

GraphRAG para la Planificación Financiera

Héctor M. Rodríguez Sosa

Universidad de La Habana

Agradecimientos

Quiero expresar mi sincero agradecimiento a todas las personas que han contribuido a la realización de esta tesis. A mi tutor, el Dr. Alejandro Piad Morffis, por su orientación experta a lo largo de este proyecto. Al profesor Roberto Marti Cedeño, por su apoyo constante en el desarrollo de mi tesis. Al profesor Daniel Alejandro Valdés Pérez, por sus consejos y apoyo sobre cómo investigar y escribir. A mi familia y amigos por su aliento y comprensión durante este viaje. A todos los que han sido parte de este proceso, gracias por su dedicación y compromiso. Este logro no habría sido posible sin su ayuda y aliento.

Resumen

Esta tesis explora el desarrollo de un sistema de preguntas y respuesta de planificación financiera utilizando un sistema de Recuperación Aumentada por Generación (RAG) basado en grafos para proporcionar asesoramiento financiero preciso y contextual. El sistema integra Neo4j para el almacenamiento de grafos, MongoDB para la recuperación basada en embeddings y la API de Gemini para la generación de lenguaje natural, con conocimiento financiero extraído y representado como nodos y aristas en un grafo para capturar relaciones complejas. Una revisión exhaustiva de las arquitecturas RAG basadas en grafos y sistemas de preguntas y respuestas fintech guió las decisiones de diseño, asegurando su alineación con los desafíos específicos del dominio. El sistema de preguntas y respuestas fue evaluado mediante consultas diseñadas a mano, y métricas como precisión, corrección factual, eficiencia y robustez demostraron su efectividad en aplicaciones financieras. Este trabajo destaca el potencial de los sistemas RAG basados en grafos para soluciones fintech inteligentes, ofreciendo un marco escalable para el desarrollo de sistemas de preguntas y respuestas y sentando las bases para futuros avances en extracción de subgrafos, integración de datos en tiempo real y adaptabilidad entre dominios.

Abstract

This thesis explores the development of a financial planner Q&A system using a graph-based Retrieval-Augmented Generation (RAG) system to deliver accurate and contextually relevant financial advice. The system integrates Neo4j for graph storage, MongoDB for embedding-based retrieval, and the Gemini API for natural language generation, with financial knowledge extracted and represented as graph nodes and edges to capture complex relationships. A comprehensive review of graph-based RAG architectures and fintech Q&A systems informed the design decisions, ensuring alignment with domain-specific challenges.

The Q&A system was evaluated using handcrafted queries, with metrics such as accuracy, factual correctness, efficiency, and robustness demonstrating its effectiveness in financial applications. This work highlights the potential of graph-based RAG systems for intelligent fintech solutions, offering a scalable framework for Q&A system development while paving the way for future advancements in subgraph extraction, real-time data integration, and cross-domain adaptability.

Tabla de contenidos

1. Introducción	7
1.1. Antecedentes y Contexto	7
1.2. Problema y Objetivos de Investigación	7
1.3. Preguntas de Investigación	8
1.4. Alcance y Delimitaciones	9
1.5. Metodología	9
1.5.1. Investigación	9
1.5.2. Diseño y Desarrollo del Sistema	10
1.6. Estructura de la Tesis	11
2. Preliminares	13
2.1. <i>Servicios de Asesoría Financiera Personalizada</i>	13
2.2. <i>Instrumentos Financieros</i>	13
2.3. <i>Tolerancia al Riesgo y Perfil de Riesgo</i>	13
2.4. <i>Cumplimiento Regulatorio en Servicios Financieros</i>	13
2.5. <i>Tipos y Fuentes de Datos Financieros</i>	14
2.6. <i>Perfilado de Clientes en Finanzas</i>	14
2.7. <i>Visión General de los Mercados Financieros</i>	14
2.8. <i>Estrategias de Inversión</i>	15
2.9. <i>Proceso de Planificación Financiera</i>	15
2.10. <i>Conceptos Financieros Clave</i>	15
2.11. <i>Indicadores Económicos</i>	16
3. Revisión de Literatura	17
3.1. Preprocesamiento y Almacenamiento de Datos	17
3.2. Recuperación	20
3.3. Generación	24
3.4. Aplicaciones y Estudios de RAG No Financieras	25
3.5. Aplicaciones y Estudios de RAG en Finanzas	27
3.6. La Necesidad de Graph RAG en la Planificación y Asesoramiento Financiero	31
3.7. Resumen	31
4. Implementación	32
4.1. Recolección y preprocesamiento de datos	32
4.2. Construcción del grafo	32
4.3. Incrustación de vectores	33
4.4. Responder a una consulta	34
4.5. Utilizando un ejército de expertos	35
5. Evaluación	36
5.1. Descripción general de la evaluación	36
5.2. Métrica ROUGE para la evaluación	36
5.3. Métricas de evaluación	36
5.4. Consultas de ejemplo	37
5.5. Resultados	37
5.5.1. Precisión	37

5.5.2. Rendimiento	38
5.5.3. Robustez	38
5.6. Resumen	38
6. Conclusiones	39
6.1. Limitaciones y trabajo futuro	39
Bibliografía	41

1. Introducción

1.1. Antecedentes y Contexto

El sector de la tecnología financiera (fintech) ha experimentado un crecimiento sin precedentes en la última década, revolucionando la forma en que las personas y las organizaciones gestionan sus finanzas. Los métodos tradicionales de planificación financiera, que a menudo implicaban la contabilidad manual, las consultas en persona con asesores financieros o las herramientas de presupuestación estáticas, están siendo reemplazados rápidamente por soluciones digitales innovadoras. Estos avances están impulsados por la creciente adopción de la inteligencia artificial (IA), el procesamiento del lenguaje natural (PLN) y las tecnologías de big data.

Hoy día, los asistentes financieros personales, los sistemas de preguntas y respuestas y los robo-asesores dominan el panorama. Empresas como Mint, YNAB (You Need a Budget) y Wealthfront proporcionan recomendaciones personalizadas y conocimientos automatizados, permitiendo a los usuarios rastrear gastos, establecer metas financieras y tomar decisiones de inversión informadas. De manera similar, los sistemas de IA conversacional como Cleo y Erica (desarrollado por Bank of America) permiten a los usuarios interactuar con sus datos financieros de manera fluida a través de interfaces basadas en chat. Estos sistemas se basan en modelos de aprendizaje automático sofisticados y en un extenso entrenamiento de datos para ofrecer asesoramiento personalizado.

Esta transición a enfoques digitales está impulsada por la demanda de conveniencia, accesibilidad y asistencia en tiempo real. A diferencia de la planificación financiera tradicional, que requería un tiempo y una experiencia significativos, los sistemas modernos democratizan la gestión financiera al hacer que las herramientas sean intuitivas y fácilmente disponibles. Los usuarios ahora pueden acceder a conocimientos desde cualquier lugar, recibir respuestas instantáneas y beneficiarse de estrategias impulsadas por IA, que anteriormente estaban limitadas a consultas de expertos. Esto es particularmente beneficioso porque empodera a todos los usuarios, independientemente de su alfabetización financiera, en tiempos pasados estos eran servicios adaptados y disponibles únicamente para una minoría adinerada de la población.

A pesar de estos avances, la mayoría de las soluciones existentes dependen de un extenso entrenamiento de modelos de IA, lo que requiere grandes conjuntos de datos y recursos computacionales. Esta dependencia introduce desafíos en la escalabilidad, la adaptabilidad a las necesidades específicas de los usuarios y el mantenimiento de bases de conocimiento actualizadas. La próxima frontera en la innovación fintech radica en aprovechar sistemas más dinámicos, eficientes y adaptables que puedan generar respuestas precisas y contextualmente relevantes sin el reentrenamiento tradicional de modelos.

