

**Universidad de las Ciencias Informáticas
Facultad 1
Centro de Ideoinformática**

**Procedimiento para recomendar problemas en el Juez
en Línea Caribeño**

Trabajo final presentado en opción al título de Máster en Informática
Aplicada

Autor: Ing. Hubert Viltres Sala

Tutor: Dr. C. Yusnier Valle Martínez

Co-Tutora: MSc. Delly Lien González Hernández

La Habana, 2014

Agradecimientos

A mi familia por todo el apoyo brindado.

A la Revolución y al Comandante en Jefe Fidel Castro Ruz.

A mis tutores Yusnier Valle Martínez y Delly Lien González Hernández por sus sugerencias y paciencia en el desarrollo de la tesis.

A José Carlos González Fernández y Alcides Cabrera Campos por su aporte a mi investigación.

A mis amigos que son el motor impulsor de este resultado.

DECLARACIÓN JURADA DE AUTORÍA

Declaro por este medio que yo Hubert Viltres Sala, con carné de identidad 84090524183, soy el autor principal del trabajo final de maestría "**Procedimiento para recomendar problemas en el Juez en Línea Caribeño**", desarrollada como parte de la Maestría en Informática Aplicada y autorizo a la Universidad de las Ciencias Informáticas a hacer uso de la misma en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

Y para que así conste, firmo la presente declaración jurada de autoría en La Habana a los días ____ del mes de diciembre del año 2014.

Ing. Hubert Viltres Sala

Autor

MSc. Delly Lien González Hernández

Co-Tutora

Dr. C. Yusnier Valle Martínez

Tutor

Índice

Resumen	IV
Introducción	1
Capítulo 1: La recomendación de información y el Juez en Línea Caribeño	8
1.1. Gestión de la información	8
1.2. Competencias de programación	10
1.3. Competencias de programación mediante jueces en línea	11
1.4. Principales Jueces en Línea para Competencias de programación	11
1.5. Herramientas para recomendar información en los jueces en línea	15
1.6. Sistemas de Recomendación	17
1.7. Clasificación de los Sistemas de Recomendación	22
1.8. Métrica de evaluación de Sistemas de Recomendación	35
1.9. Conclusiones parciales	39
Capítulo 2: Procedimiento para analizar, clasificar y recomendar problemas en el Juez en Línea Caribeño	40
2.1. Estructura general de la propuesta	40
2.2. Fuente de datos y de conocimientos	41
2.3. Representación de la información	43
2.4. Representación de los elementos	44
2.5. Procesamiento (Motor de recomendación)	45
2.6. Módulo de Recomendación de problemas en el COJ	49
2.7. Conclusiones del capítulo	50
Capítulo 3: Análisis del resultado de aplicar el procedimiento para recomendar problemas en el Juez en Línea Caribeño	51
3.1. Métricas para la validación del procedimiento	51
3.2. Preexperimento para validar el procedimiento	56
3.3. Conclusiones del capítulo	58
Conclusiones Generales	59
Recomendaciones	60
Referencias Bibliográficas	61
Anexos	67

Resumen

El presente trabajo propone un procedimiento para recomendar problemas en el Juez en Línea Caribeño, detallando cada uno de los elementos que lo componen. El estudio realizado se basa en diferentes técnicas o herramientas que ayudan al usuario en el proceso de selección de la información más relevante y ajustada a sus necesidades, teniendo en cuenta que en el escenario tecnológico actual, el crecimiento exponencial de la web plantea nuevas oportunidades y disímiles retos para los usuarios. El documento describe el resultado de la investigación sobre los sistemas de recomendación de información, el análisis de los principales enfoques y algoritmos utilizados, además de las principales métricas de evaluación de los sistemas de recomendación. Se diseña y aplica un preexperimento utilizando pruebas estadísticas que permitió comprobar la efectividad del procedimiento. La utilización de las métricas de exactitud predictiva y en clasificación evidenció la calidad de las recomendaciones obtenidas que posibilitan potenciar las habilidades de los usuarios del Juez en Línea Caribeño disminuyendo el número de intentos fallidos en la resolución de problemas.

Palabras Clave: sistema de recomendación, algoritmos, métricas, Juez en Línea Caribeño, procedimiento.

Introducción

El desarrollo alcanzado en la actualidad por las tecnologías de la información y las comunicaciones (TIC) y el rápido crecimiento de Internet en los últimos años, ha propiciado que se genere un gran volumen de contenido web. En este contexto los usuarios cuando acceden a Internet se sienten abrumados por la sobrecarga de información y no obtienen de forma fácil y rápida la que realmente necesitan.

El crecimiento exponencial de la web plantea nuevas oportunidades y disímiles retos para los usuarios cada día. La gran cantidad de información disponible en la red de redes hace más complejo el procesamiento y posterior extracción de la información útil para los usuarios. Al no poder procesar toda la información disponible debido al tiempo o la escasa experiencia en la web, muchas veces buscan ayuda en personas con sus mismos gustos o intereses para que los puedan guiar a identificar la información que necesitan.

Una solución del área de la informática a la necesidad de los usuarios de obtener información personalizada y relevante, ha sido el surgimiento de los denominados Sistemas de Recomendación (SR) (en inglés *Recommender Systems*). Según (Resnick y Varian, 1997; Burke, 2002; Resnik et al., 2010; Ricci et al., 2010) se define en la literatura disponible, y a los efectos de esta investigación se considera que un SR es una herramienta o software, que, utilizando preferencias, sugerencias y valoraciones emitidas por los usuarios, permite guiarlos en su proceso de búsqueda de información, previa selección de la información más relevante, mostrando una cantidad que sea manejable por el usuario, adaptada a sus necesidades y preferencias. La aplicación de un SR es muy diversa, en gran medida depende de la fuente de datos que utiliza para realizar la recomendación. En la actualidad la aplicación de estos sistemas a la educación ha ganado importancia en gran medida debido a los algoritmos que utilizan para analizar, clasificar y seleccionar la información relevante para el usuario y su enfoque de aprender de las preferencias y valoraciones de los usuarios. Una de las áreas de aplicación de los SR es la educación y específicamente el aprendizaje colaborativo en competencias de programación.

El Concurso Internacional Universitario ACM de Programación (ACM-ICPC por sus siglas en inglés), efectuado anualmente con la participación de varios equipos en

representación de diferentes universidades del mundo, constituye uno de los eventos más importantes y prestigiosos de programación de computadoras donde se fomenta el trabajo en equipo, la resolución de problemas y el desarrollo rápido de *software*. Dicho concurso consta de varias etapas de competición que van desde los concursos locales (efectuados en cada una de las universidades que compiten en el evento), los concursos nacionales, los regionales y el evento a nivel mundial realizado anualmente.

Cuba se ha ido insertando paulatinamente en el movimiento ACM-ICPC obteniendo buenos resultados en la región del Caribe, posibilitando que se cree en la región una sede caribeña de dicho movimiento. A través del Movimiento de Programación Competitiva "Tomás López Jiménez" (MPC-TLJ) en la Universidad de las Ciencias Informáticas (UCI) se ha fortalecido la participación cada vez mayor de concursantes en los eventos locales, nacionales y regionales de la ACM-ICPC. El Movimiento ACM-ICPC en la UCI realiza diferentes actividades enfocadas a incrementar la preparación de los participantes cubanos en todos los niveles del concurso (Ripoll, 2013; Ripoll, 2014). En la actualidad una de las principales herramientas que se utilizan para facilitar la preparación de los concursantes es el Juez en Línea Caribeño (COJ por su sigla en inglés), que mediante un amplio conjunto de problemas y la realización de competencias preparatorias constituye un referente para la preparación de los participantes en eventos ACM-ICPC (Roque, 2013; Ripoll, 2014).

Después de analizar los datos de la encuesta realizada a una muestra de los participantes en los eventos desarrollados en la universidad o usuarios registrados en el COJ y que interactúan de forma sistemática con el sistema se evidencia que el 90 % de los usuarios no siguen un orden escalonado en la selección de los problemas a resolver. El 95% selecciona para resolver los problemas que mayor cantidad de usuarios ya han resuelto, por recomendación verbal de otros usuarios o por indicación del entrenador. Más del 90% de los usuarios prefieren resolver ejercicios de las temáticas relacionadas con las categorías que ellos seleccionaron en su perfil de usuario y demuestran interés en valorar las soluciones a los problemas que resuelven ellos u otros usuarios. La encuesta realizada demuestra que los usuarios al interactuar con el COJ no explotan todas las ventajas que les brinda la herramienta, no siguen un proceso de aprendizaje escalonado y pueden abrumarse por la gran cantidad de problemas publicados.

Es muy frecuente que los usuarios al interactuar con el COJ, no hagan una selección correcta de los ejercicios publicados debido a la gran disponibilidad de ofertas y son incapaces de elegir los ejercicios que mejor se adaptan a sus preferencias, competencias y habilidades como programadores. Además el COJ propone mucho contenido (problemas categorizados, códigos fuente, competencias, etc.) que no se brinda de forma adecuada a los usuarios según sus preferencias, limitando con ello la experiencia del usuario en el sistema y el valor agregado de éste respecto a sistemas similares. Existen por tanto insuficiencias para potenciar las habilidades y competencias individuales de los usuarios, pues no se identifican sus necesidades y preferencias relacionadas con la información disponible en el COJ.

Además de la variedad y profundidad de conocimientos previos, los usuarios pueden tener diferentes estilos de aprendizaje, por lo que se requiere evaluar y aplicar técnicas que permitan seleccionar la información que más se ajusta a su perfil y esta es otra insuficiencia del COJ. Debido a la necesidad de guiar al usuario en su preparación y al gran cúmulo de información disponible se plantea como **problema de investigación**: ¿Cómo analizar y clasificar la información del Juez en Línea Caribeño de manera que permita recomendar problemas de interés a los usuarios para potenciar sus habilidades individuales?

Como parte del diseño de la investigación se define como **objeto de estudio**: proceso de búsqueda y recuperación de información.

Objetivo General: Desarrollar un procedimiento de análisis y clasificación de información para el Juez en Línea Caribeño que permita recomendar problemas de interés para potenciar las habilidades individuales de los usuarios.

Objetivos específicos:

- Establecer los fundamentos teóricos y tendencias en el análisis, clasificación y recomendación de información. Además de identificar técnicas, metodologías y algoritmos empleados en los principales Sistemas de Recomendación de Información y su integración con los Jueces en línea de programación.
- Desarrollar un procedimiento que permita analizar, clasificar y recomendar problemas y su integración con el Juez en línea Caribeño.

- Validar el procedimiento de análisis, clasificación y recomendación de problemas para el Juez en Línea Caribeño utilizando métricas de evaluación de sistemas de recomendación de información.

Hipótesis

Un procedimiento de análisis y clasificación de información en el Juez en Línea Caribeño incide favorablemente en la recomendación de problemas de interés a los usuarios para potenciar sus habilidades individuales.

Experimento

El experimento es una situación de control en la cual se manipulan, de manera intencional, una o más variables independientes (causas) para analizar las consecuencias que tal manipulación tiene sobre una o más variables dependientes (efectos) (HERNÁNDEZ, 2006). A continuación se detalla la operacionalización de las variables dependientes e independientes.

Variable independiente: Procedimiento de análisis y clasificación de información.

Variable dependiente: Recomendación de problemas de interés a los usuarios para potenciar sus habilidades individuales.

Tabla 1. Operacionalización de las variables

Variable independiente	Dimensión	Definición conceptual	Indicadores	Unidades de medida
Procedimiento de análisis y clasificación de información	Sistema de recomendación de problemas	Procedimiento de análisis y clasificación de información que permite recomendar problemas a los usuarios activos en el COJ ajustados a las preferencias de su perfil	Capacidad para recomendar problemas ajustados a las preferencias del usuario	Alta, Media, Baja
Variable dependiente	Dimensión	Definición conceptual	Indicadores	Unidades de medida
Recomendación de problemas de interés a los usuarios para potenciar sus habilidades individuales	Problemas resueltos	Recomendar problemas a los usuarios según las preferencias de sus perfiles	Cantidad de intentos fallidos por ejercicios resueltos	Porcentual

Para el desarrollo de la investigación se emplearon los siguientes métodos científicos:

Métodos Teóricos:

Analítico-sintético: A partir del análisis de los referentes teóricos y la bibliografía relacionada con la investigación se hace una síntesis de los elementos relevantes que forman parte de caracterización de los Sistemas de Recomendación.

Histórico-lógico: Para llegar a la solución del problema de investigación es necesario el análisis del surgimiento, trayectoria, evolución y estado actual de los sistemas de recomendación y otros conceptos presentes en la investigación.

Hipotético-deductivo: A través de este método se parte de la hipótesis y siguiendo reglas lógicas de deducción, se llega a la contribución principal de la investigación que posteriormente será verificada sobre la base de las métricas de evaluación de los Sistemas de Recomendación.

Métodos Empíricos:

Observación: Este método se utilizó para obtener toda la información necesaria sobre el Juez en línea Caribeño que pueda facilitar el dominio de los perfiles de los usuarios y sus necesidades al interactuar con esta aplicación.

Encuesta: Este método se utilizó para obtener toda la información necesaria sobre la interacción de los usuarios en el Juez en Línea Caribeño.

Experimental: Se utiliza con datos provenientes de la base de datos del COJ. Se aplican las métricas de evaluación (métricas de exactitud predictiva y métrica de exactitud en clasificación) debidamente fundamentadas para analizar los resultados. Se establecen indicadores adecuados que permiten realizar correctas mediciones de los resultados.

Medición: Permite medir los resultados al comparar la técnica propuesta con otros métodos de recomendación de información existentes.

La **contribución** principal del presente trabajo es el análisis y clasificación de la información en el COJ, así como la generación de recomendaciones ajustadas a las preferencias de los usuarios.

Aportes teóricos:

1. Técnica para el análisis y clasificación de la información en el COJ.
2. Marco teórico con la caracterización de los principales algoritmos y técnicas utilizadas en el desarrollo de Sistemas de Recomendación.

3. Marco teórico con la caracterización de las métricas de evaluación de los Sistemas de Recomendación.

Aportes prácticos:

4. Desarrollo del Sistema de Recomendación para el Juez en Línea Caribeño.
5. Recomendación de información acorde al perfil de los usuarios.

Estructura del documento

El presente documento está compuesto por la Introducción, tres Capítulos, Conclusiones, Recomendaciones, Bibliografía y Anexos.

Capítulo 1: La recomendación de información y el Juez en Línea Caribeño.

Contiene un análisis de los principales conceptos asociados a los sistemas de recomendación. Se realiza una revisión bibliográfica de las principales técnicas, modelos y métricas de evaluación asociados a los sistemas de recomendación y una caracterización del Juez en Línea Caribeño.

Capítulo 2: Procedimiento para analizar, clasificar y recomendar problemas en el Juez en Línea Caribeño.

Se plantea la propuesta del procedimiento para analizar, clasificar y recomendar problemas en el Juez en Línea Caribeño. Se enuncia el procedimiento propuesto, sus características y estructura general y se plantean las fuentes de datos que se utilizan para analizar y clasificar los problemas a recomendar.

Capítulo 3: Validación del procedimiento para recomendar problemas en el Juez en Línea Caribeño.

