

Representación y Razonamiento Basado en Conocimiento

# Sistema Recomendador de Libros

Dominio: Recomendación de Libros

Juan Broto Ortega y Lidia González Martín

## Contenido

Título del Proyecto: Sistema Recomendador de libros.....	<b>¡Error! Marcador no definido.</b>
1. Descripción del problema.....	2
2. Fuentes de conocimiento/datos analizadas .....	2
2.1. Fuentes evaluadas .....	2
2.2. Estructura del grafo de conocimiento .....	3
2.3. Construcción del grafo .....	4
3. Justificación del enfoque basado en conocimiento.....	4
3.1. ¿Qué pueden aportar las ontologías frente a un enfoque basado únicamente en datos?.....	5
3.2. Indica los enunciados de algunas preguntas o consultas que van a ser útiles y que permiten validar el conocimiento .....	5
4. Diseño del Sistema Recomendador.....	5
4.1. Explicabilidad .....	5
5. Dificultades encontradas.....	6
6. Aplicación Implementada.....	6
7. Conclusiones.....	8
8. Trabajo Futuro.....	8

## 1. Descripción del problema

Los usuarios buscan libros basándose en los géneros y autores de libros que han leído previamente y que les han gustado, pero la información acerca de los libros es muy amplia y dispersa, pudiendo agobiar a los usuarios que quieren una nueva lectura.

Este proyecto desarrolla un sistema recomendador enriquecido con ontologías que comprende relaciones semánticas de géneros. Permite razonar por similitud sobre los géneros. Así, el sistema puede recomendar obras similares a los usuarios basándose en sus estilos de escritura favoritos, teniendo en cuenta elementos tan relevantes como el autor del libro o la editorial.

Hemos identificado diversos problemas que justifican la necesidad de un recomendador de libros. En primer lugar, existe una sobrecarga informativa, los catálogos digitales contienen millones de títulos y las recomendaciones están basadas en la popularidad, sin tener en cuenta relaciones entre géneros.

## 2. Fuentes de conocimiento/datos analizadas

Hemos realizado el análisis de diversas fuentes de datos bibliográficos y ontologías existentes para construir un sistema de recomendación robusto y semánticamente rico. El objetivo era integrar fuentes que cubrieran dos necesidades: proporcionar estructura taxonómica completa y aportar instancias actualizadas de libros. Con respecto a la estructura taxonómica, se tuvieron en cuenta diversas fuentes:

- **Wikidata.** Empleamos wikidata como fuente principal para la extracción de la jerarquía de géneros literarios narrativos. Mediante consultas SPARQL, extrajimos la jerarquía completa de géneros narrativos partiendo de la entidad "**género narrativo**" ([Q1318295](#)). La relación "**subclase de**" ([P279](#)) permitió construir un árbol taxonómico con tres niveles de profundidad, capturando relaciones como "novela policiaca" → "ficción detectivesca" → "género narrativo".
- **Otras ontologías empleadas.** Con la finalidad de enriquecer la jerarquía obtenida de wikidata mapeamos otras ontologías externas, este es el caso de **BIBO** (Bibliographic Ontology), que aporta un estándar académico para metadatos bibliográficos, **Schema.ORG**, que aporta relaciones para las propiedades de autor, género e isbn, **Dublin Core** que aporta relaciones para las propiedades de creador, editorial y fecha de lanzamiento y **FOAF**, útil para las relaciones sociales, en este caso para autor.

### 2.1. Fuentes evaluadas

Para la obtención de libros se evaluaron varias fuentes de datos. En primer lugar, se probó **Google Books API** que con su url permite acceder de forma estructurada a múltiples datos bibliográficos, como títulos y autores, información editorial consistente, ISBN validados, fechas de publicación y descripciones y clasificaciones de edad. Google Books resultó ser una fuente muy completa con tiempo de respuesta estables.

Por otro lado, se analizó también **OpenLibrary** en la que se encontraron algunas limitaciones técnicas. Entre ellas la inestabilidad en la información de cada libro, con campos variables e información

frecuentemente incompleta, así como problemas recurrentes en los identificadores, en particular ISBN ausentes, lo mismo ocurría con la información de las editoriales.

El tiempo de las consultas también resultó un problema, la arquitectura de los end-points de OpenLibrary obliga a la realización de múltiples consultas para obtener toda la información necesaria y cada consulta tiene tiempos de respuesta prolongados. Estas limitaciones no invalidan el valor de OpenLibrary como repositorio amplio y abierto, pero sí limitan su uso, por ello decidimos no utilizarla para la creación.

