

UNIVERSIDAD CATOLICA DE SANTA MARIA

FACULTAD DE CIENCIAS E INGENIERIAS FISICAS Y FORMALES

ESCUELA PROFESIONAL DE INGENIERIAS DE SISTEMAS



**Determinando factores para el desarrollo de un sistema de
recomendación**

Tesis presentada por el Bachiller:

SOLORIO DEL CARPIO BELLIDO, JOHANA ALEJANDRA

Para obtener el Título Profesional en la escuela

Profesional de Ingeniería de Sistemas.

AREQUIPA-PERÚ

2015

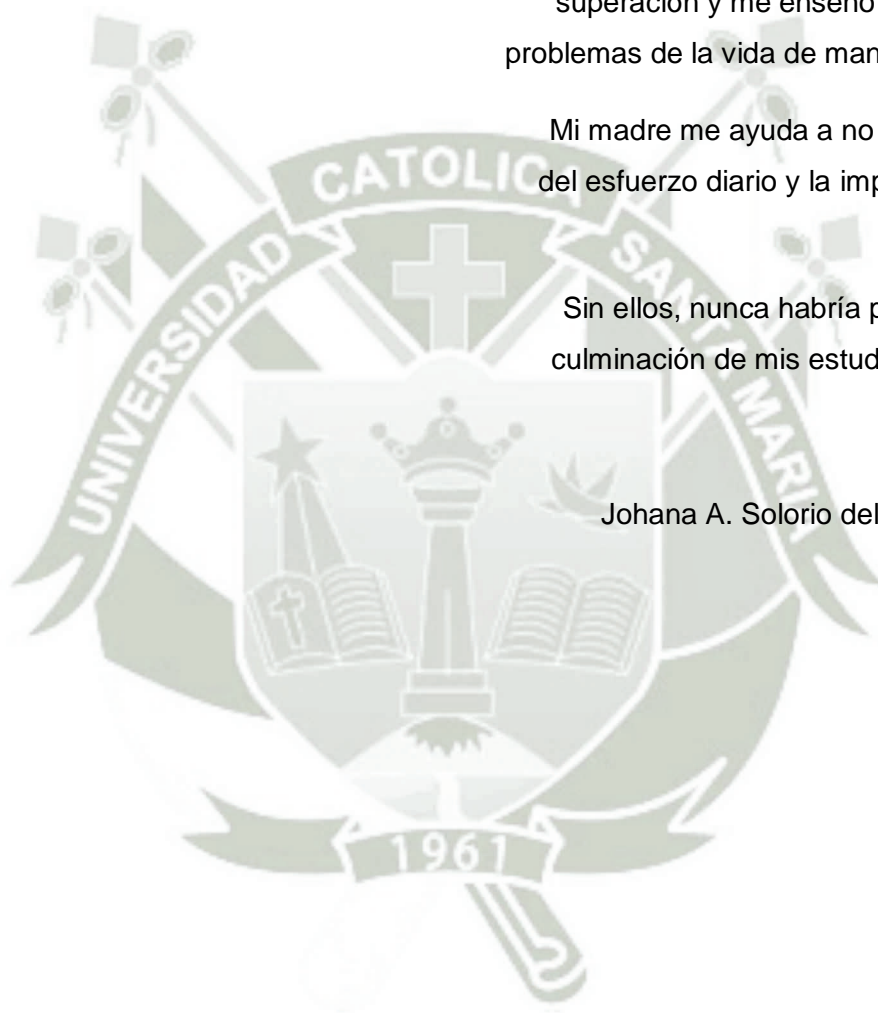
Mi carrera de ingeniería de sistemas no habría sido posible sin el enorme apoyo y aliento que recibí durante toda mi vida por parte de mis padres.

Mi padre me inculco grandes deseos de superación y me enseñó a enfrentar los problemas de la vida de manera espartana.

Mi madre me ayuda a no olvidar el valor del esfuerzo diario y la importancia de la familia.

Sin ellos, nunca habría podido lograr la culminación de mis estudios superiores.

Johana A. Solorio del Carpio Bellido



AGRADECIMIENTOS

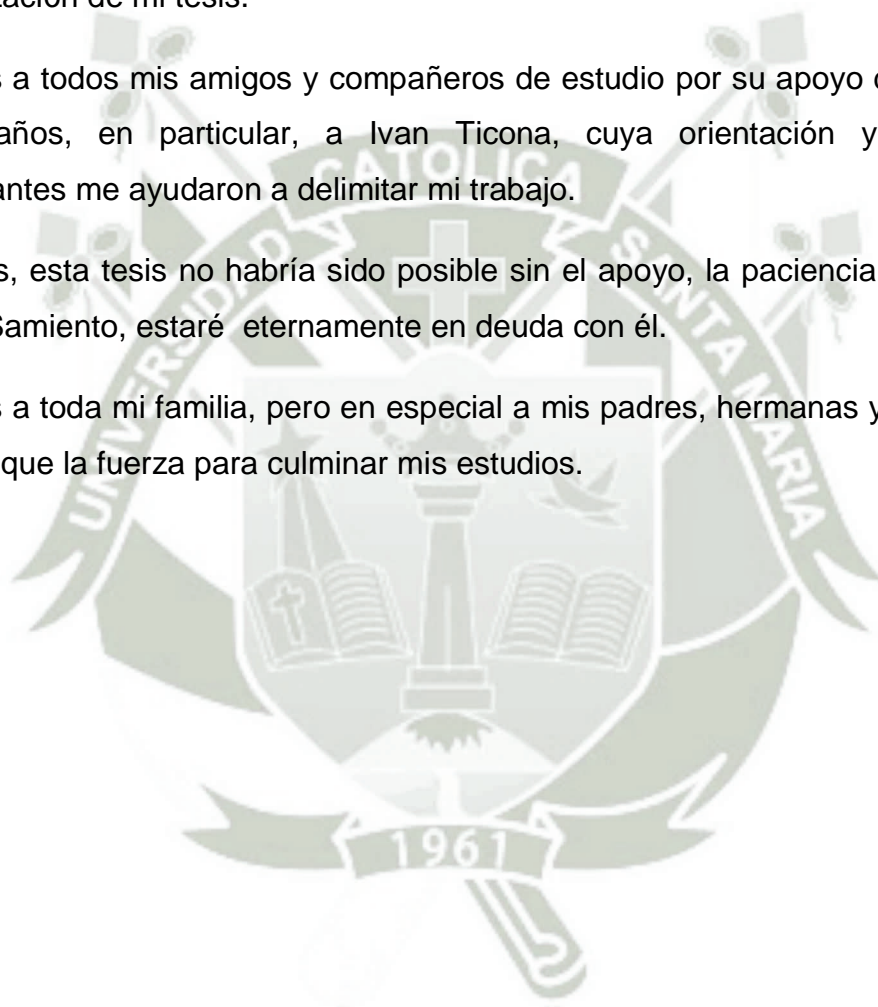
Por encima de todo, gracias a Dios por darme el privilegio de poder estudiar y la fuerza de voluntad para terminar

Gracias al Ingeniero César Baluarte Araya, por su apoyo y asesoría brindados en la realización de mi tesis.

Gracias a todos mis amigos y compañeros de estudio por su apoyo durante todos estos años, en particular, a Ivan Ticona, cuya orientación y discusiones interesantes me ayudaron a delimitar mi trabajo.

Además, esta tesis no habría sido posible sin el apoyo, la paciencia, y la guía de David Samiento, estaré eternamente en deuda con él.

Gracias a toda mi familia, pero en especial a mis padres, hermanas y sobrinos, de ellos saque la fuerza para culminar mis estudios.



LISTA DE FIGURAS

FIGURA 1. DESARROLLO DE LA INVESTIGACIÓN	8
FIGURA 2. MODELO BÁSICO DE RECOMENDACIÓN	16
FIGURA 3. TIPOS DE SISTEMAS DE RECOMENDACIÓN	28
FIGURA 4. EJEMPLO DE SISTEMAS DE RECOMENDACIÓN NO PERSONALIZADOS	30
FIGURA 5. EJEMPLO DE SISTEMAS DE RECOMENDACIÓN DEMOGRÁFICOS	31
FIGURA 6. EJEMPLO DE SISTEMAS DE RECOMENDACIÓN BASADA EN CONTENIDO	37
FIGURA 7. EJEMPLO DE SISTEMAS DE RECOMENDACIÓN HÍBRIDOS	44
FIGURA 8. ANÁLISIS DE DATOS PARA HACER LA RECOMENDACIÓN	82
FIGURA 9. ETAPAS DE DESARROLLO DEL PROTOTIPO	96
FIGURA 10. DIAGRAMA DE FLUJO DEL MODELO DE RECOMENDACIÓN HÍBRIDO	100
FIGURA 11. FUENTES DE DATOS EXTERNAS ADQUIRIDAS PARA EL PROTOTIPO	102
FIGURA 12. ARQUITECTURA GENERAL DEL SISTEMA	109
FIGURA 13. DIAGRAMA DE ENTIDAD DE RELACIÓN DEL PROTOTIPO PROPUESTO	113
FIGURA 14. CÓDIGO PARA LA IMPLEMENTACIÓN DE UN MOTOR DE	117
FIGURA 15. PROCESO DE RECOMENDACIÓN SRFC USANDO MAHOUT	117
FIGURA 16. DIAGRAMA DE SECUENCIA	118
FIGURA 17. PANTALLA DE INICIO DEL SISTEMA	119
FIGURA 18: PANTALLA PRESENTACIÓN DE ÚLTIMOS ESTRENOS	119
FIGURA 19. PANTALLA PRESENTACIÓN DE RECOMENDACIÓN	120
FIGURA 20. CÓDIGO PARA LA IMPLEMENTACIÓN DE LA	123
FIGURA 21: PELÍCULAS DEL GENERO ACCIÓN	127
FIGURA 22. PELÍCULAS DEL GENERO DRAMA	128
FIGURA 23. PERSONALIZACIÓN PARA EL USUARIO 1	129
FIGURA 24. PERSONALIZACIÓN PARA EL USUARIO 2	130
FIGURA 25. PERSONALIZACIÓN PARA EL USUARIO 3	130
FIGURA 26. OBTENCIÓN DE PREFERENCIAS POR EL USUARIO POR MEDIO DE CALIFICACIONES	131
FIGURA 27. OBTENCIÓN DE PREFERENCIAS POR EL USUARIO POR	132
FIGURA 28. RECOMENDACIÓN SIN PERSONALIZACIÓN	133
FIGURA 29. . RECOMENDACIÓN BASADA EN CONTENIDO	133
FIGURA 30. RECOMENDACIÓN BASADA EN FILTRADO COLABORATIVO	134
FIGURA 31. PRESENTACIÓN DE LA INFORMACIÓN DE PELÍCULAS	135
FIGURA 32. GRÁFICO DE RESPUESTA ACERCA DE LA IMPORTANCIA DE LA SELECCIÓN DEL TIPO DE METODOLOGÍA DE DESARROLLO	142
FIGURA 33. GRÁFICO DE RESPUESTA ACERCA DEL DOMINIO DE LA RECOMENDACIÓN	143
FIGURA 34. GRÁFICO DE RESPUESTA ACERCA DEL PROPÓSITO DE LA RECOMENDACIÓN	144
FIGURA 35. GRÁFICO DE RESPUESTA ACERCA DEL CONTEXTO DE LA RECOMENDACIÓN	145
FIGURA 36. GRÁFICO DE RESPUESTA ACERCA DEL PROPÓSITO	146
FIGURA 37. GRÁFICO DE RESPUESTA ACERCA DE LAS MÉTRICAS COMERCIALES	147
FIGURA 38. GRÁFICO DE RESPUESTA ACERCA DE LA IMPORTANCIA DE LA SELECCIÓN DEL TIPO DE METODOLOGÍA DE DESARROLLO	148
FIGURA 39. GRÁFICO DE LA RESPUESTA ACERCA DE LOS MÉTODOS	149
FIGURA 40. GRÁFICO DE RESPUESTA ACERCA DE LA IMPORTANCIA DE DEFINIR	150

FIGURA 41. GRÁFICO DE RESPUESTA DE SI LA PRESENTACIÓN DE LA RECOMENDACIÓN	151
FIGURA 42. GRÁFICO DE LA RESPUESTA ¿CUÁL DE LOS SIGUIENTES FACTORES PARA LA EVALUACIÓN DE SR LE PARECE EL MÁS ADECUADO?	152
FIGURA 43. GRÁFICO DE RESPUESTAS A LA PREGUNTA DE INTEGRACIÓN DE LOS FACTORES PRESENTADOS EN EL PROTOTIPO	153
FIGURA 44. REPUESTAS ACERCA DEL IMPACTO DE LOS FACTORES PRESENTADOS EN EL FUTURO DESARROLLO DE SR	155
FIGURA 45. GRÁFICO DE RESPUESTAS QUE BUSCA ESTABLECER LA IMPORTANCIA DE DEFINIR EL MODELO DE NEGOCIO	157
FIGURA 46. REPUESTAS A LA PREGUNTA CALIFICACIÓN DE PROTOTIPO	158
FIGURA 47. GRÁFICO DE RESPUESTA ACERCA DE LA RECOMENDACIÓN OFRECIDA POR EL SISTEMA	159
FIGURA 48. RESPUESTAS ACERCA DE LAS CARACTERÍSTICAS DEL PROTOTIPO	160
FIGURA 49. GRÁFICO DE REPUESTAS ACERCA DEL ANÁLISIS Y DISEÑO DE LAS FUNCIONALIDADES DEL PROTOTIPO.	161
FIGURA 50. GRÁFICO DE REPUESTAS ACERCA DE LA EVALUACIÓN PRESENTADA EN EL PROTOTIPO	162



LISTA DE TABLAS

TABLA 1. EJEMPLO DE UN SISTEMA DE RECOMENDACIÓN DE FILTRADO COLABORATIVO	42
TABLA 2. COMPARACIÓN DE METODOLOGÍAS	56
TABLA 3. DIFERENCIA ENTRE METODOLOGÍAS RUP Y SCRUM	63
TABLA 4. FACTORES PARA RECOMENDACIÓN DE SISTEMAS DE RECOMENDACIÓN	68
TABLA 5. TABLA COMPARATIVA DE LOS TIPOS DE RECOMENDACIÓN	84
TABLA 6. TABLA DE FASES DE DESARROLLO DEL PROTOTIPO Y FACTORES INCLUIDOS POR CADA ETAPA	94
TABLA 7. SELECCIÓN DEL TIPO DE METODOLOGÍA PARA EL DESARROLLO DEL PROTOTIPO	95
TABLA 8. DEFINICIÓN DEL MODELO DE NEGOCIO DEL PROTOTIPO DE RECOMENDACIÓN	97
TABLA 9. HISTORIA DE USUARIO: INGRESO DE PERFIL DE UN NUEVO USUARIO	104
TABLA 10. HISTORIA DE USUARIO: VISUALIZAR EL PERFIL DE UN USUARIO	105
TABLA 11. HISTORIA DE USUARIO: ACTUALIZAR EL PERFIL DEL USUARIO	105
TABLA 12. HISTORIA DE USUARIO: CALIFICACIÓN DE PELÍCULAS	106
TABLA 13. HISTORIA DE USUARIO: GESTIÓN DE COMENTARIOS	106
TABLA 14. HISTORIA DE USUARIO: DISEÑAR LA PRESENTACIÓN DE LA RECOMENDACIÓN	107
TABLA 15. HISTORIA DE USUARIO: DISEÑAR LA JUSTIFICACIÓN DE LA RECOMENDACIÓN	107
TABLA 16. DICCIONARIO DE DATOS DE LA TABLA MOVIE	110
TABLA 17. DICCIONARIO DE DATOS DE LA TABLA RATINGS	111
TABLA 18. DICCIONARIO DE DATOS DE LA TABLA USER_ROLES	111
TABLA 19. DICCIONARIO DE DATOS DE LA TABLA USERS	111
TABLA 20. DICCIONARIO DE DATOS DE LA TABLA GENRES _MOVIE	112
TABLA 21. DICCIONARIO DE DATOS DE LA TABLA GENRES	112
TABLA 22. DICCIONARIO DE DATOS DE LA TABLA ROLES	112
TABLA 23. TABLA QUE MUESTRA EL CÁLCULO DE LA CORRELACIÓN DE PEARSON	116
TABLA 24. COMPARACIÓN DE LA CALIFICACIÓN ECHA POR EL USUARIO	123
TABLA 25. FACTORES INCLUIDOS EN EL DESARROLLO DEL PROTOTIPO	126
TABLA 26. TABLA DE ACEPTACIÓN DE LAS RECOMENDACIONES POR PARTE DEL USUARIO	129
TABLA 27. COMPARACIÓN DE MÉTODOS DE CÁLCULO DE SIMILITUD	135
TABLA 28. CALCULO DE COBERTURA ALCANZADA POR EL PROTOTIPO	136
TABLA 29. TABLA DE CÁLCULO DE DIVERSIDAD DE GÉNERO PARA 40 RECOMENDACIONES	137
TABLA 30. EVALUACIÓN DE LA PRECISIÓN DE LAS RECOMENDACIONES	138
TABLA 31. TABLA DE FACTORES PARA EL DESARROLLO DE UN SR QUE FUERON EVALUADOS	141
TABLA 32. RESPUESTAS A LA PREGUNTA QUE BUSCA ESTABLECER LA IMPORTANCIA DE DEFINIR EL MODELO DE NEGOCIO	142
TABLA 33. RESPUESTA A LA PREGUNTA ACERCA DEL DOMINIO DE LA RECOMENDACIÓN	143
TABLA 34. RESPUESTA ACERCA EL PROPÓSITO DE LA RECOMENDACIÓN	144
TABLA 35. RESPUESTA A LA PREGUNTA ACERCA DEL CONTEXTO DE LA RECOMENDACIÓN	145
TABLA 36. RESPUESTA A LA PREGUNTA ACERCA DEL GRADO DE	146
TABLA 37. RESPUESTA A LA PREGUNTA ACERCA DE LA INTEGRACIÓN	147
TABLA 38. RESPUESTAS A LA PREGUNTA CREE USTED QUE ES IMPORTANTE DEFINIR EL TIPO DE RECOMENDACIÓN ANTES DE EMPEZAR EL DESARROLLO DEL SR	148
TABLA 39. RESPUESTA A LA PREGUNTA ACERCA DE LOS MÉTODOS	149
TABLA 40. RESPUESTA A LA PREGUNTA ACERCA DE LA IMPORTANCIA DE	150

TABLA 41. RESPUESTA A LA PREGUNTA ACERCA DE SI LA PRESENTACIÓN	151
TABLA 42. REPUESTAS A LA PREGUNTA ACERCA DE ¿CUÁL DE LOS FACTORES DE	152
TABLA 43. TABLA DE FACTORES QUE FUERON INTEGRADOS EN EL PROTOTIPO	153
TABLA 44. RESPUESTA A LA PREGUNTA DE INTEGRACIÓN DE LOS FACTORES CON EL PROTOTIPO	153
TABLA 45. REPUESTAS A LA PREGUNTA ACERCA DEL IMPACTO DE LOS 5 FACTORES PRESENTADOS EN EL FUTURO DESARROLLO DE UN SR	155
TABLA 46. RESPUESTA A LA PREGUNTA QUE BUSCA ESTABLECER LA IMPORTANCIA DE DEFINIR EL MODELO DE NEGOCIO	157
TABLA.47 REPUESTAS A LA PREGUNTA CALIFICACIÓN DE PROTOTIPO	158
TABLA 48. RESPUESTA A LA PREGUNTA ACERCA DE LA RECOMENDACIÓN OFRECIDA POR EL SISTEMA	159
TABLA 49. RESPUESTAS A LA PREGUNTA ACERCA DE LAS CARACTERÍSTICAS DEL PROTOTIPO	160
TABLA 50. REPUESTAS A LA PREGUNTA ACERCA DEL ANÁLISIS Y DISEÑO DE LAS FUNCIONALIDADES DEL PROTOTIPO	161
TABLA 51. REPUESTAS A LA PREGUNTA ACERCA DE LA IMPORTANCIA DE LA EVALUACIÓN DE LA CALIDAD DE LA RECOMENDACIÓN	162



LISTA DE ABREVIATURAS

SR	Sistemas de Recomendación
SRFC	Sistemas de Recomendación de Filtrado Colaborativos
SRBCont	Sistemas de Recomendación Basados en Contenido
SRD	Sistemas de Recomendación Demográficos
SRBU	Sistemas de Recomendación Basados en Utilidad
SRBCono	Sistemas de Recomendación Basados en Conocimiento
MAE	Error Absoluto Medio
MSE	Error Cuadrático Medio
DRA	Desarrollo Rápido de Aplicaciones
RUP	Rational Unified Process
XP	eXtreme Programming

INDICE

RESUMEN	xiii
ABSTRACT	xv
CAPÍTULO 1	1
PLANTEAMIENTO TEÓRICO	1
1.1. Introducción	1
1.2. Descripción del problema	3
1.3. Planteamiento del problema	4
1.4. Antecedentes de investigación	4
1.5. Objetivos de la investigación	6
1.5.1. Objetivo General	6
1.5.2. Objetivos Específicos	6
1.6. Justificación	6
1.6.1. Hipótesis	7
1.6.2. Variable Independiente	7
1.6.3. Variable dependiente	7
1.7. Desarrollo de la investigación	8
CAPITULO 2	9
MARCO TEÓRICO	9
2.1. Conceptos teóricos	9
2.1.1. Metodología	9
2.1.2. Sistemas de Información	10
2.1.3. Sistema Recomendación	10
2.1.4. Diseño metodológico	10
2.1.5. Motor de recomendación	11
2.1.6. Personalización	11
2.1.7. Sugerencia	11
2.1.8. Recomendación	11
2.1.9. Factor	12
2.2. Sistemas de recomendación	13

2.3.	Modelo Básico de un sistema de recomendación	15
2.3.1.	El proceso de recomendación	17
2.3.2.	Datos de entrada y salida del sistema.....	18
2.4.	Necesidad de Sistemas de Recomendación	20
2.4.1.	Justificación para el uso de sistemas de recomendación	21
2.5.	Problemas que enfrentan este los SR actuales.....	23
2.6.	Características que deben ser consideradas en el diseño del SR.....	26
2.7.	Tipo de sistemas de recomendación	28
2.7.1.	Sistemas de recomendación no personalizados.....	28
2.7.2.	Sistemas de recomendación demográficos (SRD)	30
2.7.3.	Sistemas de recomendación basados en conocimiento (SRBCono)	32
2.7.4.	Sistemas de recomendación basados en utilidad (SRBU)	33
2.7.5.	Sistemas de recomendación basados en confianza (SRBC)	34
2.7.6.	Sistemas de recomendación basada en contenido (SRBCont).....	35
2.7.7.	Sistemas de recomendación de filtrado colaborativo (SRFC)	37
2.7.8.	Sistemas de recomendación híbridos	42
CAPÍTULO 3.....		45
METODOLOGÍAS PARA EL DESARROLLO DE SISTEMAS DE RECOMENDACIÓN		45
3.1.	Constitución de una metodología.....	46
3.1.1.	Modelo de Cascada	46
3.1.2.	Modelo por Prototipos	49
3.1.3.	Modelo en espiral.....	51
3.1.4.	Metodología de desarrollo de software ágil.....	53
3.1.5.	Modelo de Desarrollo Rápido de Aplicaciones.....	56
3.1.6.	Modelo Incremental.....	57
3.1.7.	Proceso Unificado Rational (RUP)	59
3.1.8.	SCRUM	61
3.2.	Beneficios del uso de metodologías en el desarrollo de sistemas.....	64
3.3.	Problemas del uso de metodologías en el desarrollo de sistemas	65
CAPÍTULO 4		66
DETERMINACIÓN DE FACTORES PARA EL DESARROLLO DE UN SISTEMA DE RECOMENDACIÓN		66

4.1.	Determinación de factores para el desarrollo de un sistema de recomendación	66
4.2.	Seleccionar la metodología de desarrollo adecuada para un SR	69
4.2.1.	Factor tipo de metodología	69
4.3.	Definir el modelo de negocio de la recomendación.....	72
4.3.1.	Factor dominio de la recomendación	73
4.3.2.	Factor propósito de la recomendación	73
4.3.3.	Factor contexto de la recomendación	74
4.3.4.	Factor personalización de la recomendación	74
4.3.5.	Factor métricas comerciales	76
4.3.6.	Factor integridad de la recomendación	76
4.4.	Obtener los datos de perfil y preferencia	78
4.4.1.	Métodos de obtención de preferencias del usuario.	79
4.4.2.	Perfiles de usuario	80
4.4.3.	Información actual.....	82
4.5.	Diseñar el motor de recomendación	83
4.5.1.	Tipo de modelo de recomendación	83
4.5.2.	Proceso de recomendación	86
4.5.3.	Presentación de la recomendación	86
4.6.	Evaluar la calidad de recomendación	87
4.6.1.	Exactitud del rango	88
4.6.2.	Cobertura	88
4.6.3.	Diversidad	88
4.6.4.	Precisión.....	89
CAPÍTULO 5		92
DISEÑO DEL PROTOTIPO DE UN SISTEMA DE RECOMENDACIÓN		92
5.1.	Planeamiento.....	95
5.1.1.	Tipo de Metodología	95
5.1.2.	Justificación del sistema	96
5.1.3.	Definición del modelo de negocio para las recomendaciones.....	96
Modelo de negocio para las recomendaciones		97
5.1.4.	Tipo de modelo recomendación	97

5.1.5.	Definir las fuentes de datos	101
5.1.6.	Definición de las Tareas de desarrollo	102
5.2.	Análisis	103
5.2.1.	Métodos de obtención de preferencias del usuario	103
5.2.2.	Establecer requerimientos no funcionales.....	104
5.2.3.	Establecer las funcionalidades del sistemas mediante historias de usuarios 104	
5.3.	Diseño y codificación.....	109
5.3.1.	Diseño de la arquitectura	109
5.3.2.	Diseño de la estructura de los datos	110
5.3.3.	Diseñar el motor de recomendación.....	113
5.3.1.	Diseñar la Presentación de la recomendación.....	119
5.4.	Pruebas	121
5.4.1.	Efectuar las pruebas del sistema.....	121
5.4.2.	Evaluar la calidad de recomendación.....	122
5.4.1.	Resultados de la evaluación de la recomendación	124
5.5.	Comprobación de los factores aplicados en el desarrollo del prototipo.....	126
5.5.1.	Factor tipo de metodología	127
5.5.2.	Factor dominio de la recomendación	127
5.5.3.	Factor propósito de la recomendación	128
5.5.4.	Factor Personalización de la recomendación	129
5.5.5.	Factor métodos de obtención de preferencias del usuario	131
5.5.6.	Factor tipo de modelo de recomendación	132
5.5.7.	Factor proceso de recomendación	134
5.5.8.	Factor presentación de la recomendación	135
5.5.9.	Factor cobertura.....	136
5.5.10.	Factor diversidad.....	136
5.5.11.	Factor Precisión	137
CAPÍTULO 6		139
EVALUANDO LOS FACTORES.....		139
6.1.	Evaluando los factores de éxito planteados	139
6.1.1.	Evaluación por Expertos	139

6.1.2. Perfil de los Expertos	140
6.1.3. Procedimiento de Evaluación	140
CUESTIONARIO 1	141
CUESTIONARIO 2	156
CONCLUSIONES.....	163
RECOMENDACIONES	164
ANEXO A	165
ANEXO B.....	166
ANEXO C.....	170



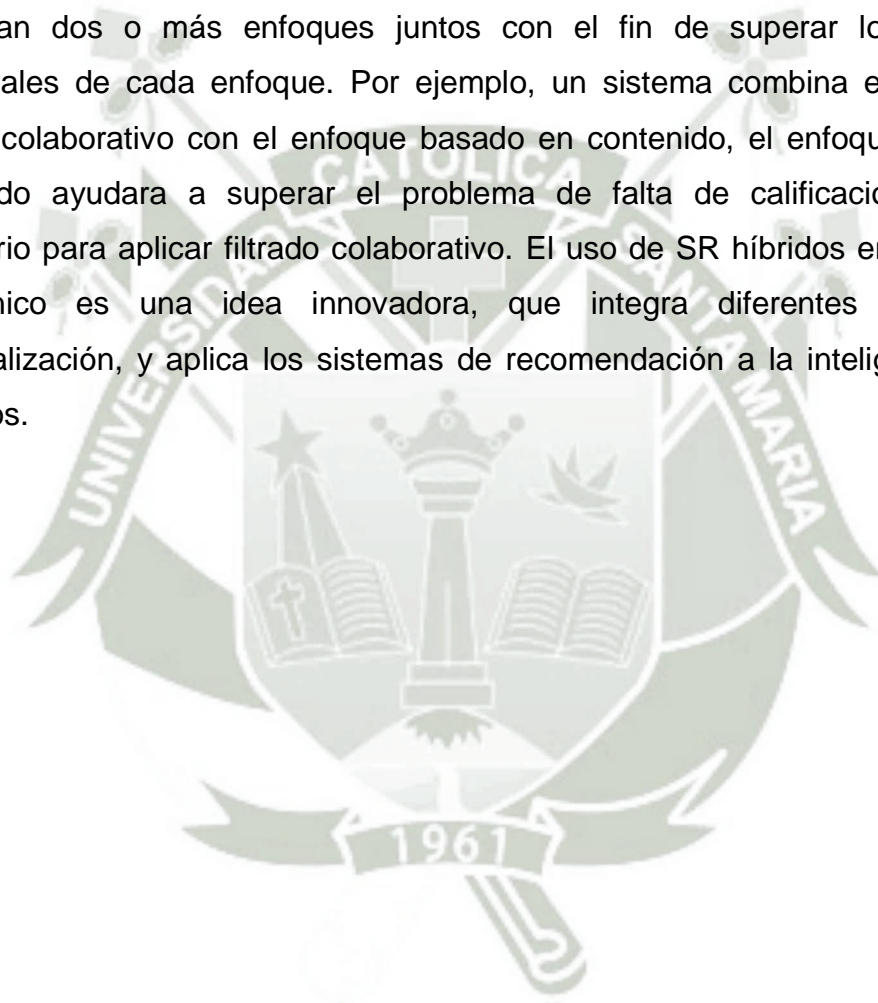
RESUMEN

La sobrecarga de información es un fenómeno que ha invadido cada campo de nuestras vidas, desde el trabajo (decidir que libros ordenar, que correo atender primero) a momentos de ocio (que películas ver, o a que restaurantes ir). Una forma fácil de resolver el problema es a través del uso de sistemas de recomendación (SR). Estos SR filtran los productos existentes y ayudan al usuario a visualizar productos que podrían ser de su interés y que van acorde con sus necesidades, y a su vez dispone de variedad y simplicidad en la selección de los productos que son de su interés. Estos sistemas están siendo desarrollados para personalizar las recomendaciones que brindan los servicios de comercio electrónico, este servicio ayuda a reducir la sobrecarga de información que los usuarios tienen al momento de buscar un producto.

La predicción de una recomendación es un proceso complejo sujeto a error, esto es debido a que no hay forma de predecir de manera exacta si un producto será del agrado de un cliente o no. Sin embargo los diferentes enfoques y métodos en los sistemas de recomendación, estudian las preferencias y el perfil del usuario, para proveer una recomendación útil y oportuna. El área de SR es muy amplia y está en creciente desarrollo, es por ello que determinar los factores para el desarrollo de SR es necesario para esclarecer el desarrollo de estos sistemas.

Existen diferentes enfoques para la generación de recomendaciones, filtrado colaborativo, basados en contenido y métodos híbridos, entre otros. Los métodos de filtrado colaborativo están basados en la similitud de preferencia de los usuarios. Se asume que los usuarios que estuvieron de acuerdo en la selección de productos en el pasado, estarán de acuerdo en la selección futura de productos. Por ejemplo, si se compara a dos usuarios en una librería, y se encuentra que la selección de libros es similar, se puede asumir que sus gustos futuros también

serán similares, por lo tanto se les puede recomendar el mismo tipo de productos. En el enfoque basado en contenido, las recomendaciones son echas basadas en el análisis de contenido. El contenido es un conjunto de términos que representa un elemento (página web, documento, correo) o lo describen, este contenido es extraído de largas descripciones del elemento. El contenido que es del interés del usuario se establece explícitamente o implícitamente mediante técnicas de aprendizaje automático. Por otro lado, los sistemas de recomendación híbridos combinan dos o más enfoques juntos con el fin de superar los problemas individuales de cada enfoque. Por ejemplo, un sistema combina el enfoque de filtrado colaborativo con el enfoque basado en contenido, el enfoque basado en contenido ayudara a superar el problema de falta de calificaciones que es necesario para aplicar filtrado colaborativo. El uso de SR híbridos en el comercio electrónico es una idea innovadora, que integra diferentes técnicas de personalización, y aplica los sistemas de recomendación a la inteligencia de los negocios.



ABSTRACT

Information overload is a phenomenon that has invaded every field in our lives, from work activities (decide which books to order, which emails to read first) to leisure time ones (which movies to see, which restaurants to go). One way to ease the problem is through the use of recommender systems, systems that try to match users and items that might interest them. There are several classic approaches for generating recommendations: collaborative filtering (CF), content-based, and hybrid methods. CF is based on similarity of user preferences. It assumes that users that agreed in the past on items they liked will probably agree on more items in the future. For example, taking one users bookshelf and crosschecking it with shelves of other users, finding those with similar books will yield several possible book recommendations for that user. To carry out such an operation, ratings of items must be gathered and stored from a large number of users. This approach is called user-user CF. In the content-based approach, recommendations are made based on content analysis. The content is a set of terms representing an item (Website, Document, Email) or describing it, usually extracted from the larger textual description of the item. To create a user model, the content that interests the target user is either explicitly given or implicitly learned through machine learning techniques. Hybrid recommender systems are systems that combine two (or more) approaches together in order for them to overcome each other's shortcomings. For example, a system that combines the collaborative filtering approach with a content-based recommender can overcome the first rater problem by matching new items using content analysis. Using hybrid recommender systems in ecommerce is an innovative idea, which integrated different personalization techniques, and applies recommender systems to business intelligence.

This thesis presents the results of the study of recommender systems, and how recommender systems should be developed.



CAPÍTULO 1

PLANTEAMIENTO TEÓRICO

1.1. Introducción

Hoy en día disponemos de una gran cantidad de información de productos y proveedores, a consecuencia de esto el número de empresas que ofrecen sus productos o servicios en Internet se multiplica cada año. Debido a esta competencia, las empresas deben ofertar una serie de servicios diferenciados que les permitan no sólo mantener su clientela sino también atraer nuevos clientes. Existen nuevas formas de alcanzar al cliente y aprovechar las oportunidades de negocio en el momento, para ello se requiere de una gran habilidad para filtrar la información útil de la que no lo es. Además se espera, un producto indicado, en el momento apropiado con un servicio personalizado. Los sistemas de recomendación (SR) surgen para solucionar este problema. Un SR es un sistema que utiliza las opiniones y comportamientos de los usuarios para ayudarles a encontrar contenidos de su gusto entre un conjunto sobrecargado de elecciones posibles (Araque, 2012). El objetivo de un buen sistema de recomendación es hacer llegar a cada usuario la información que necesita, evitando la que no le interesa. Estos sistemas estudian las características de los usuarios y mediante un

procesamiento de los datos pueden hallar un conjunto de productos que son de interés para el usuario. Sin embargo, los clásicos SR presentan diversos problemas que incluyen falta de datos iniciales, falta del perfil de usuario, falta de datos del producto, falta de puntajes, entre otros. Existe una amplia variedad de métodos y algoritmos para el desarrollo de estos sistemas. Es necesario establecer claramente las características que se van a desarrollar, tener clara comprensión del proceso de recomendación, establecer las limitaciones de los diferentes tipos de métodos de recomendación, todo ello para poder desarrollar el SR y tener éxito en la recomendación ofrecida.