1.2. Problema y Objetivos de Investigación

La adopción de sistemas de Generación Aumentada por Recuperación (RAG) en la planificación financiera ha mejorado significativamente la eficiencia y precisión de los sistemas de IA conversacional. La mayoría de las implementaciones existentes dependen de sistemas RAG basados en vectores, donde las incrustaciones de texto se indexan y buscan por similitud semántica. Aunque efectivo, este enfoque tiene limitaciones inherentes para capturar las relaciones estructuradas intrincadas que son críticas para ciertos campos como la planificación financiera, donde las interdependencias entre conceptos como gastos, inversiones, metas y riesgos son altamente relacionales.

Los sistemas RAG basados en grafos, que se centran en búsquedas de relaciones semánticas a través de estructuras de grafos, ofrecen una alternativa al codificar relaciones de manera explícita y dinámica. Esta capacidad puede potencialmente mejorar la comprensión contextual de los datos financieros, permitiendo una representación más matizada e interconectada de las metas del usuario, patrones de gasto y estrategias de inversión. Sin embargo, los sistemas RAG basados en grafos aún no se han explorado en el dominio de la planificación financiera.

Además, las soluciones de IA actuales en este dominio a menudo requieren un tiempo, recursos y experiencia sustanciales para entrenar y ajustar modelos de lenguaje grandes (LLMs). Esta dependencia crea barreras para individuos y organizaciones que no tienen acceso a vastos conjuntos de datos y poder computacional, limitando la democratización de estas herramientas avanzadas. Al investigar la viabilidad de un enfoque de “zero-shot”—confiando únicamente en la afinación de prompts sin reentrenamiento de modelos—esta investigación busca reducir estas barreras y hacer que los sistemas de preguntas y respuestas de planificación financiera sean más accesibles y asequibles.

Esta tesis aborda la brecha introduciendo un sistema RAG basado en grafos para la planificación financiera y explorando cómo un enfoque de afinación de prompts puede desempeñarse en este contexto. Se espera que los hallazgos contribuyan al desarrollo de herramientas de IA prácticas y rentables que estén verdaderamente disponibles para una audiencia más amplia, mientras se exploran las ventajas únicas que las estructuras de grafos pueden aportar a este dominio.

El objetivo principal de esta tesis es diseñar y evaluar un sistema de preguntas y respuestas de planificación financiera basado en un sistema RAG de grafos, utilizando solo la afinación de prompts para generar salidas significativas. Los objetivos incluyen:

1. Evaluar el rendimiento del sistema RAG de grafos en comparación con los sistemas RAG basados en vectores en la provisión de recomendaciones financieras relevantes y precisas.
2. Explorar la usabilidad y practicidad de un enfoque de zero-shot en aplicaciones del mundo real.
3. Identificar las ventajas y limitaciones de la búsqueda semántica basada en grafos para capturar relaciones financieras.

Al abordar estos objetivos, esta investigación busca contribuir al campo más amplio de la IA en fintech, ofreciendo ideas sobre la aplicabilidad de los sistemas RAG basados en grafos y el potencial de los métodos zero-shot para hacer que las soluciones de IA sean más accesibles e impactantes.

1.3. Preguntas de Investigación

Esta investigación busca explorar el potencial de un sistema de Generación Aumentada por Recuperación (RAG) basado en grafos para la planificación financiera abordando las siguientes preguntas clave:

1. Efectividad del Sistema RAG Basado en Grafos
 - ¿Con qué efectividad puede un sistema RAG basado en grafos capturar las relaciones semánticas necesarias para ofrecer asesoramiento financiero personalizado y contextualmente preciso?
2. Viabilidad del Ajuste de Prompts Zero-Shot
 - ¿Puede un enfoque de ajuste de prompts zero-shot generar recomendaciones financieras significativas y precisas sin requerir un extenso entrenamiento del modelo?
3. Comparación con Sistemas RAG Basados en Vectores

- ¿Cómo se compara el rendimiento de un sistema RAG basado en grafos con los sistemas basados en vectores en términos de relevancia, comprensión contextual y adaptabilidad dentro del dominio de la planificación financiera?
4. Practicidad y Escalabilidad del Sistema
- ¿Cuáles son las ventajas prácticas, limitaciones y posibles desafíos de usar un sistema RAG basado en grafos para aplicaciones de planificación financiera en el mundo real?

Estas preguntas guían la exploración y evaluación del sistema de preguntas y respuestas propuesto, ayudando a identificar tanto su potencial técnico como sus implicaciones prácticas.

1.4. Alcance y Delimitaciones

Esta investigación se centra en diseñar y evaluar un sistema de preguntas y respuestas de planificación financiera utilizando un sistema de Generación Aumentada por Recuperación (RAG) basado en grafos, confiando exclusivamente en un enfoque de ajuste de prompts “zero-shot”. Aunque las técnicas exploradas en este estudio pueden eventualmente ser útiles para desarrollar sistemas de preguntas y respuestas genéricos específicos de dominio, el enfoque principal está limitado al dominio de la planificación financiera. El alcance no se extiende a la creación de un marco generalizado para sistemas de preguntas y respuestas basados en grafos, ni esta investigación abordará la implementación de sistemas RAG utilizando arquitecturas alternativas como los enfoques basados en vectores, ya que estos están ampliamente cubiertos en la literatura existente.

En cambio, este trabajo adopta un enfoque exploratorio, experimentando con varias técnicas y configuraciones necesarias para el desarrollo del sistema. Esto incluye:

- Investigar diferentes métodos de búsqueda en grafos para mejorar la recuperación de relaciones semánticas.
- Explorar configuraciones de prompts y estrategias de ajuste para mejorar la calidad de las respuestas del sistema de preguntas y respuestas.
- Experimentar con técnicas de generación de respuestas adaptadas al contexto de la planificación financiera.

La investigación adopta una metodología de “libre para todos”, permitiendo flexibilidad en la prueba e integración de diferentes enfoques, herramientas y algoritmos según sea necesario para lograr los objetivos del proyecto. Sin embargo, el estudio se mantiene estrictamente dentro de los límites de los sistemas RAG basados en grafos y no profundiza en tecnologías o marcos no relacionados.

Al mantener este enfoque, la investigación pretende ofrecer conocimientos específicos sobre la aplicación de sistemas RAG basados en grafos para la planificación financiera, sin intentar generalizar los hallazgos más allá del dominio o arquitectura especificados.

1.5. Metodología

Este estudio se llevó a cabo en dos etapas principales: investigación y diseño y desarrollo del sistema. Cada etapa jugó un papel crucial en la realización de un sistema de preguntas y respuestas de planificación financiera basado en un sistema de Generación Aumentada por Recuperación (RAG) basado en grafos, asegurando tanto una base teórica como una implementación práctica.

1.5.1. Investigación

La fase de investigación comprendió dos objetivos principales:

1. Encuesta de Sistemas RAG Basados en Grafos

Para comprender los principios y metodologías subyacentes para implementar un sistema RAG basado en grafos, se realizó una revisión exhaustiva de la literatura académica relevante. Se analizaron críticamente artículos que discutían técnicas para la recuperación, indexación y generación de respuestas dentro de arquitecturas basadas en grafos. Esto fue esencial para obtener una comprensión integral de los enfoques existentes y su aplicabilidad para construir el sistema propuesto.

