Cultura y Censura - Knowledge Graph - Sistema General

Visitar Knowledge Graph - Cultura y Censura

Flujo de Procesamiento

1. Ingesta de Datos

Cuando subes un archivo (RDF, CSV, o SQL), el sistema:

1. **Detecta el formato** automáticamente por extensión y contenido
2. **Valida** que el archivo es procesable
3. **Verifica** si ya fue procesado anteriormente (evita duplicados)
4. **Crea una copia** en el directorio de datos permanente

2. Análisis Estructural Previo

Antes de procesar el contenido, analizamos la estructura:

Para archivos RDF:

- Analiza y prepara el grafo completo con `rdflib`
- Identifica tipos de entidades (`rdf:type`)
- Enumera predicados y sus frecuencias
- Detecta espacios y datos usados
- Analiza patrones de conexión

Para archivos CSV:

- Analiza headers para detectar columnas geográficas
- Identifica jerarquías administrativas
- Detecta coordenadas (latitud/longitud)
- Reconoce tipos de datos en cada columna

3. Extracción de Entidades y Relaciones

Usamos **Google Generative AI** (Gemini) con un prompt especializado.
El modelo no necesita saber previamente el tipo de archivo - lo detecta automáticamente por los patrones en el texto y aplica las reglas correspondientes.

1. **Recibe el contenido** estructurado del archivo
2. **Aplica reglas específicas** según el tipo de archivo para hacer eficaz el input:
    - Para RDF: Distingue entre datos contenidas en N-Triples
    - Para CSV: Gestiona las inserciones y extrae
    - Para SQL: Interpreta inserciones, foreign_keys y ids para extraer relaciones normalizadas
3. **Genera JSON estructurado** con entidades y relaciones
4. **Normaliza IDs** para consistencia entre documentos y tipos de datos especificos (por ej. latitud y longitud)

Ejemplo Tipos de Datos Soportados

Formato	Uso Típico	Ejemplo
RDF/Turtle	Metadatos Dublin Core, registros biográficos	Archivos de brigadistas internacionales
CSV	Datos tabulares geográficos	Registros de fosas comunes, sitios históricos
SQL	Dumps de bases de datos relacionales	Archivos de INSERT con datos normalizados

Relaciones que Extraemos

Documentales

- Una persona **está documentada** en un registro
- Un autor **creó** un documento
- Un documento **menciona** conceptos o lugares

Geográficas

- Un sitio **está ubicado en** un municipio
- Un municipio **pertenece a** una comarca
- Una entidad **tiene coordenadas** específicas

Sociales e Históricas

- Una persona **pertenecía a** una organización
- Personas **fueron contemporáneas**
- Individuos **participaron en** eventos históricos

Temáticas

- Entidades **están asociadas con** conceptos
- Personas **estuvieron activas en** regiones específicas
- Documentos **contienen información sobre** temas

4. Almacenamiento en Neo4j

El grafo extraído se almacena en una base de datos grafica Neo4j donde:

1. **Entidades se convierten en nodos** `Entity` con propiedades
2. **Relaciones se convierten en aristas** tipificadas
3. **Se crean embeddings** para cada chunk de texto usando Google AI
4. **Se establecen índices vectoriales** para búsqueda semántica

5. Indexación y Vínculos Cruzados

El sistema automáticamente:

- **Vincula entidades similares** entre fuentes/documentos diferentes
- **Crea relaciones de co-ocurrencia** (entidades del mismo período)
- **Establece conexiones geográficas** (por ej. personas activas en la misma región)
- **Mantiene trazabilidad** hacia documentos fuente

Ejemplo de Tipos de Relaciones Principales

Relaciones Documentales

- `DOCUMENTS` : Un registro documenta la vida de una persona
- `AUTHORED` : Autoría de documentos
- `MENTIONS` : Referencias a personas, lugares, conceptos
- `CONTAINS_INFORMATION_ABOUT` : Información detallada sobre un tema

Relaciones Personales y Sociales

- `BELONGS_TO` : Pertenencia a organizaciones o unidades
- `SERVED_IN` : Servicio militar o institucional

- `CONTEMPORARY_OF` : Personas del mismo período histórico
- `COLLEAGUE_OF` : Conexiones profesionales o militares

Relaciones Geográficas

- `LOCATED_IN` : Ubicación geográfica jerárquica
- `BORN_IN` / `DIED_IN` : Lugares de nacimiento y fallecimiento
- `ACTIVE_IN` : Regiones donde una persona estuvo activa
- `COORDINATES_AT` : Coordenadas geográficas exactas

Relaciones Temáticas

- `PARTICIPATED_IN` : Participación en eventos históricos
- `ASSOCIATED_WITH` : Asociación con conceptos o temas
- `RELATED_TO` : Relaciones temáticas generales