Esta tesis está estructurada en seis capítulos. El primer capítulo describe el problema de investigación inicial y porque este es un tema de interés. Además, presenta los objetivos de investigación, planteamiento del problema, la propuesta de investigación y la metodología de la investigación que se va seguir.

El segundo capítulo, presenta el marco teórico, muestra el trabajo realizado en el campo de sistema de recomendación, describe los enfoques más conocidos en el desarrollo de SR. El objetivo de este capítulo, es resaltar los diferentes enfoques que se pueden utilizar en el desarrollo de SR y las deficiencias que presentan.

El capítulo tres hace la revisión de diferentes metodologías de desarrollo del software con el objetivo de encontrar la más adecuada para el futuro desarrollo de un SR.

El capítulo cuatro presenta al lector los factores para el desarrollo de un SR. El objetivo de este capítulo es mostrar un análisis previo al desarrollo de un SR, con el fin de tener bases sólidas para una propuesta de desarrollo.

El quinto capítulo desarrolla un prototipo de un sistema de recomendación híbrido, en el cual se aplican los factores señalados en el capítulo anterior.

El último capítulo es la evaluación de los factores que se desarrollaron en la tesis y la evaluación del prototipo, esta evaluación será llevada a cabo por expertos del área, y se hará mediante el uso de encuestas.

1.2. Descripción del problema

El crecimiento explosivo de la World Wide Web en la década de 1990 dio lugar a un crecimiento proporcional de la cantidad de información disponible en línea, superando la capacidad de los usuarios para procesar toda esta información (Hanani et al., 2001). Surge un fuerte interés en desarrollar alguna manera de procesar grandes cantidades de información para presentar al usuario solo la información necesaria. En un intento por ayudar a los clientes en estos esfuerzos, y conservar tiempo valioso, se ofrecen a los usuarios sugerencias de productos que pueden ser de su interés.

Los sistemas de recomendación surgen para complementar los servicios de comercio electrónico, su tarea es personalizar el servicio y reducir la sobrecarga de información que tiene el usuario. Estos sistemas deben permitir seleccionar un pequeño subconjunto de los productos, de los millones de productos, que parecen adaptarse a las necesidades y preferencias de los usuarios. Aunque es casi imposible predecir con precisión las necesidades de los usuarios, un conjunto pequeño de sugerencias ayuda a limitar el número de opciones. Mediante la restricción de la cantidad de productos sugeridos, este tipo de sistemas ayudan a las personas a tomar decisiones sobre qué artículos comprar, qué noticia a leer o qué película vale la pena ver (Hanani et al., 2001).

El objetivo de un SR es ayudar a los compradores potenciales de escoger el producto adecuado para su consumo. Cuando los consumidores consumen productos recomendados por el sistema, es posible aumentar la venta cruzada, esta característica promueve un comercio electrónico más productivo. Gracias a ello se mantiene y construye la fidelidad de los clientes, se busca

convertir al visitante casual en cliente asiduo. Entonces, los sistemas de recomendación deben ayudar en la selección de productos, facilitan la toma de decisiones e incrementan la satisfacción del cliente (Musial, 2009).

1.3. Planteamiento del problema

Los sistemas de recomendación tradicionales tienen una amplia variedad de problemas como falta de datos iniciales, falta del perfil de usuario, falta de datos del producto, falta de puntuaciones, desconocimiento de las preferencias del usuario, las cambiantes tendencias de compras, la complejidad al momento de seleccionar las variables para la recomendación, estos problemas los hacen ineficientes al momento de encontrar nuevas recomendaciones (Ricci et al., 2011). Para cubrir estas carencias de información, se han desarrollado diferentes métodos y complejos cálculos probabilísticos que predicen las futuras preferencias del usuario. Se debe comprender que el proceso de selección de información es complejo, la data del entorno de recomendación muchas veces es subjetiva y difícil de medir cuantitativamente.

Una implementación de este tipo de sistemas puede enfrentarse a una abrumadora cantidad de técnicas y variables. Además, se debe identificar el modelo de recomendación a ser desarrollado para tener una recomendación que se adecue a la estrategia de negocio, esta selección puede llegar a ser complicada y difícil para una persona sin suficiente experiencia en este ámbito. Se plantea identificar los factores que son importantes para el desarrollo de un SR, estos factores son independientes del modelo de recomendación que se escoja.

1.4. Antecedentes de investigación

Diariamente las personas se enfrentan al proceso de toma de decisiones, el cual puede ser tan simple como elegir el desayuno de hoy, o tan complejo

como seleccionar la profesión que se desempeñará a lo largo de la vida. Muchas de estas decisiones se “simplifican” cuando se le pregunta su opinión a alguien más, de manera que es posible tener un parámetro que complementa el propio. De esta forma, una “recomendación” puede ayudar en la toma de decisiones. Los sistemas de recomendación, auxilian a los usuarios emitiendo sugerencias acerca de diversos objetos. Por ejemplo, Amazon.com recomienda artículos a los usuarios basándose en las búsquedas que el usuario realiza. Dado que un sistema de recomendación es considerado un auxiliar, es posible pensar en ellos como una ayuda para para filtrar productos y personalizar contenido (BÁEZ et al., 2013).

Los sistemas de recomendación han surgido hace 1990s (Inès Jomaa et al., 2011), esta recomendación no era personalizada, ni basada en las preferencias de los usuarios. Con el paso de los años se vio la necesidad de establecer nuevos criterios en las recomendaciones, y conforme las investigaciones fueron avanzando las recomendaciones comenzaron a considerar otros factores como, contexto, dominio, utilidad, entorno social del usuario, grado de personalización, integridad de la recomendación, entre otros. Nuevos algoritmos fueron creados para optimizar el resultado de la recomendación. La amplia variedad de algoritmos contribuye a la disminución de ciertas deficiencias. Para la implementación de estos sistemas se deben establecer criterios antes de diseñar el motor de recomendación y poder alcanzar los objetivos deseados. Los sistemas de recomendación modernos evolucionaron para dar recomendaciones personalizadas considerando factores más complejos.

1.5. Objetivos de la investigación

1.5.1. Objetivo General

- Determinar los factores para desarrollo de un sistema de recomendación.

1.5.2. Objetivos Específicos

- Desarrollar un prototipo que permita determinar la aplicación de los factores en el desarrollo de un sistema de recomendación.
- Visualizar los diferentes problemas que se presentan al desarrollar un SR.

1.6. Justificación

Muchos de los principales sitios web de comercio electrónico ya están utilizando SR para proporcionar sugerencias de interés para sus clientes. Estos sistemas son integrados como parte de la personalización que se le puede dar a un servicio de comercio electrónico, páginas como, Amazon.com, YouTube, Netflix, Yahoo, Tripadvisor, ya los han implementado y los utilizan rentablemente.

A nivel internacional el desarrollo de SR es ampliamente conocido e investigado, sin embargo la difusión de estos sistemas a nivel nacional y específicamente en el panorama arequipeño es poco conocido. El motivo principal para elegir este tema se justifica en la necesidad de personalizar los servicios de comercio electrónicos. Es por ello que se requiere información acerca del desarrollo de SR. Un SR implica conocimientos de ciencia de computación, ingeniería del software y estadística. No es algo que se pueda construir simplemente instalando un plug-in, sin escribir código. Se trata de conseguir un alto nivel de “personalización” para servicios de comercio

electrónico. La generación de una recomendación personalizada enfrenta diferentes retos, por ejemplo, la recomendación debe despertar el interés del usuario, dicha recomendación debe llegar a tiempo, de la forma correcta, con el producto indicado y sin interferir con el funcionamiento del sistema (Ricci et al., 2011).

Resultará más sencillo realizar el desarrollo de un SR si contamos con la información de los diferentes factores que intervienen en su desarrollo, es por ello que esta tesis recoge varios factores que diferentes autores coinciden en señalar como importantes al momento de hablar de SR. Por otra parte, este trabajo también arroja luz sobre los desafíos que pueden encontrarse durante la implementación de un sistema de recomendación basada en el código abierto Apache Mahout, una biblioteca de algoritmos estadísticos y analíticos.

1.6.1. Hipótesis

Es posible que determinando los factores para el desarrollo de sistemas de recomendación se logre brindar una adecuada recomendación, que permita al usuario tomar una decisión.

1.6.2. Variable Independiente

Factores para el desarrollo de sistemas de recomendación.

Indicador: evaluación de dichos factores.

1.6.3. Variable dependiente

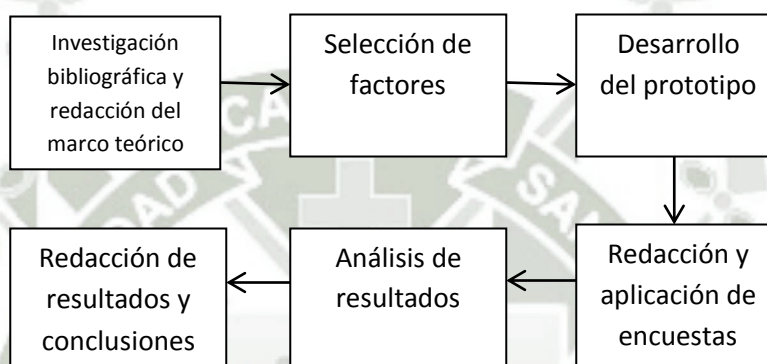
Sistema de recomendación.

Indicador: efectividad de la recomendación.

1.7. Desarrollo de la investigación

En la figura 1.1 se muestra de forma gráfica el proceso para llevar a cabo la investigación, partiendo de la investigación bibliográfica hasta la redacción de resultados.

Figura 1. Desarrollo de la investigación



Fuente: Elaboración Propia

CAPITULO 2

MARCO TEÓRICO

En este capítulo se presenta el marco teórico sobre el estado actual de los diferentes tipos de modelos de recomendación. Después de presentar los conceptos teóricos, se mostrará el modelo básico de un SR, la necesidad de los SR para formar parte de los sistemas de comercio electrónicos, los problema y características que presentan, y finalmente se muestra los diferentes enfoques que hay para el cálculo de recomendación.

2.1. Conceptos teóricos

2.1.1. Metodología

Principios usados para hacer un trabajo particular, especialidad científica o llevar a cabo una investigación académica. Las metodologías de desarrollo de sistemas son un conjunto de recomendaciones que se basan en una lógica y la filosofía específica (Lemétayer, 2010). Una metodología es aquella guía que se sigue a fin de realizar las acciones propias de una investigación. Esta guía busca escoger el conjunto de pasos que nos dirán qué y cómo hacer para obtener el resultado que hemos establecido en los objetivos de un proyecto (CORDERO, 2012).

Las metodologías de desarrollo de software son un marco que se utiliza para estructurar, planificar y controlar el proceso de desarrollo de un sistema de información. Este tipo de metodologías de desarrollo sólo están interesadas en el proceso de desarrollo de software, por lo que no implica ningún aspecto técnico, su única preocupación son obtener una planificación adecuada para el desarrollo de software (INTECO, 2009).

2.1.2. Sistemas de Información

Un sistema de información (SI) es un conjunto de elementos orientados al tratamiento y administración de datos e información. Son generados para cubrir una necesidad de información y para apoyar las actividades de una empresa o negocio (Cassidy, 1998). No siempre un sistema de información debe estar automatizado, es válido hablar de sistemas de información manuales.

2.1.3. Sistema Recomendación

Los sistemas de recomendación son software, herramientas y técnicas que proveen sugerencias de productos que pueden llegar a ser del agrado del usuario. Las sugerencias están relacionadas a varios procesos de toma de decisiones, como que producto comprar, que música escuchar, o que libro leer (Ricci et al., 2011). Su objetivo como los SR es presentar un contenido relevante, atractivo y personalizado.

2.1.4. Diseño metodológico

El diseño metodológico es la descripción de cómo se va a realizar la propuesta de intervención, son los pasos a seguir para generar la información que el proyecto requiere. El Diseño metodológico, es también una relación clara y concisa de cada una de las etapas de la intervención en un proyecto (Larson, 2014).

2.1.5. Motor de recomendación

El agente que se desarrolla para que aprenda patrones para hacer la parte de personalización y recomendación. El motor de recomendación ayuda al consumidor mediante la selección de productos que pueden llegar a ser de su agrado, esta recomendación está basada en las búsquedas previas del usuario, navegación en Internet, compras, preferencias (Konstan and Riedl, 2012).

2.1.6. Personalización

La personalización es la construcción de lealtad del cliente mediante la construcción de una relación uno a uno a través de la comprensión de las necesidades de cada individuo, además es ayudar a satisfacer un objetivo de manera eficiente y con conocimiento aborda la necesidad de cada individuo en un contexto dado. Preferencia

Circunstancia de preferir o de ser preferida por alguien una persona o cosa sobre otras personas o cosas (RAE, 2001).

2.1.7. Sugerencia

Idea que se sugiere o se propone a una persona para que la tenga en consideración o piense en ella a la hora de hacer algo (RAE, 2001).

2.1.8. Recomendación

Consejo que se da a una persona por considerarse ventajoso o beneficioso (RAE, 2001).

2.1.9. Factor

Elemento, condicionante que contribuye a lograr un resultado. Entendemos por factores a aquellos elementos que pueden condicionar una situación, volviéndose los causantes de la evolución o transformación de los hechos. Un factor es lo que contribuye a que se obtengan determinados resultados al caer sobre él la responsabilidad de la variación o de los cambios (RAE, 2001).



2.2. Sistemas de recomendación

En la vida cotidiana, la gente confía en las recomendaciones de otras personas por conversaciones, cartas de referencia, informes de noticias, encuestas generales, guías de viajes y demás. Los sistemas de recomendación ayudan y promueven este proceso social natural para ayudar a la gente seleccionar a través de los libros disponibles, artículos, páginas web, películas, música, restaurantes, chistes, productos comestibles. Un SR ayuda a encontrar información interesante y valiosa para los usuarios (Yang et al., 2014).

Los SR actuales personalizan las páginas de sus clientes y ofrecen un contenido relevante, productos y publicidad para ellos (Adomavicius et al., 2008). Gracias a esta herramienta, es posible personalizar la oferta digital mediante sugerencias construidas según las preferencias del usuario. Así se ofrece una experiencia de compra satisfactoria que fortalezca la relación entre el cliente y el comercio electrónico. Un SR es una pieza compleja de software, que a su vez es una parte de un sistema mayor.

Con el fin de comprender mejor los sistemas de recomendación, sus dimensiones y algoritmos, hay varios ejemplos útiles utilizados en sitios web populares como Netflix, Pandora, YouTube y Amazon.

- Pandora crea listas musicales personalizadas mediante la predicción de lo que vamos a querer escuchar.
- Amazon.com es quizás el sitio de comercio electrónico más famoso que implementa recomendaciones. Basado en las compras del usuario y la actividad dentro del sitio web, Amazon recomienda libros y otros artículos que pueden ser del interés de sus clientes. Amazon evalúa las preferencias de diferentes maneras, por ejemplo, se le pide al usuario valorar productos, hace seguimiento a las imágenes que el usuario agranda, utiliza lo que el usuario coloca en una lista de productos

deseados, considera las compras reales, así como los productos que el usuario busca durante su sesión. Además, utiliza la ruta que ha recorrido el usuario a través de sus sitios web: las páginas que ha visto y artículos cliqueados. Al final combina los datos para construir un perfil de preferencias. Empresas como Amazon recogen una inmensa cantidad de datos sobre sus clientes, casi cualquier acción tomada mientras se está en el sistema se almacenara para su uso futuro (Konstan and Riedl, 2012).

- Netflix recomienda igualmente películas que pueden ser de interés de sus clientes, y es famoso el premio de \$ 1.000.000 que ofreció a los investigadores que podrían mejorar la calidad de sus recomendaciones (OWEN et al., 2012).
- Sitios de redes sociales como Facebook que usan técnicas de recomendación para identificar personas con probabilidades de ser amigos, y que aún no están conectadas.

2.3. Modelo Básico de un sistema de recomendación

El modelo básico de un sistema de recomendación tiene: usuarios como la gente que tiene preferencias, productos que vamos a recomendar, y ratings o puntajes que expresan opiniones y el motor de recomendación.

Los productos son las cosas u objetos que están siendo recomendadas para un usuario. Por ejemplo, los artículos son a menudo productos, artículos de noticias, canciones o películas. Estos artículos se pueden caracterizar por sus respectivos metadatos que incluyen títulos, etiquetas o palabras clave. Por ejemplo, los artículos de noticias se pueden caracterizar por categoría de contenido, las canciones se pueden caracterizar por los artistas y el género, y las películas se pueden caracterizar por género y director (Casinelli, 2014).

Los usuarios son las personas que a menudo necesitan asistencia u orientación en la elección de un elemento dentro, el uso de recomendaciones puede ayudarles a tomar una mejor decisión (Casinelli, 2014).

Una calificación o rating es un tipo de preferencia explícita que representa una relación entre un usuario y un elemento. Todas las calificaciones pueden describir, por ejemplo, cómo un usuario se siente acerca de ciertos artículos. Un ejemplo de una calificación explícita es un usuario calificar una película con cinco estrellas (Casinelli, 2014).

El SR ofrece recomendaciones basadas en las preferencias del usuario y tal vez en las preferencias de usuarios con cierta similitud. Para que un SR pueda cumplir con su objetivo, necesita recopilar información del usuario a través de un proceso de retroalimentación implícita o explícita. En el primer caso, se evalúan los ítems sin la intervención del usuario. El análisis se realiza sin el conocimiento del consumidor mediante la captura de la información que se obtiene a través de la interacción y la huella que deja éste en la web (i.e. el historial de compra). En la retroalimentación explícita, el consumidor valora

conscientemente los contenidos mediante un sistema de calificación específico (encuesta, puntuación, ranking); expresando de manera inequívoca su interés en el ítem. Todos estos datos permiten al sistema incorporar nuevas preferencias que enriquezcan el proceso de recomendación.

El motor de recomendación es un sistema que, a través de un algoritmo, analiza las características de los ítems (productos, servicios, datos), el nivel de utilidad para el cliente y crea una recomendación basada en los gustos de éste.

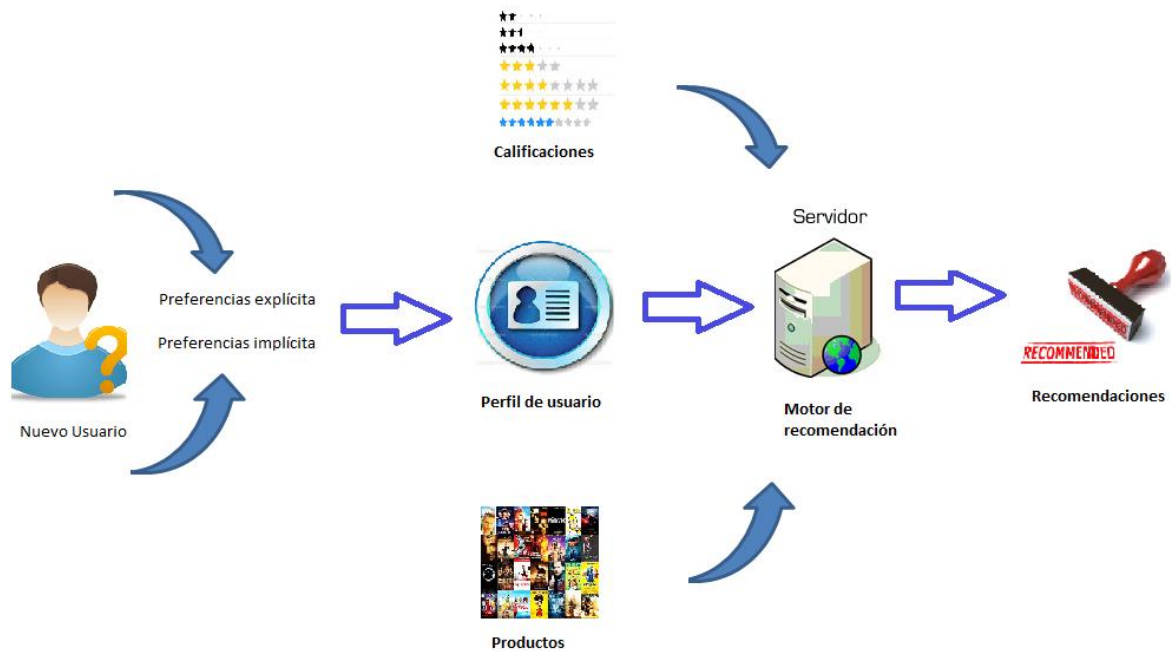


Figura 2. Modelo básico de recomendación

Fuente: Elaboración propia

2.3.1. El proceso de recomendación

2.3.1.1. Configuración de la recomendación

El SR debe tener una configuración específica para dar las recomendaciones. Es importante considerar el modelo de negocio al cual hace servicio la recomendación, que tipo de producto se está recomendado y bajo que contexto se ofrece la recomendación.

2.3.1.2. Adquirir información

El sistema debe poder adquirir información de las preferencias del usuario, esta recolección debe ser implícita o explícita. Es importante adquirir la información correcta, actualizada y que tenga calidad en los datos adquiridos. La adquisición de información es uno de los procesos más importantes para el desarrollo de las recomendaciones.

2.3.1.3. Proceso de la Recomendación

Basado en el conocimiento de preferencias del usuario el sistema busca las recomendaciones más adecuadas. El SR debe ser capaz de ofrecer recomendaciones a los usuarios, teniendo en cuenta la información contenida en su perfil. El SR debe ofrecer una serie de recomendaciones que aún no han sido vistas y que se estima sean del agrado del usuario. El sistema debe actualizar la información de recomendación cada cierto tiempo, incorporando nuevas peticiones que los usuarios hayan hecho en el servicio, y los cambios en los perfiles de usuarios (Araque, 2012).

2.3.1.4. Presentación de la recomendación

Las recomendaciones deben estar configuradas de forma que no interrumpen al usuario, deben ser una ayuda, no una fuente de molestia.

2.3.2. Datos de entrada y salida del sistema

Se debe identificar las principales características de los datos que pueden influir en el diseño y los resultados del sistema de recomendación. El proceso de recomendación comienza con la recopilación de datos a través de diferentes canales de interacción.

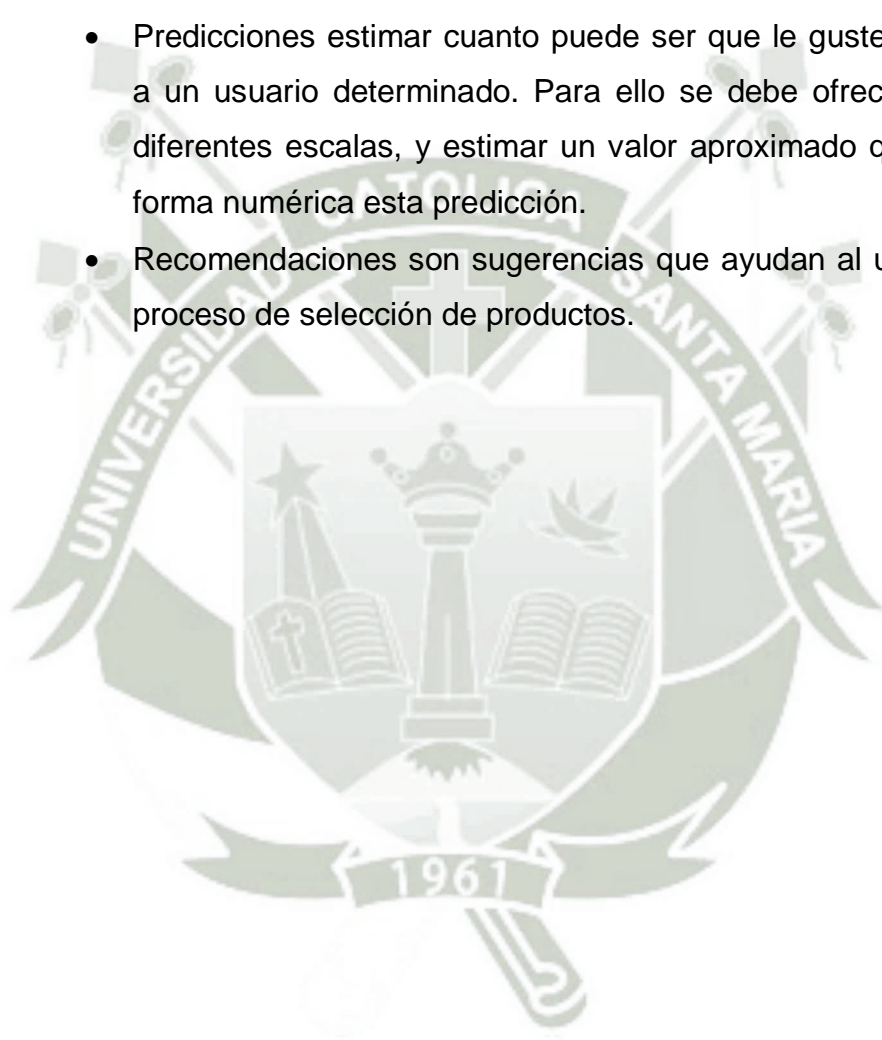
2.3.2.1. Datos de entrada

- Calificaciones (también llamados votos o valoraciones). Expresan la opinión de los usuarios sobre los productos. Las calificaciones son normalmente provistas por un usuario, y siguen una escala numérica específica. Las calificaciones pueden ser generadas de forma implícita del historial de compra del usuario, del registro de la web, de las visitas a los enlaces web, los hábitos de búsqueda o de otros accesos a la información.
- Datos demográficos: Se refieren a información como la edad, el género y la educación de los usuarios. Este tipo de datos son difíciles de obtener, y son normalmente recolectados de manera explícita del usuario (Tsihrintzis et al., 2013).
- Datos de contenido: Está basado en el análisis de contenido de productos calificados por el usuario. Las características extraídas vía este análisis son usadas como ingreso de los algoritmos basados en contenido con el objetivo de inferir un perfil de usuario más preciso (Tsihrintzis et al., 2013).

- Preferencias del usuario: Para conseguir las preferencias del usuario es necesario saber qué es lo que el usuario quiere. Un SR debe obtener preferencias de usuarios en relación con su contexto correspondiente.

2.3.2.2. Datos de salida

- Predicciones estimar cuanto puede ser que le guste un producto a un usuario determinado. Para ello se debe ofrecer al usuario diferentes escalas, y estimar un valor aproximado que refleje de forma numérica esta predicción.
- Recomendaciones son sugerencias que ayudan al usuario en su proceso de selección de productos.



2.4. Necesidad de Sistemas de Recomendación

Los sistemas de recomendación están en muchos sitios de comercio electrónico actuales. Muchos de los mayores sitios web de comercio ya están utilizando los SR para ayudar a sus clientes a realizar sus compras. Un SR aprende de los gustos del cliente y recomiendan los productos que va a encontrar más valioso de entre los productos disponibles. Los productos pueden ser recomendados en base a las ventas del sitio web, en base a los datos demográficos de los clientes, o en base a un análisis del comportamiento de compra anterior del cliente.

En términos generales, estas técnicas son parte de la personalización en un servicio web, porque ayudan al sitio web de adaptarse a cada cliente (Schafer et al., 1999).

- La mayoría de visitantes en buscadores en línea se van sin hacer ninguna compra. Los sistemas de recomendación ayudan a convertir a los visitantes casuales en compradores. Es muy posible que los usuarios dejen la página sin comprar nada, esto debido a que ellos no encontraron ningún producto de utilidad (Neumann, 2009).
- Incrementar el número de productos vendidos. Este objetivo es logrado porque los productos recomendados se aproximan más a las necesidades específicas del cliente. Además incrementa la venta cruzada mediante la recomendación de productos adicionales, es decir productos asociados con los productos en el carrito de compra.
- Venta de productos más productiva. Al momento de hacer las recomendaciones se trata de ofrecer todos los productos que existen en el stock del proveedor, por lo cual se trata de asegurar la cobertura de venta de la totalidad de los productos.

- Reduce la carga de información a los usuarios y hay un mejor entendimiento de que es lo que el usuario desea. Los sistemas de recomendación ayudan a los usuarios a filtrar más rápidamente la información y de esta manera permiten al cliente la visualización de diferentes productos que probablemente no encontraría fácilmente.
- Incrementar la satisfacción del usuario. Es bien sabido que un SR bien implementado mejora considerablemente la experiencia del usuario dentro del sistema. El usuario va a encontrar las recomendaciones interesantes y relevantes. La combinación de efectividad y precisión, harán que el usuario incremente su satisfacción (Neumann, 2009).
- Los sistemas de recomendación son una forma para automatizar la personalización en masa para los sitios de comercio electrónico. Las empresas se centran cada vez más en el valor a largo plazo de los clientes y buscan siempre mejorar el servicio (Schafer et al., 1999).

2.4.1. Justificación para el uso de sistemas de recomendación

2.4.1.1. Valor para el cliente

- Encontrar cosas que son del gusto del cliente
- Reducción de tiempos en la búsqueda de información
- Afinar el conjunto de opciones de un producto requerido
- Ayuda a la exploración de productos y nuevas opciones
- Entretenimiento

2.4.1.2. Valor para el Proveedor

- Mejora el servicio al cliente haciéndolo más personalizado.
- Incrementa la confianza del cliente y su lealtad.
- Incrementa el número de ventas cruzadas.

- Ayuda a encontrar nuevas oportunidades de negocio y persuasión.
- Ayuda a obtener más información acerca del cliente.



2.5. Problemas que enfrentan este los SR actuales

- Arranque frío

El problema de arranque en frío se produce cuando un nuevo usuario o artículo apenas ha sido ingresado en el sistema. Los nuevos artículos no pueden ser recomendados hasta que un usuario los califique, y los nuevos usuarios no pueden obtener sus recomendaciones a menos que produzcan cierta cantidad de calificaciones que ayude a determinar sus preferencias. La gravedad de este problema depende del modelo de recomendación que se use para hacer el cálculo de la recomendación.

- Cobertura

La cobertura puede ser definida como el porcentaje de artículos que el algoritmo podría proporcionar para hacer las recomendaciones. El problema de la cobertura reducida se produce cuando el número de calificaciones de los usuarios puede ser muy pequeño en comparación con el gran número de artículos en el sistema, en consecuencia el SR es incapaz de generar recomendaciones para todos los productos.

- Novedad

A partir de los elementos que un SR recomienda a los usuarios, hay elementos conocidos por los usuarios, y hay artículos que son nuevos (novedad) y desconocido para ellos. Por lo tanto, hay una competitividad entre el deseo de novedad y el deseo de recomendaciones de alta calidad. Esto significa que un el SR no debería recomendar elementos que no son deseables. Una alta tasa de novedad producirá recomendaciones de mala calidad debido a que los usuarios no serán capaces de identificar la mayor parte de los elementos de la lista de recomendaciones (Tsihrintzis et al., 2013).

- Escasez de calificaciones

El número de elementos que son calificados por los usuarios es por lo general un pequeño subconjunto de los productos totales. Por ejemplo, en Amazon, si los usuarios activos pueden haber comprado el 1% de los artículos y la cantidad total de elementos es de aproximadamente 2 millones de libros, esto significa que sólo hay 20, 000 de los libros que están calificados (Tsihrintzis et al., 2013).

- Escalabilidad

Los SR son implementados para grandes sitios de comercio electrónico y deben hacer frente a un número cada vez mayor de usuarios y artículos. Por lo tanto, se requiere una cantidad cada vez mayor de recursos computacionales conforme la cantidad de datos crece. Un método de recomendación podría ser eficiente cuando el número de datos es limitado, pero podría ser muy lento y no escalable con un número mayor de datos. Tal método sería incapaz de generar un número satisfactorio de recomendaciones a partir de una gran cantidad de datos.

- Parcialidad

Se refiere a prejuicios conscientes o inconscientes de los contribuidores. Un ejemplo de parcialidad se encuentra cuando los autores o editores contribuyen con buenas críticas y calificaciones sus propios libros y con calificaciones negativas los libros de sus competidores (Neumann, 2009). Falta de sinceridad en las calificaciones de los productos. En los casos en que cualquier persona puede proporcionar recomendaciones, las personas pueden dar numerosas recomendaciones positivas para sus propios productos y recomendaciones negativas para sus competidores.

- Credibilidad

Las recomendaciones son solo percibidas como útiles si se puede confiar en ellas. Los usuarios son muy sensibles a los problemas de credibilidad, si sienten que el SR solo busca vender productos, perderán la fe en el servicio.

- Privacidad

Los usuarios pueden preocuparse por los métodos que el SR usa para obtener la información de sus preferencias.

- Interoperabilidad

El SR debe poder ser integrado con otros servicios y cooperar con otros sistemas. Esto es importante porque el SR, va usar información de productos, clientes, ventas pasadas, estudios de mercado. El SR debe poder interactuar con los otros sistemas del negocio, y reducir problemas de interoperabilidad como acceso a las fuentes con diferentes estructuras de datos, formatos diferentes, sinonimia, protocolos de seguridad, entre otros.

- Rendimiento

Las recomendaciones deben ser ofrecidas en tiempo real, los usuarios no toleran un aumento en el tiempo de respuesta del servicio (Jose C. Cortizo et al., 2010).

2.6. Características que deben ser consideradas en el diseño del SR

- Escalabilidad

Los algoritmos deben ser capaces de manejar grandes volúmenes de datos, gran cantidad de usuarios, y los cambios en las interacciones de usuarios y productos. Algunos algoritmos funcionan relativamente bien con poca información, hasta que el sistema crece a gran escala.

- Privacidad

Los sistemas de recomendación explotan información personal del usuario, en el intento por hacer mejores sugerencias, el sistema tratará de buscar la mayor cantidad de información posible. El impacto negativo es que puede ser que los usuarios sientan que el sistema sabe demasiado de ellos. También se debe proteger la información obtenida previniendo la fuga maliciosa de información (Ricci et al., 2011).

- Diversidad

Los productos recomendados al cliente deben ser variados. El usuario debe tener variedad de selección. Es decir dentro de la cobertura de productos ofrecidos, debe haber un rango variado de propuestas para las preferencias del usuario.

- Precisión

La precisión de un SR depende de tan acertada sea la sugerencia ofrecida para el usuario (Wu et al., 2012).

- Usabilidad

Se refiere a la capacidad de un software de ser comprendido, aprendido, usado y ser atractivo para el usuario, en condiciones específicas de uso. Los principios básicos de la usabilidad son facilidad de aprendizaje, predictibilidad, tiempo de respuesta y consistencia.



2.7. Tipo de sistemas de recomendación

Cada día se crean nuevos algoritmos para hacer a los sistemas de recomendación más eficientes, se hacen modificaciones para hacerlos más escalables y eficientes.