Se presentan los resultados de las pruebas realizadas al Sistema de Recomendación de información para el Juez en Línea Caribeño. Se valida la exactitud del procedimiento utilizando la métrica exactitud predictiva y la de exactitud en clasificación. Además se realiza un diseño preexperimental para validar la propuesta.

Capítulo 1: La recomendación de información y el Juez en Línea Caribeño

El presente capítulo contiene una caracterización de los principales conceptos relacionados con el análisis, clasificación y recomendación de información. Se realiza una revisión bibliográfica de las principales técnicas, modelos y métricas empleadas en el desarrollo de los sistemas de recomendación de información. Se presenta una caracterización de los principales Jueces en Línea y se determinan los elementos más relevantes del Juez en Línea Caribeño.

1.1. Gestión de la información

Con el desarrollo alcanzado por Internet y el gran cúmulo de información que se genera diariamente se hace necesario gestionar la información y simplificarla, mostrando la más relevante para los usuarios cuando acceden a Internet; este proceso se realiza mediante la recuperación de información. El objetivo principal de la Recuperación de Información según plantean Bender y Deco (2014) es satisfacer la necesidad de información planteada por un usuario en una consulta en lenguaje natural especificada a través de un conjunto de palabras claves, también llamadas descriptores. La Recuperación de Información o *Information Retrieval* es la representación, almacenamiento, organización y acceso a *ítems* de información (Baeza et al., 1999). Después de analizado el concepto de Recuperación de Información se plantean las principales técnicas empleadas en el proceso de recuperar la información relevante para el usuario.

1.1.1. Técnicas de recuperación de información

- Sistemas de recuperación de lógica difusa

Esta técnica permite establecer consultas con frases normales, de forma que la máquina al realizar la búsqueda elimina signos de puntuación, artículos, conjunciones, plurales, tiempos verbales, palabras comunes (que suelen aparecer en todos los documentos), dejando solo aquellas palabras que el sistema considera relevantes. La recuperación se

basa en proposiciones lógicas con valores de verdadero y falso, teniendo en cuenta la localización de la palabra en el documento.

- Técnicas de ponderación de términos

Es común que en la búsqueda unos criterios tengan más valor que otros, por tanto la ponderación pretende darle un valor adecuado a la búsqueda dependiendo de los intereses del usuario. Los documentos recuperados se encuentran en función del valor obtenido en la ponderación. El valor depende de los términos pertinentes que contenga el documento y la frecuencia con que se repitan. De forma que, el documento más pertinente de búsqueda sería aquel que tenga representados todos los términos de búsqueda y además el que más valor tenga repetidos, independientemente de donde se localice en el documento.

- Técnica de *clustering*

Es un modelo probabilístico que permite las frecuencias de los términos de búsqueda en los documentos recuperados. Se atribuyen unos valores (pesos) que actúan como agentes para agrupar los documentos por orden de importancia, mediante algoritmos *ranking*.

- Técnicas de retroalimentación por relevancia

Esta técnica pretende obtener el mayor número de documentos relevantes tras establecer varias estrategias de búsqueda. La idea es que, tras determinar unos criterios de búsqueda y observar los documentos recuperados se vuelva a repetir nuevamente la consulta pero esta vez con los elementos interesantes, seleccionados de los documentos primeramente recuperados.

- Técnicas de *Stemming*

Morfológicamente las palabras están estructuradas en prefijos, sufijos y la raíz. La técnica de *Stemming* lo que pretende es eliminar las posibles confusiones semánticas que se puedan dar en la búsqueda de un concepto, para ello trunca la palabra y busca solo por la raíz.

Estas técnicas permiten definir el campo de aplicación de la búsqueda y recuperación de

información (BRI). Varios autores (Korfhage, 1997; Lancaster, 2001; Sokol, 2008) plantean que BRI tiene como principal objetivo proporcionar información relevante al usuario para satisfacer su necesidad de información, para ello se definen cinco actividades principales (localizar, seleccionar, interpretar, sintetizar y comunicar la información) que guían el proceso para obtener información relevante para el usuario. Para realizar el proceso de BRI se utilizan diferentes técnicas que permiten simplificar la información disponible y seleccionar para el usuario la más relevante. Para validar el proceso de BRI se utilizan algunas de las métricas de validación empleadas en el campo de la recuperación de información.

El análisis y clasificación de la información permite sintetizar la información más relevante para los usuarios y ayudarlos a procesar la información que realmente necesitan. Su utilización ha tenido gran aceptación y aplicación en diferentes áreas de conocimiento, una de ellas es la educación y específicamente se ha empleado en el aprendizaje colaborativo.

1.2. Competencias de programación

El Concurso de la ACM-ICPC, que se efectúa anualmente con la participación de varios equipos en representación de diferentes universidades del mundo, constituye uno de los eventos más importantes y prestigiosos de programación de computadoras donde se fomenta el trabajo en equipo, la resolución de problemas y el desarrollo rápido de *software*. Este concurso se desarrolla mediante varias etapas que van desde los concursos locales (efectuados en cada una de las universidades que compiten en el evento), los concursos nacionales, los regionales y finalmente el evento a nivel mundial realizado anualmente (Ripoll, 2014). Los eventos ACM-ICPC tienen como base fundamental el desarrollo de competencias de programación que permitan potenciar las habilidades para trabajar en equipo en la resolución de problemas de matemática, lógica y algoritmia; permitiendo incrementar su preparación para participar en cada una de las fases del movimiento ACM-ICPC. Las competencias se desarrollan utilizando jueces en línea que evalúan las habilidades adquiridas por los equipos.

1.3. Competencias de programación mediante jueces en línea

Un juez en línea es una aplicación web para entornos generalmente académicos, que incluye varios problemas de distintas materias para ser resueltos mediante técnicas de programación y, lo más importante, evalúa automáticamente las soluciones de sus usuarios en los varios lenguajes de programación disponibles. Se caracterizan por contener una interfaz bien definida para la interacción con el usuario y un alto nivel de disponibilidad, permitiendo el libre acceso a la aplicación en todo momento a las personas registradas (Junco, 2013).

La utilización de los jueces en línea ha sido diversa pero su aplicación está enfocada en potenciar las habilidades de los usuarios en la resolución de problemas, la programación de computadoras y el trabajo en equipos de programación. Una de las áreas donde más se ha utilizado es en competencias de programación competitiva, donde existen más de 50 jueces en línea a nivel internacional. La presente investigación se centra en los principales jueces en línea para competencias de programación, donde se analizan sus características y potencialidades en entornos competitivos (COJ, 2014).

1.4. Principales Jueces en Línea para Competencias de programación

Después de realizar un estudio de los principales jueces en línea (COJ, 2014), atendiendo a sus características, cantidad de problemas, competencias e información que brindan al usuario se identificaron como los jueces que más aportan a la investigación los siguientes:

1.4.1. Juez en Línea de la Universidad de Valladolid (*UVa Online Judge*)¹

El UVa es uno de los jueces en línea más antiguos y prestigiosos del mundo. Surge en el año 1995 con el propósito de ser usado como herramienta de entrenamiento para concursos de programación en formato ICPC. UVa tiene una de las mejores colecciones de problemas entre los jueces en línea. Según plantea Toledo (2013), a nivel mundial, es

¹ <http://uva.onlinejudge.org/>.

el que goza de más prestigio en la comunidad universitaria internacional, quizás por ser uno de los más antiguos y el mejor surtido en ejercicios utilizando los problemas de las competencias de los concursos ACM-ICPC. Entre sus principales características se encuentran:

- Módulo de estadísticas, que muestra información detallada respecto a los problemas, usuarios y concursos.
- Integración a la plataforma EduJudge², convirtiéndolo en uno de los pocos jueces en línea que poseen integración con el proceso de enseñanza-aprendizaje.
- Implementado sobre plataforma UNIX.

En materia del entrenamiento de concursantes ACM-ICPC así como en el aporte pedagógico el UVa es, a nivel mundial, el que goza de más prestigio en la comunidad universitaria internacional, quizás por ser uno de los más antiguos y el mejor surtido en ejercicios. Este sistema tiene como característica fundamental su organización de los problemas en extensos volúmenes nutridos normalmente de los concursos ACM-ICPC, constituyendo referencia obligada por parte de los interesados en el tema (Revilla, 2008).

- Tabla de posiciones, dedicada al desempeño de los usuarios en competencias.

Este juez en línea solo soporta el envío de soluciones para los lenguajes de programación permitidos en el ACM-ICPC (C, C++, Java, Pascal).

1.4.2. Sphere Online Judge (SPOJ)

Juez en línea desarrollado por el proyecto *Sphere Research Labs*. El SPOJ permite acceder a más de 232272 usuarios a un conjunto de problemas de alta calidad y variada dificultad que superan los 17000 problemas publicados, traducidos al ruso, portugués y vietnamita. El conjunto de problemas puede ser resuelto en cualquiera de los 48 lenguajes de programación soportados siendo los más utilizados C++ y C. Posee varias características novedosas que lo diferencian de la mayoría de los jueces en línea, tales como:

² Proyecto perteneciente a la Universidad de Valladolid que se dedica al desarrollo de herramientas con carácter pedagógico utilizando el Juez en Línea UVa.

- Posee tiempos de evaluación muy rápidos.
- Brinda un abundante *kit* de herramientas asociadas al sistema.
- Espacio de intercambio para sus usuarios mediante un foro de discusión
- Cuenta con un módulo especializado de competencia, muy versátil y eficiente.
- Presenta un sistema de juzgado especial digno de mención, dado que tiene varias implementaciones de la forma de evaluación, estrictamente necesaria para algunas tipologías de problemas (Junco, 2012).

1.4.3. Juez en Línea de Pekín (POJ)

El POJ es uno de los jueces en línea más popular entre la comunidad universitaria china, siendo uno de los más antiguos de esa región. Tiene características semejantes a otros jueces en línea de universidades chinas y de otras regiones del mundo. Durante un tiempo era posible descargar una versión disminuida y restrictiva de la aplicación. Acepta soluciones escritas en C, C + +, Pascal, Java y Fortran. Su característica más distintiva es el módulo de estadísticas, que recopila y muestra los datos de uso del sistema desde su surgimiento en el año 2003 hasta la fecha. Por otra parte, ofrece un servicio de mensajería interna y correo electrónico para los usuarios registrados, funcionalidad que se considera muy útil en este tipo de sistemas.

El POJ es la base del desarrollo de los jueces en línea en la UCI mediante la utilización de la versión descargable brindada por el POJ. Los conocimientos adquiridos en el estudio de otros jueces en línea permitió que en la universidad se desarrollaran diferentes jueces en línea entre ellos: *Xtreme Online Judge*, COJ v1, CPAV y COJ v2. La presente investigación se centra en el COJ por la relevancia que ha obtenido y además porque se pretende recomendar problemas sobre esta plataforma.

1.4.4. Juez en Línea Caribeño (COJ)

El desarrollo del sistema base (*Xtreme Online Judge*) comenzó en el año 2006 bajo la "Iniciativa Xtreme", fue publicado en Internet como COJ v1.0 el 5 de junio de 2010. El COJ posee las mismas funcionalidades que tienen los jueces que tuvieron como punto de partida el POJ, aunque en este se cambió el diseño y la arquitectura de información.

El COJ cuenta con más de 12000 usuarios registrados de 724 instituciones que representan a 144 países. Además posee un conjunto de problemas que superan los 2000 ejercicios con una variedad complejidad y alta calidad (COJ, 2014).

Posee servicios como:

- Mensajería interna para los usuarios registrados.
- Interfaz por correo electrónico para enviar soluciones y consultar el *status*.

Para el COJ se desarrolló un componente de recomendación (Yera, 2010) de contenidos que realizaba la recomendación basada en vecinos más cercanos, que en el momento de realizar esta propuesta de solución no estaba siendo utilizado por problemas de integración con la nueva versión del COJ y que presentaba como deficiencia el problema del arranque en frío.

1.4.5. Juez en Línea Académico (JELA)

Juez en línea enfocado en la docencia, desarrollado como tesis de maestría (Junco, 2014). Fue aplicado con gran aceptación a la enseñanza de la programación en la facultad 4 de la UCI. Utilizaba un enfoque de filtrado colaborativo empleando la correlación de *Person* como algoritmo a implementar. El Sistema de Recomendación es complementado con un segundo algoritmo que utiliza reglas de asociación *Apriori* para eliminar algunas de las deficiencias de los SR. Según plantea Junco (2014) el hecho de poseer un paquete de reglas de asociación como base para la generación de sugerencias posibilita enfrentar de manera exitosa muchas situaciones asociadas al arranque en frío en el caso de la existencia de nuevos usuarios. Este módulo de recomendación fue integrado al JELA y utilizado para potenciar las habilidades de los estudiantes en la enseñanza de la programación competitiva. En el momento de realizar esta investigación no se encontraba disponible para realizar una comparación aunque si se tuvo en cuenta su implementación para realizar la propuesta para el COJ.

1.5. Herramientas para recomendar información en los jueces en línea

Por la magnitud de la información manejada y para brindarle elementos (problemas, soluciones) al usuario con un alto valor y ajustado a sus necesidades, algunos jueces en línea han desarrollado herramientas con características recomendadoras sin implementar un sistema de recomendación completo. Entre estas herramientas se encuentran el *ACM Problem Grading*, de Sebastian Urbaniak, el *Next2Solve*, de Igor Naverniouk, y el *Problem classification*, en el *Sphere Online Judge*.

1.5.1. ACM Problem Grading y Next2Solve

El *ACM Problem Grading* es una herramienta implementada en apoyo al UVa por Sebastian Urbaniak que se limita a proporcionar un listado de problemas a resolver, el cual puede ser ordenado por diferentes criterios entre los que se pueden mencionar la cantidad de soluciones aceptadas para cada ejercicio, el porcentaje de soluciones correctas con respecto al total de soluciones y un *score* de simplicidad que se calcula tomando como base los dos anteriores criterios. No se enfoca en un usuario en particular, suponiendo que los problemas más sencillos a resolver para todos son aquellos que más soluciones correctas tengan. Los criterios que utiliza son completamente estáticos, siendo diametralmente opuesto a las actuales tendencias dentro de la web (Junco, 2012). No cumple con los principios de los sistemas de recomendación ya que no calcula la similitud entre los usuarios y la información que les brinda.

Por su parte, introduciendo un ligero aporte y desarrollado también para el UVa, *Next2Solve* es una herramienta basada en el *ACM Problem Grading*. Su mejor enfoque radica en que solicita la entrada del identificador de usuario para generar una sublista de la lista generada por la aplicación de Urbaniak, en la que solo aparecen los problemas que el usuario aún no ha resuelto (Junco, 2012). Presenta como mejora que está enfocada más en el usuario pero sigue sin adaptar los enfoques de un sistema de recomendación. El desarrollo de estas herramientas fue enfocado solo en el UVa y según su autor no tiene intenciones de adaptarlo a las nuevas tendencias.