2. Revisión de Sistemas de sistemas de preguntas y respuestas Fintech

Una revisión secundaria se centró en explorar trabajos existentes en el dominio de los sistemas de preguntas y respuestas fintech, particularmente aquellos que emplean sistemas RAG. Esta investigación proporcionó valiosos conocimientos sobre cómo se abordaron los desafíos específicos del dominio, como la relevancia contextual, la comprensión de la intención del usuario y la representación de datos financieros. Los hallazgos de esta revisión informaron las decisiones de diseño adaptadas al dominio de la planificación financiera.

Estas dos corrientes de investigación aseguraron que el diseño del sistema estuviera basado en técnicas establecidas mientras se abordaban los requisitos únicos de la aplicación objetivo.

1.5.2. Diseño y Desarrollo del Sistema

El sistema se implementó en varias fases, cada una abordando componentes distintos necesarios para la funcionalidad del sistema de preguntas y respuestas.

1. Extracción de Conocimiento

Se utilizaron herramientas y bibliotecas como LangChain, Beautiful Soup y APIs de Gemini y OpenAI para extraer conocimientos de planificación financiera de varias fuentes web. La información se procesó en una estructura de grafo con nodos y aristas, donde ambos contenían anotaciones textuales. Por ejemplo:

- Nodo: “ARRENDAMIENTO DE COCHE”
- Arista: “ES UNA FUENTE DE”
- Nodo Objetivo: “DEUDA”

Además, el grafo capturó múltiples relaciones por nodo, como se ilustra en:

- “ARRENDAMIENTO DE COCHE” -> “REQUIERE” -> “PAGOS MENSUALES”.

La estructura del grafo se almacenó en Neo4j, mientras que las representaciones incrustadas del contenido textual se almacenaron en una base de datos de vectores utilizando MongoDB. Esta representación dual permitió flexibilidad en las estrategias de recuperación.

2. Recuperación de Conocimiento

El enfoque principal fue recuperar conocimiento de las incrustaciones del grafo en lugar de la estructura del grafo en sí. Se emplearon técnicas como el clustering k-means y la similitud coseno para identificar nodos y aristas relevantes. Estos métodos permitieron una recuperación eficiente de puntos de datos semánticamente similares basados en las consultas de los usuarios.

3. Extracción de Subgrafos

Los nodos y aristas recuperados formaron la base para construir un subgrafo, representando el subconjunto de información más relevante para una consulta dada. Se exploraron varias técnicas para la extracción de subgrafos, que se detallarán en secciones posteriores.

4. Generación de Respuestas

El subgrafo extraído se procesó para generar respuestas en lenguaje natural. Esto se logró utilizando la API de Gemini, aprovechando varios modelos para transformar los datos estructurados del grafo en salidas coherentes y contextualmente apropiadas.

5. Evaluación y Pruebas

Para evaluar el rendimiento del sistema de preguntas y respuestas, se creó un conjunto de consultas elaboradas a mano que cubrían diversos escenarios financieros. Las respuestas del sistema se compararon con las de otros sistemas y modelos de propósito general. La evaluación se basó en las siguientes métricas:

- **Precisión:** La corrección de la información proporcionada en relación con la consulta.
- **Corrección Factual:** Si la respuesta se alineaba con el conocimiento financiero verificado.
- **Eficiencia:** La velocidad y el costo computacional de generar una respuesta.
- **Robustez:** La capacidad del sistema para manejar consultas ambiguas o incompletas.

Este marco metodológico aseguró una exploración e implementación sistemática del sistema RAG basado en grafos propuesto, proporcionando una base para contribuciones tanto técnicas como prácticas al campo de la planificación financiera.

1.6. Estructura de la Tesis

Esta tesis está organizada en varios capítulos, cada uno abordando un aspecto específico del proceso de investigación y contribuyendo al objetivo general de desarrollar un sistema de preguntas y respuestas de planificación financiera basado en un sistema de Generación Aumentada por Recuperación (RAG) basado en grafos. La estructura es la siguiente:

- **Capítulo 1: Introducción:** La introducción proporciona una visión general de la investigación, incluyendo el contexto, la declaración del problema, los objetivos de la investigación y la motivación del estudio. También se describen el alcance y las delimitaciones del trabajo, y se presentan las preguntas de investigación y las hipótesis que guían el estudio.
 - **Capítulo 2: Preliminares:** Este capítulo proporciona una descripción general de los conceptos principales de las finanzas personales y la planificación financiera comúnmente manejados en el sistema, no es necesaria su lectura para comprender el resto de la tesis.
- **Capítulo 3: Revisión de la Literatura:** Este capítulo revisa la literatura relevante en las áreas de sistemas de preguntas y respuestas fintech, sistemas RAG y representaciones de conocimiento basadas en grafos. Incluye una discusión de los enfoques existentes para el desarrollo de sistemas de preguntas y respuestas en el dominio de la planificación financiera y una encuesta de diferentes técnicas para sistemas RAG basados en grafos, proporcionando la base para la investigación.
- **Capítulo 4: Arquitectura e Implementación del Sistema:** Este capítulo proporciona una descripción detallada de la arquitectura del sistema, incluyendo los componentes involucrados en la extracción de conocimiento, la construcción de grafos, la recuperación y la generación de lenguaje natural. También se discute el proceso de implementación, con especial atención a las herramientas, marcos y bases de datos utilizados para desarrollar el sistema.
- **Capítulo 5: Evaluación y Resultados:** En este capítulo se presentan los resultados de la evaluación del sistema. Se evalúa el rendimiento del sistema de preguntas y respuestas basado en varias métricas como precisión, corrección factual, eficiencia y robustez. Los hallazgos de los experimentos de evaluación se discuten en detalle y se analizan las implicaciones de los resultados.
- **Capítulo 6: Conclusión y Trabajo Futuro:** El capítulo de conclusión resume los hallazgos clave de la investigación, reflexiona sobre la contribución del trabajo y discute sus limitaciones. También

sugiere posibles vías para futuras investigaciones, incluyendo mejoras al sistema y aplicaciones adicionales de enfoques RAG basados en grafos en otros dominios.

2. Preliminares

Esta sección proporciona una visión general de los conceptos financieros esenciales pertinentes a los servicios de asesoría financiera personalizada. Su objetivo es familiarizar a los lectores que pueden no tener antecedentes en finanzas con la terminología y los principios fundamentales relevantes de las finanzas personales para esta tesis.

2.1. *Servicios de Asesoría Financiera Personalizada*

Los servicios de asesoría financiera personalizada implican ofrecer orientación financiera a medida para individuos basada en sus situaciones financieras únicas, objetivos, tolerancias al riesgo y preferencias. A diferencia de los consejos genéricos, los servicios personalizados consideran las circunstancias específicas de cada cliente para proporcionar recomendaciones sobre inversiones, ahorros, planificación de la jubilación, estrategias fiscales y más.

2.2. *Instrumentos Financieros*

Los instrumentos financieros son activos o paquetes de capital que pueden ser negociados. Se clasifican en varias categorías:

- **Acciones (Stocks):** Representan participaciones de propiedad en una empresa. Los accionistas pueden recibir dividendos y tienen potencial de apreciación de capital.
- **Valores de Renta Fija (Bonos):** Instrumentos de deuda emitidos por corporaciones o gobiernos. Los inversionistas reciben pagos periódicos de intereses y la devolución del principal al vencimiento.
- **Derivados:** Contratos cuyo valor se deriva de activos subyacentes como acciones, bonos, materias primas o monedas. Ejemplos incluyen opciones y futuros.
- **Fondos Mutuos y Fondos Cotizados en Bolsa (ETFs):** Vehículos de inversión agrupados que permiten a los inversionistas poseer una cartera diversificada de activos.
- **Commodities:** Bienes físicos como oro, petróleo o productos agrícolas que pueden ser negociados en mercados especializados.