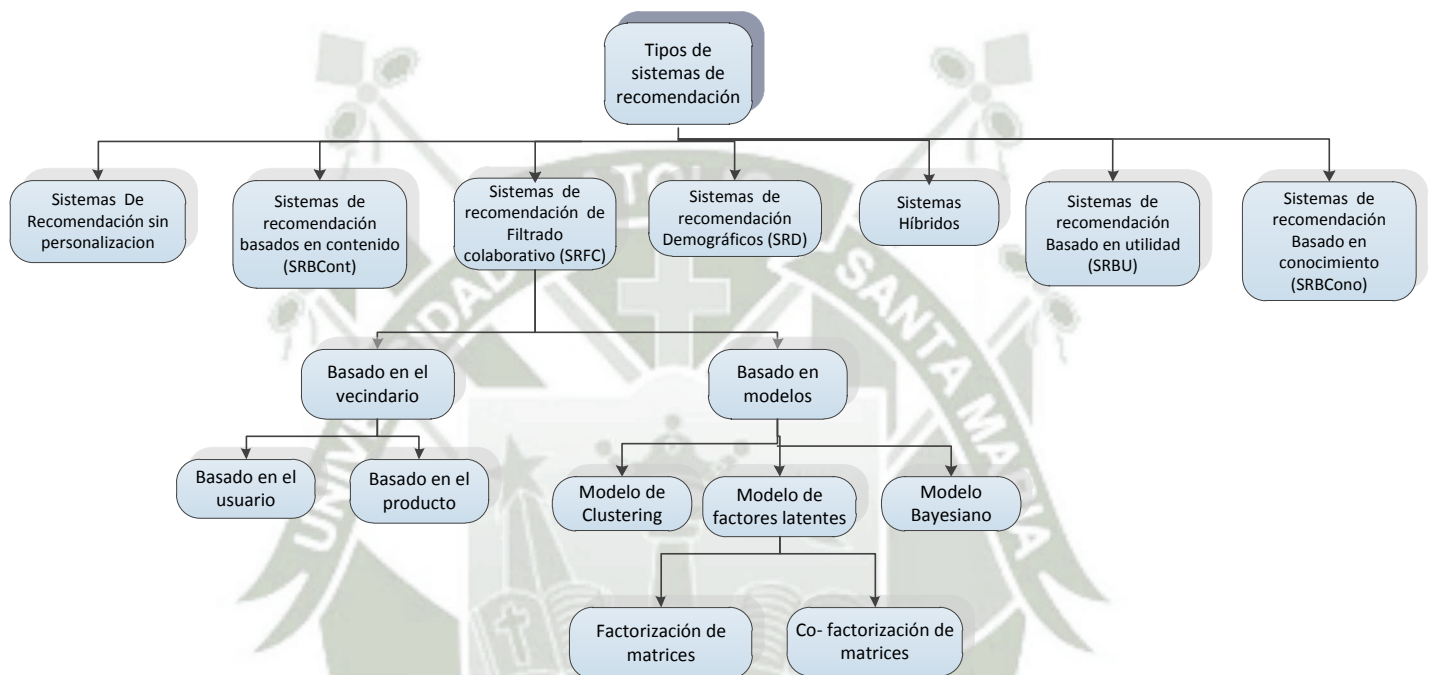


Figura 3. Tipos de sistemas de recomendación

Fuente: (Su and Khoshgoftaar, 2009)

2.7.1. Sistemas de recomendación no personalizados

Son sistemas que dan sus recomendaciones al público en general, la recomendación se basa en el promedio del gusto general, no en un gusto específico, por ejemplo el ranking de las 10 canciones más sonadas. Las

recomendaciones son independientes del cliente, por lo que cada cliente obtiene las mismas recomendaciones. Este tipo de recomendación es fácil de efectuar y puede ser muy eficiente para ciertas aplicaciones como rankings de canciones o libros (Schafer et al., 1999).

Los sistemas de recomendación no personalizados son automáticos, y requieren poco esfuerzo al cliente para generar la recomendación. El problema de estas recomendaciones es que no toman en consideración el contexto en el que se desenvuelve el usuario. Además al trabajar solo con el promedio de ratings se deja de lado ciertos aspectos, puede ser que algunos productos que sean malos solo tengan puntajes positivos, y sea sugerido como un producto de calidad. Este problema es debido a que las opiniones pueden ser muy variadas, y existen usuarios con opiniones diferentes al pensar común de esa comunidad en especial. Es por ello que surgen los sistemas de recomendación personalizados.

Ventaja

- Las recomendaciones son independientes del cliente.
- Es fácil de diseñar.

Desventaja

- Las recomendaciones no son personalizadas.



Figura 4. Ejemplo de Sistemas de recomendación no personalizados

Fuente: www.netflix.com/pe/

2.7.2. Sistemas de recomendación demográficos (SRD)

Clasifican a los usuarios en grupos demográficos basándose en ciertos atributos personales, y proporcionan recomendaciones potencialmente interesantes para cualquier persona perteneciente a dicho grupo demográfico (Torres, 2007). Su principio es la idea de que una persona con características demográficas similares como la edad, sexo, nivel de educación, domicilio tienen gustos similares a otras personas con características demográficas similares (Gallardo, 2012). Un ejemplo del uso de estos sistemas es el siguiente, el tipo de música varía mucho de persona a persona, pero si nos enfocamos en solo un grupo demográfico, podemos percibir un grado de similitud en las preferencias por un género.

Ventaja

- Si el análisis de los grupos demográficos es correcto, la recomendación ofrecida por el sistema puede llegar a ser muy efectiva.

Desventaja

- El problema principal de este tipo de técnica es la falta de datos, es difícil recoger los datos demográficos necesarios, porque las personas son resistentes a dar la información personal.
- La recolección de información es privada por lo que no se puede ofrecer una recomendación individual.
- Se necesita investigación estadística y/o social para saber cómo traducir los grupos culturales de la persona.

En la figura 5 se puede ver un ejemplo de un SRD, el sistema ofrece las siguientes recomendaciones musicales, al grupo de edades entre 55 y 65 le recomienda el álbum de Camilo Sesto y al grupo de edades entre 20 y 30 le recomienda el álbum de Enrique Iglesias.

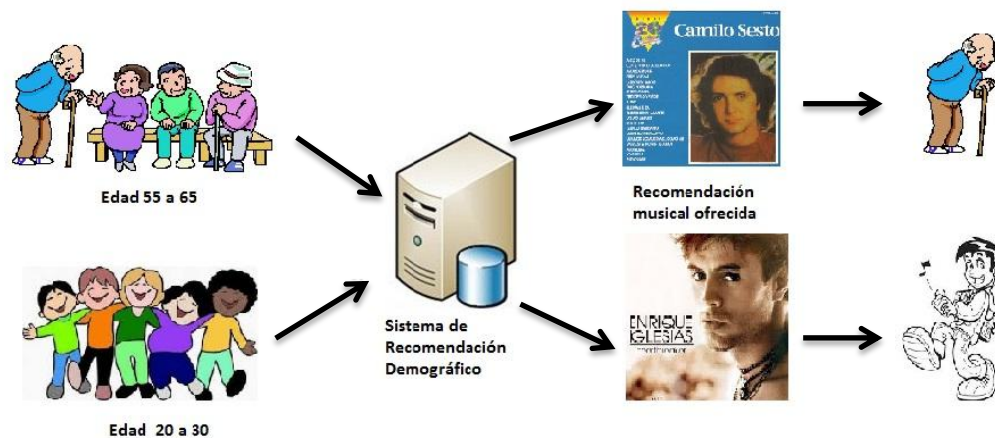


Figura 5. Ejemplo de Sistemas de recomendación demográficos

Fuente: Elaboración propia

2.7.3. Sistemas de recomendación basados en conocimiento (SRBCono)

A diferencia de otros sistemas de recomendación, los basados en conocimiento no dependen de grandes cantidades de información sobre productos puntuados (basados en contenido) y usuarios particulares (colaborativos) sino que lo único que necesitan es tener un conocimiento general sobre el conjunto de objetos y un conocimiento informal de las necesidades del usuario. A partir de conocimiento sobre los usuarios y los productos se persigue un razonamiento que indique qué producto cumple los requerimientos del usuario, dejando de lado las valoraciones que el usuario pueda hacer (Torres, 2007). Dispone de información acerca de cómo un ítem satisface una necesidad del usuario y establece relación entre una necesidad y una recomendación. El perfil puede ser cualquier estructura de datos que soporte esta relación.

Ventajas

- La ventaja principal de estos sistemas es que se puede agregar fácilmente nuevas reglas de negocio y no sufre del problema de arranque frío que tienen los sistemas tradicionales.
- Sensible a los cambios de preferencias.
- Adaptable a las necesidades de los usuarios.

Desventaja

- La principal desventaja es su complejidad al momento de ser implementados.

2.7.4. Sistemas de recomendación basados en utilidad (SRBU)

Las recomendaciones basadas en la utilidad son un caso particular de las recomendaciones basadas en el conocimiento. Su objetivo es el de crear un valor de “utilidad” (beneficio, ventaja o interés) para los ítems a recomendar para un usuario en particular. En principio el valor de “utilidad” se basa en la utilización del conocimiento (Seguido, 2009). Estos sistemas comparan la necesidad del usuario con el conjunto de opciones disponible, mediante una función de la utilidad de cada objeto para el usuario; esa función sería su perfil. Estos sistemas realizan las recomendaciones basándose en el cálculo de la utilidad de cada objeto en particular con respecto al perfil determinado de cada usuario (Torres, 2007). Añaden la necesidad de crear una “función de utilidad”, que permite obtener un valor (de “utilidad”) para cada ítem a recomendar el cual proporciona al usuario que refleja el grado de satisfacción del ítem con sus necesidades, gustos o preferencias. Luego emplea técnicas de satisfacción de restricciones para escoger la mejor opción (Ricci et al., 2011). Los sistemas basados en utilidad añaden otras características que no pertenecen a los ítems en sí. Por ejemplo: los plazos de entrega de un ítem o la garantía de éste.

Ventajas

- Eliminación del problema del arranque en frío.
- Permiten expresar al usuario todas las consideraciones, referentes a tipo de ítem, necesarias para encontrar las recomendaciones que mejor encajen en el marco de sus necesidades (Seguido, 2009).
- Adaptable a las necesidades de los usuarios.
- Incluye características que no son propias de los productos.

Desventaja

- La elaboración de funciones de utilidad más complejas implica que los usuarios realicen la medición de pesos, valoraciones y asignación de función de utilidad a las diferentes propiedades de los ítems de forma muy precisa (Seguido, 2009).

2.7.5. Sistemas de recomendación basados en confianza (SRBC)

Son parecidos a los sistemas de recomendación basados en filtrado colaborativo (SRFC), la diferencia básica es que los SRFC basan su recomendación buscando usuarios similares a los que se les asigna un peso, los sistemas de recomendación basados en confianza SRBC basan su recomendación en usuarios en los que se confía. Este tipo de SR pueden en ocasiones ser más efectivos que los SRFC. Por ejemplo dos usuarios son bastante parecidos en sus puntajes, pero uno de ellos tiene gustos completamente diferentes a los del otro usuario en un tema en específico, el SRFC fallará en dar esta recomendación fuera del promedio. El SRBC encuentra este fenómeno y determina que el lazo de confianza de dos usuarios con opiniones totalmente diferentes en un tema en específico, puede verse afectado por este fenómeno. El sistema de recomendación basado en confianza utiliza el valor de cuanto valoramos ciertos temas sobre otros y de acuerdo a eso busca el vecindario. El problema de este método es cómo calcular la confianza que existe entre determinados usuarios, la forma más simple es cuando el usuario explícitamente indica que otro usuario es de confianza. Otra forma es inferir la confianza, la regla es que si el usuario confía en alguien, confíara en los amigos de su contacto.

Ventajas

- Resuelve algunas deficiencias de lo SRFC.

- Sensible a los cambios de preferencias.
- Adaptable a las necesidades de los usuarios.

Desventaja

- Calcular el nivel de confianza entre usuarios es realmente difícil.
- Los usuarios desean mantener su nivel de confiabilidad en privado, por lo cual es mucho más difícil llevar a cabo los cálculos requeridos.
- Ingeniería del conocimiento.

2.7.6. Sistemas de recomendación basada en contenido (SRBCont)

Los sistemas basados en contenido generan recomendaciones equiparando las preferencias del usuario (expresadas por éste de forma implícita o explícita) con los metadatos o características utilizadas en la representación de los recursos o productos, ignorando de esta forma la información relativa a otros usuarios. Este tipo de SR basa la recomendación efectuando un análisis de los atributos del producto, es decir se basan en las características asociadas al producto.

La construcción de preferencias del usuario se hace en base a la búsqueda de atributos clave o palabras clave, que son del agrado del usuario, se comparan los atributos del producto con el perfil del usuario, según estas preferencias se hace un mapeo de los productos. El sistema filtra los productos que son irrelevantes para el perfil del usuario y solo muestra los más destacados al gusto del usuario. Un ejemplo referente a películas, el usuario gusta de películas de ciencia ficción, entonces el motor de recomendación buscará las películas que tengan como atributo, ciencia ficción. Este tipo de SR no son considerados sociales, debido a que no se basan en relaciones sociales sino en las características del producto. Es recomendable usar este tipo de sistema de recomendación cuando la

aplicación tiene variedad de nuevos productos, y muchos de ellos no tienen puntuación alguna.

Ventajas

- Independencia de otros usuarios, las recomendaciones basadas en contenido exploran exclusivamente calificaciones proporcionadas por el usuario activo para construir su propio perfil, no necesitan de otros usuarios.
- Las explicaciones sobre cómo funciona el sistema de recomendación son transparentes, la lista de productos recomendados es seleccionada por sus atributos, esta explicación es clara y entendible para el usuario.
- Los sistemas de recomendación basados en contenido son capaces de recomendar artículos todavía no calificado por otros usuarios. Como consecuencia, ellos no sufren el problema del arranque frío.

Desventajas

- Se basan en el análisis de contenido de texto, por lo cual se hace muy difícil hacer recomendación de productos multimedia como imágenes, video, o audio.
- Los sistemas de recomendación basados en contenido no tienen ningún método inherente para encontrar algo inesperado, elaboran recomendaciones con un grado limitado de novedad. El sistema sugiere artículos cuyas características combinan con el perfil del usuario, por lo tanto, el usuario solo obtendrá productos similares a los ya calificados (Ricci et al., 2011).
- Estos sistemas, al igual que los sociales, son poco fiables cuando se dispone de poca información sobre el usuario.

- La calidad depende de la cantidad de información.
- No diferencia ítems homogéneos (mismos atributos)(Seguido, 2009).
- El problema de sinonimia refiere a la tendencia de tener productos que tienen nombres parecidos o similares, la mayoría de SRBCont no pueden descubrir ese problema y disminuyen su performance.

Técnicas usadas en los sistemas basados en contenido

- Aprendizaje automático.
- Recuperación de la información (Information retrieval) .

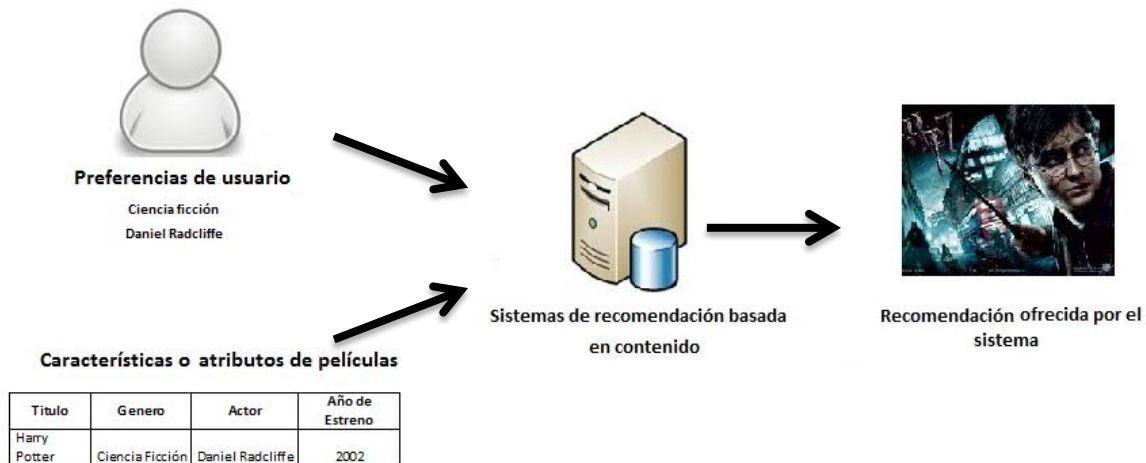


Figura 6. Ejemplo de Sistemas de recomendación basada en contenido

Fuente: Elaboración Propia

2.7.7. Sistemas de recomendación de filtrado colaborativo (SRFC)

Los sistemas de filtrado colaborativo (en adelante SRFC – collaborative filtering) son los métodos más populares hoy en día dentro de las técnicas

de recomendación. Al usuario se le recomendarán aquellos elementos elegidos anteriormente por gente con preferencias y gustos similares (Torres, 2007). La idea fundamental de los SRFC es que si el usuario 1 y el usuario 2, califican el producto n de manera similar, o tienen comportamientos similares (por ejemplo, compra, observación, gustos musicales), por lo tanto calificaran de manera similar a otros productos.

Los SRFC acumulan recomendaciones de usuarios sobre los productos, identifica similitudes entre usuarios y genera nuevas recomendaciones, es decir predice puntajes basado en la opinión de usuarios similares (Herlocker et al., 2000). Basados en las calificaciones de otros usuarios, se pueden estimar calificaciones para los productos que no han sido vistos por los usuarios nuevos a los cuales se les va proveer la recomendación. Esto quiere decir que el sistema debe tener varios usuarios, para encontrar usuarios compatibles. Esta técnica debe ser capaz de encontrar gente con intereses similares y medir el grado de similitud entre los usuarios del sistema basado en sus perfiles y así crear grupos de usuarios con características afines (Carrol, 2001). A los usuarios se les puede proveer recomendaciones de productos novedosos que hayan sido evaluados de manera positiva por los miembros de su grupo afín. El sistema en este punto es capaz de proporcionar recomendaciones personalizadas de aquellos elementos que resultan interesantes para el usuario.

La totalidad de la literatura existente está de acuerdo en que los sistemas basados en filtrado colaborativo trabajan recogiendo juicios humanos, expresados como votaciones de cada individuo para una serie de ítems en un dominio dado, y tratan de emparejar personas que comparten las mismas necesidades o gustos (Torres, 2007).

Las premisas de este método son dos, los gustos de las personas cambian en sincronía con los usuarios similares, es decir que se asume que los

gustos de ambos estarán en futura sintonía. La segunda premisa es, hay un dominio común al segmentar a los usuarios, y a su vez es posible encontrar usuarios con varios dominios comunes, es decir que les gusta por ejemplo el mismo tipo de restaurantes y también el mismo tipo de hoteles.

El principal problema de este enfoque es que sí no hay suficientes ratings para hacer las sugerencias de los productos, simplemente no son sugeridos a ningún usuario (Rodríguez, 2009).

Dentro de los sistemas de recomendación de filtrado colaborativo hay dos sub-ramas, los basados en vecindarios y los basados en modelos. Los sistemas basados en vecindarios harán la recomendación basados en las preferencias de los usuarios más cercanos. Por otro lado, los sistemas basados en modelos hacen las recomendaciones basados en modelos matemáticos y predicciones estadísticas.

a) Algoritmos basados en vecinos

Estos algoritmos utilizan la base de datos completa para generar una predicción. El funcionamiento de estos algoritmos es el siguiente: se utilizan técnicas estadísticas para encontrar un conjunto de vecinos al usuario activo y posteriormente se utilizan una serie de algoritmos que combinan las preferencias de esta vecindad para realizar las predicciones y recomendaciones. Estos algoritmos basados en vecindad son muy populares y exitosos en la práctica pero son también los que con más ferocidad sufren los problemas de escasez y escalabilidad (Zhong et al., 2013).

b) Algoritmos basados en modelos

Estos algoritmos proporcionan recomendaciones de ítems desarrollando primero un modelo de las puntuaciones de los usuarios sobre los ítems. Se utilizan técnicas estadísticas y una aproximación probabilística que

calcula el valor esperado de una predicción del usuario dados sus puntuaciones sobre otros ítems. Es decir, estos algoritmos miran en el conjunto de ítems que el usuario activo ha puntuado o evaluado y calcula cuan similar son estas puntuaciones con respecto al ítem nuevo con el fin de realizar una predicción para el mismo (Jiang et al., 2012). Las aproximaciones basadas en modelos utilizan la colección de valoraciones para aprender un modelo, el cual será utilizado a la hora de realizar las futuras predicciones.

Ventajas

- Su sencillez con respecto a otras técnicas más complejas y que ofrecen resultados similares.
- Su capacidad para tratar con elementos difíciles de analizar mediante procesos informáticos.
- El tipo de información con la que estos sistemas manejan se adapta de forma simple y directa a la información social, es decir, las calificaciones de un entorno social.
- Presentan una interesante capacidad a la hora de proporcionar unas recomendaciones personalizadas de calidad.
- Reducen la sobrecarga de información en los distintos ámbitos en los que se han utilizado, delimitando de una forma clara y sencilla las distintas alternativas.
- Proporciona una justificación concisa e intuitiva para las predicciones calculadas. Esto puede ayudar al usuario a comprender mejor la recomendación y su relevancia, y podría servir como base para un sistema interactivo donde los usuarios pueden seleccionar los vecinos y mejorar la recomendación (Kabore, 2012).

Desventajas

- Los sistemas de recomendación colaborativos necesitan una gran cantidad de datos, muchos usuarios puntuando muchos ítems similares para así poder calcular los grupos de vecinos y, en base a ellos, realizar las recomendaciones. Si en nuestra base de datos tenemos pocos usuarios o pocas puntuaciones, nuestra matriz de puntuaciones será muy escasa y los cálculos de vecindad, predicción y recomendación no pueden ser realizados con la suficiente seguridad y exactitud obteniendo por lo tanto unas recomendaciones de baja calidad.
- En los SRFC los ítems nuevos, que tienen muy pocas o, incluso, ninguna puntuación no van a ser recomendados prácticamente nunca. De la misma forma, los nuevos usuarios en el sistema recibirán muy pobres recomendaciones debido a que ellos han puntuado muy pocos ítems y se hace difícil encuadrarlos en algún grupo de vecinos.
- En los SRFC también habrá que tener en cuenta aspectos de privacidad y debido a que algunas personas no quieren que se conozcan sus hábitos o preferencias.
- El problema de la oveja negra se puede observar cuando las opiniones de un usuarios son radicalmente diferentes de otros usuarios, lo que ocasiona que las técnicas que basan sus recomendaciones en usuarios similares no funcionen (Su and Khoshgoftaar, 2009).
- La calidad de la recomendación depende de la cantidad de información (Seguido, 2009).
- Los usuarios con gustos poco comunes tiene dificultad para identificarse con un vecindario.

- El problema de la matriz dispersa surge cuando la cantidad de productos, usuarios o ambas crece de manera acelerada, pero no existen calificaciones de los usuarios a los productos, por lo cual hay espacios vacíos que demoran el proceso de recomendación.

Técnicas usadas

Algunos algoritmos colaborativos se fundamentan en:

- Uso de redes bayesianas y derivados.
- Métodos de clustering para filtrado colaborativo.
- Otro método es utilizar modelos de aprendizaje de forma que se pueda tratar el filtrado colaborativo como un problema de clasificación.

Tabla 1. Ejemplo de un Sistema de Recomendación de Filtrado Colaborativo

	Harry Potter	Star Wars	Pretty Woman	Titanic	Rocky
Johana	5	3	5	5	?
Pablo	1	3	1	1	5
Klein	2	1	1	1	2
Roger	3	4			
Mayra	5	3	5	5	3

Fuente: Elaboración Propia

El SR necesita saber si la película Rocky es recomendable para el usuario Johana. El usuario más similar es Mayra, basado en ello puede sugerir la película Rocky, indicando que la preferencia puede llegar a ser un 3.

Por otro lado, el usuario Pablo, es completamente diferente, el opuesto también puede ser usado para el cálculo de recomendación.

2.7.8. Sistemas de recomendación híbridos

El enfoque híbrido combina el enfoque basado en contenido y el filtrado colaborativo. El objetivo principal de métodos híbridos es evitar las deficiencias de los métodos anteriormente enumerados o mejorar el rendimiento del sistema. Varios estudios comparan empíricamente el rendimiento del enfoque híbrido con los métodos de colaboración y basada

en el contenido puro y demuestran que los métodos híbridos pueden proporcionar recomendaciones más precisas que los enfoques puros. Estos métodos también se pueden utilizar para superar algunos de los problemas comunes en los sistemas de recomendación, tales como arranque en frío y el problema de escasez. Hay muchas maneras diferentes de combinar el filtrado de contenidos basado y colaborativo.

- Implementar ambos métodos por separado y combinar las salidas de estos métodos.
- Añadir algunas de las características basadas en el contenido de la filtración de colaboración.
- Añadir algunas de las características de colaboración para el filtrado basado en contenido.
- Desarrollar un modelo que se aplica tanto a las características basadas en contenido y de colaboración.

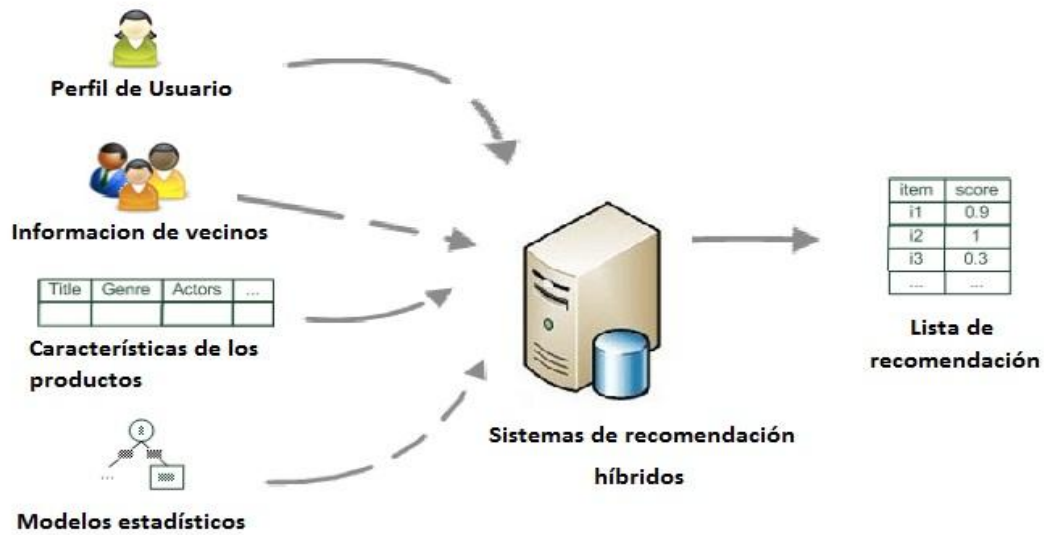


Figura 7. Ejemplo de Sistemas de recomendación híbridos

Fuente: Elaboración propia

CAPÍTULO 3

METODOLOGÍAS PARA EL DESARROLLO DE SISTEMAS DE RECOMENDACIÓN

La industria de desarrollo de software es dinámica y está en constante evolución, en particular, con el predominio de Internet y las nuevas tecnologías. Cada día se dan grandes cambios de alta velocidad, la incertidumbre y la complejidad son cada vez las características de muchos proyectos de desarrollo de software. La capacidad de crear de forma rápida y entregar software que cumpla con las necesidades del cliente se ha convertido en una ventaja competitiva innegable. Debido a esto, la selección de la metodología de desarrollo de software adecuado se ha convertido en esencial para cumplir con los requisitos de cronogramas, costos y calidad. Por el contrario la elección de una metodología inadecuada aumenta el riesgo el proyecto y retrasa el proyecto. En otras palabras, los diferentes proyectos requieren diferentes metodologías de desarrollo. Sin embargo, debido a las numerosas metodologías están disponibles en el mercado, puede ser difícil seleccionar la más apropiada. Además, la gente tiende a ser

emocional en la selección de la metodología. La selección de una metodología adecuada de desarrollo de un sistema de recomendación también es crucial, como en todo proyecto de desarrollo del software.

En el siguiente capítulo se hará una revisión de las metodologías de desarrollo de software. Se detallara sus ventajas y desventajas, además de los proyectos más adecuados para su implementación. Esta información será de suma importancia para el desarrollo del siguiente capítulo, debido a que uno de los factores críticos para hacer el desarrollo de un SR es la selección del tipo de metodología de desarrollo de software.

3.1. Constitución de una metodología

Las metodologías organizan conjuntos de reglas de comportamiento y técnicas en un enfoque coherente, que prescribe la forma de abordar los principales problemas de desarrollo. El propósito de una metodología es ayudar a desarrollar de manera satisfactoria sistemas, esto es percibido como generar, evaluar, controlar y llevar a cabo los cambios propuestos para el sistema.

3.1.1. Modelo de Cascada

El modelo de cascada es el modelo de ciclo de vida de desarrollo de software tradicional. Este modelo aclara el proceso de desarrollo de software en un flujo secuencial lineal que significa que cualquiera de las fases del proceso de desarrollo comienza sólo si la primera fase se ha completado. Este enfoque del desarrollo no promueve el proceso de volver a la fase anterior para manejar los cambios en los requisitos.

El desarrollo del proyecto es dividido en fases secuenciales, fases que se superponen e intervienen una con la otra. Se pone especial énfasis en el planeamiento de objetivos, horarios, plazos límites, presupuestos, y la implantación del sistema completo a tiempo (Cicilia, 2009).

Mantiene un control estricto a través del tiempo de vida del proyecto usando documentación dedicada, y revisiones formales al final de cada fase en donde se deberá obtener la aceptación firmada del usuario y la oficina a cargo del desarrollo del proyecto.

Ventajas

- Ideal para dar soporte a equipos con menos experiencia en el desarrollo de proyectos. El orden secuenciado de cada paso, los controles, la confiabilidad y administración del sistema, aseguran la calidad y el correcto progreso de desarrollo del software.
- Sus fases son conocidas por los desarrolladores.
- El modelo de cascada es muy simple y fácil de entender, es por eso que es muy beneficioso para el principiante o para el desarrollador novato.

Desventajas

- Es una metodología que extiende los tiempos de desarrollo del proyecto, lo que la hace más costosa.
- Es menos flexible que otros métodos, tiene poca adaptabilidad al cambio, los cambios en las últimas fases del proyecto son más costosos de implementar y pueden producir retrasos en los plazos de entrega debido a su estructura de estrictos controles.
- Durante la fase de diseño y codificación es probable encontrar requerimientos inconsistentes, componentes del sistema que no fueron considerados o aparecen nuevas necesidades de desarrollo que no fueron tomadas en cuenta en su momento. Algunos de estos problemas llegan incluso a ser descubiertos de manera tardía al momento de hacer las pruebas del sistema. Cualquier error de diseño detectado en la etapa

de prueba conduce necesariamente al rediseño y nueva programación del código afectado, aumentando los costos del desarrollo.

- El rendimiento del sistema solo puede ser probado y medido hasta que el sistema está completamente codificado.
- Produce documentación excesiva y mantener la información actualizada conforme avance el proyecto se transforma en una pérdida de tiempo.

Situaciones donde es más apropiado

- Este tipo de modelo de desarrollo funciona con mayor eficacia en los proyectos más pequeños, donde son muy bien entendidos los requisitos.
- El proyecto tiene objetivos y soluciones claros. El equipo de desarrollo del proyecto entienden completamente el negocio y la aplicación a desarrollar.
- No existe presión por una inmediata implementación.
- Los requerimientos del proyecto se puede afirmar sin ambigüedad y comprensiblemente.
- Los requerimientos del proyecto son estables o no cambia durante el ciclo de vida de desarrollo de sistemas.
- Los miembros del equipo carecen de experiencia en el desarrollo de proyectos.

Situaciones donde es menos apropiado

- Proyectos con requerimientos en continua evolución o aparecen cambios por alguna razón, cambio en las expectativas, cambio en el presupuesto o cambio en la tecnología.

3.1.2. Modelo por Prototipos

Una metodología por prototipos es el proceso de desarrollo de software que permite a los desarrolladores crear sólo el prototipo de la solución para demostrar su funcionalidad a los clientes, y hacer las modificaciones necesarias antes de desarrollar la solución final real (Certified, 2014). La mejor característica de esta metodología es que resuelve muchos problemas que a menudo se produce en un modelo de cascada tradicional.

En el modelo de prototipos, los requerimientos son cuidadosamente examinados, y sólo esos que son bien comprendidos son seleccionados para el primer incremento. Los desarrolladores construyen una implementación parcial del sistema. El prototipo es desarrollado, los usuarios lo usan, y proveen retroalimentación a los desarrolladores. Basada en esta retroalimentación, la especificación de requerimientos es actualizada, y una segunda versión del producto es desarrollada y desplegada. El proceso se repite indefinidamente (Cicilia, 2009). Este enfoque pretende resolver algunos riesgos inherentes de los proyectos de desarrollo de sistemas para ello divide el proyecto en segmentos, esto provee mayor facilidad al cambio.

Ventajas

- Cuando el prototipo se muestra a los clientes, se obtiene una clara comprensión de la funcionalidad del software.
- Como los riesgos potenciales se pueden identificar en fases tempranas, las medidas correspondientes pueden ser tomadas rápidamente.
- El usuario está involucrado a través del proceso, lo cual incrementa aceptación en la entrega final.

- Provee un entendimiento fundamental de las reglas del negocio, al final se entiende las prioridades del mismo, así se puede llegar a solucionar los problemas más urgentes primero.
- Aborda la incapacidad del usuario para especificar sus necesidades de información y la dificultad del analista de sistemas de entender el ambiente del usuario.
- Alienta innovación y diseños flexibles.

Desventajas

- Los prototipos son costosos.
- La aprobación de prototipos y el control no es estricto.
- Puede ser que el análisis de desarrollo del prototipo sea superficial y no cubra realmente las necesidades para las que fue desarrollado, resultando en un prototipo inútil para el desarrollo posterior del aplicativo final.
- Los requerimientos iniciales pueden cambiar drásticamente.
- Si los desarrolladores no llevan la adecuada documentación, y esto resulta en un nuevo análisis, el desarrollo del prototipo no será de ayuda.
- La interfaz puede ser agradable, pero no funcional, esto puede conducir a la creación de falsas expectativas, donde el cliente piense que el sistema final ya está en marcha o puede estarlo muy pronto.

Situaciones donde es más apropiada

- Los objetivos del proyecto no están claros. Este modelo es útil cuando el cliente no identifica los requisitos detallados de entrada, procesamiento y salida.

- El proyecto es grande, tiene muchos usuarios y muchas funciones.
- Existe presión por presentar avances de implementación.
- El usuario no está bien informado o no puede comunicar claramente sus necesidades.
- La composición del equipo es estable.
- El administrador del proyecto tiene experiencia.
- No hay necesidad de reducir el consumo de recursos.

Situaciones donde es menos apropiada

- El equipo del proyecto no es estable.
- Los objetivos del proyecto son claros y el modelo de negocio es bien entendido.
- Los riesgos del proyecto relacionados con la definición de requerimientos es bajo.

3.1.3. Modelo en espiral

El Modelo Espiral se centra en la identificación temprana y la reducción de los riesgos del proyecto. En este modelo, los desarrolladores se inician a pequeña escala a continuación, exploran los riesgos involucrados en el proyecto, hacen un plan para manejar los riesgos, y finalmente deciden si tomar el siguiente paso del proyecto para hacer la siguiente iteración de la espiral. El éxito de cualquier ciclo de vida Modelo espiral depende de una gestión fiable, atenta y experta del proyecto. Toma las ventajas del modelo de desarrollo en cascada y el de prototipos añadiéndole el concepto de análisis de riesgos (Cicilia, 2009).

Provee la oportunidad de evaluar los riesgos y ponderar ciertas consideraciones antes de pasar al siguiente nivel del ciclo. Cada ciclo envuelve cuatro etapas: (1) determinar los objetivos, (2) evaluar las alternativas e identificar y resolver riesgos, (3) desarrollar y verificar las entregas de la iteración, (4) planear la siguiente iteración.

Es posible tener en cuenta mejoras y nuevos requerimientos sin romper con la metodología, ya que este ciclo de vida no es rígido ni estático. Empieza cada ciclo con la identificación de interesados y sus condiciones, y termina revisando si se cumplió el compromiso.

Ventajas

- Mejora la cobertura de riesgos.
- Provee una mayor adaptabilidad al cambio durante el desarrollo del proyecto.
- Basado en el tipo de riesgo que enfrenta el proyecto, permite incorporar otros enfoques metodológicos como el cascada, el prototipo, y el incremental.

Desventajas

- En cada iteración, se debe determinar como se van a usar las metodologías o como se van a combinar para poder tener un resultado exitoso, lo que puede llegar a ser un reto.
- Hay una alta personalización de cada proyecto, esto hace que la metodología usada tenga una reusabilidad limitada para otro proyecto.
- Se requiere un administrador del proyecto con experiencia, debido a la planeación detallada de cada proyecto.