1.5.2. SPOJ Problems Classification

Utiliza un enfoque centrado en los ejercicios, definiendo para cada uno de ellos una serie de datos o etiquetas que luego son utilizadas para agruparlos por categorías. Utilizando estas categorías y a través de opciones de filtrado, los usuarios pueden obtener propuestas de problemas a resolver, basándose en criterios de búsqueda, que coinciden generalmente con los valores de las etiquetas. Presenta diferentes herramientas con enfoques recomendador como *SPOJ Problems Classification 1 y 2*, que mediante la utilización de palabras claves recomienda problemas. Una de las dificultades que presenta es que no recomienda problemas ajustados al usuario, debido a que los usuarios navegan de forma anónima en la aplicación y no se almacenan sus preferencias. También presenta herramientas relacionadas con los usuarios donde muestra información de forma estadísticas como son *SPOJ User Statistics 1, 2 y 3* (varias de estas herramientas no pudieron ser probadas por no estar disponibles).

Estas herramientas independientemente de que no siguen un enfoque de sistema de recomendación fueron desarrolladas ajustadas a las necesidades de estos jueces en línea y no pueden ser integrados al COJ. Presentan limitaciones en su implementación al no cumplir con el principio básico de los sistemas de recomendación; no utilizan algoritmos que puedan analizar el comportamiento del usuario y determinar sus gustos y preferencias. Además no permiten aprender de las recomendaciones de los usuarios y no crean un perfil de preferencias.

El estudio realizado evidencia la importancia y necesidad de los jueces en línea en competencias de programación con formato ICPC, en gran medida debido a la cantidad de información disponible y los diferentes estilos de aprendizaje de los usuarios. La utilización de un juez en línea permite potenciar el desarrollo de las habilidades de los usuarios en las técnicas de programación, el diseño y análisis de algoritmos, el trabajo en equipo y el aprendizaje colaborativo. Los usuarios al interactuar con el juez en línea, debido a la gran disponibilidad de ofertas en ocasiones son incapaces de elegir los ejercicios que mejor se adaptan a sus preferencias, competencias y habilidades como programadores. La información disponible en el COJ no se brinda de forma adecuada a los usuarios según sus preferencias, limitando con ello su aprendizaje, la posibilidad de influir con sus resultados en el aprendizaje de otros usuarios y su experiencia general en

la utilización del Juez. Para capacitar y preparar a los concursantes se hace necesario disponer de un sistema que permita recomendar problemas ajustados a las necesidades de cada usuario para poder potenciar y mejorar sus habilidades individuales y en equipo. Existen por tanto, insuficiencias para procesar la información disponible y con ello potenciar las habilidades y competencias individuales de cada usuario. Además no se identifican las preferencias y necesidades de los usuarios y tampoco se analiza su historial en el COJ. La variedad y profundidad de conocimientos previos y que los usuarios pueden tener diferentes estilos de aprendizaje requiere que se evalúen y apliquen diferentes técnicas que permitan seleccionar la información que más se ajusta a su perfil. Para aumentar las habilidades individuales y colectivas de los usuarios, la calidad y cantidad de problemas resueltos. Lo antes planteado evidencia la necesidad de utilizar diferentes técnicas que permitan analizar y clasificar los problemas disponibles, posibilitando recomendar los más relevantes para el usuario en el COJ.

1.6. Sistemas de Recomendación

Un Sistema de Recomendación tienen como particularidad que se diseña para entender las necesidades individuales de los usuarios a los que asisten, siendo claves para su adopción la percepción de los usuarios/consumidores sobre el grado en que los SR les entienden y les proporcionan información personalizada útil (Komiak y Benbasat, 2006). Los Sistemas de recomendación aglutinan una gran variedad de necesidades de los usuarios, ayudándolos en su proceso de selección de información útil. Según se plantea en la conferencia *ACM RecSys 2013*, una recomendación es una forma particular de filtrado de información, que explota los comportamientos pasados y similitudes del usuario para generar una lista de elementos de información que se adaptan personalmente a las preferencias de un usuario final (ACM RecSys, 2013). El funcionamiento básico de un Sistema de recomendación (ver figura 1) establece las relaciones entre los elementos (productos), los usuarios y sus preferencias; desarrollándose en 3 etapas (captura de las preferencias, procesamiento y recomendación de la información).



Figura 1 Esquema general de un sistema de recomendación (Formoso, 2013)

Los SR tienen como finalidad entender las necesidades de los usuarios, para ello definen un conjunto de tareas que permiten comprender mejor su funcionamiento y aplicación.

1.6.1. Tareas de un Sistema de Recomendación

Las tareas de un SR están enmarcadas desde dos puntos de vista, el primero, enfocado en el proveedor del servicio y el segundo en el usuario que obtiene la recomendación. Su auge ha llegado a los usuarios que ven representado en un SR sus metas y objetivos, estos sistemas los ayudan de forma efectiva a encontrar la información que realmente necesitan. En consecuencia un SR debe saber equilibrar las necesidades de los proveedores y los usuarios, permitiendo ofrecer un servicio que sea valioso para ambos. Para el proveedor de la recomendación, el principal objetivo de un SR es aumentar los beneficios, mientras para los usuarios es obtener información ajustada a sus necesidades y preferencias. Según se plantea en el libro *Recommender Systems Handbook* (Ricci et al, 2010) y citando a (Herlocker et al., 2004), se han definido once tareas que a grandes rasgos identifican la función de un SR.

- Encontrar un elemento relevante (*Find Some Good Items*): Recomendar a un usuario un conjunto de elementos de acuerdo a sus gustos y preferencias. Estas recomendaciones pueden ir acompañadas de una puntuación (ejemplo: en una escala de una a cinco estrellas) para determinar hasta qué punto el elemento puede satisfacer al usuario.
- Encontrar todos los elementos relevantes (*Find all good items*): Recomendar a un usuario todos los elementos que puedan satisfacer alguno de sus gustos o preferencias. En determinadas ocasiones no es suficiente con sugerir algunos elementos que sean del gusto del usuario, sino que es necesario recuperar todos los elementos que pueden ser de su interés, aún a riesgo de que el usuario tenga que invertir gran cantidad de tiempo en analizarlos (Fernández, 2014).
- Anotar según el contexto (*Annotation in context*): Analizando el contexto existente, por ejemplo, una lista de elementos, el sistema predice la preferencia de un usuario sobre un elemento determinado en función del contexto. Según dicha anotación, el usuario será capaz de decidir si el elemento en cuestión es acorde o no a sus gustos y preferencias (Fernández, 2014).
- Recomendar una secuencia (*Recommend a sequence*): En lugar de centrarse en generar recomendaciones individuales, la idea es recomendar siguiendo una secuencia de elementos que sea agradable en su conjunto para el usuario.
- Recomendar un paquete (*Recommend a bundle*): Sugerir un grupo de elementos relacionados entre sí. Por ejemplo, un plan de viaje puede estar compuesto por destinos y cada uno de ellos con varios lugares de interés y servicios de hospedaje situados en la misma zona. El usuario puede analizar estas diferentes alternativas y seleccionar un destino turístico único.
- Navegar (*Just browsing*): En ocasiones, el usuario navega por el catálogo sin intención inminente de adquirir un artículo, sino únicamente con la intención de encontrar artículos que sean de su interés. La tarea del SR es ayudar al usuario a navegar por los artículos que son más similares a caer en el ámbito de interés del usuario para esa sesión de navegación específica.
- Encontrar recomendaciones creíbles (*Find credible recommender*): En ocasiones algunos usuarios no confían en las sugerencias realizadas por los SR por lo que

juegan con ellos para comprobar que tan buenos son realizando las recomendaciones. Esta tarea consiste en ofrecer funcionalidades específicas al usuario que les permita probar el comportamiento del propio sistema para aumentar así su confiabilidad.

- Mejorar el perfil (*Improve the profile*): Esta tarea se relaciona con la capacidad del usuario de proporcionar información al sistema de recomendación sobre sus gustos. El objetivo fundamental de la tarea es obtener información del usuario para realizar recomendaciones personalizadas.
- Expresarse (*Express self*): Algunos usuarios no se preocupan por obtener recomendaciones. Más bien, lo importante para ellos es poder contribuir con sus valoraciones y expresar sus ideas y opiniones. La satisfacción del usuario puede contribuir a mantener el Sistema de Recomendación.
- Ayudar a otros (*Help others*): Algunos usuarios están dispuestos a contribuir con sus valoraciones, porque creen que la comunidad se puede beneficiar de su contribución.
- Influir en los demás (*Influence others*): En ocasiones hay usuarios cuyo objetivo principal es influir explícitamente con sus opiniones a otros usuarios sobre determinado elemento. También hay algunos usuarios malintencionados que pueden utilizar el sistema solo para promover o penalizar a ciertos productos.

Las tareas planteadas por (Herlocker et al., 2004) y a los efectos de la literatura consultada constituyen un referente teórico para implementar un Sistema de Recomendación definiendo los elementos imprescindibles que se deben tener en cuenta para desarrollar un SR. Además se plantea (Fernández, 2014) que las tareas Anotación en el contexto (Tarea de predicción) y Encontrar un elemento relevante (Tarea de recomendación) son las más estudiadas. La aplicación de un Sistema de Recomendación puede ser muy diversa y para obtener mejores resultados se hace necesario utilizar una amplia variedad de fuentes de datos y técnicas de recomendación.

1.6.2. Fuente de datos y de conocimientos

Los Sistemas de Recomendaciones procesan la información que recopilan de forma activa de varios tipos de datos con el fin de construir sus recomendaciones. Los datos

son principalmente acerca de los elementos que se sugieren y los usuarios que recibirán estas recomendaciones. Las fuentes de conocimiento disponibles para los sistemas de recomendación pueden ser muy diversas, como clasificación general, los datos utilizados se agrupan en tres grupos: elementos (*Items*), usuarios (*Users*) y transacciones (*Transactions*), es decir, las relaciones entre los usuarios y los elementos (ver figura 1).

Elementos (*Items*): Los elementos son los objetos que se recomiendan. Los elementos pueden ser caracterizados por su complejidad, su valor o utilidad. El valor de un elemento puede ser positivo si el artículo es útil o negativo si no es adecuado para el usuario.

Usuarios (*Users*): Los usuarios de un RS, como se mencionó anteriormente, pueden tener objetivos y características muy diversas. Con el fin de personalizar las recomendaciones y la interacción sistema-usuario, los RSs explotan una variedad de información sobre los usuarios. La información puede estructurarse de varias maneras y la selección de esta depende de la técnica de recomendación.

Transacciones (*Transactions*): Es la interacción registrada entre un usuario y el Sistema de Recomendación. Las transacciones son datos de registro similares que almacenan información importante generada durante la interacción persona-ordenador y que son útiles para el algoritmo de generación / recomendación que el sistema está utilizando. Por ejemplo, un registro de transacciones puede contener una referencia a la opción seleccionada por el usuario y una descripción del contexto para esa recomendación en particular. Si está disponible, la transacción puede incluir también una retroalimentación explícita que el usuario ha proporcionado, como la calificación para el elemento seleccionado. Enfocada principalmente en el historial de navegación del usuario y las valoraciones sobre los elementos que se le presentan. Posibilitando comprender mejor al usuario y el valor de cada elemento que se le muestra.

Después de realizar un estudio de los elementos que componen un SR, analizado las principales tareas y las formas de obtener los datos se plantean las principales áreas de aplicación de estos sistemas.

1.6.3. Aplicación de los Sistemas de Recomendación

La aplicación de los SR es muy diversa y se ajusta a diferentes contextos, las principales áreas de aplicación acorde a la investigación realizada son:

- Comercio electrónico: recomendación de productos a los clientes (*Amazon*).
- Entretenimiento: recomendaciones para películas, música, IPTV y vídeos (YouTube).
- Contenido: recomendación de información (documentos, libros, etc.) recomendaciones de páginas web, aplicaciones educativas.
- Servicios: recomendaciones de servicios de viajes (lugares, alojamiento, actividades, etc.), recomendación de casas para alquilar o servicios de búsqueda.

Las definiciones antes mencionadas, la utilización de diferentes fuentes de datos y las principales aplicaciones de los SR evidencian el desarrollo alcanzado en la actualidad. Seguido (2009) en su tesis de maestría sintetiza esta evolución al señalar que “los sistemas de recomendación se distinguen por dos criterios fundamentales: por un lado se encuentran aquellos que producen recomendaciones individualizadas en forma de *"output"* y, por otro, aquellos que tienen el efecto de guiar al usuario de forma personalizada para los intereses de este dentro de un dominio con grandes cantidades de elementos posibles a elegir. Se puede afirmar que los sistemas de recomendación son un tipo específico de filtro de información cuyo objetivo es mostrar elementos (libros, artículos, películas, imágenes, información, etc.) al usuario que le sean relevantes o de interés“. Para lograr realizar la recomendación se necesita procesar los datos recolectados de diferentes fuentes utilizando alguna técnica de recomendación, estas técnicas determinan la clasificación de los SR.

1.7. Clasificación de los Sistemas de Recomendación

Varias han sido las clasificaciones de los SR a lo largo de toda su evolución, estas están enfocadas principalmente hacia el área de aplicación y la forma de obtener la información para realizar la recomendación. Dependiendo de la fuente de información y del enfoque o técnica empleada varios autores (Herlocker, 2004; Burke, 2007; Font, 2009; Ricci et al., 2010; Rodríguez, 2012; Formoso, 2013; Fernández, 2014) coinciden y a los efectos de la presente investigación, que entre las clasificaciones más extendidas de los SR se encuentran: Basados en Contenido, Filtrado Colaborativo, Demográficas, Basado en el Conocimiento, Basado en la Utilidad e Híbridos.

1.7.1. Basados en contenido (del inglés *Content based CB*)

Los sistemas de recomendación basados en contenido emplean técnicas de recuperación de información, utilizando las preferencias del usuario (perfil del usuario) y la descripción del *ítem*. Según Rodríguez (2012) las recomendaciones son realizadas basándose solamente en un perfil creado, utilizan algoritmos “*ítem a ítem*” generado mediante la asociación de reglas de correlación entre ellos. Para realizar la recomendación se debe tener en cuenta el ítem, el perfil del usuario y el algoritmo de recomendación; permitiendo hacer coincidir los atributos del perfil del usuario con los atributos de los *ítems* a recomendar (Hdioud et al., 2012).

Los SR Basados en Contenidos realizan la recomendación analizando un conjunto de datos de elementos previamente valorados por el usuario y construyen un perfil de preferencias con los intereses de los usuarios en base a las características de los objetos valorados anteriormente. Este proceso de recomendación (ver figura 2 Arquitectura de alto nivel de un Sistema de Recomendación Basado en Contenido) se realiza en tres etapas (Analizador de Contenido, Construcción del Perfil y Componente de Filtrado) y describen el proceso de recomendación de información utilizando un enfoque basado en contenido. A continuación se describen cada uno de los componentes y las relaciones entre cada elemento que lo componen planteadas en *Recommender Systems Handbook* (Lops et al., 2011).

Analizador de Contenido: Cuando la información no tiene una estructura (por ejemplo, texto), se necesita una especie de etapa de pre-procesamiento para extraer la información estructurada. La principal responsabilidad de este componente es representar el contenido de los productos (por ejemplo, documentos, páginas web, noticias, descripciones de productos, etc.) procedentes de fuentes de información en un formato adecuado para los siguientes pasos del procesamiento. Esta representación es la entrada a los componentes Construcción del Perfil y Componente de Filtrado.