**DBpedia** también fue objeto de evaluación como posible fuente adicional, mediante un método específico que aprovechaba su interconexión con Wikidata a través de relaciones \*owl:sameAs\*. Sin embargo, en el ámbito concreto de los libros, especialmente los contemporáneos, se detectó una cobertura irregular y la ausencia frecuente de metadatos esenciales, como ISBN o fechas de publicación precisas. Esta falta de completitud redujo su utilidad práctica para los objetivos planteados, aunque se mantiene como una fuente relevante especialmente por su relación con Wikidata.

Finalmente, se utilizó **Wikidata** para enriquecer la ontología. Aprovechando que la jerarquía se extrajo de la misma fuente y que se conservaron sus identificadores, Wikidata se convirtió en una fuente especialmente interesante para completar con más libros. Los datos se trajeron mediante consultas SPARQL, durante cuyo proceso surgieron diversos problemas. El primero de ellos fue la limitación en la frecuencia de las consultas, que resultó ser extremadamente baja, provocando que muchas de ellas no se procesaran debido a excepciones del servidor. A esto se sumó el elevado tiempo de latencia en la ejecución de las consultas, lo que prolongó considerablemente la generación del grafo de conocimiento. Otro de los problemas detectados fue la falta de ciertos campos, como Wikidata es un grafo de conocimiento de propósito general y no específico dentro del dominio de los libros, en algunos casos las descripciones, el ISBN, la editorial u otros campos específicos se retornaron vacíos.

## 2.2. Estructura del grafo de conocimiento

A la hora de diseñar el grafo se han definido las siguientes clases:

- **LibrosXXI:** Se trata de la clase a la que pertenecen los libros que se añaden al grafo. Es equivalente a BibliographicResource en Dublin Core (DC) y Book en Schema y BIBO. Además tienen una serie de propiedades:
  - Año de publicación. Esta propiedad indica el año en que el libro se publicó por primera vez. Con equivalencias a las clases date de DC, created de BIBO y datePublished de Schema.
  - Descripción. Una sinopsis o información adicional relativa al libro. Es equivalente a la clase description de Schema y BIBO.
  - Tiene formato Ebook. Este campo indica si el libro al que pertenece se encuentra en formato de Ebook. Tiene equivalencia con la clase bookFormat de Schema.
  - ISBN. Es el indicador único de cada libro publicado publicado a partir de 1970. Tiene equivalencia con la clase isbn de Schema y BIBO.
  - Apto para menores. Este campo indica si el libro es considerado apto para personas menores de edad. Es equivalente a la propiedad contentRating de Schema.
  - tieneAutor. Esta propiedad representa al autor de un libro, se encuentra en el dominio

- de LibroXXI y en el rango del autor. Tiene como clases equivalentes autor de Schema, creator de Dublin Core (DC) y de authorList de BIBO.
- tieneGenero. Esta propiedad representa al género al que pertenece un libro, se encuentra en el dominio del LibroXXI y en el rango del género. Tiene como clases equivalentes genre de Schema, y Subject de Dublin Core (DC).
  - tieneEditorial. Esta propiedad representa la propiedad a la que pertenece un libro, se encuentra en el rango de la clase editorial y en el dominio de LibroXXI, Tiene como propiedades equivalentes publisher de Schema, Bibo y Dublin Core (DC)
- **Autor:** Clase que representa al autor de cada libro. Un libro puede tener varios autores. Es subclase de la clase de Wikidata persona (Q5), y tiene equivalencias con las clases Person de Bibo, Schema y FOAF.
  - **Género:** Clase que representa un género. Subclase de la clase de Wikidata género artístico (Q1792379). Tiene como clases equivalentes genre de Schema y DocumentPart de BIBO.
  - **Editorial:** Clase que representa a la editorial. Tiene como clases equivalentes Publisher de Schema y de Bibo.
  - **Usuario:** Representa a los usuarios de nuestro sistema, equivalente a las clase de Schema person. Cuenta con dos propiedades:
    - Le Gusta: representa la relación entre el usuario y cada libro que les gusta. Tiene como rango la clase libro y se encuentra en el dominio del usuario.
    - Edad: Edad a la que pertenece el usuario.

### 2.3. Construcción del grafo

Cada fuente se procesó de forma independiente. En primer lugar, Wikidata proporcionó la estructura taxonómica fundamental, construyendo el esqueleto del grafo a partir de su jerarquía de 1238 géneros. Por su parte las ontologías externas (BIBO, Schema.org, Dublin Core, FOAF) establecieron puentes semánticos con estándares ampliamente adoptados para todas las clases y propiedades que componen la ontología.