2.3. *Tolerancia al Riesgo y Perfil de Riesgo*

La tolerancia al riesgo se refiere a la capacidad y disposición de un inversionista para soportar la volatilidad y las posibles pérdidas en su cartera de inversiones. Está influenciada por factores como los objetivos de inversión, el horizonte temporal, la estabilidad financiera y la comodidad psicológica con el riesgo.

El perfil de riesgo es el proceso de evaluar la tolerancia al riesgo de un individuo para alinear las estrategias de inversión en consecuencia. Asegura que los consejos financieros y los productos recomendados sean adecuados para el apetito de riesgo del cliente.

2.4. *Cumplimiento Regulatorio en Servicios Financieros*

Las instituciones financieras operan bajo marcos regulatorios estrictos diseñados para proteger a los consumidores y mantener la integridad del mercado. Los conceptos regulatorios clave incluyen:

- **Conozca a su Cliente (KYC):** Procedimientos para verificar la identidad de los clientes, destinados a prevenir fraudes, lavado de dinero y otras actividades ilícitas.

- **Anti-Lavado de Dinero (AML):** Regulaciones que requieren que las instituciones financieras monitoreen, detecten y reporten transacciones sospechosas.
- **Deber Fiduciario:** Una obligación legal que requiere que los asesores financieros actúen en el mejor interés de sus clientes, priorizando las necesidades de los clientes sobre las propias.
- **Idoneidad y Apropiación:** Los asesores deben asegurar que los productos y servicios financieros sean adecuados para la situación financiera y los objetivos de inversión del cliente.

2.5. Tipos y Fuentes de Datos Financieros

La asesoría financiera efectiva se basa en diversos tipos de datos:

- **Datos de Clientes:** Información personal y financiera sobre los clientes, incluyendo ingresos, gastos, activos, pasivos y metas de vida.
- **Datos de Mercado:** Información sobre los mercados financieros, como precios de acciones, tasas de interés e indicadores económicos.
- **Datos de Productos:** Detalles sobre productos financieros, incluyendo historial de rendimiento, tarifas, términos y condiciones.
- **Actualizaciones Regulatorias:** Información sobre cambios en leyes, regulaciones y requisitos de cumplimiento.
- **Datos No Estructurados:** Información textual de fuentes como artículos de noticias, informes de analistas y redes sociales que pueden impactar las decisiones financieras.

2.6. Perfilado de Clientes en Finanzas

El **perfilado de clientes** implica crear una imagen detallada de un cliente para ofrecer servicios personalizados. Los elementos incluyen:

- **Demografía:** Edad, género, educación, ocupación y estado familiar.
- **Estado Financiero:** Patrimonio neto, nivel de ingresos, historial crediticio y flujo de efectivo.
- **Objetivos de Inversión:** Metas como preservación de capital, generación de ingresos o crecimiento.
- **Rasgos Conductuales:** Actitudes hacia el gasto, el ahorro y la inversión; respuestas a las fluctuaciones del mercado.

2.7. Visión General de los Mercados Financieros

Los mercados financieros facilitan la compra y venta de instrumentos financieros. Son cruciales para:

- **Descubrimiento de Precios:** Determinar el valor de los activos basado en la oferta y la demanda.
- **Provisión de Liquidez:** Permitir a los inversionistas comprar o vender activos rápidamente sin causar cambios significativos en el precio.
- **Asignación de Capital:** Dirigir fondos de los ahorradores a entidades que pueden usarlos para propósitos productivos.

Tipos de Mercados Financieros:

- **Mercados de Acciones:** Donde se emiten y negocian acciones.

- **Mercados de Bonos:** Para la negociación de valores de deuda.
- **Mercados de Divisas:** Donde se negocian monedas.
- **Mercados de Derivados:** Para la negociación de contratos como futuros y opciones.

2.8. Estrategias de Inversión

Las estrategias de inversión son planes diseñados para lograr objetivos financieros específicos. Las estrategias comunes incluyen:

- **Asignación de Activos:** Distribuir inversiones entre diferentes clases de activos (por ejemplo, acciones, bonos, efectivo) para equilibrar riesgo y recompensa.
- **Diversificación:** Invertir en una variedad de activos para reducir la exposición a cualquier activo o riesgo único.
- **Inversión en Valor:** Seleccionar acciones infravaloradas con sólidos fundamentos.
- **Inversión en Crecimiento:** Enfocarse en empresas que se espera que crezcan a una tasa superior al promedio.
- **Inversión en Ingresos:** Apuntar a inversiones que proporcionen ingresos regulares, como dividendos o pagos de intereses.

2.9. Proceso de Planificación Financiera

El proceso de planificación financiera es un enfoque sistemático para gestionar las actividades financieras de un individuo. Los pasos clave incluyen:

1. **Establecimiento de la Relación Cliente-Asesor:** Definir el alcance de los servicios y las responsabilidades.
2. **Recolección de Datos y Objetivos:** Recopilar información completa sobre la situación financiera y los objetivos del cliente.
3. **Análisis del Estado Financiero:** Evaluar la salud financiera actual, incluyendo flujo de efectivo, niveles de deuda y cartera de inversiones.
4. **Desarrollo de Recomendaciones:** Formular estrategias para alcanzar los objetivos financieros, considerando la tolerancia al riesgo y los horizontes temporales.
5. **Implementación del Plan:** Ejecutar las estrategias acordadas, lo que puede implicar la compra de productos financieros o el ajuste de participaciones existentes.
6. **Monitoreo y Revisión:** Evaluar regularmente el rendimiento del plan y hacer ajustes en respuesta a cambios en las circunstancias del cliente o en las condiciones del mercado.

2.10. Conceptos Financieros Clave

- **Valor Temporal del Dinero:** El principio de que una suma de dinero tiene mayor valor ahora que la misma suma en el futuro debido a su potencial de generación de ingresos.
- **Interés Compuesto:** Ganar intereses tanto sobre el principal inicial como sobre los intereses acumulados de periodos anteriores.

- **Inflación:** La tasa a la cual el nivel general de precios de bienes y servicios aumenta, erosionando el poder adquisitivo.
- **Liquidez:** La facilidad con la que un activo puede convertirse en efectivo sin afectar su precio de mercado.
- **Apalancamiento:** Uso de capital prestado para aumentar el retorno potencial de una inversión, lo que también incrementa el riesgo potencial.

2.11. Indicadores Económicos

Los indicadores económicos son estadísticas que proporcionan información sobre el rendimiento económico y las perspectivas futuras. Ejemplos incluyen:

- **Producto Interno Bruto (PIB):** Valor total de bienes y servicios producidos, indicando la salud económica.
- **Tasa de Desempleo:** Porcentaje de la fuerza laboral que está desempleada, reflejando las condiciones del mercado laboral.
- **Índice de Precios al Consumidor (IPC):** Mide los cambios en el nivel de precios de una canasta de bienes y servicios de consumo, indicando la inflación.
- **Tasas de Interés:** El costo de pedir dinero prestado, influyendo en el gasto de los consumidores y la inversión empresarial.