Construcción del Perfil: Este módulo recoge datos representativos de las preferencias del usuario y trata de generalizarlos, con el fin de construir el perfil de usuario. Por lo general, la estrategia de generalización se realiza a través de técnicas de aprendizaje automáticas que son capaces de inferir un modelo de los intereses de los usuarios a partir de los elementos que le gustó o no en el pasado.

Componente de Filtrado: Este módulo explota el perfil del usuario para sugerir elementos relevantes, haciendo coincidir el perfil del usuario con los elementos que se desean recomendar utilizando técnicas de similitud.

Según plantea Lops et al (2011) el primer paso en el proceso de recomendación es el realizado por el analizador de contenido, que representa de forma estructurada la descripción de los elementos obtenidos de la fuente de datos empleada para realizar la recomendación.

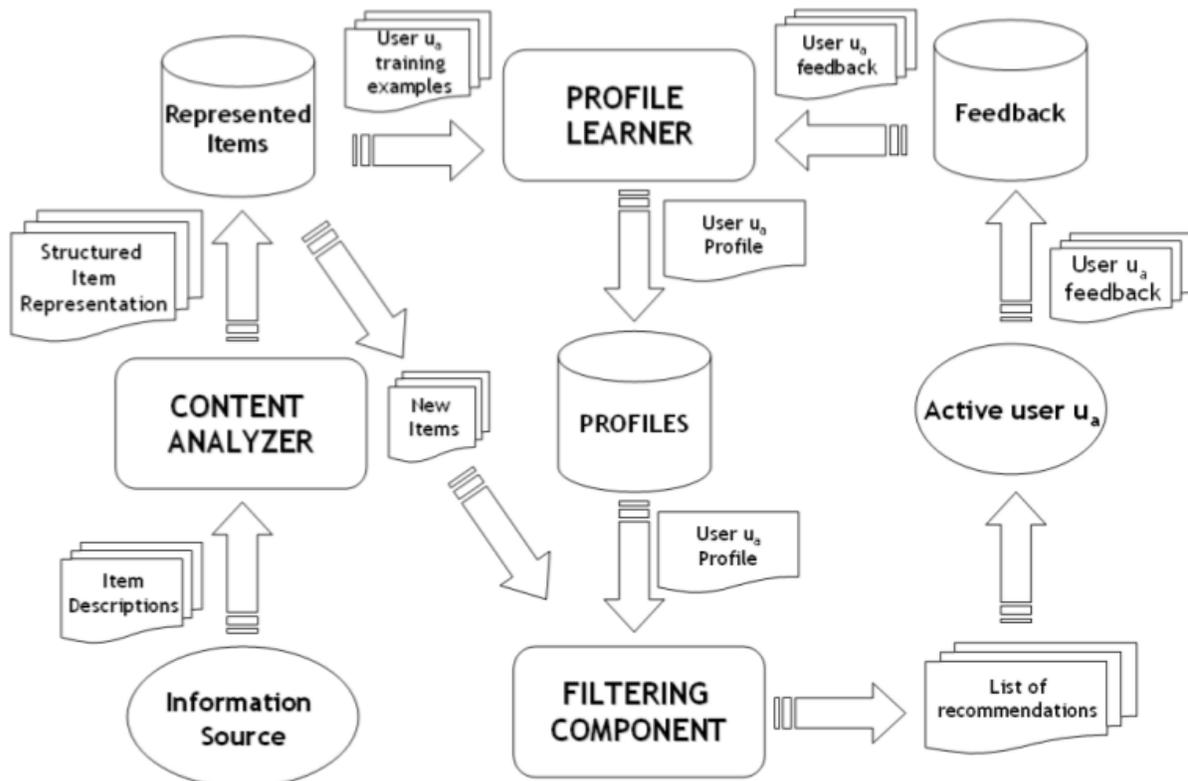


Figura 2. Arquitectura de alto nivel de un Sistema de Recomendación Basado en Contenido (Lops et al, 2011)

Posteriormente con el objetivo de construir y actualizar el perfil del usuario activo se necesita almacenar sus interacciones con los elementos en un repositorio de retroalimentación. Estas relaciones entre el usuario activo y los elementos se denominan anotaciones o retroalimentación, que junto a la representación de los elementos relacionados son explotados durante el proceso de construcción del perfil de preferencia del usuario para predecir la relevancia real de un nuevo elemento.

Los usuarios también pueden definir explícitamente sus áreas de interés a través de un

perfil inicial sin proporcionar retroalimentación. El componente construcción del perfil aplica algoritmos de aprendizaje supervisado que a partir del perfil de preferencias del usuario activo y la descripción de un nuevo elemento determina si este elemento es relevante para el usuario. El componente de filtrado permite seleccionar una lista de elementos con una relevancia para el usuario potencialmente alta. Las preferencias de los usuarios pueden cambiar con el tiempo, por lo tanto, se hace necesario mantener actualizado el perfil de preferencias del usuario.

Los sistemas basados en el contenido emplean algoritmos que aprenden las preferencias de los usuarios y filtran, de los *ítems* posibles, aquellos que más se asemejan a estas (Valero, 2010). Utilizan modelos de recuperación de información relativamente simples, entre estas técnicas según varios autores (Font, 2009; Castro, 2012) las más empleadas son:

Modelado Vectorial: Esta técnica procede del área de la recuperación de información. Para calcular las recomendaciones, el sistema calcula un vector asociado al usuario y obtiene la similitud (coseno, correlación) con los vectores asociados a los productos que el usuario no ha valorado. Finalmente se recomiendan los N productos cuya similitud con el perfil del usuario sea mayor (Font, 2009). Está basado en la utilización de espacios vectoriales de n dimensiones, es decir, con tantas dimensiones como elementos tenga el vector (Codina, 1995).

Redes Bayesianas: Una red bayesiana es un modelo probabilístico que relaciona variables aleatorias mediante un grafo no cíclico dirigido, cuyos nodos representan variables aleatorias y los arcos representan relaciones de dependencia entre las mismas. Este tipo de redes permite realizar inferencias para calcular la probabilidad de una variable condicionada al valor de las variables de las que depende (ascendiente) (Font, 2009).

Reglas de asociación: En este tipo de técnica se intenta descubrir relaciones existentes entre los elementos, que luego pueden ser utilizadas para recomendación. Estos se comparan en base al comportamiento pasado de los usuarios respecto a ellos. Está relacionada al descubrimiento de patrones en los datos. Es decir, descubre relaciones entre los atributos de una base de datos, produciendo sentencias de tipo si-entonces, relativas a los atributos y sus valores (Font, 2009). En las reglas de asociación la cantidad

de elementos por conjunto no necesariamente debe ser la misma (Monteserin, 2013). Las reglas de asociación son ampliamente utilizadas en el proceso de descubrimiento de tendencias o patrones. Este método fue ideado con el objetivo de buscar patrones interesantes (reglas) para el soporte en los procesos de toma de decisiones, permite la inclusión de diferentes tipos de valores de entrada (Monteserin, 2013).

1.7.1.1. Ventajas de un Sistema de Recomendación de Contenido

Independencia del Usuario (*User Independence*): Los SR Basados en Contenidos explotan únicamente las calificaciones proporcionadas por el usuario activo para construir su propio perfil.

Transparencia (*Transparency*): La explicación sobre cómo el sistema de recomendación obtuvo las recomendaciones que se muestran al usuario se pueden dar de manera explícita a partir de los elementos o las descripciones. Esas características son indicadores para decidir si confiar o no en la recomendación.

Elemento Nuevo (*New Item*): Los SR Basados en Contenidos son capaces de recomendar elementos que todavía no han sido calificado por ningún usuario.

1.7.1.2. Deficiencias de un Sistema de Recomendación de Basado en Contenido

Análisis de Contenido Limitado (*Limited Content Analysis*): Las técnicas basadas en contenido tienen un límite natural en el número y tipo de características que están asociadas, ya sea de forma automática o manualmente, con los elementos que ellos recomiendan. Con frecuencia se necesita conocer el dominio y las características de los elementos para determinar cuales atributos son relevantes para los elementos a recomendar.

Sobre Especialización (*Over Specialization*): Los SR Basados en Contenidos no tienen ningún método inherente para encontrar elementos inesperados. El sistema sugiere elementos cuyas valoraciones son altas cuando se compara con el perfil de usuario, por lo tanto, al usuario se le van a recomendar elementos similares a los ya valorados.

Nuevo Usuario (*New User*): El SR debe disponer de suficientes calificaciones para poder realmente entender las preferencias del usuario y ofrecer recomendaciones precisas. Por lo tanto, entre más calificaciones disponibles tenga el sistema mejores

serán las recomendaciones realizadas.

Existen varias estrategias para eliminar las desventajas de los SR Basados en Contenidos, la principal es la combinación con otros SR.

1.7.2. Filtrado colaborativo (del inglés *Collaborative Filter*)

Es una técnica que utiliza la información de preferencias y las calificaciones de un grupo de usuarios respecto a un conjunto de *ítems* con el objetivo de predecir o determinar la preferencia de un usuario sobre un *ítem*. Posibilitando generar una recomendación acertada a los demás miembros del grupo. Las recomendaciones se hacen basándose en el grado de similitud entre usuarios; los elementos que le gustan a un usuario pueden ser de interés de otros usuarios con gustos similares. Los SR colaborativos según el algoritmo que utilicen para realizar la recomendación se pueden clasificar en basados en la memoria y basados en el modelo (Gonzaga, 2008; Font, 2009; Pinho, 2010; Carrillo, 2011; Bermejo, 2011, Fernández, 2014); otros autores establecen como clasificación: algoritmos basados en usuario (*user-based algorithms*) o filtrado social y algoritmos basados en *ítems* (*item-based algorithms*) (Ruiz, 2009; Bermejo, 2011; Rodriguez, 2012; Almazro, 2010, Shahatah et al., 2010).

A los efectos de la presente investigación los algoritmos de recomendación colaborativos se clasifican en dos categorías generales: algoritmos basado en modelos y basados en memoria.

Los algoritmos basados en memoria utilizan todos los datos disponibles de una base de datos del sistema para calcular predicciones y recomendaciones.

- **Métodos basados en memoria:** emplean métricas de similitud para determinar la semejanza entre dos usuarios, estos algoritmos utilizan toda la base de datos para realizar la predicción. Para ello calculan los *ítems* que han sido votados por ambos usuarios y comparan dichos votos para calcular la similitud. La predicción de votos se fundamenta en la medida de un conjunto de pesos calculado mediante el uso de la base de datos de votaciones. Estos pesos pueden representar similitud, correlación o distancia entre el usuario activo y los demás usuarios de la base de datos. La diferencia entre los distintos sistemas de filtrado colaborativo

basados en memoria viene dada por el cálculo de estos pesos (Breese et al., 1992).

En contraste, los algoritmos basados en modelos operan derivando primero un modelo a partir de los datos del sistema, y este modelo es posteriormente utilizado en el proceso de recomendación.

- **Métodos basados en modelos:** utilizan la matriz de votaciones para crear un modelo a través del cual establecer el conjunto de usuarios similares al usuario activo. Utiliza técnicas de aproximación probabilísticas que calcula el valor esperado de la predicción del usuario dada la valoración realizada sobre otros elementos y calcula la similitud con el elemento activo. Algunos ejemplos de estos modelos son los clasificadores bayesianos, las redes neuronales, algoritmos genéticos, sistemas borrosos y la técnica de descomposición matricial basada en la técnica matemática del SVD.

En la literatura se han documentados una amplia gama de algoritmos (ver figura 3) utilizados en los SR Basados en Filtrado Colaborativo para realizar el procesamiento de la información entre los más utilizados se encuentran los algoritmos de los K vecinos.

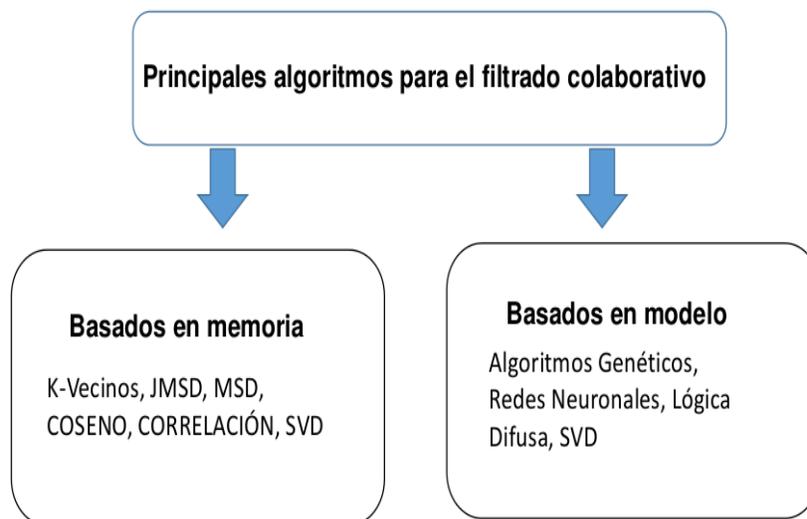


Figura 3. Algoritmos de filtrado colaborativo

El proceso general (ver figura 4) de un sistema de recomendación de filtrado colaborativo está representado por tres fases:

- **Cálculo de vecinos:** Se calcula la similitud de parejas de usuarios y se determina la vecindad del usuario activo, en relación a los usuarios con gustos o preferencias similares a las de él. Se obtiene los k usuarios más similares al usuario activo.
- **Predicción de la valoración de preferencia:** Se estima el valor de preferencia que el usuario activo daría a cada uno de los elementos que no ha valorado.
- **Recomendación de los mejores valorados:** Se ordena la lista de los elementos recomendados por su valor de preferencia predefinido y se recomiendan los N primeros elementos.



Figura 4. Proceso de recomendación en un sistema de filtrado colaborativo

Después de realizar un estudio sobre los métodos basados en memoria y modelo se determinó utilizar el método basado en memoria. Debido principalmente a que los algoritmos que utilizan este tipo de métodos son más simples de calcular, requieren menos cantidad de información, la escala que se utiliza para valorar los elementos en el COJ es un rango de 1-5 estrellas y en la actualidad la información disponible en el COJ no está totalmente valorada.

Para los métodos basados en memoria, una vez definidos los datos con los que se debe trabajar, existen tres tareas a realizar en base a generar una predicción (Adomavicius et al., 2005):

- Establecer el valor de similitud entre el usuario activo y el resto de usuarios.
- Seleccionar un conjunto de usuarios que se usarán para la predicción.
- Normalizar las valoraciones y generar una predicción.

Establecer el valor similitud entre vecinos

Para establecer la similitud entre los vecinos del usuario activo se debe definir una medida de similitud que permita evaluar el grado de semejanza entre los usuarios. Varios autores (Herlocker, 2004; Ruiz, 2009; Font, 2009; Albín, 2009) coinciden que entre las principales medidas de similitud utilizadas en los sistemas de recomendación de información se encuentran:

Coefficiente de correlación de Pearson: es una medida de relación entre dos valores (usuarios); es un índice que mide la relación lineal entre dos variables siempre y cuando ambas sean cuantitativas, el resultado es un valor en el intervalo $[-1,1]$ donde la relación entre más cercana a 1 es una correlación positiva perfecta, cuando es más cercana a -1 es correlación negativa perfecta y si es igual a 0 no existe correlación entre los usuarios. Se define en la ecuación:

$$s(a, u) = \frac{\sum_{i \in J(a,u)} (r_{ai} - \bar{r}_a.) (r_{ui} - \bar{r}_u.)}{\sqrt{\sum_{i \in J(a,u)} (r_{ai} - \bar{r}_a.)^2 \sum_{i \in J(a,u)} (r_{ui} - \bar{r}_u.)^2}}$$

Donde J es el conjunto de elementos valorados por ambos usuarios (a y u), r_{ui} es la valoración del usuario u para el elemento i .