Una vez conseguimos la base se comenzó el enriquecimiento de libros. Para ello diseñamos un algoritmo recorría cada género de manera independiente para cada una de las fuentes que se iban a integrar (ver Sección 2.1), por razones de optimización, el algoritmo contaba con un límite de 50 libros por fuente y género.

La integración se realizó mediante mapeos semánticos, estableciendo equivalencias entre propiedades y clases de diferentes vocabularios. Parte de las instancias de libros se obtuvieron de Google Books mediante consultas URL basadas en el label del género obtenido de Wikidata. No obstante, esta relación no resultó completamente estable, por lo que algunos géneros de Wikidata quedaron sin instancias asociadas. Todo ello sumado a la falta de recursos en algunas de las fuentes consultadas y la riqueza de la taxonomía generada gran parte de los géneros menos comerciales han quedado sin relacionarse con ningún libro, obteniendo finalmente una ontología con aproximadamente 2000 libros.

La obtención de libros en Wikidata se realizó mediante una consulta de SPARQL en base a la URI del género. De forma similar se realizó una consulta de SPARQL a DBpedia, aprovechando la relación entre los géneros de Wikidata y los de DBPedia basada en la propiedad OWL.sameAs.

## 3. Justificación del enfoque basado en conocimiento

### **3.1. ¿Qué pueden aportar las ontologías frente a un enfoque basado únicamente en datos?**

En un sistema de recomendación, las ontologías ayudan a superar las limitaciones de los enfoques que se basan solo en datos numéricos, porque añaden significado y reglas claras sobre cómo se relacionan los conceptos. Mientras que los modelos basados en datos se limitan a detectar patrones y coincidencias en los datos, las ontologías permiten describir de forma explícita las relaciones que explican los patrones.

Las ontologías permiten representar el conocimiento del dominio de manera estructurada. En el caso de los libros, hacen posible definir claramente la jerarquía de los géneros literarios y las relaciones entre autores, obras y géneros. Gracias a esto, el sistema puede razonar sobre las preferencias de un usuario. Por ejemplo, si una persona muestra interés por la “novela negra” y este género forma parte de la “ficción detectivesca”, el sistema puede inferir que también podrían resultarle atractivos otros géneros cercanos, como el thriller policíaco.

Otro aspecto importante es la integración de datos procedentes de distintas fuentes. Mediante relaciones semánticas estándar, es posible vincular información de orígenes diferentes, como Wikidata, Google Books o DBpedia, aunque usen estructuras o nombres distintos. Esto facilita combinar y reutilizar datos que, con procesos tradicionales, resultarían difíciles de unificar.

Además, las ontologías mejoran la explicabilidad de las recomendaciones. Al basarse en un modelo comprensible, el sistema puede justificar por qué sugiere un libro, por ejemplo, indicando que pertenece a un género relacionado con otros que el usuario ya ha leído o que comparte autor con obras previas. Esto contrasta con muchos modelos de aprendizaje automático, cuyos resultados son difíciles de interpretar.

### **3.2. Indica los enunciados de algunas preguntas o consultas que van a ser útiles y que permiten validar el conocimiento**

¿Qué libros son similares a X basándose en su género?

¿Teniendo en cuenta a los demás usuarios con gustos similares qué libros le podría gustar a un usuario A sabiendo que le gustan los libros 1, 2 y 3?

## **4. Diseño del Sistema Recomendador**

En esta sección presentamos las características más relevantes de la implementación de nuestro sistema recomendador.

### **4.1. Explicabilidad**

Como se menciona anteriormente, en este recomendador empleamos dos enfoques complementarios: Item-to-Item y filtrado colaborativo, ambos explicables y trazables.

El enfoque **Item-to-Item** se basa en la similitud entre libros, que se calcula a partir de la distancia semántica en la jerarquía de géneros. Además, se incorporan pesos adicionales por coincidencia de autor o editorial y un pequeño factor de aleatoriedad para aumentar la diversidad de las recomendaciones (serendipia). Esta aproximación permite que cada recomendación pueda ser

explicada de manera precisa, mostrando al usuario el camino semántico entre los libros, su proximidad y cualquier coincidencia de autor o editorial. De este modo, los usuarios pueden comprender por qué se sugiere un determinado libro, aumentando la transparencia y la confianza en el sistema.