3. Revisión de Literatura

Esta revisión de la literatura comenzará proporcionando una visión general de las tecnologías, marcos y metodologías de vanguardia utilizados en aplicaciones de RAG utilizando una base de conocimiento en grafo. Esto incluirá el preprocesamiento de documentos, la creación de grafos de conocimiento, la incrustación, la recuperación de información y la generación utilizando LLMs. Posteriormente, discutiremos algunos estudios relevantes sobre aplicaciones de RAG en campos no financieros, particularmente aplicaciones médicas, ya que pueden ser útiles al orientar el uso específico del dominio de la arquitectura RAG. Y después de eso, resumiremos algunos estudios sobre el uso de RAG, calidad de datos, etc., para aplicaciones FinTech, especialmente la planificación financiera, que será el enfoque de la propuesta de este documento.

3.1. Preprocesamiento y Almacenamiento de Datos

Las bases de datos de grafos han emergido como sistemas especializados diseñados para almacenar y gestionar datos en formatos de grafo, enfatizando las relaciones entre los puntos de datos. Tecnologías como Neo4j, Amazon Neptune y ArangoDB ofrecen soluciones escalables para manejar estructuras de grafos complejas. Estas bases de datos permiten consultas y recorridos eficientes de las relaciones, lo que las hace ideales para aplicaciones que requieren una recuperación y manipulación de datos interconectados de alto rendimiento, lo cual es fundamental para los sistemas Graph RAG.

El almacenamiento y la recuperación efectivos en Graph RAGs dependen de estructuras de datos optimizadas y métodos de indexación que faciliten el acceso rápido a nodos y aristas relevantes. Se emplean técnicas como listas de adyacencia, indexación de caminos y particionamiento de grafos para gestionar grandes grafos de manera eficiente. Lenguajes de consulta como Cypher y Gremlin permiten capacidades de consulta expresivas, lo que permite a los sistemas RAG recuperar subgrafos o relaciones pertinentes necesarias para la generación de contenido informada.

Basándose en estas tecnologías fundamentales, **Peng et al. (2024)** discutieron la construcción y la indexación de bases de datos de grafos en el contexto de GraphRAG, enfatizando tanto las bases de datos de grafos de código abierto como las autoconstruidas. Exploraron varios métodos de indexación, incluyendo la indexación de grafos, texto y vectores, para mejorar la eficiencia de la recuperación [1]. Su trabajo destaca la importancia de estrategias de indexación personalizadas para mejorar el rendimiento de los sistemas RAG.

Un grafo de conocimiento es una representación estructurada de entidades del mundo real y sus interrelaciones, organizada en un formato de grafo donde los nodos representan entidades y las aristas representan relaciones. Construir un grafo de conocimiento implica crear una representación estructurada de la información donde las entidades (como personas, lugares o conceptos) son nodos, y las relaciones entre ellas son aristas que conectan estos nodos. El proceso comienza con la recopilación de datos de diversas fuentes, incluyendo bases de datos estructuradas, documentos de texto no estructurados y contenido web. Estos datos diversos se preprocesan para asegurar la consistencia y la calidad, lo que implica la limpieza para eliminar inexactitudes y la normalización para estandarizar formatos [2].

El núcleo de la construcción de un grafo de conocimiento implica extraer entidades y las relaciones entre ellas utilizando técnicas de procesamiento de lenguaje natural como el reconocimiento de entidades nombradas y la extracción de relaciones. Se desarrolla una ontología o esquema para definir las categorías de entidades y las relaciones permisibles, proporcionando una estructura formal que

sustenta el grafo de conocimiento. Esta ontología sirve como un plano que guía la integración de datos y asegura la consistencia semántica a lo largo del grafo [3].

La integración de datos implica alinear y fusionar datos de diferentes fuentes, que pueden referirse a las mismas entidades de diversas maneras. Técnicas como la resolución de entidades ayudan a identificar y unificar estas referencias para mantener un grafo cohesivo. La fase de construcción implica crear la estructura del grafo instanciando nodos para entidades y aristas para relaciones según lo definido por la ontología, y enriqueciéndolos con atributos y propiedades. [3]

Una vez construido el grafo de conocimiento, la validación es crucial para asegurar su precisión y coherencia. Se pueden aplicar mecanismos de razonamiento lógico e inferencia para detectar inconsistencias y derivar nuevas ideas a partir de los datos existentes. El mantenimiento del grafo de conocimiento implica actualizaciones regulares con nueva información, escalabilidad para acomodar el crecimiento y optimización para una consulta y recuperación eficientes. [3]

La integración de grafos de conocimiento en los sistemas RAG ha mostrado beneficios significativos. **Hu et al. (2024)** demuestran cómo los grafos de conocimiento sirven como fuentes ricas de información, proporcionando entidades relevantes y sus relaciones, lo que conduce a respuestas más coherentes y precisas de los modelos de lenguaje [4]. Al aprovechar la información estructurada en los grafos de conocimiento, los sistemas RAG pueden generar contenido que es tanto contextualmente relevante como semánticamente preciso.

En dominios específicos como la medicina, la estructura de los datos del grafo se vuelve aún más crítica. **Cui et al. (2024)** exploran la aplicación de la descomposición de tareas mediante Grafos Acíclicos Dirigidos (DAG) en el dominio médico. Proponen que estructuras jerárquicas de grafos, como los DAGs, proporcionan una manera sistemática de procesar tareas médicas complejas, permitiendo que los grandes modelos de lenguaje organicen y representen de manera más eficiente las dependencias entre tareas. Esto permite una ejecución de tareas más estructurada y el uso de metadatos para los nodos de las tareas, mejorando así la gestión y comprensión del conocimiento médico. Su investigación muestra que las estructuras de DAG, especialmente en la generación aumentada por recuperación, pueden optimizar el procesamiento de tareas médicas al permitir una descomposición clara de subtareas [5].

Al considerar la elección entre diferentes sistemas de bases de datos, es esencial analizar los compromisos para asegurar un rendimiento óptimo. En comparación con las bases de datos relacionales tradicionales, las bases de datos de grafos ofrecen un rendimiento superior en el manejo de datos complejos e interconectados debido a su diseño inherente enfocado en las relaciones. Mientras que las bases de datos relacionales sobresalen en la gestión de datos estructurados y tabulares, son menos eficientes para operaciones que requieren múltiples uniones o recorridos. Esto enfatiza las ventajas de las bases de datos de grafos en el apoyo a las necesidades dinámicas de recuperación de los sistemas RAG.

Añadiendo a esta perspectiva, **Jin et al. (2024)** discuten los beneficios de rendimiento de las bases de datos vectoriales basadas en grafos, como HNSW, utilizadas en los sistemas de Generación Aumentada por Recuperación (RAG). Destacan que las bases de datos vectoriales estructuradas como grafos ofrecen ventajas significativas en términos de búsqueda y recuperación eficientes en comparación con otros tipos de bases de datos. Específicamente, se muestra que las bases de datos de grafos son más eficientes en la gestión de búsquedas vectoriales a gran escala a través de estructuras jerárquicas, reduciendo la complejidad computacional y mejorando el rendimiento de las consultas. Esto contrasta con las bases

de datos tradicionales que pueden no estar optimizadas para búsquedas vectoriales de alta dimensión. Su trabajo demuestra que incorporar bases de datos de grafos en los sistemas RAG puede mejorar la eficiencia de la recuperación de conocimiento, particularmente para aplicaciones que requieren búsquedas de similitud vectorial de alta precisión [6].