- Los ciclos continúan sin una condición de término definida, esto trae in riesgo inherente de no terminar con el cronograma establecido o con el presupuesto.

Situaciones donde es más apropiado

- Proyectos donde evitar riesgos sea prioritario.
- Proyectos donde minimizar el consume de recursos no es prioritario.
- El administrador del proyecto es altamente calificado y experimentado.
- El proyecto puede beneficiarse de la combinación de metodologías.
- Se requiere de un alto grado de precisión en el desarrollo de funcionalidades.
- El modelo es apropiado para proyectos de tamaño grande y críticos.

Situaciones donde es menos apropiado

- Proyectos donde evitar riesgos no es prioritario o no se requiere extrema precisión.
- El éxito de todo el proyecto es dependiente de la fase de análisis de riesgos por lo tanto, el fracaso en esta fase puede dañar todo el proyecto.
- Proyectos donde minimizar el consumo de recursos es absolutamente necesario.

3.1.4. Metodología de desarrollo de software ágil

La metodología de desarrollo de software ágil es un enfoque de desarrollo que se utiliza para el diseño de una disciplinada gestión de proyectos, que a su vez permite una cierta alteración frecuente en el proyecto de desarrollo.

Esta metodología es un marco conceptual para la realización de diversos proyectos de ingeniería de software. Se utiliza para minimizar el riesgo mediante el desarrollo de software en periodos cortos de tiempo que se llaman iteraciones, cada iteración generalmente duran una semana a un mes. Se dice que las metodologías ágiles son orientadas al resultado, es decir, a software funcional, y no a actividades o tareas en cierto tiempo, para esto se necesita una administración de proyecto flexible, para lo cual entendemos que nuestro plan de trabajo original puede sufrir cambios positivos o negativos buscando siempre el resultado funcional. En el caso de metodologías orientadas al plan, son conocidas las metodologías tradicionales como RUP, donde existen tareas por desarrollar durante todas las etapas del proyecto, pero muchas de ellas no entregan funcionalidad del software, solo los requerimientos de control y documentación definidos por la metodología, estas regularmente no son tan flexibles por estructura, para lo cual se tienen que hacer renegociaciones intermedias si se detecta o requiere funcionalidad nueva no solicitada en fases anteriores.

Ventajas

- Las metodologías ágiles tiene un enfoque adaptativo que es capaz de responder a las necesidades cambiantes de los clientes.
- Fomenta una comunicación directa y constante mediante retroalimentaciones del cliente, gracias a ellos no se deja espacio para las conjeturas en el sistema.
- Las metodologías ágiles permiten a los pequeños grupos de desarrollo concentrarse en la tarea de construir software fomentando prácticas de fácil adopción y en un entorno ordenado que permiten que los proyectos finalicen exitosamente.
- Las metodologías ágiles se caracterizan por su sencillez, tanto en su aprendizaje como en su aplicación.

- Gracias a la metodología ágil, se permite ajustar los ciclos de trabajo para un mayor control en los plazos de entrega de los productos. Permite, además, enfocar el talento de los profesionales en determinadas zonas de mayor o menor riesgo con la meta de que el proyecto pueda ser defendido en su totalidad.

Desventajas

- Esta metodología se centra en el software de trabajo en lugar de la documentación, por lo que puede dar lugar a la falta de documentación.
- El proyecto de desarrollo de software puede perder su objetivo primordial, si el cliente no tiene muy claro el resultado final de su proyecto.

Situaciones donde es más apropiado

- Proyectos con requerimientos cambiantes.
- El cliente es parte del equipo.
- Existe incertidumbre del cliente.

Situaciones donde es menos apropiado

- Cuando es necesario documentar formalmente los requisitos, diseño, código y probar casos durante cada fase del ciclo de vida del software.
- Cuando los entregables deben pasar a través de una cadena de aprobaciones.
- Los usuarios de dueños del proceso no son accesibles para el desarrollo del proyecto

- Un proyecto ágil generalmente requiere múltiples habilidades, a veces en las mismas personas. Se necesitará habilidades en el análisis de negocio, diseño de arquitectura, gestión de proyectos ágiles, y comunicación interpersonal. Muchas veces no se puede contar con personal con la suficiente experiencia para el desarrollo de este tipo de proyectos.

Tabla 2. Comparación de metodologías

Metodologías ágiles	Metodologías tradicionales
Se basan en heurísticas provenientes de prácticas de producción de código	Se basan en normas provenientes de estándares seguidos por el entorno de desarrollo
Preparados para cambios durante el proyecto	Cierta resistencia a los cambios
Impuestas internamente por el equipo	Impuestas externamente
Proceso menos controlado, con pocos principios	Proceso muy controlado, numerosas normas
Contrato flexible e incluso inexistente	Contrato prefijado
El cliente es parte del desarrollo	Cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (<10)	Grupos grandes
Pocos artefactos	Más artefactos
Menor énfasis en la arquitectura del software	La arquitectura del software es esencial

Fuente: Canós, J et al, 2005. Metodologías Ágiles.

3.1.5. Modelo de Desarrollo Rápido de Aplicaciones

El Desarrollo Rápido de Aplicaciones (DRA) es un modelo de proceso del desarrollo del software lineal secuencial que enfatiza un ciclo de desarrollo extremadamente corto. Las metodologías ágiles son iterativas y tienen entregas incrementales. DRA es una adaptación a “alta velocidad” en el que se logra el desarrollo rápido utilizando un enfoque de construcción basado en componentes. Se deben comprender bien los requisitos y se debe limitar el ámbito del proyecto, el proceso DRA permite al equipo de desarrollo crear un “sistema completamente funcional” dentro de periodos cortos de tiempo.

Ventajas

- Visibilidad temprana.
- Ciclos de desarrollo más pequeños.
- Son rápidas.
- Responden de manera flexible a los cambios, se adapta a las necesidades del cliente.
- Promueven la comunicación entre los miembros del equipo, los interesados y los clientes.
- Pone mayor énfasis en el trabajo en equipo.
- Se enfoca más en cumplir con las entregas programadas que en hacer documentación del proyecto. El software es auto documentado durante el proceso de desarrollo.

Desventajas

- Progreso más difícil de medir.
- Menos eficiente.
- Riesgo de empezar la programación sin el adecuado análisis.

3.1.6. Modelo Incremental

Es el modelo utilizado para el desarrollo de un proyecto de software. Este modelo plantea la implementación del proyecto a realizar en iteraciones, con lo cual se pueden definir objetivos por cumplir en cada iteración y así poder ir completando todo el proyecto iteración por iteración, con lo cual se tienen varias ventajas, entre ellas se puede mencionar la de tener pequeños avances del proyectos que son entregables al cliente el cual puede probar mientras se está

desarrollando otra iteración del proyecto, con lo cual el proyecto va creciendo hasta completarlo en su totalidad. Permite construir el proyecto en etapas incrementales en donde cada etapa agrega funcionalidad. Estas etapas, consisten en requerimientos, diseño, codificación, pruebas y entrega. Permite entregar al cliente un producto más rápido en comparación del modelo en cascada. Se realiza una serie de mini fases, donde se completan todas las partes de esa fase, una vez terminadas todas las operaciones se procede al siguiente incremento. Lo que se busca es que en cada iteración los componentes logren evolucionar agregando más opciones de requisitos y logrando así un mejoramiento mucho más completo. Una manera muy primordial para dirigir al proceso iterativo incremental es la de priorizar los objetivos y requerimientos en función del valor que ofrece el cliente.

Para apoyar el desarrollo de proyectos por medio de este modelo se han creado entornos de trabajo, de los cuales los dos más famosos son el Rational Unified Process (RUP) y el Dynamic Systems Development Method. El desarrollo incremental e iterativo es también una parte esencial de un tipo de programación conocido como Extreme Programming y los demás frameworks de desarrollo rápido de software (Cicilia, 2009).

Ventajas

- Es una combinación de la metodología en cascada y la iterativa.
- Se puede explotar el conocimiento ganado en las primeras iteraciones hasta las últimas iteraciones.
- Se mantiene un control moderado sobre la vida del proyecto a través de la documentación, revisiones formales, y firma para realizar las aprobaciones.

- Los interesados en el proyecto tienen evidencia concreta del estado del proyecto a lo largo del ciclo de vida del proyecto.
- También provee un impacto ventajoso frente al cliente con la entrega temprana de partes operativas del Software (INTECO, 2009).

Desventajas

- Cuando se reduce las fases a pequeñas partes, por lo general, se omiten consideraciones globales del negocio y de los requerimientos técnicos.
- Los problemas que se presenten y dificultades pueden ser relegados para las últimas fases.
- Requiere de mucha planeación, tanto administrativa como técnica.

Situaciones donde es más apropiado

- Se tiene metas claras para conocer el estado del proyecto.

Situaciones donde es menos apropiado

- Pequeños proyectos de corta duración.
- Cuando los riesgos de integración y de arquitectura son bajos.

3.1.7. Proceso Unificado Rational (RUP)

La metodología RUP se originó por la Corporación Rational que fue adquirida por IBM en 2003 por lo que ahora es una metodología de IBM. El método RUP incluye aspectos del desarrollo tanto reiterativo y orientado a objetos, además existen numerosas herramientas que apoyan el método. Los casos de uso y las representaciones visuales son estándar para aplicaciones RUP. El Rational Unified Process es un proceso de ingeniería de Software. Proporciona un enfoque disciplinado para la asignación de

tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es asegurar la producción de software de alta calidad que satisfaga las necesidades de sus usuarios finales, dentro de un horario predecible y presupuesto.

Ventajas

- Reconoce que los requisitos del cliente no se pueden definir completamente al principio y se deben tomar acciones adaptativas.
- Permite evaluar tempranamente los riesgos presentes en lugar de descubrir problemas en la integración final del producto de software.
- Reduce el costo del riesgo a los costos de un solo incremento.
- Acelera el ritmo del esfuerzo de desarrollo debido a que se trabaja para obtener resultados claros a corto plazo.
- Distribuye la carga de trabajo a lo largo del tiempo del proyecto ya que todas las disciplinas colaboran en una iteración.
- Facilita la reutilización de componentes teniendo en cuenta que se realizan en primeras iteraciones.

Desventajas

- La metodología puede ser vista como muy pesada debido a su contenido amplio y complejo.
- Si se desconoce que es un marco de trabajo configurable, puede parecer tedioso ya que da la impresión de que se debe hacer uso de todos los elementos (actividades, artefactos y roles).
- También se puede caer en el error de pensar que RUP no admite adaptabilidad ni extensiones dependiendo de las particularidades del

proyecto. El detalle está en conocer suficiente la metodología para estar en la capacidad de configurar el marco de trabajo.

Situaciones donde es más apropiada

- Esta metodología por su amplitud es más apropiada para proyectos grandes, de largo plazo y sobre todo cuando se trabaja con equipos de desarrollo con numerosas personas y dispersos en cuanto a ubicación geográfica.

3.1.8. SCRUM

Es un modelo de referencia que define un conjunto de prácticas y roles. Los roles principales en Scrum son el ScrumMaster, que mantiene los procesos y trabaja de forma similar al director de proyecto, el ProductOwner, que representa a los stakeholders (clientes externos o internos), y el Team que incluye a los desarrolladores. El desarrollo de software se realiza mediante iteraciones, denominadas sprints, con una duración de 30 días. El resultado de cada sprint es un incremento ejecutable que se muestra al cliente.

Ventajas

- Permite la entrega de un producto funcional al finalizar cada Sprint (iteración).
- Posibilidad de ajustar la funcionalidad en base a la necesidad de negocio del cliente.
- Se puede visualizar el proyecto día a día.
- Alcance acotado y viable.
- Potencia la formación de equipos de trabajos autosuficientes y multidisciplinarios.

Desventajas

- Scrum no genera toda la evidencia o documentación de otras metodologías.
- No es apto para todos los proyectos.

Situaciones donde es más apropiado

- Indicada para pequeños equipos de desarrollo.
- Entornos complejos.
- Proyectos donde se necesita obtener resultados pronto.
- Proyectos donde los requisitos son cambiantes o poco definidos.
- Proyectos donde la innovación, la competitividad y la productividad son fundamentales.
- Para resolver situaciones en que no se está entregando al cliente lo que necesita, cuando las entregas se alargan demasiado, los costes se disparan o la calidad no es aceptable.
- Cuando se necesita capacidad de reacción ante la competencia.
- Cuando es necesario identificar y solucionar ineficiencias sistemáticamente.
- Cuando se quiere trabajar utilizando un proceso especializado en el desarrollo de producto.

Tabla 3. Diferencia entre metodologías RUP y SCRUM

	RUP	SCRUM
Enfoque	Iterativo	Iterativo
Ciclo	El ciclo formal se define en 4 fases.	Cada iteración (sprint) es un ciclo.
Planificación	Existe un plan de proyecto formal, asociado con múltiples iteraciones.	No hay un plan específico hasta el final del proyecto. La planificación de la siguiente iteración es determinada al final de la iteración actual. El dueño del proyecto determina cuando se ha terminado con el proyecto.
Artefactos	Visión, alcance, documentación, requerimientos formales, arquitectura del sistema, plan de desarrollo, plan de pruebas, etc.	El único artefacto formal es el software funcionando.
Tipo de proyecto	Recomendado para proyectos largos, de nivel empresarial, o proyectos con complejidad media a alta.	Recomendado para proyectos de entregas rápidas, que no dependen de una fecha límite.

Fuente: Elaboración Propia

3.2. Beneficios del uso de metodologías en el desarrollo de sistemas

- La metodología incorpora una filosofía que apoya el proceso de desarrollo del sistema.
- Una metodología con pasos bien definidos provee un avanzado control en el costo de las operaciones del equipo de desarrollo y en la forma de la solución final.
- Minimización de la incertidumbre: Cada fase está claramente definida, los resultados son predecibles lo cual significa que son medibles y controlables.
- El usuario de la metodología puede empezar a profundizar más en su pensamiento crítico, a través del conocimiento y la experiencia es acumulada.
- Mejora la relación entre costos y beneficios (INTECO, 2009).
- Ayuda a la mejor comprensión de los requerimientos y define mejor su implementación.
- Hay más confianza en los plazos establecidos.
- Ayuda a mejorar el uso de los recursos que se van a usar para el desarrollo del software.

3.3. Problemas del uso de metodologías en el desarrollo de sistemas

- Las metodologías no tiene parámetros estrictamente rígidos para todos los proyectos de desarrollos, por ejemplo no se puede dar un número fijo de iteraciones para las metodologías ágiles. Muchos de los parámetros serán definidos dependiendo de la situación del proyecto.
- Un entorno de rápido cambio puede dejar en desuso a una metodología, se sabe que la única constante es el cambio. El desarrollo de sistemas de información se debe adaptar rápidamente al continuo cambio de condiciones del ambiente.
- El costo incrementa muchas veces la aplicación de los procedimientos de una metodología, su uso puede llegar a ser bastante costoso para una organización. Eso quiere decir que muchas veces las organizaciones no podrán afrontar los gastos, e implementaran sus sistemas aplicando una metodología.
- El desarrollo de una metodología extiende el tiempo de desarrollo al necesitar que se apliquen los parámetros establecidos dentro de la metodología.

CAPÍTULO 4

DETERMINACIÓN DE FACTORES PARA EL DESARROLLO DE UN SISTEMA DE RECOMENDACIÓN

4.1. Determinación de factores para el desarrollo de un sistema de recomendación

En este capítulo se recoge factores que son considerados importantes para el desarrollo de un SR. El objetivo es entender ciertas prioridades y componentes a los que se debe poner mayor atención durante el desarrollo de SR. El criterio de selección de los factores para el desarrollo de un SR se basa en la frecuencia en que estos factores fueron hallados en el material bibliográfico relacionado a SR. Además, se escogió factores específicos para el desarrollo de estos sistemas.

A continuación listamos los factores que se consideró los más importantes para poder tener una implementación de SR exitosa. Para llevar a cabo esta explicación se muestra cinco actividades que involucran la intervención de los factores.

La actividad, seleccionar la metodología de desarrollo adecuada para un SR, tiene como factor a el tipo de metodología de desarrollo. Su importancia radica

en el marco de trabajo que aporta para el desarrollo de SR. La selección del tipo de metodología es considerada un factor importante en el desarrollo de cualquier software.

La actividad definir el modelo de negocio de la recomendación, nos ayudara a definir claramente el aporte estratégico que dará la recomendación al negocio, recordemos que un SR por lo general es un componente de un sistema mayor. Las recomendaciones pueden estar orientadas a mostrar las mejores ofertas o pueden ser parte de la campaña publicitaria del negocio que las implemente, por esta razón es importante alinear esta estrategia de recomendación a la estrategia de negocio. Los factores para esta actividad son: definir el dominio, propósito, contexto, grado de personalización, métricas comerciales, y la integridad de la recomendación. La selección de estos factores está basada en la importancia de considerar el modelo de negocio de cual es parte la recomendación.

La actividad obtener los datos de perfil y preferencia es parte esencial en el proceso de un SR, debido a que no se puede hacer recomendaciones si no se conoce el perfil del usuario y sus preferencias, es por ello que se desarrolla los factores: métodos de obtención de preferencias del usuario, perfiles de usuarios, e información actual. Estos factores son mencionados repetidas veces en diferentes fuentes bibliográficas, esto se debe a que la información de entrada del SR es de vital importancia para producir una recomendación.

La cuarta actividad diseñar el motor de recomendación es el núcleo del desarrollo de un SR, para ello son importantes los factores: tipo de modelo de recomendación, proceso de recomendación y presentación de recomendación. Para desarrollar estos tres factores es importante definir con qué tipo de recomendación se va a trabajar, después se puede definir el proceso de recomendación y la presentación de la misma. El diseño del motor de recomendación es la actividad básica de cualquier SR.

La última actividad, evaluar la calidad de la recomendación nos dirá la efectividad de la recomendación producida por el SR. Los factores son: exactitud de rango, cobertura, diversidad y precisión. Durante la revisión bibliográfica, se encontraron diferentes formas de evaluar las recomendaciones, se seleccionó las 4 anteriores debido a que son las más comunes, y están implementadas en las librerías de desarrollo de SR.

Todos estos son factores clave para un buen desarrollo de un SR. Existen, también, otros factores, no tan evidentes que no fueron considerados porque estaban más ligados a modelos de recomendación específicos.

Tabla 4. Factores para recomendación de sistemas de recomendación

ACTIVIDAD	FACTORES
Seleccionar la metodología de desarrollo adecuada para un SR	Tipo de metodología de desarrollo
Definir el modelo de negocio de la recomendación	Dominio de la recomendación
	Propósito de la recomendación
	Contexto de la recomendación
	Grado de personalización de la recomendación
	Métricas comerciales
Obtener los datos de perfil y preferencia	Integridad de la recomendación
	Métodos de obtención de preferencias del usuario
	Perfiles de usuarios
Diseñar el motor de recomendación	Información actual
	Tipo de modelo de recomendación
	Proceso de recomendación
Evaluar la calidad de recomendación	Presentación de la recomendación
	Exactitud de rango
	Cobertura
	Diversidad
	Precisión

Fuente: Elaboración Propia

4.2. Seleccionar la metodología de desarrollo adecuada para un SR

Una práctica común en el desarrollo del software, es empezar el desarrollo de un sistema, sin hacer el adecuado análisis del mismo, partiendo solo de los requerimientos que son conocidos (INTECO, 2009). Esta acción tiene drásticas consecuencias en el desarrollo de proyectos de software, por mencionar algunas, falta de entendimiento de los requerimientos, falta de alineación entre el software y los procesos de la empresa, falta de integración del software con otros sistemas. Es necesario seguir un proceso ordenado, haciendo uso de una metodología. Sin importar el tamaño del proyecto, una metodología proporciona un marco de trabajo coherente a través del ciclo de vida del software y optimiza el rendimiento del equipo.

La selección de la metodología es un factor importante, ya que todo el esfuerzo de desarrollo se hará sobre un marco de trabajo. Según las características de sistema podremos implementar de una u otra forma el desarrollo del mismo y conseguir cumplir las expectativas funcionales y de negocio esperadas.

4.2.1. Factor tipo de metodología

Se debe tener en cuenta que en general se puede escoger entre metodologías lineales, incrementales, iterativas, adaptativas y ágiles. Considerando esto se puede seleccionar la metodología que más se adapte a la implementación del SR.

Se debe seleccionar el tipo de metodología, entender sus lineamientos, políticas, técnicas y procedimientos para el seguimiento del desarrollo del SR. Para la selección del tipo de metodología, podemos tomar como referencia los siguientes parámetros.

4.2.1.1. Tipo de Recomendación

La metodología seleccionada no tiene dependencia con la selección del tipo de recomendación escogido. El tipo de recomendación es más dependiente del tipo de tecnología que se va a usar para su desarrollo, que de la metodología que se va utilizar para su desarrollo.

4.2.1.2. Tiempo y presupuesto

El tiempo y presupuesto asignados para el desarrollo del SR sí pueden influenciar en la selección de la metodología. Para una metodología tradicional es necesario tener el tiempo y presupuesto adecuado, se debe considerar un costo mayor en horas hombre para hacer la documentación, analizar y definir las etapas de dicha metodología. Además son recomendables en proyectos donde los equipos de trabajo son grandes y el personal cuenta con perfiles multifuncionales. En proyectos donde el presupuesto y tiempo son pequeños en relación al alcance funcional del proyecto, se recomienda el uso de metodologías ágiles (Sillitti et al., 2011).

4.2.1.3. Orientación al resultado o al plan

Las metodologías ágiles son orientadas al resultado, a software funcional, y no a actividades, por eso tienen una administración de proyecto flexible, por lo tanto, entendemos que el plan de trabajo original puede sufrir cambios. En el caso de metodologías orientadas al plan, como RUP, existen tareas por desarrollar durante todas las etapas del proyecto, sin embargo muchas de estas tareas no entregan funcionalidad del software, no son tan flexibles, se tienen que hacer renegociaciones si se requiere alguna nueva funcionalidad (Sillitti et al., 2011).

4.2.1.4. Colaboración del cliente

Establecer si se necesita la colaboración activa del cliente para poder desarrollar el sistema a medida. En un equipo de alta experiencia, el 35% o más del equipo, pueden ejecutar muy bien las metodologías ágiles. Por debajo del 15% la opción adecuada es la metodología planificada tradicional (Sillitti et al., 2011).

4.2.1.5. Cambios de funcionalidad

Establecer la cantidad de cambios de requerimientos que pueden aparecer durante el desarrollo del SR. En el caso de volatilidad de los requerimientos (40% de cambio por mes) las metodologías adecuadas son ágiles. Por el contrario, si los requisitos son estables (1% de cambios al mes) se adaptan mejor las metodologías tradicionales (INTECO, 2009).

4.2.1.6. Tamaño del equipo

En general equipos grandes, 300 a más personas, sugieren una metodología planificada tradicional, mientras que proyectos con equipos inferiores a 10 personas funcionan muy bien con metodologías ágiles (Sillitti et al., 2011).

4.3. Definir el modelo de negocio de la recomendación

Las organizaciones sufren de falta de alineación entre las tecnologías de información y el negocio. Este desajuste se debe, entre otras cosas, a una estrategia ineficaz de las tecnologías de información. Los negocios que necesitan de SR no están ajenos a esta realidad, tienen las mismas necesidades, es por ello que para el desarrollo efectivo de estos sistemas también se debe considerar estar alineado con la estrategia de negocio. Un SR es un componente de un sistema mayor, por lo tanto la alineación estratégica entre el SR y el sistema del cual forma parte es vital.

Es necesario entender el negocio, para realizar la implementación de un nuevo sistema con precisión. El SR debe funcionar de acuerdo al conjunto de reglas de negocio, esto ayuda a asegurar que las recomendaciones son a la vez útiles para el usuario y rentables para el negocio (Herlocker et al., 2000). Las reglas de negocio en torno a estos sistemas están diseñadas para prevenir sugerencias necias y también para ayudar a los vendedores en línea a maximizar las ventas. Una regla del sistema podría ser no vender productos de alta demanda debajo del precio de costo, o proveer recomendaciones de los productos que tienen exceso de stock. Se debe definir como actuarán las reglas de negocio en relación con la honestidad de la recomendación, por ejemplo no recomendar algo que no esté en el stock actualmente. Un sistema que simplemente empuja productos de baja salida no va a ganar la confianza de los clientes. Un ejemplo es Amazon, en setiembre del 2000, varios clientes presentaron quejas al descubrir que se les impuso precios más altos por ser clientes regulares, Amazon negó la acusación diciendo que solo fue un examen de precios al azar (Konstan and Riedl, 2012).

El objetivo principal de definir el modelo de negocio que se va seguir es definir el software (la solución) que satisfaga las necesidades de negocio (el problema).

Este análisis del modelo de recomendación plantea una serie de preguntas:

- ¿Qué tipos de usuarios buscan la recomendación?
- ¿Cuál es el nivel de especialización que se debe dar a la recomendación?
- ¿Cómo se obtienen las preferencias?
- ¿Qué incentivos se puede dar a los usuarios para fomentar el uso de las recomendaciones y las retroalimentaciones?
- ¿Cuál sería una buena estrategia de recomendación?
- ¿Qué preferencias van a ser usadas para hacer la recomendación?

4.3.1. Factor dominio de la recomendación

- Definir el tipo de producto que se va recomendar, por ejemplo, noticias, productos, películas, promociones y paquetes, música, recomendar gente a más gente (Herlocker et al., 2000, Ekstrand et al., 2014).
- Definir las reglas de recomendación específicas para ese dominio, por ejemplo, si son películas, sugerir las películas de los estrenos más recientes.
- Definir la novedad que se pretende que aporte la recomendación, dentro de este dominio.

4.3.2. Factor propósito de la recomendación

Es necesario determinar claramente el propósito que busca la recomendación que se ofrece. Los sistemas de recomendación no son considerados como parte del núcleo del negocio, se les considera un sistema adicional. En consecuencia, el SR debe estar en coherencia con el sistema mayor del que es parte. Sí el sistema mayor es un sistema de comercio electrónico, entonces el propósito del SR es la venta de productos (Herlocker et al., 2000, Ekstrand et al., 2014). Para este caso, el SR debe

ser desarrollado como una ayuda al usuario para la selección de productos. Generalmente el propósito de la recomendación es:

- Vender productos.
- Reducir tiempo de búsqueda.
- Soporte a las decisiones del usuario.

4.3.3. Factor contexto de la recomendación

El contexto de la recomendación es la situación, circunstancia o momento en la que se va a ofrecer la recomendación. El contexto es importante para ofrecer la recomendación de forma adecuada, se puede definir primero la actividad que está llevando a cabo el usuario como por ejemplo haciendo compras, conociendo gente o escuchando música (Herlocker et al., 2000, Ekstrand et al., 2014). También, se puede considerar el entorno que rodea al usuario, si el SR recomienda actividades culturales, podría considerar el clima, al momento de hacer sus sugerencias.

4.3.4. Factor personalización de la recomendación

4.3.4.1. Recomendación sin personalización

La recomendación sin personalización es el tipo más antiguo de recomendación, es cuando los SR proporcionan las mismas recomendaciones a todos los usuarios (Adomavicius et al., 2008, Herlocker et al., 2000). Dichas recomendaciones estarán basadas en selecciones manuales, resúmenes estadísticos u otras técnicas similares. No hay ningún tipo de personalización en el servicio ofrecido. Es efectivo cuando no se tiene información del usuario y tiene bajo costo de desarrollo.

4.3.4.2. Recomendación personalizada

Los sistemas de recomendación que ofrecen el mayor grado de personalización son los que usan personalización, ofreciendo recomendaciones distintas para distintos usuarios, incluso cuando estén buscando el mismo ítem (Adomavicius et al., 2008, Herlocker et al., 2000). Estos sistemas están basados en el perfil de los usuarios, por lo que hacen uso de métodos de filtrado colaborativo, filtrado basado en contenidos o correlaciones entre ítems. Esta recomendación tiene un costo de implementación y mantenimiento mayor, pero es más efectiva.

4.3.4.3. Recomendación Semi-personalizada

Los SR que tienen en cuenta la información actual del usuario objeto de las recomendaciones, proporcionan personalización efímera, puesto que las recomendaciones son respuesta al comportamiento y acciones del usuario en su sesión actual de navegación (Herlocker et al., 2000). Se basa en lo que el usuario está haciendo, escuchando o buscando en el momento. No necesitan mucha información de las preferencias del usuario debido a que solo hace el análisis actual de su búsqueda para ofrecer los productos.

4.3.4.4. Recomendación Demográfica

La recomendación es producida para un grupo objetivo específico, por ejemplo el grupo puede estar definido por género o por edad (Ricci et al., 2011).

4.3.5. Factor métricas comerciales

La razón por la que medimos el rendimiento de una recomendación es para poder evaluar si la recomendación ofrecida por el sistema tiene algún impacto en el negocio (Applegate et al., 2009). Para la selección de métricas adecuada se debe tener claros los objetivos que el SR persigue.

- Porcentaje de incremento de ventas
- Porcentaje de incremento de ventas cruzadas
- Ofertas aceptadas por el usuario gracias a una recomendación
- Porcentaje de satisfacción del cliente

4.3.6. Factor integridad de la recomendación

4.3.6.1. Proteger la precisión de la recomendación y la neutralidad de la misma (Herlocker et al., 2000, Ekstrand et al., 2014).

- Evitar la creación de cuentas falsas, los usuarios maliciosos pueden querer promover la venta de un producto o evitar su venta.
- Evitar la manipulación de puntajes.

4.3.6.2. Establecer los niveles de privacidad que va a tener el SR

Las preferencias obtenidas ofrecen gran cantidad de información sobre los usuarios y representan una valiosa fuente de información. Se necesita el consentimiento del usuario con respecto al uso de sus datos de carácter personal (Musial, 2009).

- Definir cuanta información personal necesita el sistema para poder redactar claramente los términos de condición de uso, junto con una política de privacidad comprensible.

- Proteger la privacidad del usuario, de ataques o de otros usuarios del sistema.
- Proteger la información que el servicio del sistema recupera de los usuarios.
- Al momento de hacer la explotación de la información de un cliente, se debe considerar las leyes de privacidad vigentes en el país de uso del sistema.
- Evitar hacer mal uso de la información registrada por el sistema.



4.4. Obtener los datos de perfil y preferencia

El componente clave de un sistema de recomendación son los datos. El proceso de obtención de datos servirá para construir el perfil del usuario y poder establecer sus preferencias. El sistema de recomendación debe poder predecir preferencias, pero este proceso solo será eficiente si la información que se tiene del usuario es correcta y apta para ser procesada. Esta información de las preferencias del usuario puede ser obtenida por una variedad de medios tales como calificaciones de los clientes a los productos, retroalimentación, comentarios, perfil del usuario, y preferencias explícitas e implícitas. Estos datos serán la base para realizar nuevas recomendaciones a los usuarios.

El siguiente conjunto de preguntas ayudará en esta tarea:

- ¿Qué tipo de datos va a manipular el sistema de recomendación?
- ¿Cuál es la calidad y cantidad de los metadatos?
- ¿Cuál es la cantidad de productos?
- ¿Cómo es la diversidad de los productos?
- ¿Son las valoraciones de los usuarios implícita o explícita?
- ¿Son votaciones binarias o multinivel?

Además se debe considerar:

- Los registros de actividad deben tener información útil.
- La data debe poder ser extraída de manera eficiente y precisa.
- Los datos extraídos deben tener la calidad necesaria para ser procesados.
- La información debe estar al día, de lo contrario la sugerencia ofrecida no es relevante.

4.4.1. Métodos de obtención de preferencias del usuario.

Con el fin de generar un perfil de usuario es necesario recopilar las preferencias del usuario (Marco de Gemmis et al., 2009). Se debe escoger información que es relevante a la vista del usuario y a su vez al servicio ofrecido, no tiene sentido recuperar datos que no son del interés del usuario o que lo son pero no están relacionados al servicio que se ofrece. El resultado de una mala selección de información proporcionará una recomendación inútil (Musia, 2006). Uno de los principales problemas de la recolección de datos es que la información acerca de las preferencias e intereses del usuario es limitada. Se debe implementar la recolección de información por medio de dos formas, de manera implícita y explícita.

4.4.1.1. Recolección de preferencias explícitas

La recolección se obtiene explícitamente cuando el usuario ingresa directamente datos en su perfil. El enfoque explícito está basado en preguntas hechas de forma directa al usuario, se debe implementar preguntas directas acerca de la opinión de este usuario, mediante puntuaciones, evaluaciones, cuestionarios, encuestas y votos. Además, identificar el momento oportuno para pedir las calificaciones, puede ser mientras el producto es visualizado en el momento, o puede llevarse a cabo después de que el producto fue visualizado.

Considere los siguientes problemas con las preferencias explícitas

- La opinión del mismo usuario en diferentes momentos suele ser diferente, debido a esto las puntuaciones no son siempre exactas.
- Las preferencias cambian con el tiempo o son fácilmente influenciadas por el entorno.

- La información explícita donde el usuario provee de forma directa sus preferencias es escasa o nula. (Tsihrintzis et al., 2013).

4.4.1.2. Recolección de preferencias implícitas

Se obtiene de un proceso de retroalimentación recolectado de las interacciones que el usuario tiene con el sistema, el usuario no es consciente en ningún momento del proceso que se está llevando a cabo. El sistema debe seguir sus acciones durante la interacción con el servicio, para poder identificar sus posibles gustos. La recolección de preferencias implícitas se hace mediante toda la recolección de información implícita disponible sobre el consumidor, incluyendo información demográfica, historial de navegación, compras, y otra información transaccional. Esta información se recoge a partir de múltiples fuentes, se procesa y se transforma en conocimiento y se almacena en los perfiles de consumo. Un ejemplo de este tipo de información es la interacción entre el SR y el consumidor incluyendo, compras, actividades de navegación, información visualizada durante la sesión, imágenes que son agrandadas, productos que han sido vistos varias veces, tiempo que el usuario utiliza para ver los productos, entre otros. Este conocimiento se utiliza posteriormente para determinar cómo será la siguiente recomendación.

Considere los siguientes problemas con las preferencias implícitas:

- Se puede atentar en contra de la privacidad del usuario.

4.4.2. Perfiles de usuario

Una vez recogidos los datos, uno de los temas clave en las aplicaciones de personalización es la integración de estos datos y la construcción de perfiles de consumidores. Para definir el perfil del consumidor se debe

entender a los consumidores mediante la recopilación de información acerca de ellos y convertirla en conocimiento de perfiles de consumo (Adomavicius et al., 2008). Se debe reunir una imagen de los diferentes grupos de usuarios a través de la información registrada en el sistema como edad, el género, área de trabajo, nacionalidad, idiomas que se hablan, e investigación de preferencias. Uno de los supuestos fundamentales en la personalización es que las preferencias de los consumidores tiene cierta estabilidad a través del tiempo y que las actividades de consumo pasados se pueden utilizar para predecir sus posibles acciones futuras (Adomavicius et al., 2008). La comprensión de esto permite que el equipo de desarrollo pueda empezar a construir una relación con los usuarios y obtener una apreciación de sus necesidades. Esta comprensión de los grupos de usuarios puede permitir la construcción de recomendaciones simples. Al momento de construir el perfil de los clientes no se debe abrumar al usuario con muchas preguntas, es importante ser breve y directo, e inferir la mayor cantidad de datos y preferencias posibles.