Por su parte, el **filtrado colaborativo** aprovecha las preferencias de otros usuarios para identificar libros potencialmente atractivos. La similitud entre perfiles se calcula mediante la métrica de Jaccard, que considera únicamente los libros en los que existe superposición entre usuarios. Esto permite identificar qué elementos compartidos entre usuarios originan la recomendación, ofreciendo un nivel de explicabilidad sobre la influencia de otros perfiles en las sugerencias.

En conjunto, generan un modelo explicable, en el que las recomendaciones pueden justificarse tanto por relaciones semánticas entre libros como por patrones de gustos de usuarios afines. Esta explicabilidad es un elemento clave frente a modelos basados únicamente en datos o embeddings vectoriales, que carecen de interpretación directa y no permiten justificar la relación entre ítems y usuarios de manera comprensible.

## 5. Dificultades encontradas

Además de las dificultades mencionadas anteriormente relacionadas con las diferentes fuentes de datos analizadas (ver Sección 2.1) nos surgieron las siguientes dificultades durante la implementación del sistema:

- **Similitud Item-To-Item.** Al inicio nuestra taxonomía de géneros estaba estructurada de tal manera que los libros eran RDF.type de los géneros, pero esto resultó ser problemático ya que al navegar por los diferentes géneros de la jerarquía el algoritmo no conseguía llegar a los mismos. Por ello tuvimos que redefinir la estructura para que los libros pasasen a ser RDF.type de una clase denominada Librosxxi y establecer una propiedad con dominio Librosxxi y rango géneros llamada tieneGenero.
- **Filtrado colaborativo.** Los usuarios se relacionan con los libros a través de una propiedad llamada leGusta que tiene como dominio a los propios usuarios y de rango a la clase Librosxxi. Inicialmente calculábamos la similitud entre dos perfiles a través de la métrica de Jaccard. Sin embargo, ante un volumen de datos medio y perfiles con baja superposición, esta métrica no permitía obtener valores de similitud suficientemente elevados entre usuarios, lo que dificultaba la generación de recomendaciones. Por ello decidimos optar por una adaptación de la métrica en la que se ponderó el denominador de forma que la fórmula únicamente tuviera en cuenta el perfil con menor número de libros.
- **Falta de Serendipia.** Una de las dificultades que encontramos fue la falta de serendipia en las recomendaciones. Esta falta de diversidad vino originada por la forma en la que calculamos la similitud Item to Item. Para resolverlo se decidió agregar pesos de forma que se tuviera en cuenta autores o editoriales en común, dando más peso al autor (0.5) que a la editorial (0.2). Se agregó a su vez un factor de aleatoriedad que suma aleatoriamente un factor de hasta 0.1.

## 6. Aplicación Implementada

Todas las funcionalidades fueron integradas en una aplicación que carga la ontología completa y

realiza operaciones sobre ella. Dividida en dos secciones, la aplicación permite obtener recomendaciones de libros basadas en un único libro o en un perfil completo de usuario.

La recomendación basada en un único libro está basada en la recomendación Item-To-Item devolviendo al usuario una serie de recomendaciones junto con una justificación de la misma, indicando la similitud entre ambos libros con un score, con el path entre el libro recomendado y el inicial, la distancia entre ambos y si tiene editorial o autor en común.

The screenshot shows the main interface of the 'Sistema de Recomendación de Libros'. On the left, there's a sidebar with sections for 'Modo de búsqueda' (Search mode) and 'Información' (Information). Under 'Modo de búsqueda', it says 'Selecciona cómo quieras obtener recomendaciones:' with two options: 'Por libro específico' (radio button selected) and 'Por perfil de usuario'. Under 'Información', it says 'Este sistema te ayuda a encontrar libros que podrían interesarte.' followed by two bullet points: 'Por libro específico: Introduce un libro que te guste y obtén recomendaciones similares.' and 'Por perfil de usuario: Comparte tus preferencias para obtener recomendaciones personalizadas.' The main content area has a title 'Sistema de Recomendación de Libros' and a subtitle 'Descubre tu próximo libro favorito basado en tus preferencias'. Below this is a section titled 'Recomendaciones basadas en un libro' with a sub-section 'Introduce el título de un libro que te haya gustado:' containing the text 'amanecer'. A blue button labeled 'Buscar recomendaciones' is below the input field. Further down, another section titled 'Recomendaciones para: Amanecer' shows a result for 'Día de verano (Score: 1.091)' with details: 'Camino de género: Amanecer → philosophical poem → Día de verano', 'Distancia de género: 2', 'Bonus Autor: Coincide', and 'Bonus Editorial: Coincide'.

Existe la opción de una recomendación basada en un perfil de usuario, donde se indica el nombre, una edad y los libros que te han gustado. Este perfil será guardado en el grafo y utilizado en recomendaciones de futuros usuarios.