Además, la organización de los datos dentro de los grafos puede mejorar aún más los procesos de recuperación. **Chang et al. (2024)** introducen CommunityKG-RAG, un nuevo marco que aprovecha las estructuras comunitarias en grafos de conocimiento (KGs) para mejorar los sistemas de generación aumentada por recuperación. Enfatizan que los sistemas RAG tradicionales sufren de una pobre integración del conocimiento estructurado, lo que limita su efectividad en tareas de verificación de hechos. Al centrarse en la detección de comunidades dentro de los KGs, CommunityKG-RAG proporciona un proceso de recuperación más contextualmente rico en comparación con las bases de datos estándar o las estructuras de conocimiento planas. Este enfoque mejora significativamente la recuperación de información relevante al utilizar las relaciones de múltiples saltos en los KGs, mostrando las ventajas de las estructuras de datos basadas en grafos sobre otros tipos de bases de datos en aplicaciones complejas de verificación de hechos [7].

Un aspecto esencial para mejorar la recuperación en los sistemas RAG es el uso de incrustaciones semánticas. Las incrustaciones semánticas representan palabras, frases o entidades en espacios vectoriales continuos, capturando sus significados y relaciones. En los sistemas RAG, estas incrustaciones son cruciales para el mapeo entre el modelo de lenguaje y la base de conocimiento. Permiten que el sistema recupere información semánticamente relevante basada en el contexto de la entrada, facilitando una generación más coherente y precisa.

Para generar estas incrustaciones, se emplean diversas técnicas. Métodos tradicionales como Word2Vec y GloVe capturan relaciones semánticas basadas en la co-ocurrencia de palabras, mientras que modelos contextuales como BERT y GPT consideran el significado de las palabras basado en el contexto circundante. Para datos de grafos, métodos como Node2Vec y GraphSAGE generan incrustaciones que capturan información estructural dentro del grafo. Combinar estos enfoques permite que los sistemas RAG creen representaciones ricas que reflejan tanto propiedades lingüísticas como relacionales. [1]

En el contexto de grafos de conocimiento textuales—donde tanto los nodos como las aristas tienen notaciones textuales—los modelos de lenguaje preentrenados resultan particularmente beneficiosos. **He et al. (2024)** afirmaron que usar modelos como **SentenceBERT** puede mejorar la capacidad del sistema para entender y recuperar información relevante [8]. Este enfoque aprovecha las fortalezas de los modelos de lenguaje para capturar matices semánticos, mejorando así el proceso de recuperación en los sistemas RAG.

Las incrustaciones semánticas no solo mejoran la relevancia de la recuperación, sino que también permiten una mejor generalización a través de diferentes contextos. Las aplicaciones se extienden a la búsqueda semántica, sistemas de recomendación y personalización. Los beneficios incluyen respuestas más precisas, mayor satisfacción del usuario y la capacidad de manejar consultas ambiguas o novedosas de manera efectiva. Al integrar incrustaciones semánticas, los sistemas RAG se vuelven más aptos para manejar las complejidades del lenguaje natural y las intrincadas de la intención del usuario.

Concluyendo esta discusión, **Peng et al. (2024)** elaboraron además sobre cómo GraphRAG mejora el rendimiento de la recuperación al incorporar incrustaciones semánticas derivadas de estructuras de grafos. Esta integración permite una comprensión más matizada de las relaciones dentro de los datos,

mejorando la calidad de la recuperación de información para tareas posteriores [1]. Sus ideas refuerzan el papel crítico de las incrustaciones semánticas en el avance de las capacidades de los sistemas RAG.

3.2. Recuperación

La recuperación es un componente crítico en los sistemas de Generación Aumentada por Recuperación en Grafos (Graph RAG), determinando la relevancia y precisión de la información extraída de la base de conocimiento para el proceso de generación. A diferencia de los sistemas RAG tradicionales que recuperan texto no estructurado, los Graph RAG aprovechan las propiedades estructurales de los grafos, recuperando no solo información textual sino también datos relacionales incrustados dentro de nodos, aristas y subgrafos. Esto permite respuestas más matizadas, esenciales para aplicaciones que requieren una comprensión contextual profunda y conocimiento interconectado. Métodos de recuperación eficientes son esenciales para asegurar que el sistema pueda procesar efectivamente estructuras de grafo grandes y complejas mientras mantiene la precisión y velocidad de las respuestas.

La etapa de procesamiento de consultas en los sistemas Graph RAG está diseñada para mapear la entrada de lenguaje natural del usuario a estructuras de grafo relevantes. Esto implica transformar la entrada en una consulta que pueda ser entendida por la base de datos de grafos subyacente. Técnicas como algoritmos de recorrido de grafos, enlace de entidades y coincidencia semántica son frecuentemente empleadas para extraer nodos, aristas o subgrafos relevantes para la consulta [1]. La estructura única de los grafos permite métodos de recuperación que pueden considerar tanto relaciones directas como indirectas entre entidades, proporcionando así un contexto más rico para el modelo de generación.

Por ejemplo, **He et al. (2024)** proponen un enfoque de recuperación de k-vecinos más cercanos (k-NN) que identifica nodos y aristas relevantes midiendo la similitud del coseno entre la consulta y las incrustaciones de los elementos del grafo. Este método recupera los k nodos y aristas más relevantes, asegurando que la consulta se mapee a estructuras y relaciones de grafo significativas, proporcionando al sistema datos precisos para tareas de generación subsiguientes [8]. De manera similar, **Hu et al. (2024)** demuestran cómo WeKnow aprovecha el conocimiento estructurado de los grafos para mejorar la precisión y relevancia de la información recuperada por los modelos de lenguaje, especialmente para consultas complejas que requieren comprensión relacional. Proponen un modelo RAG específico de dominio basado en un mecanismo de scraping, haciendo de la base de conocimiento del grafo el dominio desde el cual extraer información relevante. Este enfoque facilita el ensamblaje de una base de conocimiento más grande con menos recursos como memoria o poder de cómputo [9].

En el ámbito de la verificación de hechos, los procesos de recuperación demandan aún mayor precisión y conciencia contextual. **Chang et al. (2024)** contribuyen a mejorar los procesos de recuperación integrando grafos de conocimiento (KGs) con sistemas RAG usando comunidades. Introducen un mecanismo de recuperación consciente de la comunidad que aprovecha las vías de múltiples saltos dentro de los KGs para mejorar la precisión de la información recuperada. Esto realza la relevancia y exactitud contextual de los datos recuperados para tareas de verificación de hechos, superando los métodos de recuperación tradicionales. Su sistema también funciona bien en configuraciones de aprendizaje sin supervisión (zero-shot), adaptándose a nuevas consultas sin ajustes adicionales, lo que lo hace altamente escalable [7].

Debido al tamaño potencialmente grande de las bases de datos de grafos, especialmente en aplicaciones del mundo real, son necesarias estrategias de optimización para mejorar la eficiencia de la recuperación.

Técnicas como la indexación de grafos, la búsqueda aproximada de vecinos más cercanos y métodos de recuperación de múltiples etapas se utilizan para reducir el espacio de búsqueda y disminuir la carga computacional. En Graph RAGs, el equilibrio entre la precisión de la recuperación y la velocidad es crucial, particularmente en aplicaciones que requieren respuestas en tiempo real. Los métodos de recuperación híbridos, que combinan técnicas paramétricas (por ejemplo, modelos neuronales) y no paramétricas, son cada vez más populares para equilibrar el rendimiento con la escalabilidad.

Para abordar estos desafíos, **He et al. (2024)** introducen un método que construye subgrafos basados en el algoritmo de Árbol de Steiner con Recompensas (Prize-Collecting Steiner Tree, PCST). Este método asigna puntuaciones de relevancia (o “recompensas”) a nodos y aristas, construyendo un subgrafo que maximiza la relevancia mientras minimiza el costo de recuperación. La adaptación de PCST para nodos y aristas permite una construcción eficiente de subgrafos que mantienen un tamaño manejable sin sacrificar datos relacionales importantes [8]. Esta estrategia es particularmente efectiva para asegurar que solo se recupere información pertinente, reduciendo el ruido y la carga computacional.