$\bar{r}_u.$ la valoración media dada por el usuario u .

$\bar{r}_a.$ promedio de los votos del usuario actual.

$\bar{r}_a.$ es la valoración media dada por el usuario a .

Coefficiente Coseno: mide la distancia que existe entre dos usuarios en función del ángulo que forman entre ellos. Se considera cada elemento como un vector dentro de un espacio vectorial de m dimensiones y se calcula la similitud como el coseno del ángulo que forman tomando dos vectores ($a;u$). El resultado es un valor en el intervalo $[0,1]$

donde la relación entre más cercana a 1 mayor similitud existe entre los usuarios. Se define en la ecuación:

$$s(a, u) = \frac{\sum_{j \in \mathcal{J}(a, u)} r_{aj} r_{uj}}{\sqrt{\sum_{k \in \mathcal{J}(a)} r_{ak}^2} \sqrt{\sum_{k \in \mathcal{J}(u)} r_{uk}^2}}$$

Para calcular la similitud también existen otros algoritmos como son Coeficiente de Pearson forzado, Coeficiente de *Pearson* ponderado, Coseno ajustado, Diferencia Cuadrática Media (Herlocker, 2004; Rodríguez, 2012; Fernández, 2014).

Selección de los K-vecinos

Después de calculada la similitud entre el usuario activo y el resto de los usuarios se selecciona entre todos los usuarios similares cuales se utilizarán para calcular las predicciones y generar una recomendación. Para calcular la similitud se plantean diferentes técnicas:

Umbral de correlación: Consiste en seleccionar solo aquellos usuarios cuya similitud con el usuario activo sobrepase un determinado umbral.

Máximo número de vecinos: Consiste en seleccionar los k usuarios que sean más similares al usuario activo, donde k será un parámetro del algoritmo.

A los efectos de la presente investigación uno de los métodos más utilizados es el de los vecinos más cercanos (Máximo número de vecinos o kNN) el cual será implementado para seleccionar los K vecinos más cercanos al usuario activo. Una vez calculado el conjunto de elementos más similares para cada uno de los elementos de la base de datos se hace necesario predecir que elementos se le deben recomendar al usuario activo.

Algoritmos de Predicción

De los algoritmos que se utilizan para realizar la predicción según varios autores (Sarwar et al., 2001; Papagelis, 2004; Herlocker, 2004; Ruiz, 2009) y a los efectos de la presente investigación los más empleados en los SR son:

Suma ponderada (*Weighted sum*): Calcula la predicción de un elemento i por parte del usuario activo u como la suma de las evaluaciones del usuario u sobre elementos similares a i . Cada una de estas evaluaciones está ponderada por la correspondiente similitud $s(i, j)$ entre los elementos i y j . Se puede expresar esta técnica mediante de la siguiente ecuación (Sarwar et al., 2001):

$$p(u, i) = \frac{\sum_{n=1}^k s(u, x) v(x, i)}{\sum_{n=1}^k |s(u, x)|}$$

Indicando k los k elementos más similares al elemento i . Esencialmente, esta técnica intenta captar cómo evalúa el usuario activo a los elementos vistos por los usuarios similares y que no hayan sido vistos por él. Es necesario ponderar estas evaluaciones con la similitud para asegurarse que la predicción entra dentro del rango previamente definido.

Suma media ajustada (*Weighted adjusted sum*): Esta técnica presupone que una predicción para un usuario activo sobre un elemento es igual al valor medio de votaciones del usuario activo más un ajuste, que viene a ser la suma ponderada de las evaluaciones hechas por el usuario similar a él y su respectiva similitud con el usuario activo (Papagelis, 2004).

1.7.2.1. Limitaciones de los sistemas de contenidos colaborativos

Cobertura limitada: La similitud entre dos usuarios se determina solo para aquellos usuarios que hayan valorado los mismos elementos. Esta suposición es muy limitante ya que los usuarios pueden haber valorado muy pocos o ningún elemento en común y aun así pueden ser similares. Otro problema es que solo los elementos valorados por los vecinos serán aquellos que un futuro podrán ser recomendados.

Sensibilidad a la escasez de datos: Es un problema común en la mayoría de los sistemas de recomendación propiciado por el hecho de que los usuarios normalmente solo valoran una pequeña proporción de los elementos disponibles. Esto se intensifica para los elementos recién añadidos al sistema, por lo general no tienen ninguna valoración en lo absoluto, problema que se conoce como arranque en frío.

1.7.3. Demográficas (del inglés *Demographic*)

Las recomendaciones se realizan basándose en la información demográfica del usuario (edad, género, nacionalidad).

1.7.4. Basadas en el conocimiento (del inglés *Knowledge-based*)

Las recomendaciones se realizan en base a la utilidad o la necesidad del *ítem* para el usuario. Se recomiendan *ítem* basados en el conocimiento de un dominio específico de cómo ciertas características de productos satisfacen las necesidades y preferencias de los usuarios y, en última instancia, cómo el producto es útil para el usuario.

1.7.5. Híbridos (del inglés *Hybrid*)

Un SR Híbrido es aquel que combina múltiples técnicas en un único sistema logrando aprovechar los beneficios de la combinación de más de un método de recomendación. Aunque se pueden combinar técnicas utilizando el mismo enfoque, se obtienen mejores resultados cuando se emplean diferentes tipos de métodos. En (Burke 2007) se definen siete métodos para combinar estrategias de recomendación:

Ponderado (*Weighted*): El resultado de las valoraciones de las diferentes técnicas de recomendación que componen el sistema se combinan y se obtiene una puntuación única. La salida de cada una de estas técnicas se combinan utilizando una función ponderada lineal. El elemento con mejor puntuación será el que se ofrezca al usuario.

Conmutados (*Switching*): Se emplea un criterio de selección para determinar que técnica utilizar en cada momento. Existen dos métodos para realizar la recomendación, la primera, analiza los resultados obtenidos por los SR y selecciona que resultados mostrar; mientras que el segundo método selecciona primero la técnica de recomendación antes que la información.

Combinado (*Mixed*): Se procesan simultáneamente diferentes técnicas y se les muestra

una única recomendación al usuario.

Combinación de características (*Feature combination*): La información obtenida de una o más técnicas de recomendación (habitualmente colaborativa) se utiliza como una característica adicional asociada a los datos. Sobre estas características, combinadas con la información general, se utiliza otra técnica de recomendación (Seguido, 2009).

Cascada (*Cascade*): Una técnica refina las recomendaciones realizadas por otra técnica siguiendo un proceso secuencial, donde la entrada de la segunda técnica es la salida de la primera.

Potenciación de características (*Feature augmentation*): Las salidas de una de las técnicas es utilizada como entrada por otra técnica solo considerada como una característica más.

Meta nivel (*Meta-level*): El modelo aprendido es utilizado como entrada de otra técnica. Los SR híbridos, son una solución a las limitaciones de los SR basados en contenidos y de filtrado colaborativo, donde se busca aprovechar los beneficios de las combinaciones de ambos métodos. Las variantes ponderado, combinado y combinación de características procesan los resultados de recomendaciones de diferentes métodos; otros utilizan a la vez diferentes tipos de datos como la combinación de características, y otras son la unión de dos o más SR como el cascada, aumento de características y meta nivel.

Después de analizar cada una de las técnicas planteadas se determina que por la gran cantidad de datos, de usuarios y las relaciones que se establecen entre ellos se hace necesario utilizar más de una técnica para realizar recomendaciones más efectivas, ajustadas a las necesidades y preferencias del usuario y que se pueda potenciar las habilidades individuales. Para ello se selecciona desarrollar un procedimiento que utilizando un sistema de recomendación híbrido en cascada pueda mitigar las deficiencias de los SR basados en contenidos y filtrado colaborativo. El procedimiento planteado debe ser evaluado utilizando las métricas de evaluación para determinar la calidad de las recomendaciones que se realizan a los usuarios, a continuación de describen estas métricas.

1.8. Métrica de evaluación de Sistemas de Recomendación

Para realizar la evaluación de un sistema de recomendación se pueden utilizar diferentes métricas que permitan determinar la exactitud de las predicciones, la tendencia general es medir la precisión de los algoritmos. Según Herlocker (2004) una métrica para la exactitud de un sistema de recomendación mide, empíricamente, que tan cerca está el ordenamiento de *ítems* predichos para un usuario por el sistema con respecto al ordenamiento verdadero que el usuario haría, según su preferencia, de los mismos *ítems* (Herlocker, 2004). Además (Rodríguez, 2012; Fernández, 2014) se plantea que las métricas de evaluación pueden categorizarse en tres grupos: métricas de exactitud predictiva, métricas de exactitud en clasificación y métricas de exactitud en ordenamiento.

1.8.1. Métricas de exactitud predictiva

Son aquellas métricas que miden qué tan cerca están las puntuaciones predichas por el sistema con respecto a las puntuaciones reales dados por el usuario. Varios autores (Salazar, 2006; Rodríguez, 2012; Fernández, 2014) coinciden que entre las principales métricas de exactitud predictiva se encuentran:

Error Medio Absoluto (*Mean Absolute Error*): Esta métrica mide la media de la diferencia en valor absoluto entre el valor predicho por el algoritmo y la puntuación real dada por el usuario, cuanto más pequeño sea el error mayor es la eficiencia del Sistema de Recomendación. Debido a su facilidad de cálculo y sus propiedades estadísticas (Herlocker et al., 2004) se ha convertido en una de las métricas más populares para evaluar sistemas de recomendación. Se define en la ecuación:

$$MAE = \frac{\sum^N |r_{ui} - \hat{r}_{ui}|}{N}$$

r_{ui} es la predicción, \hat{r}_{ui} es el valor de preferencia real y N es el número de predicciones que se calculan.

Error Medio Cuadrático (*Root Mean Squared Error*): Esta métrica de precisión en la predicción da más valor a errores grandes, puesto que pueden ser los que más influyan en la decisión del usuario. Se define en la ecuación:

$$RMSE = \sqrt{\frac{\sum^N (r_{ui} - \hat{r}_{ui})^2}{N}}$$

Existen otras métricas como Error Medio Absoluto Normalizado.

1.8.2. Métricas de exactitud en clasificación

Las métricas de exactitud, también llamadas métricas de decisión, evalúan la efectividad de un sistema de predicción, ayudando al usuario a seleccionar los elementos de mayor calidad. Para calcular la exactitud se utiliza la siguiente nomenclatura:

	Recomendados	No recomendados
Relevante	Verdadero Positivo (VP)	Falso Negativo (FN)
No relevante	Falso Positivo (FP)	Verdadero Negativo (VN)

Las principales métricas son:

Precisión (*Precision*): es la probabilidad de que realmente el elemento recomendado sea útil para el usuario. Es el ratio entre los productos que el algoritmo considera relevantes y que realmente lo son, entre el número total de productos que para el algoritmo son relevantes. Se define en la ecuación:

$$P = \frac{VP}{VP + FP}$$

Exhaustividad (*Recall*): Representa el número de productos relevantes recomendados frente al número total de productos relevantes. Se define en la ecuación:

$$R = \frac{VP}{VP + FN}$$

Las métricas Precisión y Exhaustividad están inversamente relacionadas. Cuando aumenta el tamaño de la lista de elementos recomendados también lo hace la Exhaustividad y disminuye la Precisión. Por tanto es necesario tener en cuenta la relación entre ambas métricas para evaluar la calidad de un algoritmo en cuanto a la clasificación.

Las curvas ROC (*Receiver Operating Characteristic*): Según (Herlocker et al., 2004) esta métrica representa gráficamente la capacidad que tiene un determinado algoritmo de distinguir productos relevantes de los que no lo son. Es necesario establecer un umbral de relevancia de tal modo que los productos cuya puntuación sea mayor que el umbral se consideren relevantes, y no relevantes en otro caso.

1.8.3. Métricas de exactitud en ordenamiento

En las métricas de exactitud de ordenamiento están aquellas que miden la experiencia de un Sistema de Recomendación para producir un ordenamiento de *ítems* recomendados que coincida con el ordenamiento que un usuario haría de los mismos *ítems*. Entre estas métricas están la correlación predicción-evaluación, métrica de utilidad *half-life* y la medida NDPM.

Utilidad de vida media (*Half-life utility*): Se basa en la suposición de que los productos al principio de la lista de recomendación tienen mayor probabilidad de ser consultados por el usuario, probabilidad que decrece exponencialmente según se baja en la lista.

1.8.4. Otras métricas

Varios autores (Herlocker et al., 2000; Ziegler et al., 2005; Konstan et al., 2006; Fornoso, 2011) recomiendan utilizar otras métricas dependiendo del contexto para medir la exactitud de las recomendaciones, entre ellas se encuentran:

Cobertura (*coverage*): Mide la capacidad del sistema para generar recomendaciones o predicciones. En el caso de la recomendación, se puede reportar la cobertura en el espacio de usuarios (es decir, el porcentaje de usuarios para los cuales el sistema es capaz de recomendar productos), y en el espacio de productos (es decir, el porcentaje de productos que el sistema es capaz de recomendar) (Fornoso, 2011).

Recomendaciones novedosas o inesperadas: La novedad de una recomendación (Konstan et al., 2006) mide el grado de familiaridad de un usuario con la recomendación del sistema, es decir, si el usuario ya conoce los productos recomendados. Una posible mejora a esta métrica es el considerar no solo si un producto es o no conocido por el usuario, sino también si es inesperado.

Confianza: También es importante conseguir que el usuario confíe en los resultados del sistema. Errores en la recomendación pueden hacer que el usuario deje de confiar en las recomendaciones del sistema, pero limitarse siempre a recomendar productos muy similares al perfil del usuario, limita el grado de novedad y utilidad del sistema. El sistema puede incorporar a la recomendación información adicional que indique al usuario el riesgo de ciertas recomendaciones, o bien explicar al usuario el porqué de una determinada recomendación (Herlocker et al., 2000).

Diversidad: Mide la diferencia entre los productos recomendados. Lo ideal es un sistema que recomiende productos variados para que el usuario tenga diferentes opciones para elegir (Ziegler et al., 2005).

Eficiencia y escalabilidad: Miden la eficiencia y la escalabilidad de los algoritmos de recomendación es un factor clave para su uso en entornos reales.

1.9. Conclusiones parciales

Los referentes teóricos sobre la búsqueda y recuperación de información, la recomendación de información y los jueces en línea evidenció la importancia y necesidad de desarrollar un procedimiento que permita analizar, clasificar y recomendar problemas en el COJ.

Las tres herramientas con enfoque de recomendación empleadas en los jueces en líneas estudiados tienen como principal insuficiencia que no cumplen con el principio de los sistemas de recomendación, siendo necesario recomendar información ajustada al perfil del usuario.