Ejemplo Tipos de Entidades Principales

Tipo	Descripción	Propiedades Típicas
Person	Individuos históricos, brigadistas	`full name`, `nationality`, `military_unit`, `birth_date`
Document	Registros, archivos, metadatos	`title`, `content_type`, `author`, `creation_date`
Organization	Unidades militares, instituciones	`name`, `type`, `founding_date`, `historical_context`
Location	Lugares geográficos	`name`, `administrative_level`, `coordinates`, `country`
Site	Sitios históricos específicos	`name`, `type`, `conservation_state`, `historical_period`
Concept	Ideas, temas, períodos históricos	`name`, `description`, `domain`, `significance`
Event	Eventos históricos	`name`, `date`, `location`, `participants`

Ejemplo Ingestando RDF

Reconocimiento de Patrones RDF

Input RDF:
:record_123 dc:title "Antonio Fernández" ;
dc:type "Brigadista" ;
dc:creator "Archivo Municipal" ;
dc:subject "XV Brigada Internacional" .

**Sistema detecta:

- Patrón: título + tipo → Persona documentada
- Crea: Entidad Person("Antonio Fernández")
- Crea: Entidad Document("record_123")
- Crea: Entidad Organization("XV Brigada Internacional")
- Vincula: Document DOCUMENTS Person
- Vincula: Person BELONGS_TO Organization**

Ejemplo de Procesamiento CSV

Input CSV:
Nom | Municipi | Comarca | Província | Latitud | Longitud
Fossa Gran | Ascó | Ribera d'Ebre | Tarragona | 41.1889 | 0.5722

Sistema crea:

- Site("Fossa Gran") con coordinates
- Location("Ascó") nivel=municipi
- Location("Ribera d'Ebre") nivel=comarca
- Location("Tarragona") nivel=província
- Jerarquía: Site →LOCATED_IN→ Municipi →LOCATED_IN→ Comarca →LOCATED_IN→ Província
- Coordinates: Site →COORDINATES_AT→ (41.1889, 0.5722)

Ejemplos de Uso Chatbot Conversacional

Contexto de Investigación Histórica

"¿Qué brigadistas estuvieron activos en Tarragona?"
→ Encuentra personas → Sigue relaciones geográficas → Muestra conexiones históricas

Contexto Análisis Geográfico

Archivo CSV de fosas comunes
→ Crea jerarquía administrativa → Vincula coordenadas → Conecta con contexto histórico

Detección de Entidades Cross-Documento

El sistema identifica automáticamente:

- **Misma persona en diferentes archivos** usando nombre y contexto
- **Lugares mencionados** en múltiples fuentes
- **Organizaciones recurrentes** y sus variantes
- **Períodos temporales comunes**

Motor de Consultas

Búsqueda Semántica

Cuando haces una pregunta, el sistema:

1. **Convierte tu pregunta** en embedding vectorial
2. **Busca chunks similares** en el índice vectorial
3. **Identifica entidades relevantes** en esos chunks
4. **Expande el contexto** siguiendo relaciones en el grafo
5. **Genera respuesta** usando el contexto expandido

Navegación de Grafo

Para explorar relaciones:

1. **Seleccionas una entidad** (persona, lugar, concepto)
2. **Sistema consulta Neo4j** para encontrar conexiones directas
3. **Filtra por tipo de relación** relevante (geográfica, temporal, temática)
4. **Expande gradualmente** el grafo para mantener claridad visual
5. **Agrupa nodos similares** para evitar saturación visual

Gestión de Memoria

- **Procesamiento por chunks** para archivos grandes
  (límite 100 chunks por proceso)

Rendimiento

- **Índices especializados** en Neo4j para diferentes tipos de búsqueda
- **Consultas parametrizadas** para reutilización
- **Paginación automática** para resultados grandes

Escalabilidad

- **Procesamiento asíncrono** de archivos grandes
- **Detección de duplicados** adaptativo
- **Normalización de IDs** para consistencia

Visualización

Grafo de Relaciones

- **Algoritmo de layout** que agrupa entidades relacionadas
- **Colores semánticos** por tipo de entidad
- **Tamaños proporcionales** al número de conexiones
- **Filtros dinámicos** por tipo de relación

Análisis Estructural

- **Detección automática** de comunidades temáticas
- **Métricas de centralidad** para identificar entidades clave
- **Análisis de densidad** de conexiones por área geográfica
- **Visualización temporal** de evolución del grafo

Robustez

Validación de Datos

- **Verificación de formatos** en ingesta
- **Normalización automática** de valores
- **Detección de inconsistencias** entre fuentes
- **Logs detallados** para trazabilidad

Control de Relaciones

- **Validación cruzada** entre múltiples fuentes
- **Métricas de completitud** por tipo de entidad
- **Detección de entidades huérfanas**

Recuperación ante Errores

- **Procesamiento resiliente** que continúa ante errores menores (se calcula una tasa de error de 1%)
- **Rollback automático** de operaciones fallidas
- **Reintentos inteligentes** con ingesta adaptativas

Proximos Pasos

- Implementacion de cache dinamico
- Ampliacion ventana de contexto (escalar chunks)
- Generar estrategias especificos para tipos de datos complejos de alta prioridad (coordenadas, lugares comunes frecuentes)