Una de las limitaciones de los sistemas Graph RAG es el razonamiento de múltiples saltos, que implica llegar a respuestas que requieren varios pasos de inferencia dentro del grafo de conocimiento. **Fang et al. (2024)** proponen una solución a este problema creando cadenas de razonamiento basadas en el suelo —cadenas de nodos y aristas utilizadas para construir una respuesta a una consulta. Esencialmente, construyen una cadena de pensamiento donde, dado cualquier punto en la cadena, se puede avanzar, siendo el último nodo la respuesta a la consulta [10]. Este método mejora la capacidad del sistema para manejar consultas complejas que requieren la comprensión de múltiples relaciones.

En dominios especializados como la medicina, la necesidad de una recuperación precisa y eficiente es aún más pronunciada. **Cui et al. (2024)** contribuyen a mejorar la recuperación en sistemas RAG proponiendo el marco Bailicai. Destacan la importancia de refinar los procesos de recuperación para reducir el ruido de documentos irrelevantes y optimizar el tiempo de recuperación en grandes modelos de lenguaje (LLMs). Al utilizar estrategias adaptativas, como la identificación de límites de auto-conocimiento, Bailicai asegura que la recuperación solo se invoque cuando sea necesario, mejorando la eficiencia. Este enfoque aborda el desafío del ruido en los documentos y mejora la precisión en tareas de recuperación médica, mejorando significativamente la integración del conocimiento externo en los LLMs [5].

De manera similar, **Jin et al. (2024)** realizan contribuciones significativas a la recuperación en sistemas RAG introduciendo el marco RAGCache. Se enfocan en optimizar los procesos de recuperación mediante el almacenamiento en caché de estados intermedios de conocimiento, reduciendo efectivamente los cálculos redundantes. Al aprovechar estrategias de almacenamiento en caché dinámico multinivel, RAGCache mejora la eficiencia de la recuperación, especialmente para solicitudes de documentos de alta frecuencia, que se almacenan en caché para minimizar la latencia de recuperación. El estudio también discute los cuellos de botella de rendimiento en los pasos de recuperación en RAG y cómo el almacenamiento en caché de resultados intermedios puede reducir drásticamente los costos de recuperación y computacionales [6].

En el campo médico, **Wu et al. (2024)** introducen un método U-retrieve, que mejora la eficiencia de recuperación de los sistemas RAG en aplicaciones médicas. U-retrieve equilibra la conciencia global y la eficiencia de indexación generando resúmenes en diferentes niveles del grafo y realizando coincidencias de arriba hacia abajo. Esto asegura que el sistema recupere entidades altamente relevantes y sus

relaciones a través de múltiples capas de conocimiento médico. Su método permite respuestas precisas y basadas en evidencia a consultas médicas, mejorando dramáticamente la confiabilidad de los modelos de lenguaje aumentados por recuperación en escenarios médicos críticos [11].

Un componente esencial que mejora la recuperación en los sistemas Graph RAG es la utilización de Redes Neuronales de Grafos (GNNs). Las GNNs son una clase de redes neuronales diseñadas para procesar datos estructurados en grafos capturando las dependencias entre nodos a través de mecanismos de paso de mensajes. Extienden las redes neuronales tradicionales al incorporar la topología de los grafos en el proceso de aprendizaje, lo que permite a los modelos aprender representaciones que reflejan tanto las características de los nodos individuales como su contexto estructural dentro del grafo.

Varias arquitecturas de GNN, como las Redes Neuronales Convolucionales de Grafos (GCNs), las Redes de Atención de Grafos (GATs) y las Redes Neuronales Recurrentes de Grafos (GRNNs), son particularmente relevantes para los sistemas RAG. Estas arquitecturas difieren en cómo agregan y propagan la información a través del grafo. Por ejemplo, las GATs aprovechan los mecanismos de atención para ponderar la importancia de los nodos vecinos, lo que puede mejorar el proceso de recuperación en los RAG al enfocarse en relaciones más relevantes.

Peng et al. (2024) destacan el papel de las GNNs en los sistemas GraphRAG, explicando que las GNNs se utilizan para representar estructuras de grafos y mejorar los procesos de recuperación y generación. Los recuperadores basados en GNN son particularmente adecuados para tareas que involucran relaciones complejas en datos estructurados en grafos [1]. En la práctica, las GNNs se utilizan para codificar bases de conocimiento en grafos en incrustaciones que pueden integrarse con modelos de lenguaje. Las aplicaciones incluyen la mejora del reconocimiento de entidades, la extracción de relaciones y la provisión de respuestas conscientes del contexto. Las GNNs permiten que el sistema comprenda interacciones complejas dentro del grafo, lo que conduce a una recuperación y generación de información más precisas en respuesta a las consultas de los usuarios.

Por ejemplo, **He et al. (2024)** utilizan una Red de Atención de Grafos (GAT) para codificar la respuesta del subgrafo de la fase de recuperación utilizando una operación de pooling, seguida de un perceptrón multicapa y, finalmente, utilizando un incrustador de texto para convertir la salida en forma textual. Esto resulta en una menor cantidad de tokens necesarios para realizar una operación de pregunta-respuesta, mejorando la eficiencia [8]. De manera similar, **Fang et al. (2024)** proponen el uso de una GNN en la fase de recuperación del sistema RAG para recuperar los tripletes más relevantes (cabeza, borde, cola) en el grafo de conocimiento, mejorando la capacidad del sistema para manejar consultas complejas [10].

El procesamiento eficiente de consultas es crítico para el rendimiento de los sistemas Graph RAG. Se emplean técnicas como algoritmos de recorrido de grafos, estrategias de indexación y búsqueda aproximada de vecinos más cercanos para recuperar información relevante rápidamente. Optimizar estos procesos asegura que el componente de recuperación no se convierta en un cuello de botella en la línea de generación. **Hu et al. (2024)** destacan la importancia de la consulta eficiente de grafos y el aprendizaje de representaciones, incrustando entidades y relaciones en un espacio compartido con el modelo de lenguaje para facilitar la interacción fluida entre datos estructurados y no estructurados [4].

Las estrategias de optimización en Graph RAG se centran en reducir la latencia y la sobrecarga computacional. Los métodos incluyen precomputar incrustaciones, usar mecanismos de caché y

emplear procesamiento paralelo. Además, la poda de partes irrelevantes del grafo y el aprovechamiento de estructuras jerárquicas pueden mejorar la eficiencia de la recuperación. **Peng et al. (2024)** exploran cómo GraphRAG mejora el procesamiento de consultas a través de técnicas como la expansión y descomposición de consultas. Estos métodos aseguran recuperaciones más precisas refinando la consulta original con contexto adicional o descomponiéndola en partes más pequeñas y manejables [1].

Construyendo sobre estas estrategias, **Cui et al. (2024)** contribuyen a la optimización de consultas dentro del contexto de los sistemas RAG al introducir mecanismos que determinan si una consulta médica requiere recuperación externa o puede resolverse utilizando el conocimiento interno del modelo. Su módulo de identificación de límites de conocimiento propio optimiza el proceso de recuperación al evitar la recuperación de datos externos innecesarios, reduciendo el tiempo computacional y el consumo de recursos. La combinación de este método con la descomposición de tareas asegura que cada consulta se maneje de manera eficiente, ya sea a través del conocimiento interno o accediendo a los documentos externos relevantes. Este proceso de recuperación dinámico y dirigido ofrece mejoras significativas en el procesamiento de consultas para aplicaciones médicas [5].