Se analizaron las principales técnicas de recomendación de información lo que permitió identificar las principales características de los sistemas de recomendación, seleccionándose un Sistema de Recomendación de información híbrido en cascada utilizando como base un SR Colaborativo y después un SR Basado en Contenidos.

Se identificó la fuente de datos que se utilizará en las recomendaciones y los elementos que conformarán el perfil de preferencias del usuario.

Se seleccionaron las métricas de exactitud en la predicción y exactitud en clasificación para validar la eficiencia y calidad de las recomendaciones realizadas.

Capítulo 2: Procedimiento para analizar, clasificar y recomendar problemas en el Juez en Línea Caribeño

En el presente capítulo se presentan los componentes que sustentan el procedimiento para analizar, clasificar y recomendar problemas en el Juez en Línea Caribeño. Se describe y fundamenta el proceso de recomendación (Análisis, Procesamiento y Recomendación), así como los componentes de cada técnica de recomendación. Se realiza un análisis detallado de cada fuente y de las técnicas empleadas.

2.1. Estructura general de la propuesta



Figura 5. Descripción general del procedimiento

El procedimiento para recomendar problemas en el COJ está sustentado en el proceso de recomendación empleado por los principales Sistemas de Recomendación analizados. Como se plantea en la estructura general (ver figura 5) el procedimiento se ha dividido en tres componentes fundamentales (entrada, procesamiento y salida).

Las entradas (Perfil de preferencias del usuario e información a procesar): definen los elementos relevantes del usuario para construir su perfil de preferencias y además se seleccionan los metadatos de los problemas para realizar el procesamiento y posterior selección de los elementos relevantes para el usuario.

El procesamiento (SR basado en filtro colaborativo, SR basado en contenido y la combinación de ambas técnicas en un SR Híbrido en cascada): define cada uno de los pasos a seguir para implementar el sistema de recomendación Híbrido en cascada.

Salidas (listado de las recomendaciones): Muestran un número reducido de recomendaciones ajustadas a las preferencias del usuario.

La propuesta general para realizar la recomendación está enfocada en dos técnicas en función del tipo de usuario al que se le recomienda la información. La primera de ellas al usuario anónimo utilizando un SR Basado en Contenido recomendando los problemas mejor valorados por los usuarios registrados. La segunda emplea la combinación de una técnica híbrida en cascada donde primero se procesan los datos utilizando un SR Basado en Filtrado Colaborativo, estos resultados son la fuente de entrada del SR Basado en Contenido permitiendo ajustar la técnica y procesar la información simplificando los datos que se mostrarán al usuario registrado.

2.2. Fuente de datos y de conocimientos

Como se explicó en el Capítulo 1 la fuente de datos es uno de los componentes más importantes en los sistemas de recomendación ya que aglutinan los elementos necesarios para realizar la recomendación. Los principales elementos de una fuente de datos son los *Items* (información disponible en el COJ), los Usuarios (usuarios registrados) y las relaciones que se establecen entre ellos (problemas resueltos, mejores soluciones, valoraciones de problemas).

Los elementos (*items*) que más información brindan en el COJ son los problemas publicados y sus principales componentes son:

Metadatos de los problemas

- Descripción
- Especificación de entrada
- Especificación de salida
- Ejemplo de entrada
- Ejemplo de salida
- Sugerencia(s)
- Creado por
- Adicionado por
- Fecha de adición
- Tiempo límite (ms)
- Prueba límite (ms)
- Memoria límite (kb)
- Salida límite (mb)
- Tamaño límite (bytes)
- Lenguajes activados

Además están los datos del usuario que conforman su perfil, estos datos se dividen en datos implícitos y datos explícitos:

- Nombre de usuario
- Alias
- Nombre
- Apellidos
- Correo-e
- País (Selección)
- Institución (Selección)
- Lenguaje de la Interfaz (Selección)
- Lenguaje de Programación (Selección)
- Contraseña

- Confirmación de la Contraseña
- Género (Selección)
- Fecha de nacimiento (Selección)
- Mostrar fecha de nacimiento (Marcar)
- Mostrar correo-e (Marcar)
- Recibir notificaciones vía correo-e (Marcar)
- Pregunta de seguridad (Selección)
- Respuesta de seguridad

Aunque estos datos implícitos no abarcan toda la información que el usuario puede brindar sobre sus preferencias y habilidades en la resolución de problemas, posibilita crear las bases necesarias para procesar la información. También se analizan las preferencias que se pueden inferir sobre el historial (problemas resueltos, valorados) del usuario en el COJ que brindan de forma detallada más información sobre las habilidades y las preferencias del usuario.

2.3. Representación de la información

Con el fin de construir y actualizar el perfil del usuario activo su interacción con los elementos disponibles en el COJ se deben almacenar en el sistema. La tarea del componente de Representación de la información radica en:

- Representación de los ítems (usuario o problemas).
- Determinar y almacenar los k-vecinos para cada ítem.
- Representación de las interacciones y perfiles de usuarios.

Estas interacciones, llamadas anotaciones o retroalimentación, junto con las descripciones de los elementos relacionados, son explotadas en el proceso de aprendizaje de un modelo útil para predecir la relevancia real de los elementos recién presentados. Por lo general, es posible distinguir entre dos tipos de retroalimentación por relevancia: la información positiva (inferir características del gusto del usuario) y de la información negativa (inferir que no le gusta al usuario).

Existen dos técnicas que se pueden utilizar para almacenar la retroalimentación de los usuarios. La primera cuando el sistema requiere que el usuario evalúe explícitamente los elementos (problemas publicados mediante la asignación de un valor mediante el voto), esta técnica se conoce generalmente como "retroalimentación explícita". La otra técnica,

llamada "retroalimentación implícita", no requiere ninguna intervención del usuario activo, la regeneración se deriva del seguimiento y análisis de las actividades del usuario en el COJ.

Las evaluaciones explícitas indican qué tan relevante o interesante es un elemento para el usuario (Ricci et al., 2010). Hay tres enfoques principales para obtener retroalimentación relevancia explícita:

- gusta / disgusta - los elementos se clasifican como "relevante" o " no relevante" mediante la adopción de un sencilla escala de clasificación binaria, como en (Ricci et al., 2010).
- Calificación: se adopta una escala numérica discreta para valorar el elemento, puede ser mediante la utilización de estrellas. Se aplica esta escala en el COJ para valorar los problemas.
- Texto comentarios: Los comentarios acerca de un solo tema se recogen y se presentan a los usuarios como un medio de facilitar el proceso de toma de decisiones.

2.4. Representación de los elementos

Los elementos que se pueden recomendar a un usuario están representados por un conjunto de características, también llamado atributos o propiedades. Estos atributos son las opciones seleccionadas en su perfil de usuario y los metadatos y las valoraciones de los problemas publicados en el COJ.

Primeramente se construye y almacena el perfil de preferencias del usuario. Para construir el perfil de preferencias del usuario, se utilizan dos formas o métodos en la recolección de características (implícitas o explícitas).

Explícita: Se recolectan los datos mediante las valoraciones realizadas por los usuarios sobre los problemas publicados en el COJ.

Implícita: El sistema obtiene retroalimentación implícita capturando la interacción del usuario sin que él lo note.

Posteriormente se representa como un vector con pesos sobre las descripciones de los elementos. Y se obtiene un modelo de los elementos que son de interés para ese usuario. Cuando se realiza la representación de los elementos se almacena la información relacionada con los problemas resueltos, los votos emitidos sobre los problemas

valorados, la cantidad de intentos para resolver un problema y la fecha de la última vez que se actualizó su perfil de preferencia. Después de representados los datos se procede a procesar la información utilizando primero un SR basado en filtrado colaborativo.

2.5. Procesamiento (Motor de recomendación)

Se emplean las técnicas para procesar las entradas (fuente de información) y realizar la recomendación de los elementos más relevantes para el usuario activo. Se utiliza primero un Sistema de Recomendación Basado en Filtrado Colaborativo y a continuación un Sistema de Recomendación Basado en Contenidos.

2.5.1. Componente filtrado colaborativo

Como se plantea (1.5.6 **Filtrado colaborativo**) el filtrado colaborativo es una técnica que utiliza la información de preferencias y las calificaciones de un grupo de usuarios respecto a un conjunto de *ítems* con el objetivo de predecir o determinar la preferencia de un usuario sobre un *ítem*. Posibilitando generar una recomendación acertada a los demás miembros del grupo. Las recomendaciones se hacen basándose en el grado de similitud entre usuarios; los elementos que le gustan a un usuario pueden ser de interés de otros usuarios con gustos similares.



Figura 6. Proceso de recomendación en un SR Basado en Filtrado Colaborativo

Según plantean (Shardanand y Maes; Herlocker et al., 2004) se pueden definir tres etapas fundamentales en el funcionamiento de un sistema de recomendación colaborativo:

1. **Cálculo de la similitud:** se mide el grado de similitud entre el usuario activo y el resto los usuarios del COJ y se crean grupos de usuarios con características similares.

Para seleccionar los usuarios con gustos similares se utilizó el Coeficiente de correlación de Pearson. Se determinan un listado (ver figura 7) de usuarios similares al usuario activo. Se seleccionan los K vecinos más similares al usuario activo.

		productos				
						
usuarios						
						
						

Figura 7. Cálculo de la similitud en algoritmos basados en usuario

2. **Selección de los vecinos:** se selecciona el conjunto de usuarios más similares al usuario activo. Para seleccionar los usuarios más similares al usuario activo se utilizó **Máximo número de vecinos**.

3. **Cálculo de la predicción:** el sistema calcula la predicción con la información obtenida en los pasos anteriores. A cada usuario se le recomendarán problemas que no ha resuelto anteriormente y que obtuvieron los mayores valores para dicha predicción. Para realizar la predicción se utiliza **Suma media ajustada**, porque los usuarios pueden utilizar diferentes escalas de puntuación sobre el mismo elemento y este algoritmo brinda una valoración más acorde a las preferencias del usuario.

Después de calcular la predicción se obtiene un listado de recomendaciones para cada usuario ajustado a sus preferencias. Estas recomendaciones son la entrada al sistema de recomendación basado en contenidos.

2.5.2. Motor de recomendación basado en contenido

Siguiendo el mismo principio de los sistemas de recomendación basados en contenido primero se analizan las fuentes de datos (recomendaciones realizadas por el componente de recomendación basado en filtrado colaborativo) y el perfil de preferencia del usuario activo. Los sistemas de recomendación basados en contenido emplean técnicas de recuperación de información, utilizando las preferencias del usuario (perfil del usuario) y la descripción del *ítem (problemas)*. Como se explicó anteriormente la entrada de esta técnica es el resultado obtenido de aplicar filtrado colaborativo, permitiendo disponer de un perfil de preferencia con las recomendaciones realizadas por este enfoque. Inicialmente se representan los datos utilizando el perfil de preferencia del usuario (Tabla 2.) y la descripción de los problemas (Tabla 3.) publicados en el COJ.

Tabla 2. Perfil de preferencia del usuario

Nombre del usuario	Preferencias	ID de los problemas resueltos
Nombre del usuario 1	Valor de los votos emitidos, preferencias sobre los tipos de problemas que le gustan.	1001, 1002, 1003,1004
Nombre del usuario 2	Valor de los votos emitidos, preferencias sobre los tipos de problemas que le gustan.	1001, 1007, 1003,1004
Nombre del usuario 3	Valor de los votos emitidos, preferencias sobre los tipos de problemas que le gustan.	1001, 1017, 1003,1020
Nombre del usuario 4	Valor de los votos emitidos, preferencias sobre los tipos de problemas que le gustan.	1001, 1002, 1003,1004
		2.5.3.

Tabla 3. Representación de los problemas

Nombre del problema	Descripción	Categoría	Complejidad
1002 - <i>New House</i>	<i>Johan wants to build a new house and he wants his house as large as it can. Given an $N \times N$ grid land, find the largest square size that fit in the free area.</i>	Fuerza Bruta	Alta
1004 - <i>Traversing Grid</i>	<i>Starting at the top left corner of an $N \times M$ grid and facing towards the right, you keep walking one square at a time in the direction you are facing. If you reach the boundary of the grid or if the next square you are about to visit has already been visited, you turn right. You stop when all the squares in the grid have been visited. What direction will you be facing when you stop? For example: Consider the case with $N = 3, M = 3$. The path followed will be $(0,0) \rightarrow (0,1) \rightarrow (0,2) \rightarrow (1,2) \rightarrow (2,2) \rightarrow (2,1) \rightarrow (2,0) \rightarrow (1,0) \rightarrow (1,1)$. At this point, all squares have been visited, and you are facing right.</i>	AD-Hoc	Media
1017 - <i>Three powers</i>	<i>Consider the set of all non-negative integer powers of 3. $S = \{1, 3, 9, 27, 81, \dots\}$ Consider the sequence of all subsets of S ordered by the value of the sum of their elements. The question is simple: find the set at the n-th position in the sequence and print it in increasing order of its elements.</i>	Álgebra	Media

Se calcula la similitud entre el perfil de preferencias del usuario y el listado de posibles elementos a recomendar utilizando la categoría de los problemas como medida de semejanza. La similitud se calcula en base al ángulo entre los dos vectores utilizando la

similitud del coseno. Mediante la función de similitud, se obtienen los elementos que son más similares a los gustos del usuario activo. Posteriormente se ordena la lista de posibles problemas a resolver por la relevancia obtenida en el paso anterior. Como último paso del proceso de recomendación se seleccionan los 5 problemas mejor valorados y que más se ajustan a las preferencias del usuario.

2.6. Módulo de Recomendación de problemas en el COJ

Para aplicar el procedimiento se implementó el módulo de recomendación de problemas que se integró al COJ. El módulo realiza recomendaciones a los usuarios dependiendo de sus preferencias. Para realizar la recomendación utiliza el procedimiento descrito en la presente investigación siguiendo un enfoque Híbrido en cascada. Los datos del perfil del usuario, su historial y la descripción de los problemas son la fuente de datos que se utiliza en cada una de las técnicas propuestas para analizar y clasificar los problemas, seleccionando los más relevantes para el usuario activo. El proceso de recomendación (ver figura 8.) se realiza cuando el usuario accede a uno de los problemas publicados en el sistema. El módulo de recomendación analiza la descripción del problema, el perfil de preferencia del usuario y la semejanza con los usuarios más similares y predice que problemas se ajustan más al usuario activo.