El SR requiere analizar los perfiles sobre los que basará sus recomendaciones. El análisis de perfiles se puede entender mejor observando la figura 8, este es un ejemplo de SRFC. Los usuarios 1 y 5 parecen tener gustos similares. Ambos gustan del libro 101, del 102 un poco menos, y como 103 menos todavía. Lo mismo ocurre con los usuarios 1 y 4, ya que parece que les gusta 101 y 103 de forma idéntica. Por otro lado, los usuarios 1 y 2 tienen gustos que parecen ir en contra de uno al otro, 1 le gusta 101 mientras que al usuario 2 no, y al usuario 1 le gusta 103 pero al usuario 2 es todo lo contrario (OWEN et al., 2012). Hacer recomendaciones se trata de descubrir cosas nuevas. La intuición sugiere que debido a que los usuarios de 4 y 5 son similares a 1, sería bueno recomendar algo que el usuario 4 o 5 digan que es de su agrado. Eso deja a los libros 104, 105 y 106 como posibles recomendaciones. Para la

creación de recomendaciones es necesario escoger alguno de los algoritmos de recomendación e implementarlo para que pueda llevar a cabo las recomendaciones de todos los usuarios del sistema.

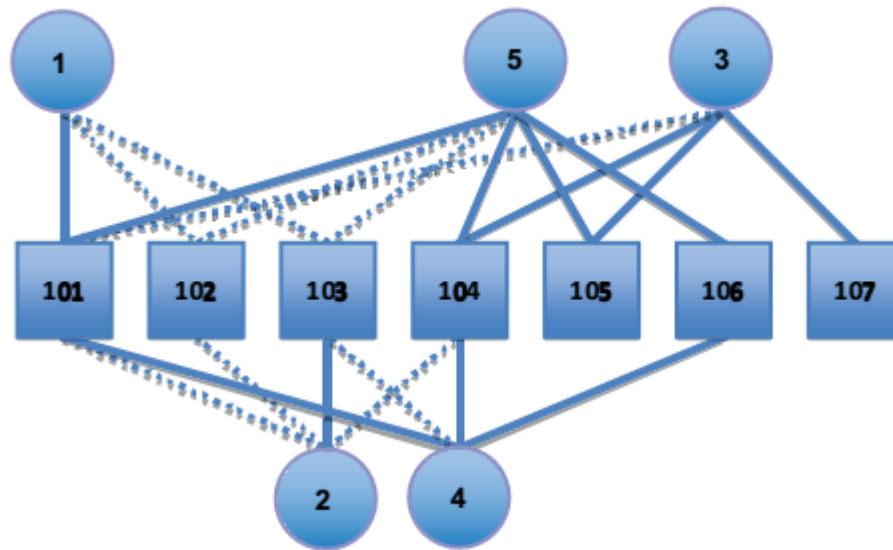


Figura 8. Análisis de datos para hacer la recomendación

Fuente: (Owen et al., 2011)

4.4.3. Información actual

Es crucial mantener la base de datos de productos actualizados para que el sistema pueda dar recomendaciones de productos actuales. La recomendaciones no podrán ser novedosas ni diversas, si no hay un adecuado mantenimiento de la base de datos de productos. Además, las nuevas preferencias del usuario deben ser añadidas al perfil de usuario cada vez que sea necesario para mantener el perfil del usuario, esto se debe a que las preferencias del usuario no son estáticas, cambian con el tiempo (Herlocker et al., 2000).

4.5. Diseñar el motor de recomendación

Después de la recolección de datos, los sistemas de recomendación utilizan algoritmos para encontrar similitudes y afinidades entre los productos y los usuarios.

El principio de un motor de recomendación es que los gustos de la gente varían, aunque estos gustos siguen patrones específicos y que son predecibles. Las personas tienden a preferir productos que son similares a otros productos que les gustaron en el pasado. Estos patrones se pueden utilizar para predecir tales gustos y disgustos. La recomendación tiene que ver con la predicción de estos patrones de gusto, para descubrir cosas nuevas y deseables, cosas que ni el mismo usuario sabe que serán de su agrado.

El motor de recomendación debe calcular la recomendación personalizada basada en el conocimiento acerca de cada usuario, según los datos almacenados en los perfiles de los consumidores y en la información sobre los productos. Su función es encontrar las recomendaciones personalizadas, identificar cuáles son los productos más relevantes para cada uno de los consumidores, en el contexto especificado, y entregarlos de la mejor manera posible, en el momento más adecuado (s) y por medio de canales apropiados (Adomavicius et al., 2008).

4.5.1. Tipo de modelo de recomendación

La pregunta inicial al desarrollar el motor de recomendación es que tipo de recomendación se debe usar, en este momento el mayor desafío es averiguar la mejor manera de seleccionar el algoritmo indicado. Escoger el algoritmo de recomendación puede llegar a ser una tarea complicada dada la amplia variedad de ellos conocidos actualmente. La comparación de diferentes tipos de recomendación ayudara al desarrollador a encontrar el tipo más adecuado a su necesidad y que derive en una recomendación de calidad para el usuario.

Se puede hacer un sistema de recomendación híbrido para que cumpla con las diferentes necesidades que se tiene. Muchos servicios de comercio electrónico combinan diferentes tipos de modelos de recomendación para poder satisfacer más necesidades y en diferentes contextos.

Tabla 5. Tabla comparativa de los tipos de recomendación

ENFOQUE	VENTAJAS	DESVENTAJAS
RECOMENDACIONES NO PERSONALIZADAS	<ul style="list-style-type: none"> • Las recomendaciones son independientes del cliente. • Es fácil de diseñar. 	<ul style="list-style-type: none"> • Las recomendaciones ofrecidas no son personalizadas por cliente.
RECOMENDACIÓN DEMOGRAFICA	<ul style="list-style-type: none"> • Proporcionan recomendaciones interesantes para cualquier persona perteneciente al grupo demográfico. 	<ul style="list-style-type: none"> • Falta de datos demográficos. • La recolección de información es privada, no se puede ofrecer una recomendación individual.
BASADOS EN CONTENIDO	<ul style="list-style-type: none"> • Independencia del usuario. • Las explicaciones sobre cómo funciona el SR son transparentes. • No sufren el problema del arranque frío de productos. 	<ul style="list-style-type: none"> • Es difícil hacer recomendación de productos multimedia. • No existen recomendaciones inesperadas. • No es un enfoque social. • La calidad depende de la cantidad de información. • No diferencia ítems, homogéneos (mismos atributos).
BASADO EN CONOCIMIENTO	<ul style="list-style-type: none"> • Agregar fácilmente nuevas reglas de negocio. • Sensible a los cambios de preferencias. • Adaptable a las necesidades de los usuarios. 	<ul style="list-style-type: none"> • Complejidad al momento de ser implementados.

<p>BASADOS EN UTILIDAD</p>	<ul style="list-style-type: none"> • No sufren el problema del arranque frío de productos. • Permiten al usuario encontrar las recomendaciones que mejor encajen en el marco de sus necesidades. • Adaptable a las necesidades de los usuarios. • Incluye características que no son propias de los productos. 	<ul style="list-style-type: none"> • La elaboración de funciones de utilidad implica que los usuarios realicen la medición de pesos, valoraciones y asignación de función de utilidad a las diferentes propiedades de los ítems.
<p>FILTRADO COLABORATIVO (CF)</p>	<ul style="list-style-type: none"> • Implementación sencilla. • Proporciona una justificación concisa e intuitiva • Puede llegar a tener un enfoque social. • Son los más comunes y cuentan con diversas librerías que los implementan. 	<ul style="list-style-type: none"> • Necesitan una gran cantidad de calificaciones. • Tiene el problema del arranque frío de usuarios y productos. • Problemas de privacidad. • Escalabilidad • No hay un buen manejo de sinónimos. • El problema de la oveja negra.
<p>HIBRIDOS</p>	<ul style="list-style-type: none"> • Aportan una mejora considerable con respecto a los CF y a los basados en contenido. • Mejora el rendimiento de las predicciones. • Resuelve el problema del arranque frío. 	<ul style="list-style-type: none"> • Incrementa la complejidad de implementación. • Requiere más información de productos y usuarios.

Fuente: Elaboración Propia

4.5.2. Proceso de recomendación

La forma de procesar la recomendación varía para cada enfoque.

- Se debe definir es el tipo de modelo de recomendación que se va a utilizar. Dependiendo del enfoque que se seleccione, se implementará el proceso de recomendación.
- Seleccionar los datos de entrada, que necesita el enfoque de recomendación seleccionado. Un SRFC necesita la calificación de varios usuarios y hallar un vecindario para un usuario y poder producir la recomendación, mientras un SRBC necesita la información de preferencias de usuarios y atributos de productos. Muchas veces el tipo de información que se tenga inicialmente definirá el modelo de recomendación que puede ser implementado.
- Dentro de un mismo tipo de modelo de recomendación pueden existir una variedad de algoritmos para implementar la recomendación, es importante probar diferentes algoritmos para ver el rendimiento del sistema. Para este proceso se puede hacer uso de librerías como Apache Mahout, que ya cuentan con el código desarrollado.

4.5.3. Presentación de la recomendación

Las recomendaciones deben ser entregadas y presentadas al consumidor de la mejor manera posible, es decir, en el momento(s) más apropiada, a través de los canales más adecuados, y en la forma más adecuada (Herlocker et al., 2000, Ekstrand et al., 2014). Se debe definir cómo será la presentación de la recomendación, explícita o será una interacción pasiva. Los métodos explícitos llegan a un consumidor que actualmente no está interactuando con el sistema, explícita es la sugerencia directa de ofertas para ver productos, por ejemplo, mediante el envío de un mensaje de correo electrónico. Los métodos pasivos muestran esta información sólo cuando el consumidor lo solicita abiertamente (Adomavicius et al., 2008).

4.6. Evaluar la calidad de recomendación

La evaluación de la recomendación proporciona información acerca de las deficiencias de los métodos de recomendación. Esta información sirve como una realimentación para posibles mejoras. Se deben desarrollar técnicas de evaluación de acuerdo a la selección del enfoque adoptado y los objetivos planteados. Teniendo en cuenta algunas deficiencias de los enfoques de los SR, los investigadores han estado trabajando en establecer ciertas métricas estándar (Casinelli, 2014). No es necesario aplicar todas las evaluaciones existentes, pero si es necesario aplicar alguna evaluación. La forma más fácil de evaluar un método de recomendación es medir la diferencia entre sus predicciones y las votaciones reales que el usuario provee. Medir el impacto de recomendación mediante la determinación de cuan satisfecho está el usuario con la recomendación (Adomavicius et al., 2008).

Cuando se trabaja con SR es común probar diferentes algoritmos, se debe utilizar la misma muestra y la misma escala para la comparación para que los resultados de la evaluación sean efectivos. El resto de este capítulo se hace una pausa para estudiar la manera de evaluar un sistema de recomendación.

Un motor de recomendación es una herramienta, un medio para responder a las siguientes preguntas:

- ¿Cuáles son las mejores recomendaciones para un usuario?
- ¿Qué es exactamente una buena recomendación?
- ¿Qué debe estar en la lista de recomendación?
- ¿Cómo sabemos que la lista de recomendación provista es una buena recomendación?

4.6.1. Exactitud del rango

Asegurar que el ranking proporcionado por el sistema es realmente efectivo. Dentro de esta métrica de debe considerar hacer buenas recomendaciones y que dentro de estas recomendaciones queden fuera productos sin valor para el usuario.

4.6.2. Cobertura

La cobertura es la proporción de artículos disponibles que el sistema de recomendación recomienda a los usuarios (Robillard et al., 2014). Evalúa el número de elementos para los que el sistema podría proporcionar recomendaciones (Good et al., 1999). Una de las principales reglas de negocio es permitir que los usuarios vean los productos que existen en toda la tienda. Como podemos deducir, un sistema con una baja cobertura será poco valioso, no podrá cubrir las necesidades de cualquier usuario, puesto que se limitaría enormemente el espectro de recomendaciones o alternativas que el sistema propondría a la hora de tomar una decisión. Cuando intentamos predecir la valoración de un producto mediante el SRFC y ningún vecino del usuario activo ha valorado dicho producto, se produce un fallo de cobertura. Intuitivamente, si el número de vecinos elegido en este algoritmo es mayor, la cobertura aumenta (Gallardo, 2012).

4.6.3. Diversidad

Es la medida de cuan diferentes son los productos que son recomendados uno del otro. Para implementar esta medida los productos ofrecidos dentro del catálogo deben estar clasificados, para que pueda ser más fácil identificar productos similares (Herlocker et al., 2000, Ekstrand et al., 2014).

4.6.4. Precisión

4.6.4.1. Precisión:

Fracción de objetos relevantes entre todos los objetos recomendados. Para encontrar el número de buenas recomendaciones se debe hacer entrevistas con el usuario, para conocer su opinión acerca de la cantidad de buenas recomendaciones que el SR le proporciona (Herlocker et al., 2000, Ekstrand et al., 2014).

$$Precision = \frac{\text{Nro. de buenas recomendaciones}}{\text{Total de recomendaciones}}$$

Fuente: (Jannach et al., 2010)

4.6.4.2. Recall:

Fracción de objetos relevantes entre todos los objetos buenos. Para encontrar el número de buenas recomendaciones se debe hacer entrevistas con el usuario, para conocer su opinión acerca de la cantidad de buenas recomendaciones que el SR le proporciona (Herlocker et al., 2000, Ekstrand et al., 2014).

$$Recall = \frac{\text{Nro. de buenas recomendaciones}}{\text{Total de buenas recomendaciones}}$$

Fuente: (Jannach et al., 2010)

4.6.4.3. Error absoluto medio

Es considerado como una medida estadística para la estimación de la exactitud con la que el sistema realizará las predicciones. Este tipo de medidas tratan de verificar con qué grado de exactitud el sistema predice las valoraciones con respecto a las verdaderas valoraciones del usuario (Torres, 2007).

Error absoluto medio, en este caso particular es la desviación entre la predicción y el puntaje real. Se entiende por la fórmula P-R, el resultado puede ser negativo o positivo. El valor absoluto se encarga de remover la dirección del error, para darnos un valor dentro de la escala. Finalmente se obtiene el promedio. La sumatoria del valor absoluto de la predicción menos el puntaje, se divide por la cantidad de puntajes que están siendo evaluados. Para llevar a cabo la evaluación de las medidas se puede hacer el promedio de ratings por usuario, o hacerlos por el total del usuario. El resultado de ambos puede llegar a variar considerablemente, si el error absoluto medio para el promedio total de ratings es mayor que el promedio por usuario, entonces se puede afirmar que el algoritmo trabaja mejor con usuarios que han hecho una gran cantidad de ratings. Si el resultado de comparación anterior es similar en ambos casos, podemos decir que el algoritmo funciona de manera similar para usuarios con pocos y muchos ratings.

Siendo

P: predicción

R: Rating

Fuente: (Jannach et al., 2010)

$$\frac{\sum \text{ratings} |P - R|}{\# \text{ ratings}}$$

4.6.4.4. Error cuadrático medio

Esta medida de precisión es también bastante popular en la evaluación de los sistemas de recomendación. El beneficio es que gracias a la potencia se elimina el signo automáticamente, el valor

absoluto queda obsoleto. Y ayuda a identificar errores grandes. La desventaja es que la escala del error cuadrático medio no es intuitiva.

Siendo
P: predicción
R: Rating
Fuente: (Jannach et al.,
2010)

$$\frac{\sum ratings(P - R)^2}{\# ratings}$$

4.6.4.5. Experimentos offline y online

El análisis offline se usa una técnica o algoritmo para predecir ciertos valores retenidos de un conjunto de datos y los resultados son analizados mediante una o varias métricas de error. Este análisis offline tiene la ventaja de ser rápido y económico pero también tiene dos desventajas importantes: el problema de la escasez de datos y el problema de obtener sólo como resultado la bondad de la predicción.

Hacer experimentos online. Mediante la realización de pruebas con usuarios reales. El análisis online permite obtener otros resultados como son la actuación de los usuarios participantes, su satisfacción o su participación. En su defecto es más lento y caro que el análisis offline.

4.6.4.6. Pruebas de laboratorios

Hacer experimentos controlados como por ejemplo, dividir el conjunto de datos en varias partes, y usar estos datos para hacer el entrenamiento del sistema. Después se puede probar la efectividad de la recomendación haciendo la comparación de los datos originales con la salida de datos del sistema de recomendación.

CAPÍTULO 5

DISEÑO DEL PROTOTIPO DE UN SISTEMA DE RECOMENDACIÓN

En este capítulo se hará el desarrollo e implementación de un prototipo de sistema de recomendación, llamado MovieAdvisor. El prototipo es un modelo a escala de las características básicas que debería tener un SR, para ello se tomará el caso de recomendar películas y demostrar la aplicabilidad del mayor grado de factores determinados. Para la realización de este prototipo se consideró parte de los factores desarrollados en esta tesis, los factores contemplados son: tipo de metodología, alcance de la recomendación, dominio de la recomendación, propósito de la recomendación, contexto de la recomendación, personalización de la recomendación, tipo de recomendación, métodos de obtención de preferencia, análisis de datos, proceso de recomendación, presentación de la recomendación, cobertura y precisión.

Durante la fase la de planteamiento se establecerá el modelo de negocio de la recomendación que se plantea implementar. En la fase de análisis se obtendrá los requisitos básicos del sistema, y se especificará las funcionalidades que se requieren en el prototipo. La fase de diseño y codificación se usa para establecer la arquitectura del sistema, interfaces, bases de datos y el motor de

recomendación. La fase final de pruebas, serán las pruebas del sistema y la evaluación de la recomendación que el prototipo produce.



Tabla 6. Tabla de fases de desarrollo del prototipo y factores incluidos por cada etapa

Fuente: Elaboración propia

FASES	DESARROLLO DEL PROTOTIPO	FACTORES DESARROLLADOS EN EL PROTOTIPO
Planteamiento	Definición de la metodología	Tipo de metodología de desarrollo
	Justificación del sistema	Tarea propia del desarrollo del prototipo
	Definición del modelo de negocio para las recomendaciones	Dominio de la recomendación, Propósito de la recomendación, Contexto de la recomendación, grado de personalización
	Definición del modelo híbrido de recomendación	Tipo de recomendación
	Definición de las tareas de desarrollo	Tarea propia del desarrollo del prototipo
	Definir fuentes de datos	Tarea propia del desarrollo del prototipo
Análisis	Analizar de los datos	Métodos de obtención de preferencias del usuario
	Establecer requerimientos no funcionales	Tarea propia del desarrollo del prototipo
	Establecer las funcionalidades del sistemas mediante historias de usuarios	Tarea propia del desarrollo del prototipo
Diseño y codificación	Diseño de la arquitectura	Tarea propia del desarrollo del prototipo
	Diseño de la estructura de los datos	Tarea propia del desarrollo del prototipo
	Diseñar el motor de recomendación	Proceso de recomendación
	Diseñar la presentación de la recomendación	Presentación de la recomendación
Pruebas	Efectuar las pruebas del sistema	Tarea propia del desarrollo del prototipo
	Evaluar la calidad de recomendación	Cobertura Precisión
	Resultados de la evaluación de la recomendación	Tarea propia del desarrollo del prototipo

5.1. Planeamiento

5.1.1. Tipo de Metodología

La metodología seleccionada para el desarrollo del prototipo es SCRUM, entre algunas de las ventajas ya mencionadas se espera que esta metodología aporte simplicidad de diseño y codificación, y produzca entregables rápidamente. El prototipo será el resultado de la primera iteración. La retroalimentación del mismo se hará simultáneamente al desarrollo.

Tabla 7. Selección del tipo de metodología para el desarrollo del prototipo

PARAMETROS PARA LA SELECCIÓN DE LA METODOLOGIA		SELECCION
Tipo de recomendación	Modelo de recomendación híbrido	No influye en la selección de la metodología.
Tiempo y presupuesto	Corto	Metodología ágil
Orientación al resultado o al plan	Se espera resultados inmediatos (prototipo funcional).	Metodología ágil
Colaboración del cliente	100%	Metodología ágil
Cambios de funcionalidad	Cambios pequeños	Metodología ágil o Metodología Tradicional
Tamaño del equipo	Menos de 10	Metodología ágil

Fuente: Elaboración Propia



Figura 9. Etapas de desarrollo del prototipo

Fuente: Elaboración propia

5.1.2. Justificación del sistema

Este prototipo será desarrollado para mostrar la aplicabilidad de los factores mencionados en el capítulo anterior.

5.1.3. Definición del modelo de negocio para las recomendaciones

Se proporcionará una descripción de alto nivel del sistema de recomendación que se desea desarrollar, especificando sus características, y su propósito. En esta etapa se debe definir el propósito de la recomendación.

Tabla 8. Definición del modelo de negocio del prototipo de recomendación

Modelo de negocio para las recomendaciones	
Alcance del SR	Desarrollar un prototipo funcional de un sistema de recomendación híbrido para la recomendación de películas, que aplique los factores comentados en esta tesis. Se hará uso de bases de datos que ya cuentan con información de películas, usuarios, y ratings. La BD será descargada de https://movielens.org/ y se complementó con la información de themoviedb.com , ambas bases de datos han sido desarrolladas con propósitos de investigación. Principalmente, se debe hacer uso de tres enfoques diferentes, filtrado colaborativo, basado en contenido y recomendaciones no personalizadas. Se debe evaluar la eficiencia de la recomendación en dos casos específicos, para un nuevo usuario y para un usuario con varias calificaciones. El prototipo realizado no será de uso comercial y su funcionalidad tendrá un fin académico.
Dominio de la recomendación	El dominio de la recomendación del prototipo será la recomendación de películas.
Propósito de la recomendación	Reducir el tiempo de búsqueda de películas y dar soporte a las decisiones del usuario.
Contexto de la recomendación	El usuario no tiene el propósito de comprar, ni visualizar las películas, el contexto es probar que los factores ayudaron al desarrollo del prototipo.
Grado de personalización	El grado de personalización del sistema será mixto entre una recomendación sin personalización, que mostraran. Un usuario nuevo no tendrá personalización debido a que aún no están bien definidas sus preferencias, conforme el sistema aprenda más del usuario podrá visualizar contenido más personalizado.

Fuente: Elaboración propia

5.1.4. Tipo de modelo recomendación

El modelo de recomendación usado será híbrido, este modelo permitirá cubrir diferentes necesidades. Para lo cual, se está integrando recomendación no personalizada, sistemas de recomendación basado en contenido y sistemas de recomendación de filtrado colaborativo. Se deben

establecerá claramente las políticas de recomendación para un nuevo usuario y un nuevo producto. El tratamiento de un nuevo usuario es complicado debido a que no hay preferencias, ni perfil que ayuden a proveer la recomendación personalizada.

5.1.4.1. Recomendación no personalizada

Esta recomendación será integrada para todos los usuarios, sean nuevos o antiguos. El objetivo de esta recomendación es que el usuario siempre tenga sugerencias de películas. Estas sugerencias sin personalización, son solo estrenos recientes y películas populares.

5.1.4.2. Sistema de Recomendación basado en contenido

Esta recomendación será integrada para todos los usuarios que tengan alguna preferencia de género seleccionada en su perfil. El enfoque de un sistema de recomendación basado en contenido se basa en comparar los atributos de una película y el perfil del usuario. En el caso del prototipo desarrollado, será solo una comparación del género seleccionado en el perfil del usuario y las películas que pertenezcan a ese género.

5.1.4.3. Sistema de Recomendación de filtrado colaborativo

Esta recomendación está habilitada solo para los usuarios que ya tienen un registro de calificaciones. Esto se debe a que los sistemas de recomendación de filtrado colaborativo necesitan comparar el perfil de preferencias del usuario con sus vecinos cercanos. Esta recomendación se basa en la premisa de que el usuario comparte gustos con sus usuarios vecinos, de esta manera se puede predecir sus posibles gustos.

5.1.4.4. Tratamiento de nuevas películas

Una nueva película que no ha sido calificado por ningún usuario puede quedar aislado y no participar de ninguna recomendación, deben tener un tratamiento especial. Las nuevas películas se visualizaran como parte de la recomendación no personalizada y de manera aleatoria, para que se pueda abarcar mejor el rango que se muestra al usuario. Además los nuevos estrenos serán parte de uno de los complementos, que son parte de la pantalla principal.



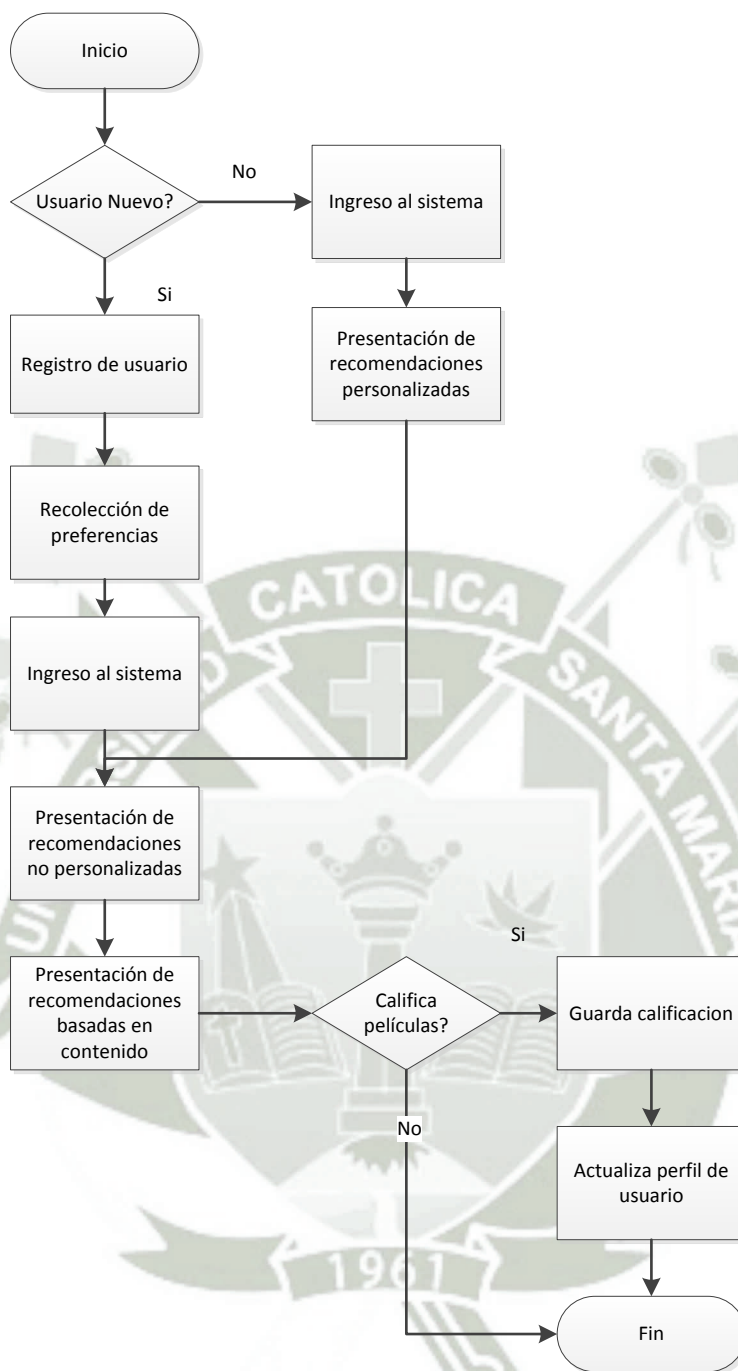


Figura 10. Diagrama de flujo del modelo de recomendación híbrido

Fuente: Elaboración propia

5.1.5. Definir las fuentes de datos

El prototipo de sistema desarrollado en esta tesis, está basado en la información de películas, usuarios y calificaciones sacados de dos bases de datos disponibles en Internet para hacer investigación académica

- GroupLens es un grupo de investigación especializado en SR que ha recogido y preparado conjuntos de datos de calificaciones, usuarios y películas. MovieLens es un sistema de recomendación de películas online basado en filtrado colaborativo. Desarrollado por el GroupLens Research de la Universidad de Minnesota (<http://www.grouplens.org>), recolecta puntuaciones sobre películas de sus usuarios y en base a esos datos agrupa los usuarios de similares gustos. Los conjuntos de datos han sido recogidos durante varios períodos de tiempo. Atendiendo a las puntuaciones de todos los usuarios dentro de un grupo se intenta predecir para cada usuario individual su opinión sobre películas que todavía no ha visto (Inès Jomaa et al., 2011).
- ThemovieDB es una api que permite descargar información de películas desde el sitio <https://www.themoviedb.org/>, de aquí se descarga la información adicional de las películas, como su votación promedio, imágenes, popularidad, etc.

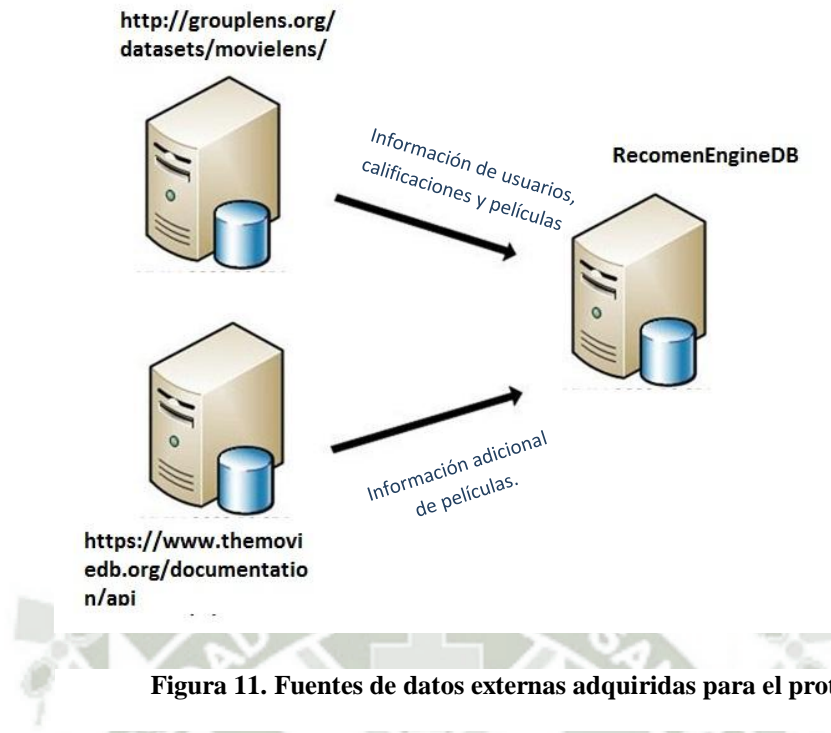


Figura 11. Fuentes de datos externas adquiridas para el prototipo

Fuente: Elaboración Propia

5.1.6. Definición de las Tareas de desarrollo

- Hacer historias de usuario con la información acerca de lo que el usuario espera del prototipo, añadir dependencia de tareas, número de la iteración nivel de dificultad, descripción y validez.
- Diseño de pantallas, e interfaces.
- Encontrar y probar los datos de películas, rankings y usuarios para poner en marcha el prototipo, con datos de prueba.
- Establecer la arquitectura del sistema.
- Definir la estructura de la Base de datos.
- Implementar el motor de recomendación y la aplicación web.
- Diseñar la evaluación de recomendación.

5.2. Análisis

En esta etapa se debe definir el propósito de la recomendación, objetivos, funcionalidades, capacidades y características del sistema. Llevar a cabo el análisis preliminar de requisitos con la finalidad de conseguir una descripción de los mismos que sea fácil de entender y ayude a estructurar el sistema de recomendación. Para ello, se definen los requisitos generales y se comienzan a definir las historias de usuario. Conforme avance el proceso de ingeniería de requisitos, los requisitos generales se irán detallando en requisitos más específicos e irán evolucionando hacia su forma detallada.

Para crear un sistema de recomendación se debe analizar los datos que se tiene para trabajar y considerar qué algoritmo de recomendación podría ser adecuado para 100.000 votos y 9.000 películas valorados por 700 usuarios.

El conjunto de datos se ha pre procesado de la siguiente manera: los usuarios que tienen menos de 20 calificaciones no están incluidos. Además, se excluyen a los usuarios que parecen haber evaluado cada perfil con el mismo valor, presumiblemente debido a que puede ser spam o un intento poco serio en la calificación. Se conservan los datos de usuarios con más calificaciones; su aportación es útil, en comparación con las votaciones de los usuarios menos comprometidos. Varias métricas de similitud y definiciones de vecindario están disponibles en Mahout. Para tener una idea de lo que funciona y lo que no funciona, se puede probar muchas combinaciones. Para nuestros propósitos aquí, se usaran recomendación de filtrado colaborativo basado en el usuario, distancia Euclídea y vecinos cercano.

5.2.1. Métodos de obtención de preferencias del usuario

La obtención de preferencias desarrollada en este prototipo será del tipo explícito. El usuario indicara de forma explícita sus gustos en cuanto a géneros de películas.

5.2.2. Establecer requerimientos no funcionales

Los requisitos no funcionales son las restricciones impuestas a los requisitos funcionales del sistema.

- La respuesta del sistema debe ser rápida, y las recomendaciones ofrecidas deben aparecer en el menor tiempo posible. Es decir es importante tener en cuenta la velocidad del sistema, y el rendimiento de las consultas a la base de datos.
- El sistema debe implementar una interfaz gráfica intuitiva y de fácil uso, estas características deben promover una interacción efectiva.
- La aplicación debe ser diseñada para que sea tolerante a fallos.

5.2.3. Establecer las funcionalidades del sistemas mediante historias de usuarios

5.2.3.1. Ingreso del perfil de un nuevo usuario

Tabla 9. Historia de usuario: Ingreso de perfil de un nuevo usuario

Nombre de Historia:	Ingreso del perfil de un nuevo usuario		
Numero de tarea:	1	Riesgo en desarrollo:	Baja
Prioridad :	Alta	Iteracion asignada:	1
Nivel de dificultad:	2 (1-5)	Dependencia:	0
Descripción:	El sistema debe proporcionar un mecanismo para que el usuario pueda registrarse en el sistema. El sistema puede preguntar las preferencias del usuario como parte del registro de nuevo usuario. Y además debe aceptar los términos de condición de uso.		
Validación :	El sistema debe guardar correctamente la nueva información del usuario.		

Fuente: Elaboración Propia

5.2.3.2. Visualizar perfil del usuario

Tabla 10. Historia de usuario: Visualizar el perfil de un usuario

Nombre de Historia:	Visualizar perfil del usuario		
Numero de tarea:	2	Riesgo en desarrollo:	Baja
Prioridad :	media	Iteracion asignada:	1
Nivel de dificultad:	1 (1-5)	Dependencia:	1
Descripción:	El sistema debe permitir al usuario visualizar la información de su perfil, incluyendo sus preferencias.		
Validación :	Visualización correcta de la información del usuario.		

Fuente: Elaboración Propia

5.2.3.3. Actualizar el perfil del usuario

Tabla 11. Historia de usuario: Actualizar el perfil del usuario

Nombre de Historia:	Actualizar el perfil del usuario		
Numero de tarea:	3	Riesgo en desarrollo:	Baja
Prioridad :	media	Iteracion asignada:	1
Nivel de dificultad:	1 (1-5)	Dependencia:	2
Descripción:	El usuario debe poder modificar sus preferencias y sus datos. Si el usuario lo desea, debe poder darse de baja del sistema.		
Validación :	Actualización correcta de la información del usuario.		

Fuente: Elaboración Propia

5.2.3.4. Calificar de películas

Tabla 12. Historia de usuario: Calificación de películas

Nombre de Historia:	Calificación de películas		
Numero de tarea:	4	Riesgo en desarrollo:	Baja
Prioridad :	media	Iteracion asignada:	1
Nivel de dificultad:	2 (1-5)	Dependencia:	0
Descripción:	Se necesita implementar alguna manera de asignarle valores numéricos a las puntuaciones que el usuario provee acerca de las películas. Desarrollar una escala de calificación 1- malo a 5-excelente.		
Validación :	El sistema ha de permitir al usuario evaluar numéricamente las películas, y debe poder modificar esta puntuación si lo desea.		

