico que se demanda del usuario.
- Exhaustividad: Capacidad del sistema de presentar todos los ítems relevantes.
- Precisión: Capacidad del sistema de ocultar ítems que no sean relevantes

Así mismo se precisa poder observar con más detalles cada ejecución, para ver que tipos de resultados se presentan en cuanto a los pesos (ponderaciones de las características) que se evolucionan en este enfoque.

Referencias

- [1]. Aggarwal, Ch. C., Wolf, J. L., Wu, K-L., and Yu, P. S. Horting hatches an egg: A new graph-theoretic approach to collaborative filtering. In Knowledge Discovery and Data Mining, 1999. pp. 201-212.
- [2] Belew, R. K. Adaptive information retrieval. In Proceedings of the Twelfth Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval, , June 1989, Cambridge, MA,. pp 11-20.
- [3]. Breese, J.S., Heckerman, D. and Kadie, C. Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence 1998. pp. 43-52.
- [4] Christakou, C., Stafylopatis, A. A hybrid movie recommender system based on neural networks. In Proceedings 5th International Conference on Intelligent Systems Design and Applications, 2005. ISDA '05., Sept. 2005, pp 500 – 505.
- [5] Cleverdon, C., Mills, J., Keen, M. Factors Determining the Performance of Indexing Systems , Vol. 2--Test Results. ASLIB Cranfield Res. Proj., Cranfield, Bedford, England, 1966.
- [6]. Geyer-Schulz, A., Hahsler, M., Jahn, M. myVU: A Next Generation Recommender System Based on Observed Consumer Behavior and Interactive Evolutionary Algorithms. In: W. Gaul, O. Opitz, M. Schader (Eds.): Data Analysis – Scientific Modeling and Practical Applications, Studies in Classification, Data Analysis, and Knowledge Organization, Vol. 18, 2000. Springer, Heidelberg, 447-457.
- [7] Heckerman, D., Chickering, D., Meek, C., Rounthwaite, R., Kadie, C. Dependency Networks for Density Estimation, Collaborative Filtering, and Data Visualization. *Journal of Machine Learning Research*. 1:49-75, 2000.
- [8] Herlocker, J.L., Konstan, J.A., Borchers, A. and Riedl, J.. An Algorithmic Framework for Performing Collaborative Filtering. In SIGIR '99: proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 230-237, 1999.

- [9] Kwok, K. L. A neural network for probabilistic information retrieval. In Proceedings of the Twelfth Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval, June 1989, Cambridge, MA., pp 21-30
- [10]. Malone, T.W., Grant, K.R., Turbak, F.A., Brobst, S.A. and Cohen, M.D. Intelligent information sharing systems, -Communications of the ACM, 30(5) 1987, 390-402.
- [11]. Min Tjoa, A., Höfferer, M., Ehrentraut, G., Untersmayr, P. Applying Evolutionary Algorithms to the Problem of Information Filtering. DEXA Workshop 1997: 450-458.
- [12]. Moukas, A., Maes., P. Amalthea: an evolving multi-agent information filtering and discovery system for the WWW. Autonomous Agents and Multi-agent Systems, 1(1) 1998, pp 59-88.
- [13] Nasraoui, O., and Pavuluri, M. Accurate Web Recommendations Based on Profile-Specific URL-Predictor Neural Networks. In Proceedings of the International World Wide Web Conference, New York, NY, May. 2004.
- [14] Nichols, D. M. Implicit Rating and Filtering. In Proceedings of the Fifth DELOS Workshop on Filtering and Collaborative Filtering, Nov. 1997, ERCIM: pp.31-36.
- [15] Salton, G., and McGill, M.J. Introduction to Modern Information Retrieval. McGraw-Hill, New York, 1983.
- [16] Sarwar, B., Karypis, G., Konstan, J. and J. Riedl. Analysis of recommendation algorithms for e-commerce. In Proceedings of ACM E-Commerce, 2000.
- [17] Sebastiani, F. Machine Learning in Automated Text Categorisation. Technical Report IEIB4-31-1999, Consiglio Nazionale delle Ricerche, Pisa, Italy, 1999.
- [18] Sheth, B., Maes, P. Evolving agents for personalized information filtering. In Proc on Artificial Intelligence for Applications 1993. US, 345-352.
- [19] Ujjin, S. and Bentley, P. J. Learning User Preferences Using Evolution. In Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution And Learning (SEAL'02) 2002. Singapore.
- [20] Ungar, I., Foster, D. Clustering Methods for Collaborative Filtering (1998). Proceedings of the Workshop on Recommendation Systems