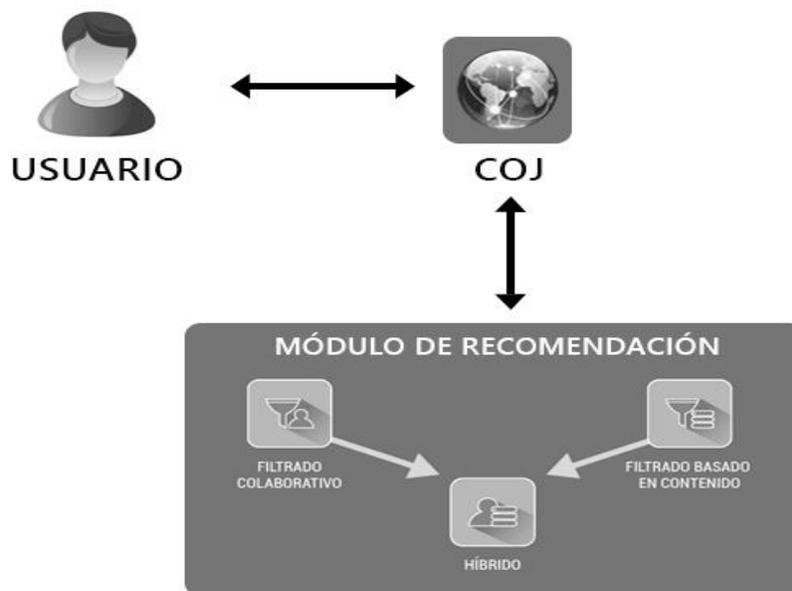


Figura 8. Funcionamiento del módulo de recomendación de problemas en el COJ

Después de realizada la recomendación para cada problema visitado por el usuario se procede actualizar el perfil de preferencias del usuario activo y se registra la fecha en la que se generó la última actualización. El módulo fue desarrollado siguiendo la arquitectura establecida en el COJ empleando el *framework Spring*, además se implementaron los algoritmos necesarios para calcular la similitud y los k vecinos a utilizar para realizar la recomendación al usuario activo.

2.7. Conclusiones del capítulo

- Se describe el procedimiento para realizar recomendaciones a los usuarios utilizando un SR híbrido en cascada, permitiendo mitigar las deficiencias de los SR basados en contenidos y SR colaborativos.
- El procedimiento propuesto permite crear un perfil de preferencias del usuario y representar la información disponible en el COJ para seleccionar los problemas más relevantes ajustados a las preferencias del usuario activo.
- Se implementó el módulo de recomendación que permite calcular la similitud y determinar el tamaño de la vecindad del usuario activo que se debe utilizar para realizar la recomendación.

Capítulo 3: Análisis del resultado de aplicar el procedimiento para recomendar problemas en el Juez en Línea Caribeño

En este capítulo se realiza la validación del procedimiento para recomendar problemas en el Juez en Línea Caribeño mediante la utilización de las métricas de exactitud predictiva (MAE) y exactitud en la clasificación (*Precision* y *Recall*), que permiten comprobar la efectividad de las recomendaciones obtenidas. Además se realiza un preexperimento para validar el comportamiento de los indicadores definidos en las variables operacionales y se expone el resultado de la validación.

3.1. Métricas para la validación del procedimiento

Para comprobar el funcionamiento y la validez del procedimiento para el análisis, clasificación y recomendación de problemas en el COJ, se realizó un experimento que permitió, mediante diferentes técnicas, medir la exactitud y el valor de las recomendaciones generadas. Para validar la exactitud y la precisión de las recomendaciones se puede utilizar la evaluación mediante un experimento *online*, *offline* o el estudio de usuario (Shani y Gunawardana, 2011).

Experimento *online*: se utiliza un entorno con usuarios reales para el experimento.

Experimento *offline*: se simula el comportamiento de los usuarios en un entorno real y se utiliza un conjunto de datos previamente almacenado. Este tipo de evaluación es rápida y económica, además permite validar el procedimiento antes de presentar el Sistema de Recomendación al usuario.

Experimento mediante estudio de usuario: se selecciona un reducido número de usuarios y se les pide que interactúen con el sistema y después se les pregunta sobre su experiencia en el mismo.

En la presente investigación se utiliza el método de evaluación *offline* por las ventajas que ofrece de probar el sistema y optimizar los resultados antes de mostrárselo al usuario final. Para realizar la validación se utilizan diferentes métricas (sección 1.5) las cuales evalúan la precisión de los algoritmos utilizados en el motor de procesamiento por cada uno de los enfoques empleados. El objetivo general de estas métricas es medir que tan

cerca están las recomendaciones ofrecidas al usuario con respecto al ordenamiento que él haría según sus preferencias.

Para validar la exactitud en la predicción se seleccionó un conjunto de datos compuesto por 1000 problemas, 1000 usuarios y un total de 70.000 calificaciones en una escala de 1 a 5 (1 = Muy malo, 2 = Malo, 3 = Regular, 4 = Bueno, 5 = Muy bueno) donde cada usuario ha evaluado al menos 700 problemas. Partiendo del principio que a un usuario se le deben recomendar problemas que se ajusten a su perfil de preferencias se asignaron valoraciones a los problemas resueltos por cada usuario en una escala de 4 a 5 estrellas simulando el voto que él usuario realizaría sobre los problemas que le gustan.

3.1.1. Métricas de exactitud predictiva

Se utilizan en la tarea de predicción y miden el error o diferencia entre la predicción realizada por el algoritmo y la puntuación real dada por el usuario. Para la realización de las pruebas utilizando la métrica de exactitud predictiva se utilizó MAE. Donde los valores que puede tomar el MAE, en este caso, van desde 0 hasta 1. Un MAE de 0 indicaría que las predicciones coincidieron perfectamente con el valor real. La tabla 4. muestra el resultado de aplicar el Error Absoluto Medio (MAE) calculado para diferentes valores de K (tamaño de la vecindad).

Tabla 4. Resultados de aplicar el Error Absoluto Medio

K	MAE
5	0,8357
10	0,8291
15	0,8023
20	0,7719
25	0,7492

También se analizó con qué valor de k el algoritmo obtendría menor MAE, por lo que cada iteración se repitió por cada valor de k , los cuales estaban en un intervalo de 5

hasta 25. Como se evidencia en la tabla 4. se obtiene un mejor MAE para K igual 25, que es cuando el valor de MAE está más cercano a 0.

3.1.2. Métricas de exactitud en clasificación

Se utilizan para evaluar la efectividad de un sistema de predicción, ayudando al usuario a seleccionar los elementos de mayor calidad. A los efectos de la presente investigación dentro del grupo de las métricas de exactitud según varios autores (Herlocker, 2004; Rodríguez, 2012; Fernández, 2014) las más utilizadas con Precisión y Exhaustividad y son las que se aplican para evaluar la efectividad de las recomendaciones. La precisión selecciona solo los elementos más relevantes mientras la *exhaustividad* selecciona todos los *elementos* relevantes para mostrárselos a los usuarios.

Como parte de la evaluación *offline*, se tienen los problemas que cada usuario ha resuelto y valorado con anterioridad. Para realizar la prueba, se selecciona un usuario específico, y se divide el conjunto de problemas asociados (sus problemas resueltos), en dos subconjuntos, llamados conjunto de pre-recomendaciones y conjunto de pruebas. Seguidamente al conjunto de pre-recomendaciones se le aplica el algoritmo a validar, y se seleccionan las N-primas recomendaciones generadas por este. A su vez, los elementos que aparecen tanto en el conjunto de pruebas como en el de las N-primas recomendaciones se consideran como parte de un conjunto llamado *hit set*. Dentro de este contexto, Precisión (*precision*) y Exhaustividad (*recall*) se definen de la siguiente manera (Cristache, 2009):

$$recall = \frac{\text{tamaño del hit set}}{\text{tamaño del conjunto de prueba}}$$

$$precision = \frac{\text{tamaño del hit set}}{N}$$

El conjunto de pruebas, para cada uno de los usuarios, estuvo conformado por el 20% del total de sus ejercicios resueltos, seleccionados de manera aleatoria. Para cada uno de los usuarios se ejecutaron varias corridas en las que se varió, desde 5 hasta 25 el

número de vecinos. A continuación se muestran los resultados obtenidos para K = 5 (ver Tabla 5.) rango inicial de la vecindad y K = 20 (ver Tabla 6.) cuando se obtuvo mejor precisión y exhaustividad en las recomendaciones brindadas por el procedimiento de recomendación de problemas.

Tabla 5. Resultados de aplicar Precisión y Exhaustividad, el procedimiento para K = 5

Usuario	Precisión	Exhaustividad
MarX	1	1
Raschach	0.91	0.83
humbertodiaz	0.98	0.92
Cyrax	0.90	0.83
dglima	0.95	0.83
eliogovea	0.99	0.91
teruel	0.95	0.85
MindFreaky	0.90	0.83
codeoverflow	0.88	0.83
jpg	1	1
huhumope	0.57	0.92
angelmh	0.74	0.83
kassandra	0.59	0.93
isaacvr	0.89	0.99
baphomet	0.81	0.83
leninx	0.73	0.83
rpgomez	0.99	0.91
Kazekage	0.73	0.83

Tabla 6. Resultados de aplicar Precisión y Exhaustividad, el procedimiento para $K = 20$

Usuario	Precisión	Exhaustividad
MarX	1	1
Raschach	0.92	0.83
humbertodiaz	0.99	0.92
Cyrax	0.91	0.83
dglima	0.95	0.83
eliogovea	1	0.91
teruel	0.95	0.85
MindFreaky	0.91	0.83
codeoverflow	0.89	0.83
jcj	1	1
huhumope	0.58	0.92
angelmh	0.74	0.83
kassandra	0.59	0.93
isaacvr	0.90	0.99
baphomet	0.82	0.83
leninx	0.73	0.83
rpgomez	1	0.91
Kazekage	0.73	0.83

En la tabla 7 se muestra el resumen de aplicar la métrica de *precision* y *recall* al procedimiento planteado en la investigación para el rango de K establecido en el experimento.

Tabla 7. Resumen de los resultados de aplicar Precisión y Exhaustividad

K	Precisión	Exhaustividad
5	0,848	0,876
10	0,851	0,876
15	0,852	0,876
20	0,857	0,876
25	0,852	0,867

Partiendo del supuesto que la *precision* es la posibilidad que un elemento relevante sea recomendado y que después de aplicar el procedimiento se obtuvo una precisión de 0,857 (86%) indica que de cada 10 recomendaciones 8.5 son buenas. Además se obtuvo para *recall* un valor de 0.876 demostrando que de cada 10 elementos recomendados 8.7 son relevantes. El análisis del resultado de aplicar las métricas de exactitud en clasificación determinó que para K = 20 (tamaño de la vecindad) se obtienen los mejores resultados. Es válido mencionar que los valores obtenidos de *precision* y *recall* están en correspondencia con los obtenidos por otros sistemas de recomendación como los descritos en (Guzmán, 2009; Castro, 2012; Rodríguez, 2012; Fernández, 2014).

3.2. Preexperimento para validar el procedimiento

Caracterización de la población y la muestra seleccionada

La población seleccionada en la investigación se corresponde con los 300 usuarios activos del COJ y que resolvieron al menos 1 problema en el período antes del 1 de julio de 2014 y posterior al 1 de julio de 2014 (cuando se desplegó el procedimiento).

La muestra seleccionada está compuesta por 72 usuarios del COJ que representan un 24.00% con respecto a su población.

3.2.1. Análisis de la variable dependiente

La variable dependiente fue operacionalizada a partir del promedio de intentos fallidos por ejercicios resueltos antes y después de desplegar el procedimiento. Con esta variable se pretende evaluar el incremento de las habilidades individuales de los usuarios del COJ mediante la disminución de los intentos fallidos en la resolución de problemas.

Para evaluar si los datos cumplían el supuesto de normalidad fue realizada la prueba de *Shapiro – Wilk*, resultando significativo con valor del estadígrafo $W = 0,7575$ y valor de probabilidad $p < 0.05$, lo que permitió rechazar la hipótesis de normalidad de los datos. Para realizar el análisis de los resultados obtenidos primeramente se aplicó la prueba *t de Student*, con el objetivo de conocer si existen diferencias significativas en la muestra y minimizar la cantidad de pruebas a realizar de ser significativas estas diferencias. Como consecuencia de este resultado se procedió a aplicar la Prueba no paramétrica de **rangos con signos** de **Wilcoxon**, donde se obtuvo valor del estadígrafo $W = 5.7633$ y probabilidad $p < 0.05$, lo que permite concluir que la mediana de los datos es diferente de cero, lo que significa que existen diferencias significativas entre las muestras de los promedios de los intentos aceptados antes y después de aplicar el procedimiento para recomendar problemas en el Juez en Línea Caribeño. El análisis de los datos evidenció una disminución considerable en el promedio de intentos fallidos por ejercicios resueltos en el período donde se aplicó el procedimiento para recomendar problemas en el Juez en Línea Caribeño.

3.2.2. Análisis de la variable independiente

La variable independiente fue operacionalizada a partir de la efectividad de las recomendaciones obtenidas. Con esta variable se pretende evaluar la calidad de las recomendaciones, midiendo la precisión y la exhaustividad en las recomendaciones realizadas a un usuario activo. Para comprobar estos indicadores se utilizaron las métricas de exactitud en clasificación, donde se obtuvo una precisión de 0,8535 (85%), indicando que de cada 10 recomendaciones 8.5 son buenas, además para exhaustividad un valor de 0.8794. Los resultados de precisión y exhaustividad permiten evaluar como Alta la calidad de las recomendaciones obtenidas después de aplicar el procedimiento.

3.3. Conclusiones del capítulo

- Se demostró la capacidad del procedimiento de brindar recomendaciones cercanas a las preferencias de los usuarios.
- Se demostró la calidad de las recomendaciones realizadas al usuario activo mediante los resultados obtenidos de aplicar las métricas de exactitud predictiva y exactitud en la clasificación.
- Se aplicó un preexperimento que permitió comprobar que disminuyó el promedio de intentos fallidos por problema aceptado.

Conclusiones Generales

- El procedimiento desarrollado permite combinar diferentes técnicas para implementar el módulo de recomendación de problemas en el COJ.
- El conjunto de métodos científicos utilizados para la validación del procedimiento permitió comprobar la calidad de las recomendaciones obtenidas.
- El procedimiento propuesto, desarrollado y validado permite recomendar problemas ajustados a las preferencias de los usuarios del COJ.

Recomendaciones

- Incluir en el procedimiento el procesamiento de información utilizando el enfoque basado en conocimiento.
- Aplicar algoritmos mejorados para predecir y realizar recomendaciones más ajustadas a las preferencias del usuario.
- Utilizar otras métricas que puedan medir la exactitud teniendo en cuenta la novedad, confianza y satisfacción del usuario con las recomendaciones realizadas.
- Implementar la métrica de evaluación exactitud en el ordenamiento.

Referencias Bibliográficas

- ACM Recommender Systems community. PACM RecSys 2013. [En línea]. 2013 [Consultado el: 12 de septiembre de 2013]. Disponible <http://recsys.acm.org/recsys13/>
- Álvarez Arrieta, F. TravelCobber Técnicas para la integración de recomendaciones en Facebook. Máster en Investigación en Informática de la Facultad de Informática, Facultad de Informática, Universidad Complutense de Madrid (UCM), Madrid, 2010.
- Albín Rodríguez, A. P.; Martínez López, Lu. y Pérez Cordón, L. G. 2009. Sistema de Recomendación Colaborativo basados en Algoritmos de Filtrado Mejorados. Jaén: Universidad de Jaén, 2009.
- Barraza-Urbina, A.; Carrillo Ramos, A. Vizier: un sistema de recomendación genérico y multidimensional. Revista Avances en Sistemas e Informática, 2011, Vol.8 No.3.
- Bermejo Rivera, J F. Sistema de Recomendación de Recursos Docentes. Tesis Doctoral, Departamento de Informática, Escuela Politécnica Superior de Jaén, Universidad de Jaén, Jaén, 2011.
- Betancur, D.; Moreno, J. y Ovalle, D., Modelo para la recomendación y recuperación de objetos de aprendizaje en entornos virtuales de enseñanza/aprendizaje. Revista Avances en Sistemas e Informática Vol 6 No 1. (2009)
- Breese, J. S.; Hecherman, D. y Kadie, C. 1998. Empirical Analysis of Predictive Algorithms for Collaborative Filtering. Paper read at Uncertainty in Artificial Intelligence. Proceedings of the Fourteenth Conference.
- Burke, R.: Hybrid web recommender systems. In: The Adaptive Web, pp. 377–408. Springer
Berlin / Heidelberg (2007)
- Burke. R. Hybrid recommender systems: Survey and experiments. User modeling and user-adapted interaction. 2002 pag: 331-370.