Fuente: Elaboración Propia

5.2.3.5. Gestión de comentarios

Tabla 13. Historia de usuario: Gestión de comentarios

Nombre de Historia:	Gestión de comentarios de películas		
Numero de tarea:	5	Riesgo en desarrollo:	Baja
Prioridad :	Baja	Iteracion asignada:	1
Nivel de dificultad:	1 (1-5)	Dependencia:	4
Descripción:	Permitir el ingreso, editado, eliminado, visualización y guardado de los comentarios que el usuario ingrese acerca de las películas comentadas.		
Validación :	El sistema debe permitir la fácil gestión de comentarios		

Fuente: Elaboración Propia

5.2.3.6. Diseñar la presentación de la recomendación

Tabla 14. Historia de usuario: Diseñar la presentación de la recomendación

Nombre de Historia:	Diseñar la presentación de la recomendación		
Numero de tarea:	6	Riesgo en desarrollo:	Alta
Prioridad :	Alta	Iteracion asignada:	2
Nivel de dificultad:	1 (1-5)	Dependencia:	5
Descripcion:	Presentar la recomendación de forma amigable para el usuario, de manera que el usuario se sienta atraído a las recomendaciones personalizadas y de su opinión.		
Validación :	El usuario se siente cómodo con las recomendaciones que se le ofrecen, y hace uso de las calificaciones y comentarios del sistema.		

Fuente: Elaboración Propia

5.2.3.7. Diseñar una forma de justificación de recomendación para el usuario

Tabla 15. Historia de usuario: Diseñar la justificación de la recomendación

Nombre de Historia:	Justificación de la recomendación		
Numero de tarea:	7	Riesgo en desarrollo:	Media
Prioridad :	Media	Iteracion asignada:	2
Nivel de dificultad:	1 (1-5)	Dependencia:	6
Descripcion:	Ofrecer al usuario explicaciones sencillas de la recomendación, haciendo uso de gráficos simples y con información comprensible para el usuario. Esto se hace con el objetivo de proveer a los usuarios con información detallada acerca de la predicción hecha, en lugar de mostrar solo un puntaje. Esta parte del sistema ayuda a los usuarios a entender porque el SR escogió esa película, y de esta forma el usuario puede indicar si el		

	razonamiento del SR es correcto.
Validación :	Explicación sencilla de porque se escogió esas recomendaciones para el usuario.

Fuente: Elaboración propia



5.3. Diseño y codificación

Diseño de actividades, se iniciara el diseño con una vista amplia de la arquitectura del sistema y se termina con un diseño mucho más específico de la arquitectura

5.3.1. Diseño de la arquitectura

5.3.1.1. Herramientas

- IDE : Eclipse
- VCS version control system : GIT
- Base de datos mysql
- Servidor web : Tomcat
- Java Jdk 8
- Mokaups para hacer el diseño de las pantallas

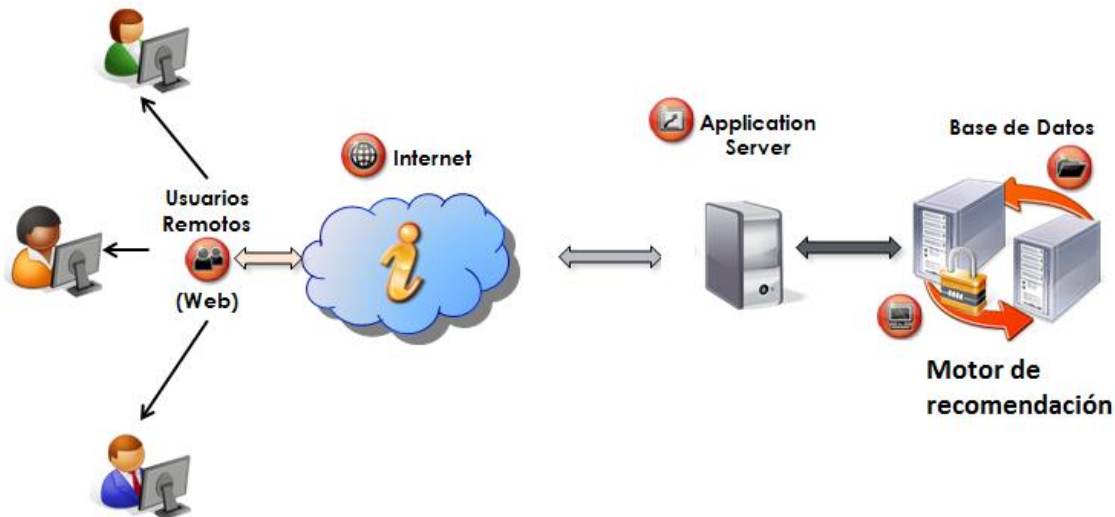


Figura 12. Arquitectura general del sistema

Fuente: Elaboración propia

5.3.2. Diseño de la estructura de los datos

5.3.2.1. Datos de los usuarios:

Almacena la información personal de cada usuario, cada usuario tiene que registrarse en el sistema.

5.3.2.2. Datos de películas:

Almacena las películas y sus atributos.

5.3.2.3. Datos de preferencias:

Almacena las calificaciones que los usuarios proveen de cada película

Tabla 16. Diccionario de datos de la tabla Movie

TABLA	CAMPO	TIPO DE DATO	P K	N N	DESCRIPCION
Movie	MovieID	INTEGER(11)	x	x	Identificador de películas
Movie	backdrop_path_orig	VARCHAR(255)			Portada de la película
Movie	original_title	TEXT			Título original
Movie	overview	TEXT			Reseña de la película
Movie	homepage	VARCHAR(255)			Página oficial de la película
Movie	release_date	VARCHAR(255)			Fecha de estreno
Movie	poster_path_orig	VARCHAR(255)			Afiche oficial
Movie	rating	INTEGER(11)			Calificación promedio de la película
Movie	vote_average	FLOAT			Calificación promedio de la película
Movie	vote_count	INTEGER(11)			Numero de persona que calificaron la película
Movie	insertion_date	TIMESTAMP			Fecha de inserción a la base de datos
Movie	update_date	TIMESTAMP			Fecha de actualización
Movie	popularity	FLOAT			Popularidad

Fuente: Elaboración propia

Tabla 17. Diccionario de datos de la tabla Ratings

TABLA	CAMPO	TIPO DE DATO	P K	N N	DESCRIPCION
Ratings	UserId	INTEGER(11)	x	x	Identificador del usuario que hizo la calificación
Ratings	MovieID	INTEGER(11)	x	x	Identificador de la película que se califico
Ratings	Rating	Decimal (11,0)		x	Calificación del usuario para esta película, este valor debe estar entre el 1 y 5.
Ratings	Timestamp	INTEGER(11)		x	Fecha en la que se llevó a cabo la calificación

Fuente: Elaboración Propia

Tabla 18. Diccionario de datos de la tabla user_roles

TABLA	CAMPO	TIPO DE DATO	PK	NN	DESCRIPCION
user_roles	user_role_id	INTEGER(11)	x	x	Identificador de usuario
user_roles	username	VARCHAR(45)		x	Nombre del usuario
user_roles	ROLE	VARCHAR(45)		x	Tipo de rol que tiene este usuario

Fuente: Elaboración propia

Tabla 19. Diccionario de datos de la tabla users

TABLA	CAMPO	TIPO DE DATO	P K	N N	DESCRIPCION
Users	Userld	INTEGER(11)	x	x	Identificador único de usuario
Users	Name	Varchar(50)		x	Nombre de usuario
Users	Password	Varchar(50)		x	Contraseña de usuario
Users	Country	Varchar(50)		x	Ciudad del usuario
Users	Email	Varchar(50)		x	Correo electrónico

Fuente: Elaboración propia

Tabla 20. Diccionario de datos de la tabla genres _movie

TABLA	CAMPO	TIPO DE DATO	P K	N N	DESCRIPCION
Genres_movi e	genre_tmdb_id	INTEGER(11)		x	Id de genero
Genres_movi e	movie_tmdb_id	INTEGER(11)		x	Id de película

Fuente: Elaboración propia

Tabla 21. Diccionario de datos de la tabla genres

TABLA	CAMPO	TIPO DE DATO	PK	NN	DESCRIPCION
Genres	id	INTEGER(11)	x	x	Identificador del genero
Genres	tmdb_id	INTEGER(11)		x	Identificador de the movie DB
Genres	name	VARCHAR(255)		x	Nombre del genero

Fuente: Elaboración propia

Tabla 22. Diccionario de datos de la tabla roles

TABLA	CAMPO	TIPO DE DATO	PK	NN	DESCRIPCION
Roles	id	INTEGER(11)	x	x	Identificador del rol
Roles	Descripción	VARCHAR(20)			Descripción del rol

Fuente: Elaboración propia

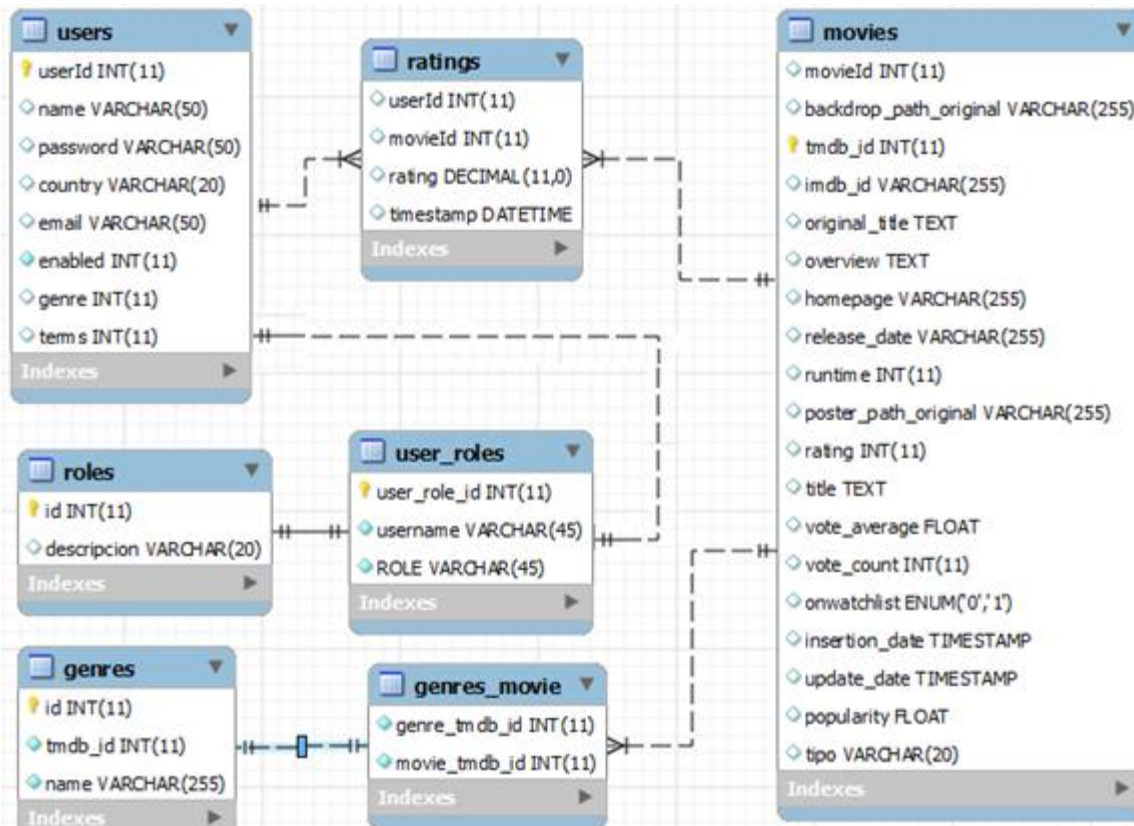


Figura 13. Diagrama de entidad de relación del prototipo propuesto

Fuente: Elaboración propia

5.3.3. Diseñar el motor de recomendación

El enfoque que se va a utilizar para el desarrollo del prototipo del motor de recomendación es un enfoque híbrido que combina los métodos basados en contenido y el filtrado colaborativo. La razón por la que optamos por este enfoque reside en la necesidad de combinar varios modelos de recomendación para poder cubrir las limitaciones propias de cada modelo. Los modelos de recomendación implementados en las páginas web de servicio de comercio electrónico por lo general combinan varios tipos de modelos de recomendación.

El prototipo será capaz de proporcionar recomendaciones utilizando la biblioteca Apache Mahout, además se implementará la base de datos en MySQL, con los conjuntos de datos extraídos de MovieLens para recomendar películas.

Mahout es una biblioteca de aprendizaje abierto de Apache, es altamente escalable y permite a los desarrolladores utilizar algoritmos como el filtrado colaborativo. Apache Mahout se está convirtiendo en una de las bibliotecas más popular para los proyectos de recomendación.

Mahout significa principalmente motores de recomendación de filtrado colaborativo, agrupación y clasificación.

5.3.3.1. Proceso de recomendación del SRFC

5.3.3.1.1. Acceso a datos

Clase en Mahout: DataModel

Guarda y proporciona acceso a todas las preferencias de usuario, y los datos de artículos necesarios en el cálculo.

5.3.3.1.2. Cálculo de vecinos

Se realiza un cálculo del vecindario, esto es, los usuarios con gustos o necesidades más parecidas al usuario (Gallardo, 2012). Aplicada a la recomendación, el algoritmo de vecinos cercanos calcula los usuarios más parecidos al usuario activo (vecindario) y realiza una predicción de la valoración de preferencia que el usuario daría a un producto utilizando las valoraciones de preferencia que otorgaron sus vecinos más cercanos (Cover and Hart, 2006). Las variables a tener en cuenta son el número de vecinos que se utilizan (k), la medida de similitud a utilizar para el cálculo de los vecinos (distancia Euclídea, medida del coseno, similitud de Pearson, tanimoto) y la técnica de predicción que se

aplica para agregar las valoraciones de preferencia de los vecinos (suma ponderada, suma ponderada ajustada con la media)(Gallardo, 2012).

Clase en Mahout : UserNeighborhood

Define una noción de un grupo de usuarios que están y son más similares. Debido a que la recomendación está basada en la opinión de otros usuarios, se debe encontrar el vecindario más apropiado para el usuario. La selección del vecindario tiene un gran impacto en el proceso de la recomendación, y puede definir el éxito o el fracaso de la recomendación generada (Herlocker et al., 2000).

5.3.3.1.3. Cálculo de similitud entre usuarios

En el caso de los sistemas de recomendación, la correlación de Pearson se usa para medir la tendencia de los valores de preferencia de dos usuarios a moverse juntos, si es relativamente alta o relativamente baja, respecto a los mismos artículos. La correlación de Pearson es un número entre -1 y 1 que mide la tendencia de dos series de números, emparejado uno-a-uno, a moverse juntos. Mide la tendencia de los números a moverse juntos de forma proporcional, de tal manera que hay una relación aproximadamente lineal entre los valores de una serie y la otra. Cuando esta tendencia es alta, la correlación es cercana a 1. Cuando parece que hay poca relación en absoluto, el valor está cerca de 0 (OWEN et al., 2012). La correlación de Pearson se calcularía como se muestra en la tabla 6. La interpretación sería la siguiente, el usuario 2 y el usuario 1 tienen una similitud de -0.76.

Tabla 23. Tabla que muestra el cálculo de la correlación de Pearson

	Item 101	Item 102	Item 103	Correlation with user 1
User 1	5.0	3.0	2.5	1.000
User 2	2.0	2.5	5.0	-0.764
User 3	2.5	.	.	.
User 4	5.0	.	3.0	1.000
User 5	4.0	3.0	2.0	0.945

Fuente:(OWEN et al., 2012)

Clase en Mahout: UserSimilarity

Proporciona una idea de cuán similares son dos usuarios, esto podría estar basado en una de las muchas métricas o cálculos posibles. Sin una noción fiable y eficaz de cálculo de similitud entre usuarios, este enfoque se desmoronaría. El ejemplo ha utilizado la aplicación PearsonCorrelationSimilarity, que es una métrica de similitud sobre la base de la correlación de Pearson.

5.3.3.1.4. Recomendación de elementos

Una vez obtenidos los vecinos se realiza una predicción que estima el valor de preferencia que el usuario activo daría a cada uno de los productos que no ha valorado (Gallardo, 2012). Se ordena la lista de los productos recomendados por su valor de preferencia predicho en orden descendente y se recomiendan los N primeros productos de la lista (Gallardo, 2012).

Clase en Mahout: Recommender

Coloca todos los componentes anteriores juntos para procesar la recomendación.

```

class RecommenderIntro {
    public static void main(String[] args) throws Exception {
        DataModel model =
            new FileDataModel (new File("intro.csv"));      ← Load data file

        UserSimilarity similarity =
            new PearsonCorrelationSimilarity (model);

        UserNeighborhood neighborhood =
            new NearestNUserNeighborhood (2, similarity, model);

        Recommender recommender = new GenericUserBasedRecommender (
            model, neighborhood, similarity);                ← Create
                                                           recommender engine
        List<RecommendedItem> recommendations =
            recommender.recommend(1, 1);

        for (RecommendedItem recommendation : recommendations) {
            System.out.println(recommendation);
        }
    }
}
    
```

Figura 14. Código para la implementación de un motor de recomendación usando Mahout

Fuente:(OWEN et al., 2012)

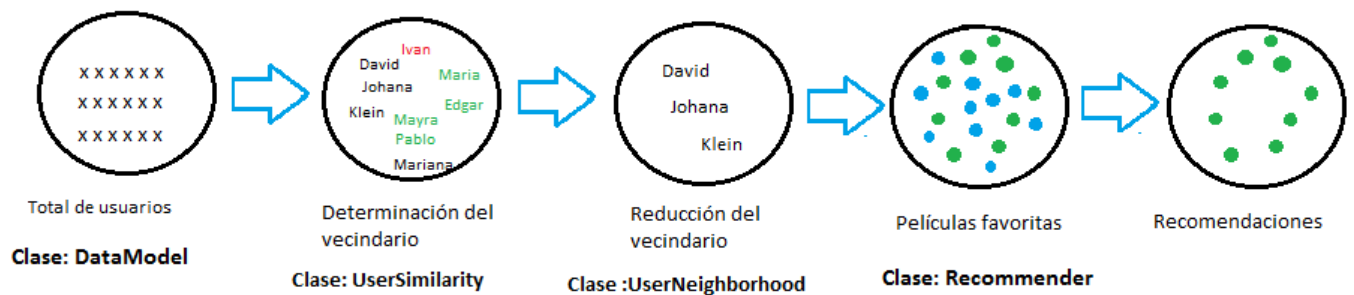


Figura 15. Proceso de recomendación SRFC usando Mahout

Fuente: Elaboración Propia

5.3.3.1.5. Diagrama de secuencia del SRFC

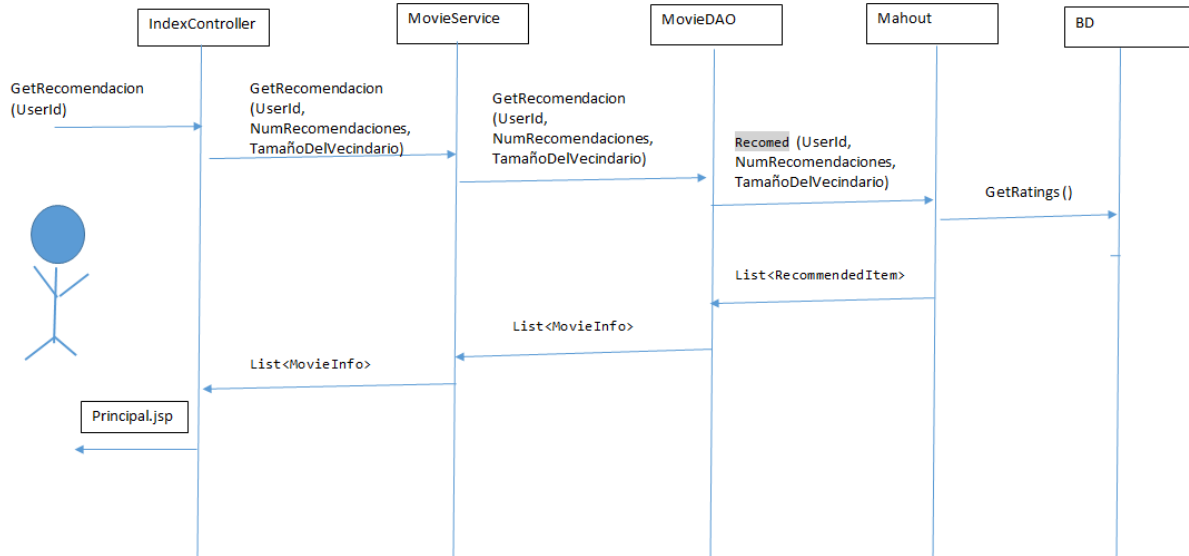


Figura 16. Diagrama de secuencia

Fuente: Elaboración propia

5.3.3.1. Proceso de recomendación del SRBC

Para iniciar este proceso de recomendación basado en SRBC el usuario debe indicar al momento de su registro el género de su preferencia, después esta información se cruzara con los géneros de películas registrados en la BD. El usuario tendrá como recomendación todas las películas con el género que el indicó en su registro.

5.3.1. Diseñar la Presentación de la recomendación

En el caso del prototipo la presentación de la recomendación es el núcleo del sistema por lo cual, la presentación será explícita.



Figura 17. Pantalla de inicio del sistema

Fuente: Elaboración propia

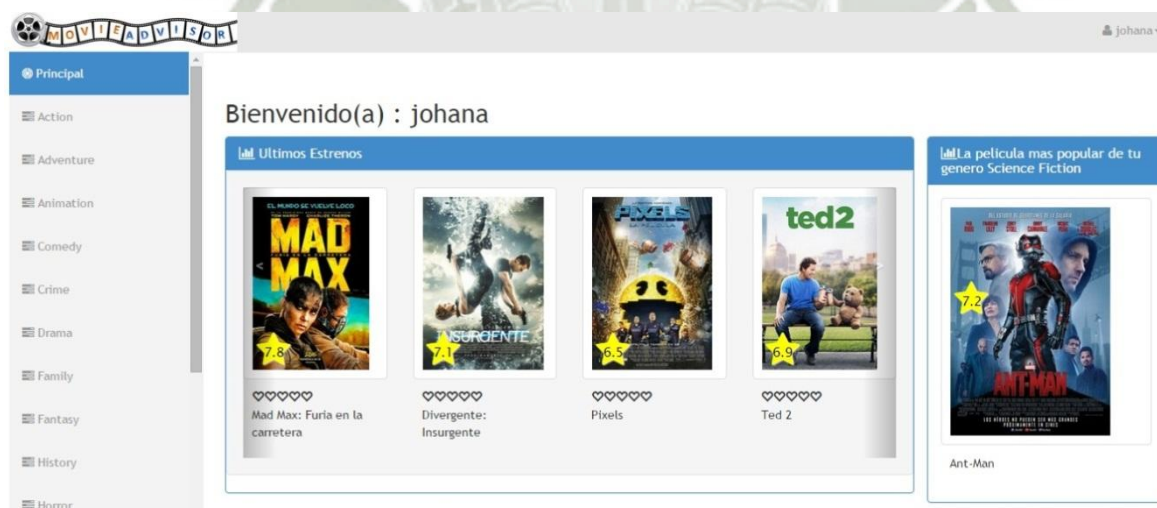


Figura 18: Pantalla presentación de últimos estrenos

Fuente: Elaboración propia

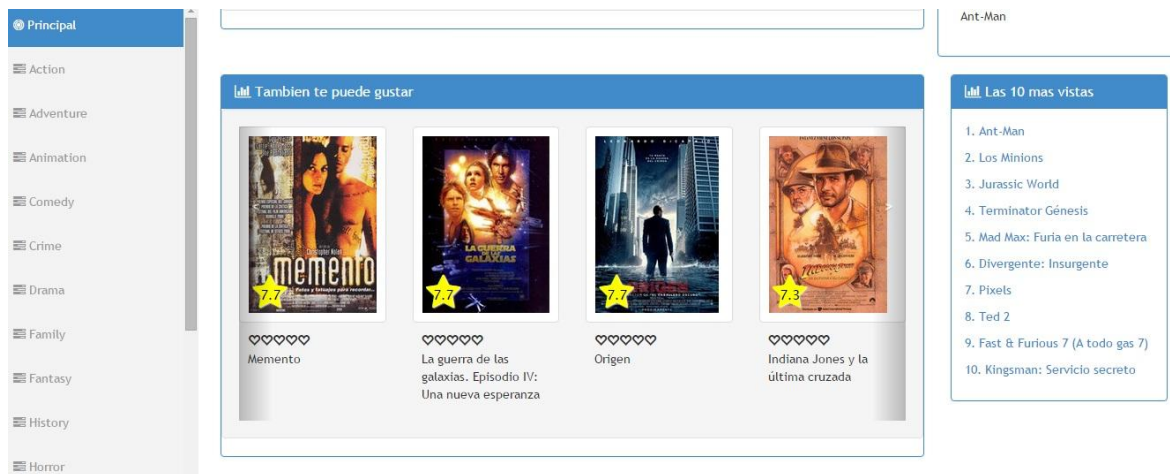


Figura 19. Pantalla presentación de recomendación

Fuente: Elaboración propia



5.4. Pruebas

La etapa consistirá en la realización de diferentes pruebas para comprobar el correcto funcionamiento del sistema y, sobretodo, del algoritmo de recomendaciones. Además, se procederá a realizar las diferentes evaluaciones para comprobar el rendimiento del sistema. De esta etapa, obtendremos los resultados de las pruebas realizadas y los resultados de la evaluación del sistema.

5.4.1. Efectuar las pruebas del sistema

Verificar los resultados de la implementación

5.4.1.1. Pruebas con el usuario

Realizar pruebas directas con el usuario es complejo, en el desarrollo de SR es más complejo aun porque involucra las preferencias del usuario y su comportamiento. El usuario debe tener una breve explicación de cómo se hizo la recomendación, esto para ayuda a los usuarios a entender porque el SR escogió la película mostrada, de esta forma el usuario puede indicar si el razonamiento del SR es correcto. No es necesario una explicación completa de cómo funciona el sistema.

Con este fin se pueden usar diferentes técnicas:

- Registros de uso del sistema para saber cuáles son las características que más usan los usuarios, y cuál es el resultado de este uso. Por ejemplo, cuantos usuarios hacen compras después de ver una recomendación, o saber si el usuario ignora la recomendación.
- Encuestas, cuestionarios, grupos de enfoque.
- Experimentos de campo.

5.4.1.2. Evaluar la satisfacción del cliente respecto al aplicativo

- Cobertura
- Satisfacción con la recomendación

5.4.2. Evaluar la calidad de recomendación

Debe quedar muy claro que los algoritmos y optimizaciones que aquí resulten más exactos y efectivos pueden no funcionar igual en otro ámbito y/o con otro tipo de datos; es algo característico de los algoritmos de filtrado colaborativo y conocido en toda la literatura que este tipo de algoritmos es muy dependiente del dominio concretos en el que actúan.

Se hicieron pruebas con la información extraída la base de datos movilens, primero se usara el 70% de los datos para entrenar al motor de recomendación, el 30% de los datos se usaran para comparar los resultados emitidos por el sistema con las calificaciones que se tienen en la base de datos. El sistema debe predecir el valor de la recomendación, este resultado se debe comparar con el valor real, que se encuentra con la base de datos, haciendo estas pruebas se debe poder evaluar la calidad de la recomendación. El sistema de recomendación estimó las preferencias que faltaban, y las estimaciones se compararon con los valores reales.

- Conjunto de entrenamiento = 70%
- Conjunto de test = 30%

Se entrenó el sistema con el 70% de los datos y se trató de predecir el 30% restante. Se tomó una muestra de 5 usuarios. Los resultados muestran que para esta muestra, la diferencia entre la calificación real dada por el usuario y la predicción calculada por el sistema es 0.75.

**Tabla 24. Comparación de la calificación hecha por el usuario
y la calificación dada por el sistema**

USUARIO	CALIFICACIÓN	PREDICCIÓN	DIFERENCIA
Usuario 1	3	2,5	0,5
Usuario 2	4	5	1
Usuario 3	4	3	1
Usuario 4	5	4,5	0,5
		MAE	0,75

Fuente: Elaboración propia

```

RandomUtils.useTestSeed();
DataModel model = new FileDataModel (new File("intro.csv"));
RecommenderEvaluator evaluator =
    new AverageAbsoluteDifferenceRecommenderEvaluator ();
RecommenderBuilder builder = new RecommenderBuilder() {
    @Override
    public Recommender buildRecommender(DataModel model)
        throws TasteException {

        UserSimilarity similarity = new PearsonCorrelationSimilarity (model);
        UserNeighborhood neighborhood =
            new NearestUserNeighborhood (2, similarity, model);
        return
            new GenericUserBasedRecommender (model, neighborhood, similarity);
    }
};
double score = evaluator.evaluate(
    builder, null, model, 0.7, 1.0);
System.out.println(score);
    
```

← Generates repeatable results

← Builds recommender

← Trains with 70% of data; tests with 30%

**Figura 20. Código para la implementación de la
evaluación de la recomendación con Mahout**

Fuente:(OWEN et al., 2012)

5.4.1. Resultados de la evaluación de la recomendación

En términos de eficiencia y exactitud se debe eligió un subconjunto de usuarios para el cálculo de la predicción en vez de usar el conjunto completo.

5.4.1.1. Problemas con la determinación del vecindario

Se encontró entonces un problema de clasificación, en el que se trata de agrupar a los usuarios en grupos o clases: lo que sirven para elaborar la predicción, y los que no son considerables. En pocas palabras la elección de los mejores k vecinos, afecta considerablemente el proceso de recomendación. Al variar el umbral del vecindario se pudo observar que colocando un valor alto para la selección del vecindario puede haber algunos problemas, como por ejemplo que no haya un número suficiente de vecinos, provocando problemas de cobertura, o bien que el número de vecinos elegidos sea excesivamente grande, computacionalmente hablando. La elección del vecindario es importante ya que si elegimos un número muy grande y los datos está muy dispersos entonces no habrá una buena calidad en la predicción, pero si el número es demasiado pequeño, la cobertura se reducirá y habrá muchas sugerencias que podrían perderse.

5.4.1.2. Problemas de escalabilidad

Para que el sistema pueda realizar las predicciones usando el filtrado colaborativo, debe haber varias valoraciones hechas por los usuarios. El sistema hace predicciones más exactas si hay varias valoraciones, a su vez con un mayor número de usuarios se hace más complejo seleccionar el vecindario.

5.4.1.3. Problemas del nuevo usuario

Un nuevo usuario debe hacer varias calificaciones al comienzo para que el sistema pueda comprender sus preferencias.



5.5. Comprobación de los factores aplicados en el desarrollo del prototipo

Los factores propuestos para el desarrollo de SR, dieron la pauta para el desarrollo del prototipo, el cual es funcional. El prototipo de SR es capaz de ofrecer recomendaciones a los usuarios nuevos y antiguos. La calidad de la recomendación puede ser mejorada entrenando más al SR, sin embargo la recomendación resultante es aceptable.

Tabla 25. Factores incluidos en el desarrollo del prototipo

ACTIVIDAD	FACTORES INCLUIDOS EN EL PROTOTIPO	
Seleccionar la metodología de desarrollo adecuada para un SR	Tipo de metodología de desarrollo	Si
Definir el modelo de negocio de la recomendación	Dominio de la recomendación	Si
	Propósito de la recomendación	Si
	Contexto de la recomendación	No
	Grado de personalización de la recomendación	Si
	Métricas comerciales	No
	Integridad de la recomendación	No
Obtener los datos de perfil y preferencia	Métodos de obtención de preferencias del usuario	Si
	Perfiles de usuarios	No
	Información actual	No
Diseñar el motor de recomendación	Tipo de modelo de recomendación	Si
	Proceso de recomendación	Si
	Presentación de la recomendación	Si
Evaluar la calidad de recomendación	Exactitud de rango	No
	Cobertura	Si
	Diversidad	Si
	Precisión	Si

Fuente: Elaboración propia

5.5.1. Factor tipo de metodología

El requisito de realizar un prototipo que efectuó recomendaciones, fue completado en la primera iteración satisfactoriamente, gracias a la aplicación de la metodología.

5.5.2. Factor dominio de la recomendación

El dominio de la recomendación son películas. Como se puede apreciar en las siguientes figuras, el dominio esta sesgado a 9000 películas de todos los géneros.

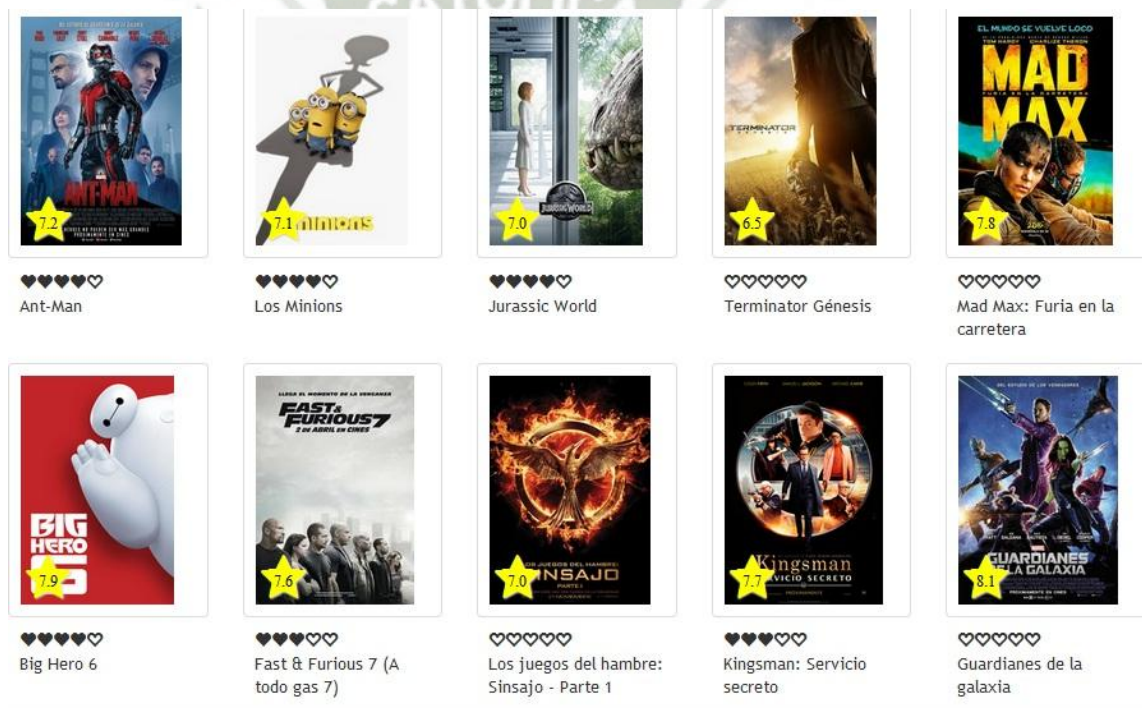


Figura 21: Películas del genero acción

Fuente: Elaboración propia

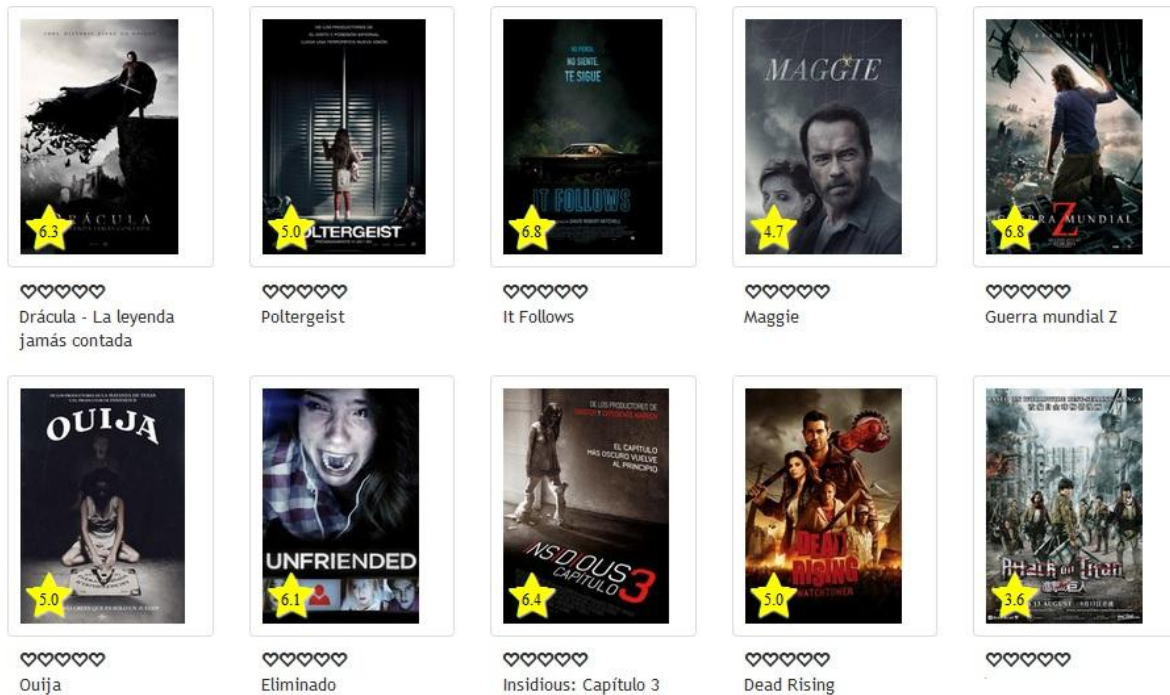


Figura 22. Películas del genero drama

Fuente: Elaboración propia

5.5.3. Factor propósito de la recomendación

El propósito del prototipo es demostrar la aplicación de los factores mencionados. El propósito de la recomendación emitida por el prototipo, ayudara a los usuarios en la selección de películas que pueden ser de su agrado, es decir dará soporte a sus decisiones.