This screenshot shows the user profile setup section of the system. The sidebar remains the same. The main content area has a title 'Sistema de Recomendación de Libros' and a subtitle 'Descubre tu próximo libro favorito basado en tus preferencias'. Below this is a section titled 'Recomendaciones basadas en tu perfil' with a sub-section 'Tu nombre:' containing 'Juan\_Perez' and 'Tu edad:' containing '76'. A blue button labeled 'Obtener recomendaciones personalizadas' is at the bottom. Below this, there's a section titled 'Tus libros favoritos' with a sub-section 'Introduce algunos libros que hayas disfrutado (uno por línea):' containing 'Ej: El señor de los anillos Harry Potter y la piedra filosofal 109.4'. A scroll bar indicates more content is available.

En la recomendación por usuarios obtenemos una serie de libros cuya recomendación se basa en el filtro colaborativo explicado en apartados anteriores. En esta recomendación, de cada libro obtenemos la puntuación obtenida del Jaccard combinado junto con toda la información que tenemos del libro, autor, editorial, descripción, isbn, disponibilidad en ebook, restricción de edad, descripción y género.

The screenshot shows a user interface for a book recommendation system. On the left, there's a sidebar with sections for 'Modo de búsqueda' (Search mode), 'Información' (Information), and two options for getting recommendations: 'Por libro específico' (By specific book) and 'Por perfil de usuario' (By user profile). The main area displays personalized recommendations for 'Juan\_Perez'. It shows two books: 'All Families Are Psychotic' and 'Absent in the Spring', each with a star rating and a score of 0.6. Detailed information for each book is provided, including author, genre, year, publisher, ISBN, classification, EPUB accessibility, and description.

**Modo de búsqueda**

Selecciona cómo quieras obtener recomendaciones:

- Por libro específico
- Por perfil de usuario

**Información**

Este sistema te ayuda a encontrar libros que podrían interesarte.

- **Por libro específico:** Introduce un libro que te guste y obtén recomendaciones similares.
- **Por perfil de usuario:** Comparte tus preferencias para obtener recomendaciones personalizadas.

**Recomendaciones personalizadas para Juan\_Perez**

**Libros recomendados para ti:**

**All Families Are Psychotic** (★ Score: 0.6)

Autor: Douglas Coupland  
Género: novel  
Año: Año  
Editorial: Random House of Canada  
ISBN: 0-679-31140-8  
Clasificación: No especificado  
Accesibilidad EPUB: No  
Descripción: Sin descripción disponible

**Absent in the Spring** (★ Score: 0.6)

Autor: Agatha Christie  
Género: tragedy  
Año: Año  
Editorial: Farrar & Rinehart

## 7. Conclusiones

En este trabajo se ha desarrollado un sistema recomendador de libros basado en conocimiento, con el objetivo de reducir la sobrecarga informativa presente en los catálogos digitales actuales. El uso de ontologías ha permitido crear jerarquía de géneros literarios y las relaciones entre libros, autores y editoriales, posibilitando un razonamiento semántico que va más allá de las recomendaciones basadas en popularidad o coincidencias directas.

La integración de múltiples fuentes de datos, Wikidata, Google Books API, y DBpedia junto con ontologías estándar como BIBO, Schema.org, Dublin Core y FOAF, ha dado lugar a un modelo interoperable y escalable. Este enfoque ha facilitado la generación de recomendaciones más coherentes y explicables.

Asimismo, la combinación de similitud basada en ítems y filtrado colaborativo, junto con ajustes en las métricas de similitud y la introducción de factores de diversidad, ha permitido mejorar la calidad de las recomendaciones, abordando problemas como la baja superposición entre perfiles y la falta de serendipia. En conjunto, los resultados obtenidos confirman la idoneidad de los enfoques basados en conocimiento para el dominio de la recomendación de libros, habiendo alcanzado con nuestra implementación los objetivos planteados inicialmente.

## 8. Trabajo Futuro

Como trabajo futuro, se propone ampliar y refinar la ontología incorporando nuevos géneros y relaciones semánticas, así como mejorar los mecanismos de adquisición y actualización de datos para reducir problemas de latencia y falta de completitud.

También sería de interés explorar enfoques híbridos que combinen el razonamiento ontológico con técnicas de aprendizaje automático, con el fin de capturar similitudes no explícitas y mejorar la diversidad de las recomendaciones. Finalmente, la evaluación del sistema con usuarios reales permitiría comprobar su utilidad y precisión.