- Casali, A. et al. Sistema inteligente para la recomendación de objetos de aprendizaje. Revista Generación Digital Vol. 9 No. 1. Edición 16. (2011)
- Castells, P, et al. Recuperación y almacenamiento de información en la web, Escuela Politécnica Superior Universidad Autónoma de Madrid, 2011
- Castro Gallardo, J. Un nuevo modelo ponderado para Sistemas de Recomendación Basados en Contenido con medidas de contingencia y entropía. Departamento de Informática, Escuela Politécnica Superior de Jaén, Universidad de Jaén, Jaén, 2012.
- Carrillo G., Ochoa X. 2011. Recomendación de Objetos de Aprendizaje basado en perfil de usuario y la información de atención contextualizada. Disponible en <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/18962/6/gcarrilloTFM0113memoria.pdf>. Consultado en Mayo 2013.
- Cechinel, C. et al. Evaluating collaborative filtering recommendations inside large learning object repositories. Information Processing & Management. (2012)
- Corniel, M.; Gil, R., et al. Re-diseño de un sistema recomendador de estudios basado en minería web semántica. Revista Avances en Sistemas e Informática, 2011, Vol.8 No.2.
- Cristache, Alex. 2009. Hybrid recommender system using association rules. Auckland, New Zealand : s.n., 2009.
- Desrosiers, C.; Karypis, G.: A Comprehensive Survey of Neighborhood-based Recommendation Methods. (2011)
- Duran Dorado, D.; Arciniega Herrera, J. Algoritmo de Filtrado Híbrido-Mixto para Recomendación de Contenidos Audiovisuales a Comunidades Virtuales. Scientia et Technica Año XVIII, 2013, Vol.18 No.1.
- Espinilla, M et al., SR-REJA. Sistema de recomendación híbrido. Georeferenciado, Departamento de Informática Campus “Las Lagunillas”, Universidad de Jaén, (España), 2009, pp. 8.

- Fernández Iglesias, D. Uso de técnicas de recomendación en sistemas dispersos. Tesis Doctoral, Departamento de Tecnologías de la Información y las Comunicaciones, Universidad de la Coruña, Coruña, España. 2014.
- Galán Nieto, S. M. Filtrado Colaborativo y Sistemas de Recomendación, Inteligencia en Redes de Comunicaciones,5 Ingeniería de Telecomunicación, Universidad Carlos III de Madrid
- Guzmán Álvarez, C. A. Sistema de recomendación y planificación turística de la ciudad de Valencia vía web. Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, España. 2009.
- Hdioud, F., Frikh, B., y Ouhbi, B. (2012). A comparison study of some algorithms in Recommender Systems. Information Science and Technology (CIST), Colloquium, 130–135. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6388076.
- Herlocker, J. et al. (2004). "Evaluating collaborative filtering recommender systems." ACM Trans. Inf. Syst(5– 53).
- Howe, A.E., Forbes, R.D.: Re-considering neighborhood-based collaborative filtering parameters in the context of new data. In: CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge management, pp. 1481–1482. ACM, New York, NY, USA (2008)
- Junco Vázquez, T., et al. El Jurado en Línea, una nueva forma de ejercitación y evaluación en las asignaturas de programación. Ciudad de la Habana, Cuba: Informática 2009, 2009. ISBN-978-959-286-010-0.
- Junco Vázquez, T., et al. JELA: HERRAMIENTA DE APOYO A LA ENSEÑANZA DE LA PROGRAMACIÓN. Ciudad de la Habana, Cuba: Informática 2013, 2013. ISBN-978-959-7213-024.
- Lara Navarra, P. y Martínez Usero, J.Á.; (2004) *Agentes inteligentes en la búsqueda y recuperación de información*. Manual. Planeta UOC, Barcelona. (No publicado)

- Lobaina. J.C, Roque.J.L. Desarrollo de la versión 2 del Juez en línea Caribeño. 2012 [En línea] [Accedido 5 de diciembre del 2013] Disponible en: http://bibliodoc.uci.cu/RDigitales/2012/octubre/31/TD_05129_12.pdf
- Lops, P. et al.: Content-based Recommender Systems: State of the Art and Trends (2011)
- Martinez, L. "Modelos de recomendación con falta de información. Aplicaciones al sector turístico". Tesis Doctoral.
- Martínez Mimbrena, F.J. Sistema de Recomendación Actualizable y con Gestión de usuarios. Departamento de Informática, Escuela Politécnica Superior de Jaén, Universidad de Jaén, Jaén, 2008.
- Melville, P., Mooney, R.J., Nagarajan, R.: Content-boosted collaborative filtering for improved recommendations. In: 18th National Conf. on Artificial intelligence, pp. 187–192. American Association for Artificial Intelligence, Menlo Park, CA, USA (2002)
- Mondelo Hernández, Y. Caribbean online judge como herramienta de trabajo colaborativo y entrenamiento. La analítica web como base estratégica en el crecimiento del movimiento ACM-ICPC en el Caribe. En Congreso Internacional de Información, INFO2012. La Habana. Abril de 2012.
- Montiel Araque, D. Especificación de un Sistema de Recomendación basado en Etiquetado Social. Tesis de Maestría, UNIVERSIDAD DE HUELVA, Huelva, 2012
- Moya. R. SVD aplicado a Sistemas de recomendación basados en filtrado colaborativo. 2013 [En línea] [Accedido 9 de abril del 2014] Disponible en: <http://informatico.ricardomoya.es/proyectoMaster.html>.
- Moya. R, Invarato. R. Sistemas de recomendación basados en Filtrado Colaborativo (K-Vecinos): Jarroba. 2014b [En línea] [Accedido 25 de febrero del 2014] Disponible en: <http://jarroba.com/sistemas-de-recomendacion-basados-en-filtrado-colaborativo-k-vecinos/>.
- Núñez-Valdéz, E.; Joyanes Aguilar, L., et al. Plataforma de recomendación de contenidos para libros electrónicos inteligentes basada en el comportamiento de los usuarios. [En línea].2012 [Consultado el: 12 de junio de 2013]. Disponible en: technologyjournallac.files.wordpress.com/2012/02/25-40.pdf

- Ripoll Méndez, D. ACM-ICPC Caribe: principales resultados y perspectivas. Conferencia Magistral en el marco del Taller Virtualización de la Educación. Universidad 2014, La Habana, Cuba, 2014.
- Pérez Cordón, L.G. Modelos de recomendación con falta de información. Aplicaciones al sector turístico. Tesis Doctoral, Departamento de Informática, Escuela Politécnica Superior de Jaén, Universidad de Jaén, Jaén, 2008.
- Rodríguez Buitrago, O. Evaluación del Sistema de Recomendación de Patrones Pedagógicos (SRPP) en cursos de Geometría Euclidiana. Tesis de Maestría: Maestría en Enseñanza de la Matemática – Énfasis en Matemática Computacional, Departamento de Matemáticas, Universidad Tecnológica de Pereira, Pereira, 2012.
- Rodríguez Marín, P.A. Modelo de recomendación adaptativa de objetos de aprendizaje en el marco de una federación de repositorios, apoyado en agentes inteligentes y perfiles de usuario. Tesis de Maestría, Facultad de Minas, Departamento de Ciencias de la Computación y de la Decisión Medellín, Universidad Nacional de Colombia, Colombia, 2013.
- Pinho Lucas, J. Métodos De Clasificación Basados en Asociación Aplicados a Sistemas de Recomendación. Tesis Doctoral, Departamento de Informática y Automática, Universidad de Salamanca, Salamanca, 2010.
- Resnick, P. y Varian, H.R. “Recommender systems,” Communications of the ACM, vol. 40, 1997, pág. 57.
- Resnick, P. et al. “GroupLens: An open architecture for collaborative filtering of netnews”. 1994.
- Ruiz Iniesta, A. Estrategias de recomendación aplicadas a repositorios de recursos educativos. Máster en Investigación en Informática de la Facultad de Informática, Facultad de Informática, Universidad Complutense de Madrid (UCM), Madrid, 2009.
- Shardanand, U. y Maes, P. Social information filtering: algorithms for automating “word of mouth”. Proceedings of the SIGCHI conference on Human factors in computing systems, ACM Press/Addison-Wesley Publishing Co, 1995.

- Schein, A.I. et al. Methods and metrics for cold-start recommendations. In: SIGIR '02: Proc. of the 25th Annual Int. ACM SIGIR Conf. on Re-search and Development in Information Retrieval, pp. 253–260. ACM, New York, NY, USA (2002)
- Seguido Font, M. Sistemas de recomendación para webs de información sobre la salud. Tesis de Máster, Departament de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, Catalunya, 2009.
- Sergio Navarro Gil, S. SriLanda: Sistema de recomendaciones inteligente para música, Escuela superior de Informática, Departamento de Informática, Universidad de Castilla-La Mancha, 2014.
- Smyth, B. (2007). "Case-Based Recommendation. The Adaptive Web." Springer Berlin / Heidelberg 4321: 342-376.
- Valero Cubas, S. Arquitectura de Búsqueda Basada en Técnicas Soft Computing para la Resolución de Problemas Combinatorios en Diferentes Dominios de Aplicaciones. Tesis Doctoral, Departamento de Tecnologías de la Información y las Comunicaciones, Universidad Politécnica de Valencia, Valencia, España. 2010.
- Xiaoyuan Su and Taghi M. Khoshgoftaar. A survey of collaborative filtering techniques. Advances in Artificial Intelligence, 2009:1–20, 2009.
- Zapata, A. et al. A Hybrid Recommender Method for Learning Objects. Design and Evaluation of Digital Content for Education Proceedings published by International Journal of Computer Applications. (2011).

Anexos

Anexos 1. Encuesta para la conocer la interacción de los usuarios con el COJ

Pregunta 1. Selecciona usted los problemas a resolver atendiendo a:

- Complejidad
- Indicación del entrenador
- Recomendación de otros usuarios
- Ejercicios que más usuarios han resuelto

Pregunta 2. ¿Sigue usted un proceso escalonado para resolver los problemas, seleccionando primero los de menor complejidad?

- Sí
- No
- No sé

Pregunta 3. ¿Valoraría usted los problemas publicados en el COJ?

- Sí
- No
- No sé

Anexos 2. Datos de los intentos para resolver un problema

Usuario	Antes del 01-07-2014			Después del 01-07-2014		
	Total de envíos	Total de EA	Promedio A	Total de envíos	Total de EA	Promedio A
MarX	1063	438	2,43	712	359	1,98
Raschach	1111	431	2,58	596	278	2,14
humbertodiaz	761	469	1,62	514	383	1,34
Cyrax	401	126	3,18	455	208	2,19
dglima	1593	393	4,05	443	108	4,1
eliogovea	1214	544	2,23	399	166	2,4
teruel	539	313	1,72	393	207	1,9
MindFreaky	273	125	2,18	393	206	1,91
codeoverflow	1486	479	3,1	377	123	3,07
jcg	770	583	1,32	357	247	1,45
huhumope	285	200	1,43	346	239	1,45
angelmh	497	29	17,14	326	56	5,82
kassandra	16	4	4	309	22	14,05
isaacvr	566	303	1,87	308	184	1,67
baphomet	74	21	3,52	289	102	2,83
leninx	30	5	6	269	51	5,27
rpgomez	1354	635	2,13	256	85	3,01
Kazekage	6	1	6	254	127	2
Abdo	1804	713	2,53	251	133	1,89
NickMadworld	488	215	2,27	244	110	2,22
iTzRyu	5	2	2,5	239	107	2,23
rancherosonny	129	12	10,75	235	75	3,13

2013i11annel	636	244	2,61	160	72	2,22
<u>jacalistre</u>	1120	337	3,32	160	52	3,08
<u>ibestard</u>	749	368	2,04	158	129	1,22
<u>YhaYhaXD</u>	40	4	10	157	72	2,18
<u>Orlandol say97</u>	46	29	1,59	156	94	1,66
<u>otero91</u>	514	193	2,66	155	105	1,48
<u>Kino</u>	1756	674	2,61	155	96	1,61
<u>dcordb</u>	757	316	2,4	154	83	1,86
<u>ddwebx</u>	11	2	5,5	154	59	2,61
<u>Avid</u>	466	92	5,07	153	41	3,73
<u>Oreste</u>	1557	672	2,32	152	61	2,49
<u>Tang</u>	59	19	3,11	150	81	1,85
<u>renneweed</u>	371	103	3,6	148	60	2,47
<u>irc</u>	91	50	1,82	144	76	1,89
<u>christian08</u>	5	1	5	143	44	3,25
<u>yordanlaborde</u>	695	275	2,53	142	130	1,09
<u>arosalesa</u>	61	8	7,63	141	50	2,82
<u>Kyler</u>	143	36	3,97	140	76	1,84
<u>Mozzart</u>	661	267	2,48	140	68	2,06
<u>Davidthebest</u>	4155	1548	2,68	138	66	2,09
<u>shirley</u>	43	9	4,78	138	64	2,16
<u>raudyagdel</u>	124	57	2,18	136	80	1,7
<u>EdPazK</u>	917	512	1,79	135	97	1,39
<u>Siervo</u>	535	188	2,85	133	43	3,09

Locos	2931	812	3,61	229	81	2,83
novax	1325	262	5,06	225	64	3,52
jabetancourt	487	174	2,8	215	126	1,71
jtavara23	64	16	4	210	109	1,93
JoelCz	194	92	2,11	206	78	2,64
EmmanuelUPIR	58	6	9,67	205	28	7,32
HectorA	100	11	9,09	203	25	8,12
Keyfu	2094	1047	2	202	83	2,43
Oskar	7	2	3,5	199	95	2,09
GhostUplink13	374	129	2,9	185	91	2,03
adejongh93	158	58	2,72	183	82	2,23
Charlie	3134	1586	1,98	182	103	1,77
mikemandujas	468	179	2,61	182	98	1,86
NiuBer	3	4	0,75	179	68	2,63
KeGiRoZa	326	113	2,88	175	66	2,65
ypizarroza	2184	987	2,21	174	74	2,35
yuri	576	273	2,11	172	117	1,47
osvaldocuevas	35	8	4,38	171	43	3,98
miguel550	31	8	3,88	168	94	1,79
Encarnacion	92	17	5,41	167	37	4,51
daylor	350	73	4,79	164	66	2,48
alda	896	530	1,69	161	91	1,77
WIL	1276	450	2,84	161	59	2,73
rvargas	1189	508	2,34	161	58	2,78

danito	409	199	2,06	132	93	1,42
allen	883	281	3,14	132	71	1,86
adelrisco	918	322	2,85	130	61	2,13
rvarona	362	100	3,62	127	60	2,12