La siguiente tabla muestra la aceptación de las recomendaciones por parte del usuario, calificadas del 1 al 10, siendo 1 el usuario piensa que la recomendación no cumple su propósito de ayudar a su decisión, y siendo 10 es una recomendación completamente aceptable que debería ser tomada en cuenta. La aceptación de la recomendación fue de un 4.6.

Tabla 26. Tabla de aceptación de las recomendaciones por parte del usuario

USUARIO	ACEPTACIÓN
Usuario1	3
Usuario2	4
Usuario3	3
Usuario4	5
Usuario5	8
	4,6

Fuente: Elaboración propia

5.5.4. Factor Personalización de la recomendación

Como se puede apreciar la recomendación ofrecida por el sistema es diferente para los tres usuarios, lo que demuestra que la personalización es por usuarios.

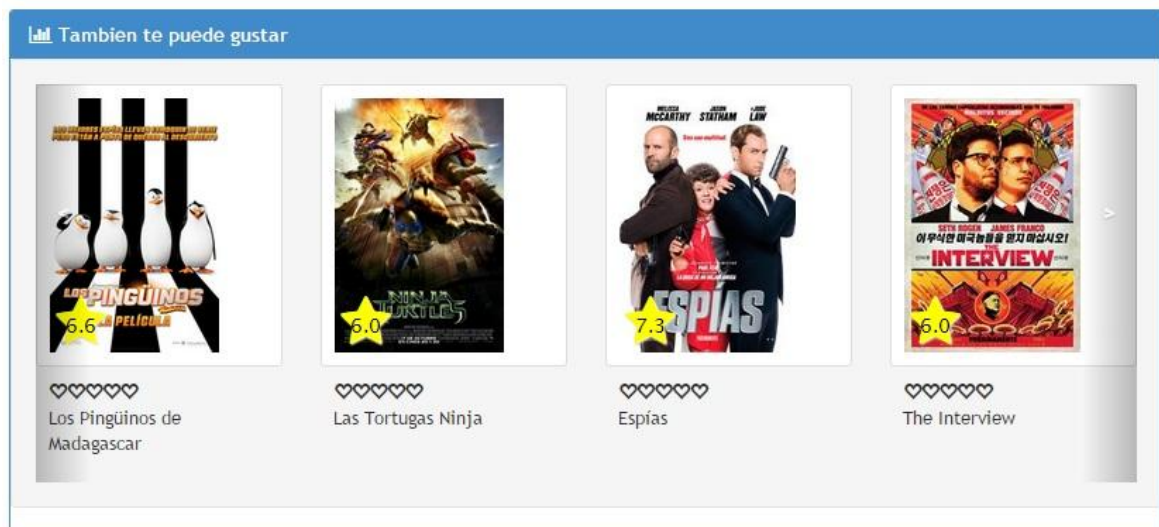


Figura 23. Personalización para el usuario 1

Fuente: Elaboración propia

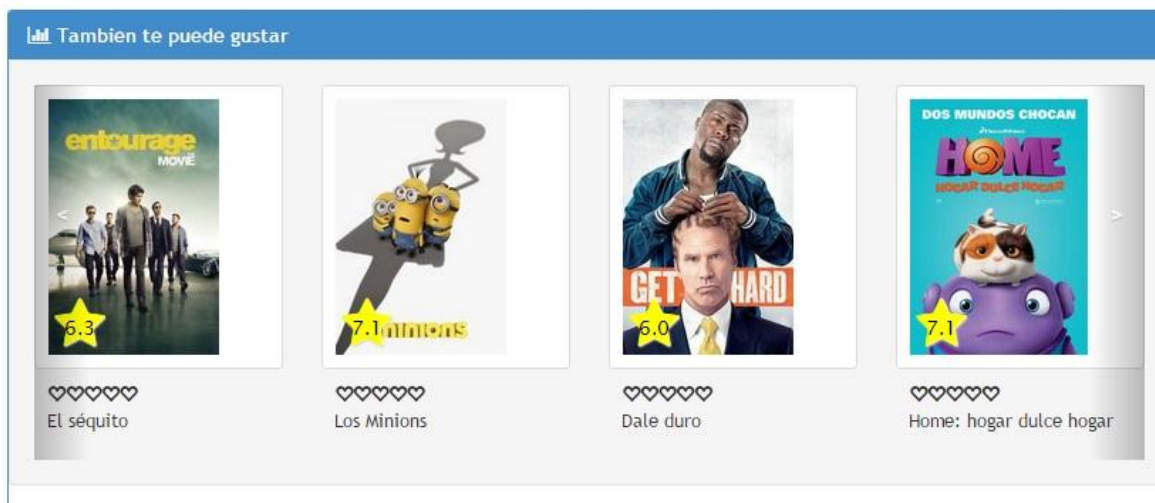


Figura 24. Personalización para el usuario 2

Fuente: Elaboración propia

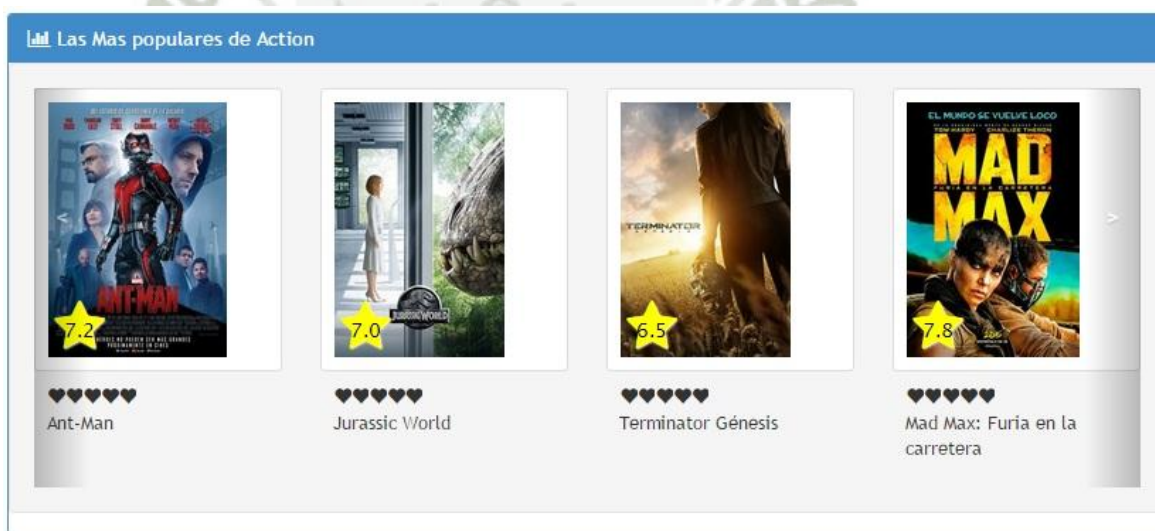


Figura 25. Personalización para el usuario 3

Fuente: Elaboración propia

5.5.5. Factor métodos de obtención de preferencias del usuario

La obtención de las preferencias del usuario son explícitas. Se le pide al usuario su opinión acerca de la calificación que le daría a la película, estos valores podrían ser calificaciones en una escala de 1 a 5, donde 1 indica los elementos que el usuario no disfruta en absoluto, y 5 indica favoritos.

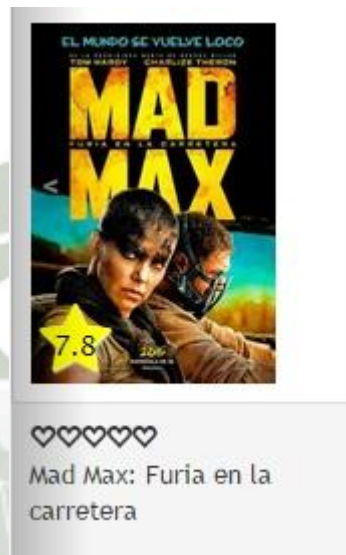


Figura 26. Obtención de preferencias por el usuario por medio de calificaciones

Fuente: Elaboración propia

A un nuevo usuario se le recomendará películas basadas en las preferencias seleccionadas de su perfil.

1. Selección de
Preferencia

2. Recomendaciones
ofrecidas de acuerdo
a la selección de
preferencia del
usuario

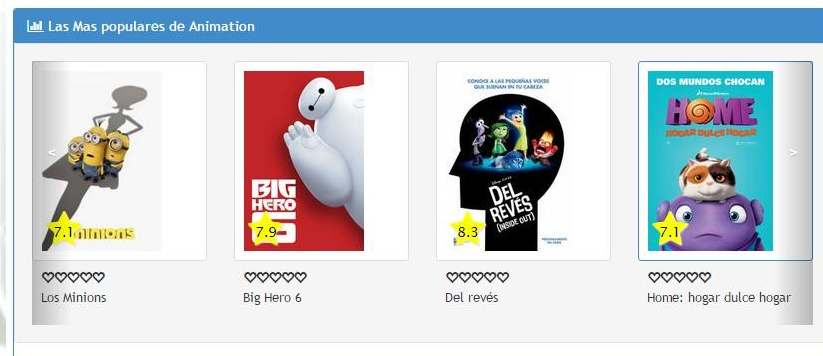


Figura 27. Obtención de preferencias por el usuario por medio de selección de género de preferencia

Fuente: Elaboración propia

5.5.6. Factor tipo de modelo de recomendación

La aplicación del SR híbrido solucionó el problema del arranque frío de usuarios, y ayudo a proporcionar diferentes tipos de recomendación.

Las recomendaciones no personalizadas, son los rankings de las películas más vistas.



Figura 28. Recomendación sin personalización

Fuente: Elaboración propia

Las recomendaciones basadas en contenido usan información del género de la película seleccionado por el usuario.



Figura 29. . Recomendación basada en contenido

Fuente: Elaboración propia

Las recomendaciones basadas en el modelo de filtrado colaborativo se aplican a los usuarios que ya tiene un perfil más definido.

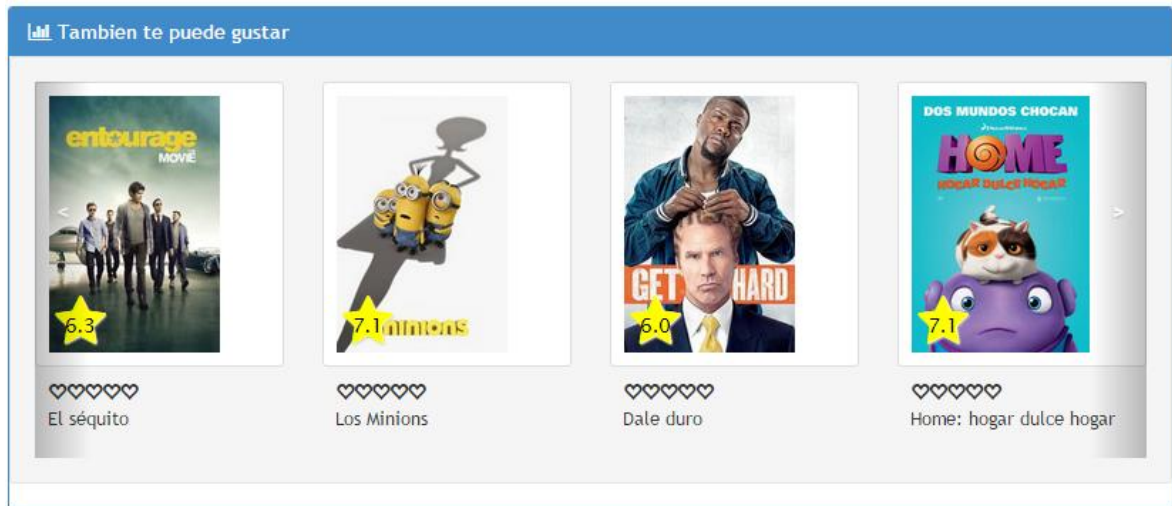


Figura 30. Recomendación basada en filtrado colaborativo

Fuente: Elaboración propia

5.5.7. Factor proceso de recomendación

El proceso de recomendación fue medido y evaluado con diferentes métodos. El error absoluto medio es la medida estadística seleccionada para la estimación de la exactitud con la que el sistema realizará las predicciones.

Durante el proceso de evaluación se varió de métodos de cálculo de similitud entre usuarios, los resultados fueron: la distancia euclidiana tuvo menos precisión que los otros métodos, sin embargo el tiempo de proceso fue más corto y mostraba más recomendaciones por usuarios. El método de correlación de Pearson fue que el demoro menos, sin embargo no produjo tantas recomendaciones como distancia euclidiana. Loglikelihood fue el método que más demoro, y causo problemas con el tiempo de

respuesta de la aplicación. Tanimoto tuvo el mejor puntaje en cuanto precisión en el cálculo de similitud, pero también el tiempo de ejecución fue mayor que el método de correlación de Pearson y distancia Euclídea.

Tabla 27. Comparación de métodos de cálculo de similitud

MÉTODOS	MAE	TIEMPO
Distancia Euclídea	0.13710711932763814	89ms
Correlación Pearson	0.5424198646567864	84ms
Loglikelihood	0.797738729602	661ms
Tanimoto	0.808423418490	424ms
Cálculo de similitud evaluando 700 usuarios		

Fuente: Elaboración propia

5.5.8. Factor presentación de la recomendación

La recomendación ofrecida será el núcleo del prototipo, además se podrá visualizar información adicional de las películas recomendadas.

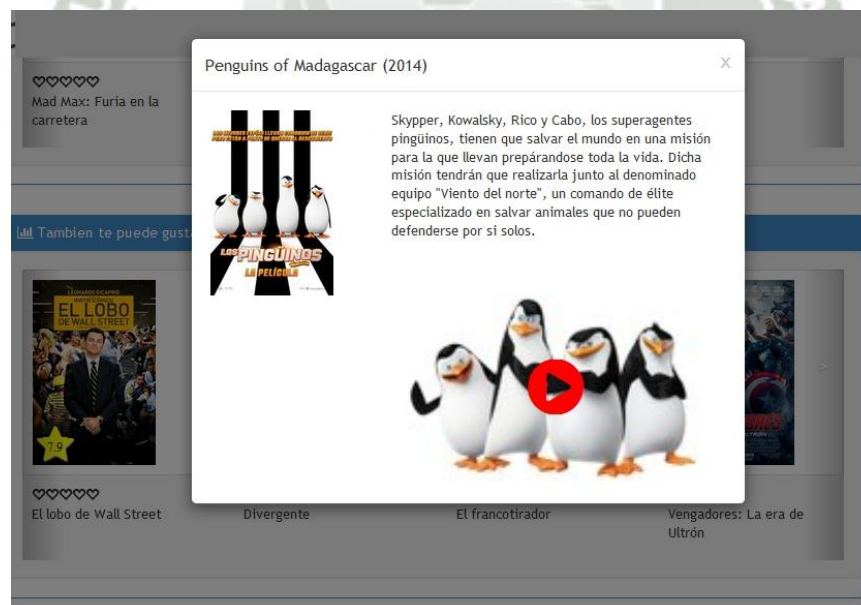


Figura 31. Presentación de la información de películas

Fuente: Elaboración propia

5.5.9. Factor cobertura

Se hizo una simulación para tratar de averiguar cuántos artículos del catálogo se recomienda. La simulación calculó 10 recomendaciones para 700 usuarios, teniendo un universo de 9000 películas para poder proveer las recomendaciones. El cálculo varió en cada simulación. El resultado fue el siguiente, la cobertura promedio de películas recomendadas es de un 10,06%.

Tabla 28. Calculo de cobertura alcanzada por el prototipo

NRO ITERACION	TOTAL DE PELICULAS	NRO DE PELICULAS RECOMENDADAS
1	9000	877
2	9000	900
3	9000	950
4	9000	850
Promedio		894,25
Cobertura 9000/894,25		10,06429969

Fuente: Elaboración propia

5.5.10. Factor diversidad

Se hizo la evaluación de las recomendaciones producidas para 4 usuarios, y se evaluó la diversidad de las recomendaciones que había en la muestra. El resultado muestra que de 40 recomendaciones de películas, 15% de películas recomendadas son de acción, siendo este el género más recomendado. Los géneros menos recomendados son: aventura, animación, y drama con una participación de solo el 5%.

Tabla 29. Tabla de cálculo de diversidad de género para 40 recomendaciones

Género	Usuario1	Usuario2	Usuario3	Usuario4	Sumatoria por genero	Diversidad por genero
Acción	1	3	1	1	6	15%
Aventura		1		1	2	5%
Animación	2				2	5%
Comedia		3	1		4	10%
Crimen		2		4	6	15%
Drama			2		2	5%
Familia	2		2		4	10%
Fantasia		1	2		3	7,5%
Horror	4				4	10%
Misterio				4	4	10%
Romance	1		2		3	7,5%
TOTAL	10	10	10	10	40	100%

Fuente: Elaboración propia

5.5.11. Factor Precisión

Otra forma de calcular la precisión fue preguntando a los usuarios el número de buenas recomendaciones incluidas en su recomendación. Esta evaluación se hizo con usuarios reales, y tuvo una precisión promedio de 0.34, ningún usuario dijo que tenía más de la mitad de buenas recomendaciones del total de recomendaciones ofrecidas por el sistema.

Tabla 30. Evaluación de la precisión de las recomendaciones

Usuario	Total de recomendaciones	Nro. de buenas recomendaciones	Precisión
Usuario 1	10	5	0,5
Usuario 2	10	4	0,4
Usuario 3	10	5	0,5
Usuario 4	10	3	0,3
Usuario 5	10	2	0,2
Promedio			0,38

Fuente: Elaboración Propia



CAPÍTULO 6

EVALUANDO LOS FACTORES

En el sexto capítulo se presenta la ejecución de los instrumentos de investigación. En primer lugar, se describen los datos recogidos a través de entrevistas con el fin de perfeccionar el conjunto de variables que se miden en la encuesta. A continuación, los datos obtenidos de los cuestionarios se presentan y estudian. Los análisis estadísticos de los datos de la encuesta se realizan a continuación. Por último, las limitaciones de las pruebas realizadas son presentadas y se presenta un resumen de las principales conclusiones.

6.1. Evaluando los factores de éxito planteados

Uso de grupos focales

6.1.1. Evaluación por Expertos

El panel de expertos puede definirse como un grupo de especialistas independientes y reputados en al menos uno de los campos concernidos por el sistema que se va a evaluar, al que se reúne para que emita un juicio colectivo y consensuado sobre dicho programa. Según se les solicite, el juicio emitido puede hacer referencia a la puesta en práctica o a los efectos del conjunto o de una parte del sistema.

6.1.2. Perfil de los Expertos

Profesionales con conocimiento en:

- Sistemas de información
- Sistemas de recomendación
- Comercio electrónico y desarrollo de aplicaciones web
- Ciencias de la computación

Experiencia no menor de 3 años en desarrollo de sistemas de información.

6.1.3. Procedimiento de Evaluación

Para la tesis propuesta se hará uso de dos encuestas. Primero, se encuestara a los expertos acerca de los factores para el desarrollo de SR, es importante resaltar que en el desarrollo del prototipo no se integraron todos los factores desarrollados en esta tesis, por esta razón durante la encuesta solo se consideraran los factores desarrollado en el prototipo. La segunda encuesta será la evaluación del prototipo de sistema de recomendación.

CUESTIONARIO 1

EVALUACIÓN DE LOS FACTORES PARA EL DESARROLLO DE UN SISTEMA DE RECOMENDACIÓN

- Los expertos recibirán una breve explicación acerca de los factores que fueron considerados determinantes en el desarrollo de sistemas de recomendación.
- Los evaluadores recibirán un cuestionario con preguntas cerradas.
- A los evaluadores se les dará un tiempo no mayor a 5 horas para la evaluación.
- La evaluación se realizara en 1 sesiones.
- Por último se interpretaran los cuestionarios resueltos por los evaluadores.

La siguiente tabla resume los factores que fueron considerados durante la encuesta y el número de pregunta a la que están relacionados.

Tabla 31. Tabla de factores para el desarrollo de un SR que fueron evaluados

FACTORES DESARROLLADOS EN EL PROTOTIPO	Nro. de pregunta
Tipo de metodología de desarrollo	Pregunta 1
Dominio de la recomendación	Pregunta 2
Propósito de la recomendación	Pregunta 3
Contexto de la recomendación	Pregunta 4
Grado de personalización	Pregunta 5
Tipo de recomendación	Pregunta 6
Métodos de obtención de preferencias del usuario	Pregunta 7
Proceso de recomendación	Pregunta 9
Presentación de la recomendación	Pregunta 10
Cobertura	Pregunta 11
Precisión	Pregunta 11

Fuente: Elaboración Propia

INTERPRETACIÓN DE LA EVALUACIÓN DE EXPERTOS

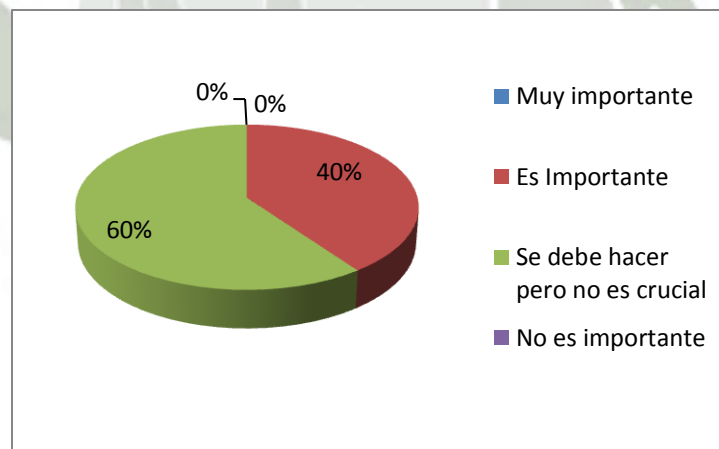
1. ¿Cree usted que la selección de tipo de metodología podría ser un factor de éxito para el desarrollo de un sistema de recomendación?

Tabla 32. Respuestas a la pregunta que busca establecer la importancia de definir el modelo de negocio

Respuesta	Frecuencia	Porcentaje
Muy importante	0	0%
Es Importante	2	40%
Se debe hacer pero no es crucial	3	60%
No es importante	0	0%
Total	5	100%

Fuente: Elaboración Propia

De los 5 expertos encuestados, 2 expertos que representan el 40% les pareció que es importante la selección de la metodología para llevar a cabo el sistema, 3 expertos que representan el 60% cree que la selección de la metodología se debe hacer pero no es crucial.



Fuente: Elaboración propia

Figura 32. Gráfico de respuesta acerca de la importancia de la selección del tipo de metodología de desarrollo

2. ¿Cree usted que es importante definir claramente el dominio de la recomendación?

Tabla 33. Respuesta a la pregunta acerca del dominio de la recomendación

Respuesta	Frecuencia	Porcentaje
Si	5	100%
No	0	0%
Total	5	100%

Fuente: elaboración propia

De los 5 expertos encuestados, 5 expertos que representan el 100% cree que definir el dominio de la recomendación es importante, 0 expertos creen que definir el dominio de la recomendación no es importante.

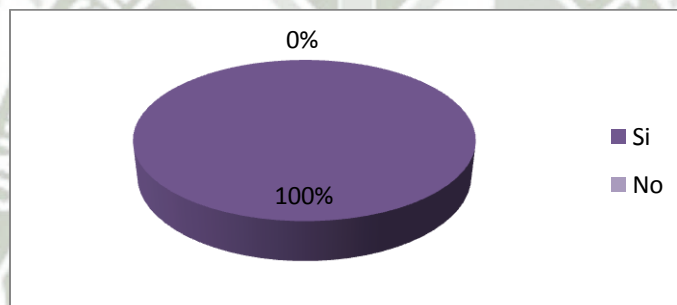


Figura 33. Gráfico de respuesta acerca del dominio de la recomendación

Fuente: Elaboración propia

3. ¿Cree usted que es importante definir claramente el propósito de la recomendación?

Tabla 34. Respuesta acerca el propósito de la recomendación

Respuesta	Frecuencia	Porcentaje
Si	3	60%
No	2	40%
Total	5	100%

Fuente: Elaboración propia

De los 5 expertos encuestados, 3 expertos que representan el 60% creen que el propósito de la recomendación es importante, 2 expertos que representan el 40% creen que definir claramente el propósito de la recomendación no es importante.

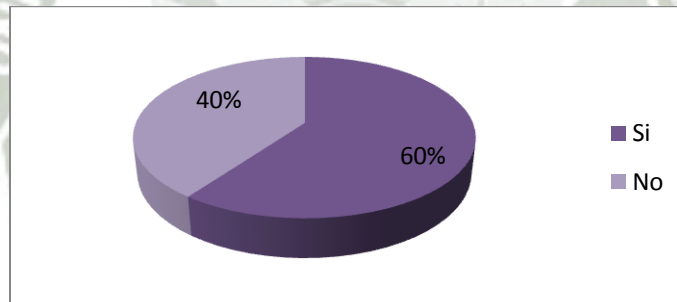


Figura 34. Gráfico de respuesta acerca del propósito de la recomendación

Fuente: Elaboración propia

4. ¿Cree usted que es importante definir claramente el contexto de la recomendación?

Tabla 35. Respuesta a la pregunta acerca del contexto de la recomendación

Respuesta	Frecuencia	Porcentaje
Si	5	100%
No	0	0%
Total	5	100%

Fuente: Elaboración propia

De los 5 expertos encuestados, 5 expertos que representan el 100% cree que definir el contexto de la recomendación es importante, 0 expertos creen que definir el contexto de la recomendación no es importante.

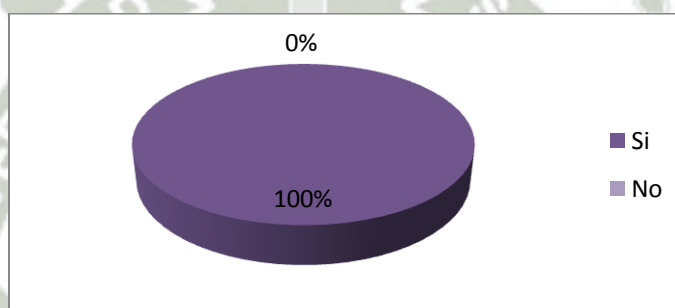


Figura 35. Gráfico de respuesta acerca del contexto de la recomendación

Fuente: Elaboración propia

5. ¿Cree usted que es importante definir el grado de personalización de la recomendación?

Tabla 36. Respuesta a la pregunta acerca del grado de personalización de la recomendación

Respuesta	Frecuencia	Porcentaje
Si	3	60%
No	2	40%
Total	5	100%

Fuente: Elaboración propia

De los 5 expertos encuestados, 3 expertos que representan el 60% creen que la personalización de la recomendación es importante, 2 expertos que representan el 40% creen que definir la personalización no es importante.

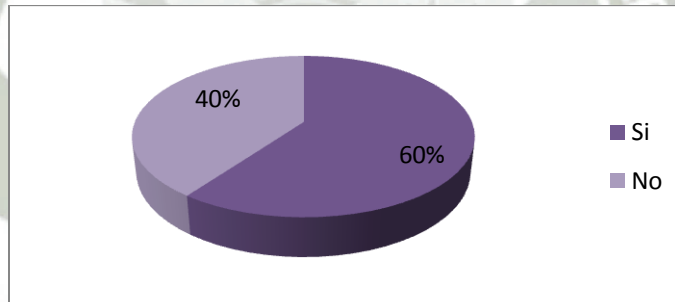


Figura 36. Gráfico de respuesta acerca del propósito de la recomendación

Fuente: Elaboración propia

6. ¿Cree usted que se debe integrar más de un tipo de recomendación en la realización de un SR?

Tabla 37. Respuesta a la pregunta acerca de la integración de más de un tipo de recomendación en SR

Respuesta	Frecuencia	Porcentaje
Si	5	100%
No	0	0%
Total	5	100%

Fuente: Elaboración propia

De los 5 expertos encuestados, 5 expertos que representan el 100% creen que se debe integrar más de un tipo de recomendación en la realización de un SR, 0 expertos creen que no debe haber integración de más de 1 tipo de recomendación.

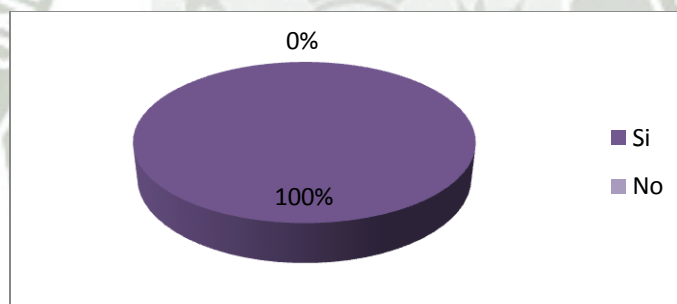


Figura 37. Gráfico de respuesta acerca de las métricas comerciales

Fuente: Elaboración propia.

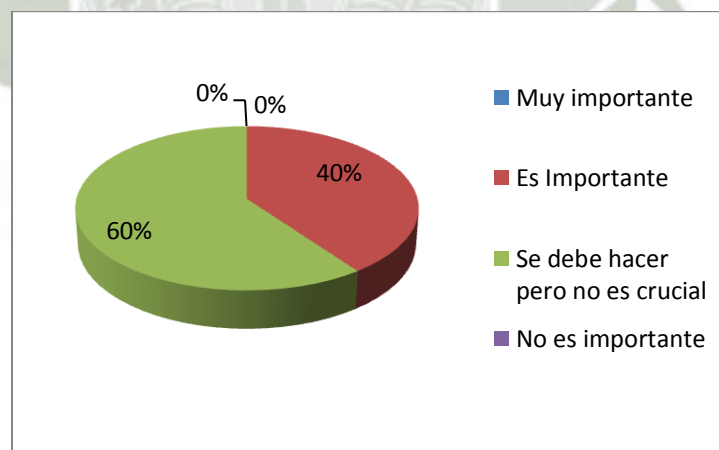
7. ¿Cree usted que es importante definir el tipo de recomendación que se va a usar, antes de empezar el desarrollo del SR?

Tabla 38. Respuestas a la pregunta cree usted que es importante definir el tipo de recomendación antes de empezar el desarrollo del SR

Respuesta	Frecuencia	Porcentaje
Muy importante	3	60%
Es Importante	2	40%
Se debe hacer pero no es crucial	0	0%
No es importante	0	0%
Total	5	100%

Fuente: Elaboración Propia

De los 5 expertos encuestados, 3 expertos que representan el 60% les pareció que es importante llevar a cabo la selección del tipo de recomendación antes de empezar el desarrollo del sistema, 2 expertos que representan el 40% cree que la selección del método de recomendación es importante antes de llevar a cabo el desarrollo de sistema.



Fuente: Elaboración propia

Figura 38. Gráfico de respuesta acerca de la importancia de la selección del tipo de metodología de desarrollo

8. ¿Cree usted que es importante implementar métodos de obtención de preferencias de usuario para poder establecer el perfil del usuario?

Tabla 39. Respuesta a la pregunta acerca de los métodos de obtención de preferencias de usuario

Respuesta	Frecuencia	Porcentaje
Si	4	100%
No	1	0%
Total	5	100%

Fuente: Elaboración Propia

De los 5 expertos encuestados, 4 expertos que representan el 80% creen que implementar métodos de obtención de preferencias de usuarios es importante para poder establecer el perfil de usuarios, 1 experto que corresponde al 20% cree que no es importante.

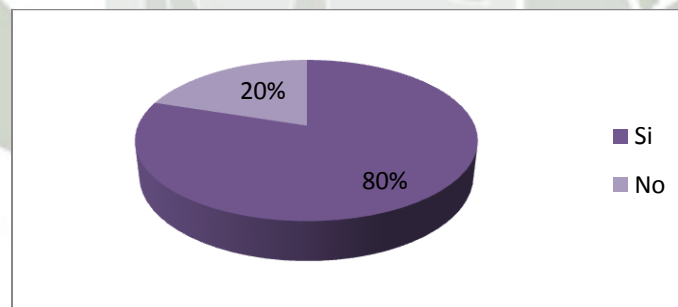


Figura 39. Gráfico de la respuesta acerca de los métodos de obtención de preferencias de usuario

Fuente: Elaboración Propia

9. ¿Cree usted que es importante definir el proceso de recomendación antes de empezar con el desarrollo del sistema?

Tabla 40. Respuesta a la pregunta acerca de la importancia de definir el proceso de recomendación

Respuesta	Frecuencia	Porcentaje
Si	5	100%
No	0	0%
Total	5	100%

Fuente: Elaboración propia.

De los 5 expertos encuestados, 5 expertos que representan el 100% creen que es importante definir el proceso de recomendación antes de empezar con el desarrollo del sistema, 0 expertos creen que no es importante.

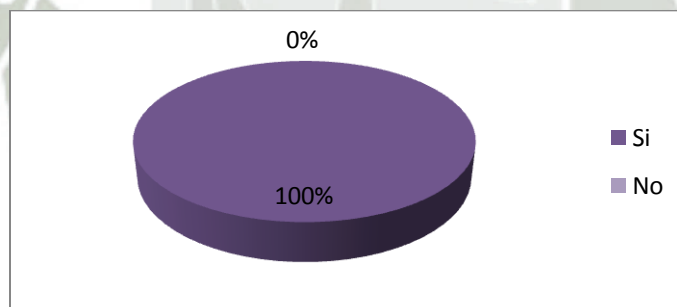


Figura 40. Gráfico de respuesta acerca de la importancia de definir el proceso de recomendación

Fuente: Elaboración propia.

10. ¿Cree usted que la presentación adecuada de la recomendación influye en la percepción del usuario?

Tabla 41. Respuesta a la pregunta acerca de si la presentación de la recomendación influye en la percepción que el usuario tiene de la recomendación

Respuesta	Frecuencia	Porcentaje
Si	5	100%
No	0	0%
Total	5	100%

Fuente: Elaboración propia

De los 5 expertos encuestados, 5 expertos que representan el 100% creen que la presentación de la recomendación si influye en cómo percibe el usuario la recomendación, 0 expertos creen que no es importante.

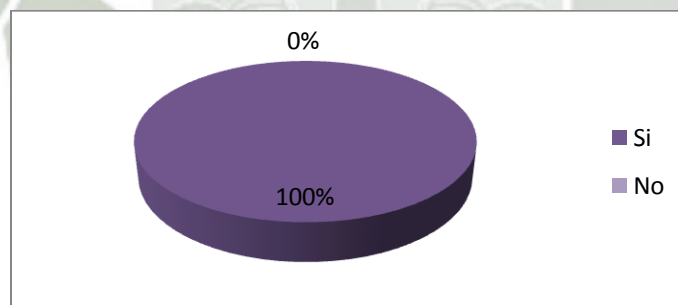


Figura 41. Gráfico de respuesta de si la presentación de la recomendación influye en la percepción que el usuario tiene de la recomendación

Fuente: Elaboración propia

11. ¿Cuál de los siguientes factores para la evaluación de SR le parece el más adecuado?

Tabla 42. Respuestas a la pregunta acerca de ¿cuál de los factores de evaluación le parece más adecuado?

Respuesta	Frecuencia	Porcentaje
Exactitud de rango	0	0%
Cobertura	1	20%
Diversidad	2	40%
Precisión	2	40%
Total	5	100%

Fuente: Elaboración propia

De los 5 expertos encuestados, 1 experto que representa el 20% cree que el factor más importante para hacer la evaluación de la recomendación es la cobertura, 2 expertos que representan el 40% creen que el factor más importante es la diversidad, 2 expertos que representan el 40% creen que el factor más importante es la precisión.

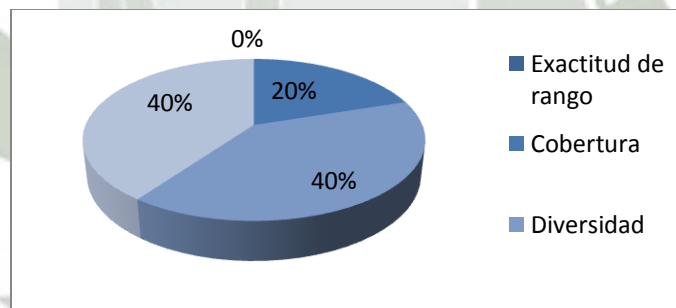


Figura 42. Gráfico de la respuesta ¿Cuál de los siguientes factores para la evaluación de SR le parece el más adecuado?

Fuente: Elaboración propia

12. ¿Considera que el prototipo diseñado integra los siguientes factores de manera correcta?

Tabla 43. Tabla de factores que fueron integrados en el prototipo

FASES	DESARROLLO DEL PROTOTIPO	FACTORES DESARROLLADOS EN EL PROTOTIPO
Definición de la metodología	Metodología de desarrollo	Tipo de metodología de desarrollo
Planteamiento	Definición del modelo de negocio para las recomendaciones	Dominio de la recomendación, Propósito de la recomendación, Contexto de la recomendación, Grado de personalización
	Definición del modelo híbrido de recomendación	Tipo de recomendación
Análisis	Analizar de los datos	Métodos de obtención de preferencias del usuario
Diseño y Codificación	Diseñar el motor de recomendación	Proceso de recomendación
	Diseñar la presentación de la recomendación	Presentación de la recomendación
Pruebas	Evaluar la calidad de recomendación	Cobertura, Precisión

Fuente: Elaboración propia

Tabla 44. Respuesta a la pregunta de integración de los factores con el prototipo

Respuesta	Frecuencia	Porcentaje
Si	4	80%
No	1	20%
Total	5	100%

Fuente: Elaboración propia

De los 5 expertos encuestados, 4 expertos que representan el 80% consideran que el prototipo diseñado integra los factores presentados de manera correcta, y 1 experto que representa el 20% no creen que el prototipo presentado integra los factores presentados de manera correcta .

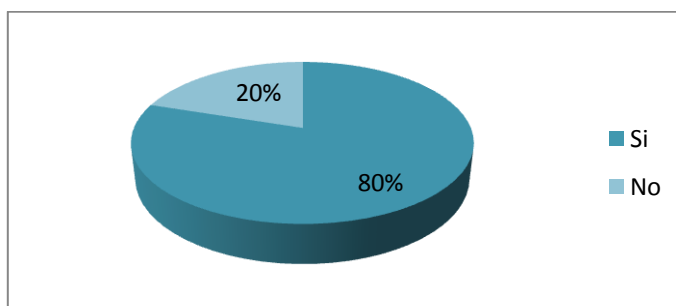


Figura 35. Gráfico de repuestas acerca de integración de los factores con el prototipo

Fuente: Elaboración propia

13. ¿Considera que los factores planteados para el desarrollo de sistemas de recomendación, serán de ayuda en el futuro desarrollo de un SR?

Tabla 45. Respuestas a la pregunta acerca del impacto de los 5 factores presentados en el futuro desarrollo de un SR

Respuesta	Frecuencia	Porcentaje
No ayudan	0	0%
Sirven de referencia	2	40%
Son necesarios	3	60%
Total	5	100%

Fuente: Elaboración Propia

De los 5 expertos encuestados, 3 expertos que representan el 60% creen que los factores planteados para el desarrollo de sistemas de recomendación, son necesarios en el futuro desarrollo de un SR y 2 expertos que representan el 40% creen que los factores planteados pueden ser usados como referencia en un futuro desarrollo.

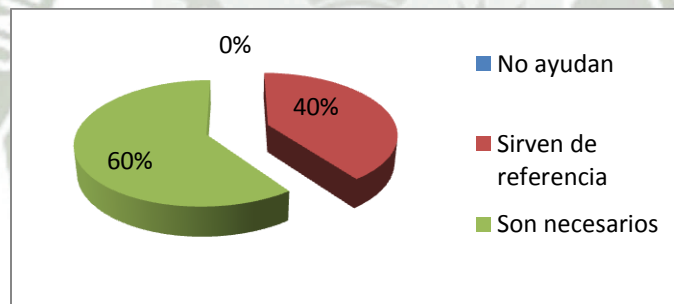


Figura 44. Respuestas acerca del impacto de los factores presentados en el futuro desarrollo de SR

Fuente: Elaboración propia

CUESTIONARIO 2

Evaluación el prototipo de sistema de recomendación

- Los expertos recibirán el manual de uso del prototipo del sistema de recomendación.
- Los evaluadores recibirán un cuestionario con preguntas cerradas.
- A los evaluadores se les dará un tiempo no mayor a 3 horas para la evaluación.
- La evaluación se realizara en 1 sesiones.
- Por último se interpretaran los cuestionarios resueltos por los evaluadores.



INTERPRETACIÓN DE LA EVALUACIÓN DE EXPERTOS

1. ¿Cree usted que es importante definir claramente el modelo de recomendación para llevar a cabo el desarrollo del prototipo?

Tabla 46. Respuesta a la pregunta que busca establecer la importancia de definir el modelo de negocio

Respuesta	Frecuencia	Porcentaje
Muy importante	2	40%
Es Importante	2	40%
No es crucial	1	20%
No es importante	0	0%
Total	5	100%

Fuente: Elaboración propia

De los 5 expertos encuestados, 2 expertos que representan el 40% les pareció que es muy importante definir claramente el modelo de negocio para llevar a cabo el sistema, 2 expertos que representan el 40% respondieron que les parece importante definir claramente el modelo de negocio y 1 experto que representa el 20% indica definir el modelo de negocio para llevar a cabo el sistema de recomendación se debe hacer pero no es crucial.

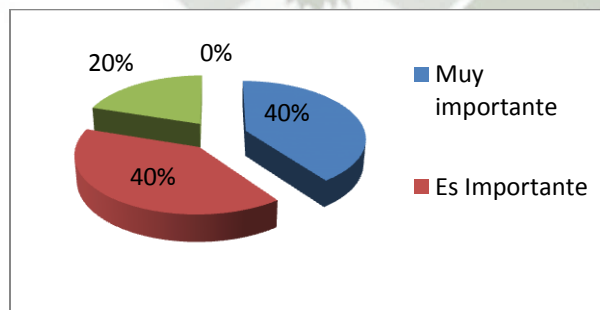


Figura 45. Gráfico de respuestas que busca establecer la importancia de definir el modelo de negocio

Fuente: Elaboración propia

2. ¿Cómo califica el diseño del prototipo?

Tabla.47 Respuestas a la pregunta calificación de prototipo

Respuesta	Frecuencia	Porcentaje
Muy bueno	1	20%
Bueno	2	40%
Regular	2	40%
Malo	0	0%
Total	5	100%

Fuente: Elaboración propia

De los 5 expertos encuestados, 1 experto que representan el 20% cree que el prototipo diseñado es muy bueno, 2 expertos que representan el 40% creen que el prototipo diseñado es bueno, y 2 expertos que representan el 40% creen que es regular

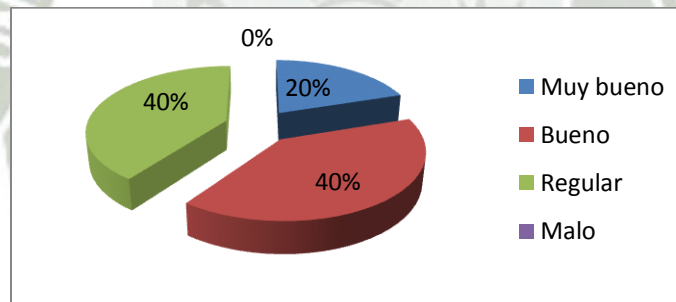


Figura 46. Respuestas a la pregunta calificación de prototipo

Fuente: Elaboración propia

3. ¿Considera que la recomendación ofrecida es correcta?

Tabla 48. Respuesta a la pregunta acerca de la recomendación ofrecida por el sistema

Respuesta	Frecuencia	Porcentaje
Muy bueno	1	20%
Bueno	3	60%
Regular	1	20%
Malo	0	0%
Total	5	100%

Fuente: Elaboración propia

De los 5 expertos encuestados, 1 experto que representan el 20% cree que la recomendación ofrecida es muy buena, a 3 expertos que representan el 60% creen que la recomendación ofrecida es buena, mientras que a 1 experto que representan el 20% respondió que las recomendaciones ofrecidas por el sistema son regulares.

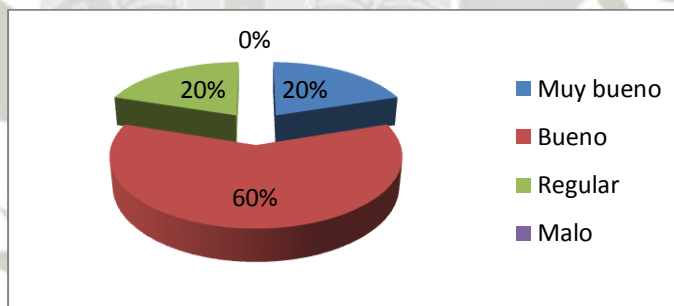


Figura 47. Gráfico de respuesta acerca de la recomendación ofrecida por el sistema

Fuente: Elaboración Propia

4. ¿Considera que las características del prototipo desarrollado son adecuadas?

Tabla 49. Respuestas a la pregunta acerca de las características del prototipo

Respuesta	Frecuencia	Porcentaje
Muy bueno	1	20%
Bueno	3	60%
Regular	1	20%
Malo	0	0%
Total	5	100%

Fuente: Elaboración Propia

De los 5 expertos encuestados, 1 experto que representan el 20% consideran que las características del prototipo desarrollado son muy buenas, a 3 expertos que representan el 60% creen que las características presentadas son buenas, mientras que a 1 experto que representan el 20% cree que las características del prototipo son regulares.

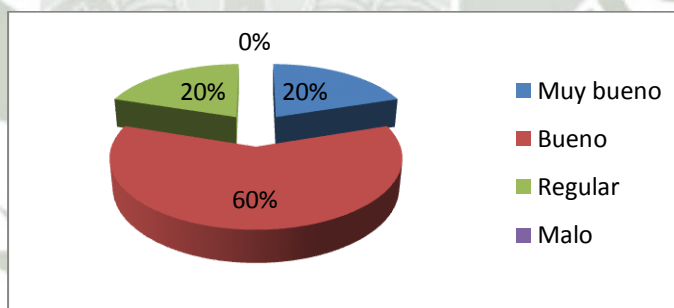


Figura 48. Respuestas acerca de las características del prototipo

Fuente: Elaboración Propia

5. ¿Considera correcto el análisis y diseño de las funcionalidades del prototipo?

Tabla 50. Respuestas a la pregunta acerca del análisis y diseño de las funcionalidades del prototipo

Respuesta	Frecuencia	Porcentaje
Si	4	80%
No	1	20%
Total	5	100%

Fuente: elaboración propia

De los 5 expertos encuestados, 4 expertos que representan el 80% consideran correcto el análisis y diseño de las funcionalidades del prototipo, y 1 experto que representa el 20% cree que el prototipo no clarifica en su totalidad las funcionalidades que debería tener un SR.

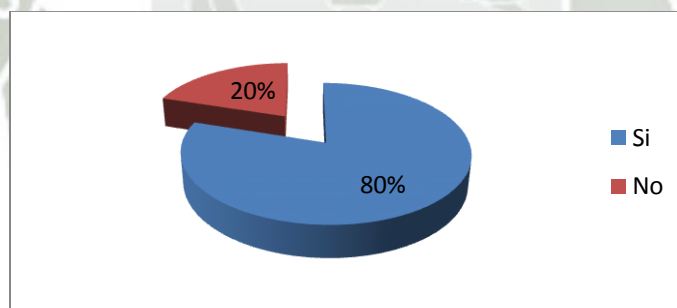


Figura 49. Gráfico de repuestas acerca del análisis y diseño de las funcionalidades del prototipo.

Fuente: Elaboración propia

6. ¿Cree usted que la evaluación del prototipo presentado es adecuada?

Tabla 51. Respuestas a la pregunta acerca de la importancia de la evaluación de la calidad de la recomendación

Respuesta	Frecuencia	Porcentaje
Si	5	100%
No	0	0%
Total	5	100%

Fuente: Elaboración propia

De los 5 expertos encuestados, 5 expertos que representan el 100% indican que la evaluación de la recomendación es adecuada.

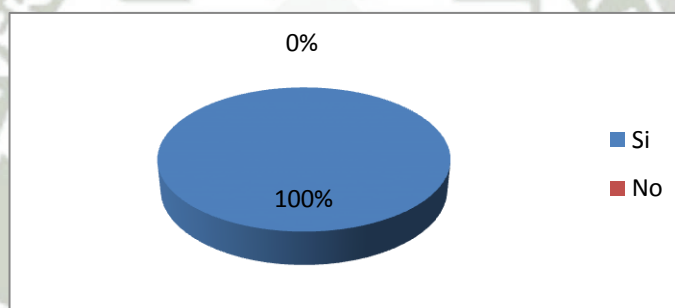


Figura 50. Gráfico de repuestas acerca de la evaluación presentada en el prototipo

Fuente: Elaboración propia

CONCLUSIONES

PRIMERA.- Se ha logrado demostrar que mediante la determinación adecuada de factores para el desarrollo de un sistema de recomendación, este logre brindar una buena recomendación. La recomendación ofrecida servirá como orientación para que el usuario tome una mejor decisión.

SEGUNDA.- Los factores para el desarrollo de un sistema de recomendación determinados ayudan a tener un marco de trabajo adecuado para el desarrollo de un sistema de recomendación, y mantienen la alineación del sistema de recomendación, con el negocio del que son parte.

TERCERA.- Para el análisis de estos factores se llevó a cabo un prototipo, donde se aplicaron la mayor parte de los factores mencionados. El resultado fue un prototipo funcional que cumplía con los requerimientos establecidos y ofrece recomendaciones coherentes.

CUARTA.- La evaluación de las recomendaciones ofrecidas demostró que las recomendaciones eran efectivas. De esta forma, quedo demostrado que la aplicación de los factores para el desarrollo de sistemas de recomendación, que fueron mencionados en este trabajo, son útiles en la implementación de sistema de recomendación.

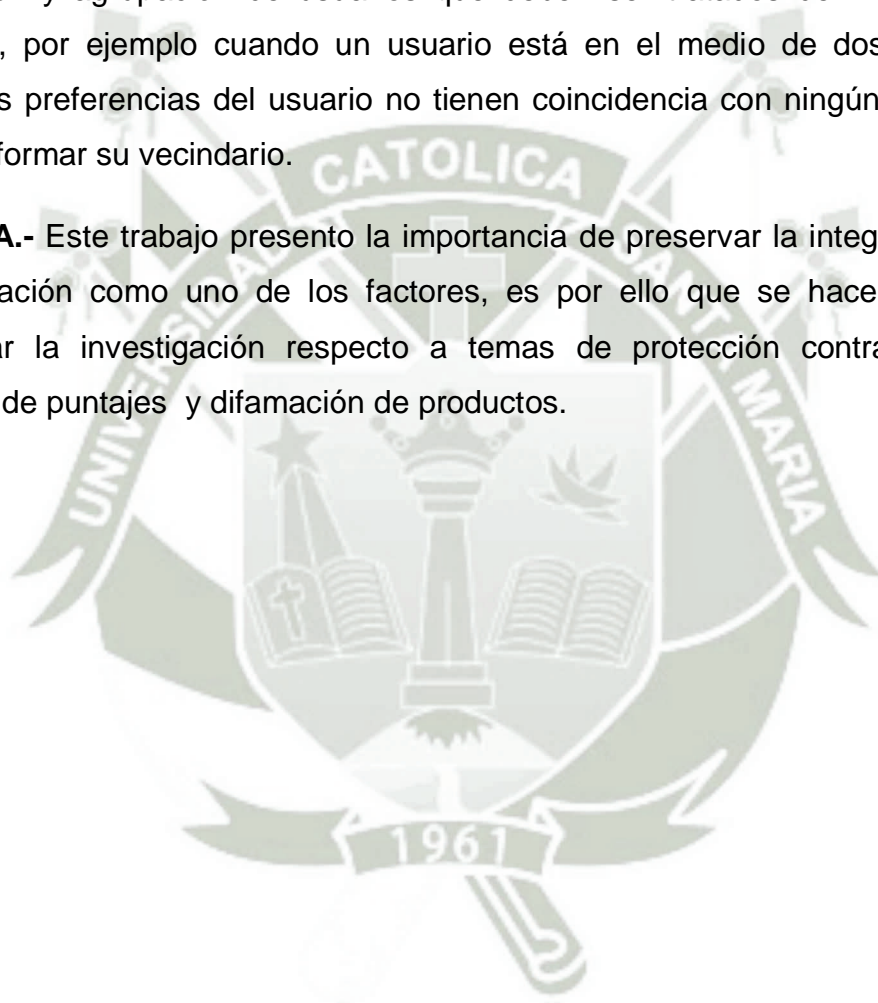
QUINTA.- Durante la realización del presente trabajo se logró exponer los diferentes problemas que tienen los sistemas de recomendación, y se pudo constatar algunos de los problemas más comunes durante las pruebas del prototipo.

RECOMENDACIONES

Luego de haber realizado este estudio, se prevé la necesidad de extender las siguientes áreas de estudio no han sido agotados en este trabajo.

PRIMERA.- Para conseguir que el motor de recomendación tenga un mejor desempeño, es necesario ahondar más en el estudio de los algoritmos que soportan las técnicas de recomendación. Existen ciertos problemas durante la clasificación y agrupación de usuarios que deben ser tratados de manera más específica, por ejemplo cuando un usuario está en el medio de dos grupos o cuando las preferencias del usuario no tienen coincidencia con ningún otro, y no se puede formar su vecindario.

SEGUNDA.- Este trabajo presento la importancia de preservar la integridad de la recomendación como uno de los factores, es por ello que se hace necesario profundizar la investigación respecto a temas de protección contra ataques, alteración de puntajes y difamación de productos.



ANEXO A

DATOS DE LOS EXPERTOS CON QUIENES SE REALIZARON LAS ENCUESTAS

Experto: Ing. David Sarmiento Patrón

Experiencia: Ingeniero de Sistema con 7 años de experiencia en desarrollo de software

Móvil: 959234828

CIP: 168910

Experto: Ing. Edgard Frank Lizárraga Ugarte

Experiencia: Ingeniero de Sistema con 7 años de experiencia en desarrollo de software

Móvil: 959132542

CIP: 134101

Experto: Ing. Klein Wilfredo Pinto Arcaya

Experiencia: Ingeniero de Sistema con 7 años de experiencia en desarrollo de software

Móvil: 993065498

CIP: 160352

Experto: Ing. Carlos Rodríguez Palacios

Experiencia: Ingeniero de Sistema con 8 años de experiencia en administración de base de datos

Móvil: 959132542

Experto: Ing. Rosangela Carpio Ojeda

Experiencia: Ingeniero de Sistema con 6 años de experiencia en desarrollo de software

Móvil: 95913254

ANEXO B

CUESTIONARIO 1

Junto con saludarle, deseo invitarle a responder el presente cuestionario. Sus respuestas tienen por objeto recoger su importante opinión con la finalidad de evaluar y optimizar el desarrollo del presente proyecto.

Por favor, marque con una X su respuesta.

1. **¿Cree usted que la selección de tipo de metodología podría ser un factor de éxito para el desarrollo de un sistema de recomendación?**

 Muy importante
 Importante
 Se debe hacer pero no es crucial
 No es importante
2. **¿Cree usted que es importante definir claramente el dominio de la recomendación?**

 Si No
3. **¿Cree usted que es importante definir claramente el propósito de la recomendación?**

 Si No
4. **¿Cree usted que es importante definir claramente el contexto de la recomendación?**

 Si No

5. **¿Cree usted que es importante definir el grado de personalización de la recomendación?**

Si No

6. **¿Cree usted que se debe integrar más de un tipo de recomendación en la realización de un SR?**

Si No

7. **¿Cree usted que es importante definir el tipo de recomendación que se va a usar, antes de empezar el desarrollo del SR?**

Muy importante

Importante

Se debe hacer pero no es crucial

No es importante

8. **¿Cree usted que es importante implementar métodos de obtención de preferencias de usuario para poder establecer el perfil del usuario?**

Si No

9. **¿Cree usted que es importante definir el proceso de recomendación antes de empezar con el desarrollo del sistema?**

Si No

10. **¿Cree usted que la presentación adecuada de la recomendación influye en la percepción del usuario?**

Si No

11. ¿Cuál de los siguientes factores para la evaluación de SR le parece el más adecuado?

- Exactitud de rango
- Cobertura
- Diversidad
- Precisión

12. ¿Considera que el prototipo diseñado integra los siguientes factores de manera correcta?

- Si No

FASES	DESARROLLO DEL PROTOTIPO	FACTORES DESARROLLADOS EN EL PROTOTIPO
Definición de la metodología	Metodología de desarrollo	Tipo de metodología de desarrollo
Planteamiento	Definición del modelo de negocio para las recomendaciones	Dominio de la recomendación, Propósito de la recomendación, Contexto de la recomendación, Grado de personalización
	Definición del modelo híbrido de recomendación	Tipo de recomendación
Análisis	Analizar de los datos	Métodos de obtención de preferencias del usuario
Diseño y Codificación	Diseñar el motor de recomendación	Proceso de recomendación
	Diseñar la presentación de la recomendación	Presentación de la recomendación
Pruebas	Evaluar la calidad de recomendación	Cobertura, Precisión

13. ¿Considera que los factores planteados para el desarrollo de sistemas de recomendación, serán de ayuda en el futuro desarrollo de un SR?

- No ayudan
- Sirven de referencia
- Son necesarios



ANEXO C

CUESTIONARIO 2

Junto con saludarle, deseo invitarle a responder el presente cuestionario. Sus respuestas tienen por objeto recoger su importante opinión con la finalidad de evaluar y optimizar el desarrollo del presente proyecto.

Por favor, marque con una X su respuesta.

1. ¿Cree usted que es importante definir claramente el modelo de recomendación para llevar a cabo el desarrollo del prototipo?

- Muy importante
- Importante
- Se debe hacer pero no es crucial
- No es importante

2. ¿Cómo califica el diseño del prototipo?

- Muy bueno
- Bueno
- Regular
- Malo

3. ¿Considera que la recomendación ofrecida es correcta?

- Muy bueno
- Bueno
- Regular
- Malo

4. **¿Considera que las características del prototipo desarrollado son adecuadas?**

Muy bueno

Bueno

Regular

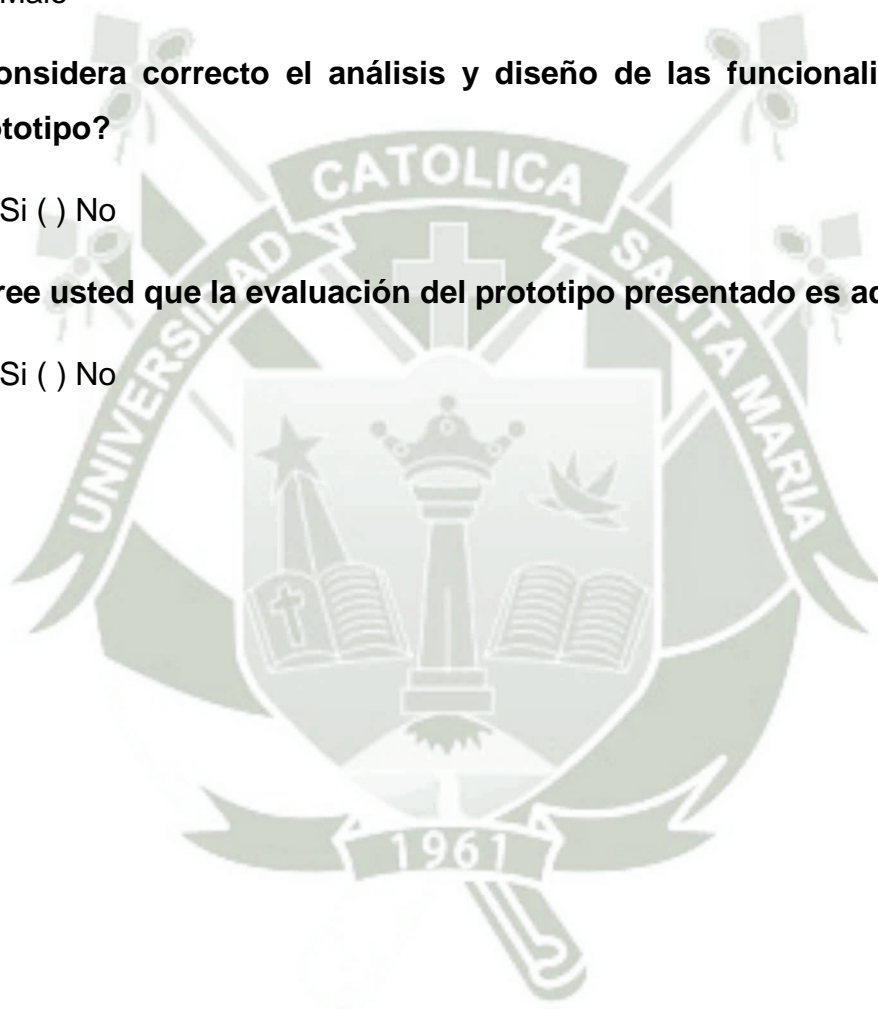
Malo

5. **¿Considera correcto el análisis y diseño de las funcionalidades del prototipo?**

Si No

6. **¿Cree usted que la evaluación del prototipo presentado es adecuada?**

Si No



BIBLIOGRAFÍA

- ADOMAVICIUS, G., HUANG, Z. & TUZHILIN, A. 2008. Personalization and Recommender Systems. *State-of-the-Art Decision-Making Tools in the Information-Intensive Age*.
- APPLEGATE, L., AUSTIN, R. & SOULE, D. 2009. *Corporate Information Strategy and Management: Text and Cases*, McGraw-Hill, Inc.
- ARAQUE, D. M. 2012. *Especificación de un Sistema de Recomendación basado en Etiquetado Social*. UNIVERSIDAD DE HUELVA.
- BÁEZ, L. C., OROZCO, R. P. & LEDESMA, E. R. 2013. Los Sistemas de Recomendación en la Toma de Decisiones. *SISTEMAS, CIBERNÉTICA E INFORMÁTICA VOLUMEN 10 - NÚMERO 2*.
- CARROL, J. 2001. Beyond Recommender Systems: Helping People Help Each Other. In: ADDISON-WESLEY (ed.) *HCI In The New Millennium*.
- CASINELLI, P. 2014. *Evaluating and Implementing Recommender Systems As Web Services Using Apache Mahout*. Boston College.
- CASSIDY, A. 1998. *A Practical Guide to Information Systems Strategic Planning*, CRC Press, Inc.
- CERTIFIED, M. G. 2014. *Top 12 Software Development Methodologies and Its Advantages/Disadvantages* [Online]. Available: <http://www.tatvasoft.com/blog/top-12-software-development-methodologies-and-its-advantages-disadvantages> [Accessed 18 de Junio del 2015 2014].
- CICILIA, C. B. 2009. *MONOGRAFIA SOBRE LA METODOLOGIA DE DESARROLLO DE SOFTWARE, RATIONAL UNIFIED PROCESS (RUP)*. INGENIERO EN CIENCIAS DE LA COMPUTACIÓN, UNIVERSIDAD DON BOSCO.
- CORDERO, J. A. 2012. *METODOLOGÍA PARA ADMINISTRAR PROYECTOS DE TECNOLOGÍA BASADOS EN ARQUITECTURA ORIENTADA A SERVICIOS*. TITULO DE MASTER EN ADMINISTRACIÓN DE PROYECTOS, UNIVERSIDAD PARA LA COOPERACION INTERNACIONAL.
- COVER, T. & HART, P. 2006. Nearest neighbor pattern classification. *IEEE Trans. Inf. Theor.*, 13, 21-27.
- EKSTRAND, M. D., HARPER, F. M., WILLEMSSEN, M. C. & KONSTAN, J. A. 2014. User perception of differences in recommender algorithms. *Proceedings of the 8th ACM Conference on Recommender systems*. Foster City, Silicon Valley, California, USA: ACM.
- GALLARDO, J. C. 2012. *Un modelo ponderado para sistemas de recomendacion basados en contenido con medidas de contigencia y entropía*. Universidad de Jaen.
- GOOD, N., SCHAFFER, J. B., KONSTAN, J. A., BORCHERS, A., SARWAR, B., HERLOCKER, J. & RIEDL, J. 1999. Combining collaborative filtering with personal agents for better recommendations. *Proceedings of the sixteenth national conference on Artificial intelligence and the eleventh Innovative applications of artificial intelligence conference innovative applications of artificial intelligence*. Orlando, Florida, USA: American Association for Artificial Intelligence.
- HERLOCKER, J. L., KONSTAN, J. A. & RIEDL, J. 2000. Explaining collaborative filtering recommendations. *Proceedings of the 2000 ACM conference on Computer supported cooperative work*. Philadelphia, Pennsylvania, USA: ACM.
- INÈS JOMAA, EMILIE POIRSON, CUNHA, C. D. & PETIOT, J.-F. 2011. A methodology for design a recommender system based on customer preferences. *INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED11*. TECHNICAL UNIVERSITY OF DENMAR.
- INTECO 2009. *INGENIERÍA DEL SOFTWARE: METODOLOGÍAS Y CICLOS DE VIDA* España.
- JANNACH, D., ZANKER, M., FELFERNIG, A. & FRIEDRICH, G. 2010. *Recommender Systems: An Introduction*, Cambridge University Press.

- JIANG, M., CUI, P., WANG, F., YANG, Q., ZHU, W. & YANG, S. 2012. Social recommendation across multiple relational domains. *Proceedings of the 21st ACM international conference on Information and knowledge management*. Maui, Hawaii, USA: ACM.
- JOSE C. CORTIZO, CARRERO, F. M. & MONSALVE, B. 2010. An Architecture for a General Purpose Multi-Algorithm Recommender System. *Workshop on the Practical Use of Recommender Systems*.
- KABORE, S. C. 2012. *Design and implementation of a recommender system as a module for Liferay portal*. Master in Information Technologies Barcelona School of Computing (FIB).
- KONSTAN, J. A. & RIEDL, J. 2012 *How Amazon and Netflix predict your preferences and prod you to purchase* [Online]. IEEE Spectrum. Available: <http://spectrum.ieee.org/computing/software/deconstructing-recommender-systems> [Accessed 29 de mayo del 2015].
- LARSON, E. W. 2014. *Project management: the managerial process*, North Ryde, N.S.W., McGraw-Hill education.
- LEMÉTAYER, J. 2010. *IDENTIFYING THE CRITICAL FACTORS IN SOFTWARE DEVELOPMENT METHODOLOGY FIT*. Degree of Master of Management Studies, Victoria University of Wellington.
- MARCO DE GEMMIS, LEO IAQUINTA, PASQUALE LOPS, CATALDO MUSTO, FEDELUCIO NARDUCCI & SEMERARO, G. 2009. *Preference Learning in Recommender Systems*. University of Bari "Aldo Moro".
- MUSIAL, K. 2009. *Recommender System for Online Social Network*. LAP Lambert Academic Publishing.
- NEUMANN, A. W. 2009. *Recommender Systems for Information Providers*, Germany, Physica-Verlag A Springer Company.
- OWEN, S., ANIL, R., DUNNING, T. & FRIEDMAN, E. 2011. *Mahout in Action*, Manning Publications Co.
- OWEN, S., ANIL, R., NG, T. D. & FRIEDMAN, E. 2012. *Mahout in Action*, United States of America, SHELTERISLAND.
- RAE 2001. *Diccionario de la lengua española* Madrid, España.
- RICCI, F., ROKACH, L. & SHAPIRA, B. 2011. *Recommender Systems Handbook*, London, Springer.
- ROBILLARD, M. P., MAALEJ, W., WALKER, R. J. & ZIMMERMANN, T. 2014. *RECOMMENDATION SYSTEMS IN SOFTWARE ENGINEERING*, Berlin Springer Berlin Heidelberg.
- RODRÍGUEZ, A. P. A. 2009. *Sistema de Recomendación Colaborativo Basado en Algoritmos de Filtrado Mejorados*. Universidad de Jaen.
- SCHAFER, J. B., KONSTAN, J. & RIEDL, J. 1999. Recommender systems in e-commerce. *Proceedings of the 1st ACM conference on Electronic commerce*. Denver, Colorado, USA: ACM.
- SEGUIDO, M. 2009. *Sistemas de Recomendacion para websde informacion sobre la salud*. Master en computacion, Universidad de Cataluña.
- SILLITTI, A., HAZZAN, O., BACHER, E. & ALBALADEJO, X. 2011. Agile Processes in Software Engineering and Extreme Programming. *12th International Conference, XP 2011*. España.
- SU, X. & KHOSHGOFTAAR, T. M. 2009. A survey of collaborative filtering techniques. *Adv. in Artif. Intell.*, 2009, 2-2.
- TORRES, E. J. C. 2007. *EVALUACIÓN DEL USO DE ALGORITMOS COLABORATIVOS PARA ORIENTAR ACADÉMICAMENTE AL ALUMNADO EN BACHILLERATO*. DIPLOMA DE ESTUDIOS AVANZADOS Universidad de Jaén.
- TSIHRINTZIS, G. A., VIRVOU, M. & JAIN, L. C. 2013. *Multimedia Services in Intelligent Environments Advances in Recommender Systems*, Australia, Springer.

- WU, W., HE, L. & YANG, J. 2012. Evaluating Recommender Systems
- YANG, X., GUO, Y., LIU, Y. & STECK, H. 2014. A survey of collaborative filtering based social recommender systems. *Computer Communications*, 41, 1-10.
- ZHONG, Y., DU, L. & YANG, J. 2013. Learning Social Relationship Strength via Matrix Co-Factorization with Multiple Kernels. *In: LIN, X., MANOLOPOULOS, Y., SRIVASTAVA, D. & HUANG, G. (eds.) Web Information Systems Engineering – WISE 2013*. Springer Berlin Heidelberg.