Jin et al. (2024) introducen varias optimizaciones clave para el procesamiento de consultas en sistemas RAG. Una de las principales contribuciones es el desarrollo de pipeline especulativo dinámico, que superpone las etapas de recuperación e inferencia para minimizar el tiempo de procesamiento de consultas. Esta técnica asegura que, a medida que se generan los resultados de recuperación, se utilicen inmediatamente en la inferencia, reduciendo la latencia. Además, su programación consciente de la caché y una política de reemplazo consciente del prefijo aseguran que los documentos recuperados con frecuencia sean rápidamente accesibles, mejorando el rendimiento general de las consultas. Se demuestra que este enfoque reduce significativamente el tiempo total de consulta, ofreciendo una solución escalable para sistemas que manejan grandes volúmenes de consultas complejas [6].

Además, **Huang et al. (2023)** introducen estrategias de optimización como Retriever como Clasificador de Respuestas (RAC) y Similitud de Conocimiento Denso (DKS), que se enfocan en recuperar pasajes que son relevantes en conocimiento. Esta optimización asegura que el sistema recupere los pasajes más significativos para una consulta dada, mejorando tanto el rendimiento de la recuperación como de la generación. Su trabajo mejora significativamente la eficiencia de las consultas al enfocarse en la coincidencia de conocimiento denso en lugar de depender únicamente de la coincidencia a nivel de tokens [12].

Además, **Chang et al. (2024)** optimizan el proceso de consulta al introducir un sistema de recuperación jerárquico y dirigido por la comunidad dentro de su marco de grafo de conocimiento. Este enfoque selecciona dinámicamente las comunidades y oraciones más relevantes basadas en su relación con la consulta, mejorando tanto la velocidad como la precisión de la verificación de hechos. La capacidad de cero disparo de su sistema significa que la optimización de consultas ocurre sin la necesidad de reentrenamiento, haciendo que la solución sea eficiente y escalable para diversas tareas de verificación de hechos y recuperación [7].

Wu et al. (2024) también introducen varias optimizaciones en el proceso de consulta a través de su mecanismo U-retrieve, que implica la coincidencia de consultas de arriba hacia abajo con estructuras basadas en grafos. Este enfoque minimiza el espacio de búsqueda al indexar y coincidir dinámicamente las consultas con nodos relevantes del grafo, permitiendo una recuperación más rápida y precisa. Además, el uso de estructuras de grafos jerárquicas permite que el sistema recupere información en

distintos niveles de detalle, asegurando que incluso las consultas médicas complejas sean manejadas de manera eficiente y efectiva [11].

Evaluar el rendimiento de las técnicas de recuperación en Graph RAGs implica evaluar tanto la precisión de los subgrafos recuperados como la eficiencia del proceso de recuperación. Se utilizan métricas como precisión, recall y F1-score para medir la relevancia de la recuperación. Además, se consideran métricas específicas relacionadas con la recuperación basada en grafos, como la calidad del subgrafo y la relevancia de los caminos. Las métricas de velocidad, incluyendo la latencia de las consultas y el tiempo de recuperación, también son críticas para aplicaciones donde se requiere generación en tiempo real. El benchmarking en conjuntos de datos estándar ayuda a asegurar que los métodos de recuperación puedan generalizarse a varios dominios mientras mantienen el rendimiento.

En conclusión, la recuperación en los sistemas Graph RAG es un proceso multifacético que se beneficia de la integración de técnicas avanzadas como las GNNs, estrategias de optimización y métodos eficientes de procesamiento de consultas. Las contribuciones colectivas de varios investigadores destacan los avances continuos en este campo, con el objetivo de mejorar la precisión, eficiencia y escalabilidad de los procesos de recuperación. Estas mejoras son esenciales para el desarrollo de sistemas Graph RAG sofisticados capaces de manejar aplicaciones complejas del mundo real, como servicios de planificación y asesoramiento financiero.

3.3. Generación

En los sistemas Graph RAG, el componente de generación es responsable de producir salidas en lenguaje natural que están informadas por los datos del grafo recuperados. Esto implica integrar la información estructurada del grafo de conocimiento en el proceso de generación del modelo de lenguaje. Técnicas como la generación condicional, donde el modelo condiciona su salida en los datos recuperados, son comúnmente empleadas. La fusión de recuperación y generación permite que los modelos produzcan salidas que están enriquecidas contextualmente y son factualmente consistentes con la base de conocimiento externa.

El análisis semántico juega un papel crucial en esta integración al convertir el lenguaje natural en representaciones estructuradas comprensibles por máquinas. Los Graph RAG contribuyen a este proceso al recuperar e incorporar estructuras de grafos relevantes que reflejan el significado semántico de la entrada. Esta integración permite una interpretación más precisa de las consultas y la generación de respuestas que están alineadas con la base de conocimiento subyacente. Al mapear las entradas de los usuarios a entidades y relaciones específicas dentro del grafo, el sistema puede generar salidas que son tanto semánticamente precisas como contextualmente relevantes.

Las representaciones basadas en RAG mejoran los modelos de NLP al incrustar el conocimiento recuperado directamente en el proceso de generación. Se exploran técnicas como el entrenamiento conjunto y los mecanismos de atención sobre el contenido recuperado para maximizar la sinergia entre los componentes de recuperación y generación. Por ejemplo, los mecanismos de atención permiten que el modelo se enfoque en las partes más relevantes de los datos del grafo recuperados durante la generación, mejorando la coherencia y relevancia de la salida. Esta fusión asegura que las respuestas generadas no solo sean fluidas sino también fundamentadas en la base de conocimiento externa.

Además, se emplean modelos avanzados como las arquitecturas basadas en Transformer en el paso de generación para manejar patrones de lenguaje y dependencias complejas. Modelos de lenguaje preentrenados como GPT-3 y BERT pueden ser ajustados para incorporar conocimiento estructurado en

grafos, mejorando su capacidad para generar texto contextualmente rico y factualmente preciso. Herramientas y bibliotecas como Hugging Face Transformers proporcionan marcos para integrar modelos de recuperación y generación, facilitando el desarrollo de sistemas Graph RAG sofisticados.

Peng et al. (2024) abordaron las limitaciones de los modelos tradicionales de NLP al aprovechar estructuras de grafos para recuperar conocimiento relacional. Demostraron que GraphRAG supera a los sistemas RAG convencionales al incorporar tanto información textual como estructural, lo que permite respuestas más conscientes del contexto [1]. Su trabajo resalta la importancia de integrar la recuperación basada en grafos con técnicas avanzadas de generación para mejorar el rendimiento general de los modelos de lenguaje.

Además de estos enfoques, se emplean técnicas de generación de texto a partir de grafos para traducir datos estructurados en grafos a lenguaje natural. Esto implica el uso de modelos que pueden interpretar los nodos y bordes de un grafo y generar descripciones o explicaciones textuales coherentes. Las Redes Neuronales de Grafos (GNNs) juegan un papel significativo en este proceso al codificar estructuras de grafos en incrustaciones que capturan tanto información semántica como relacional. Estas incrustaciones son luego utilizadas por modelos de secuencia a secuencia para generar la salida textual final [1], para mejorar las capacidades de modelado para datos de grafos, mejorando así el rendimiento en tareas posteriores como la clasificación de nodos, la predicción de bordes, la clasificación de grafos y otras.

La evaluación del componente de generación en los sistemas Graph RAG es crucial para asegurar la calidad y fiabilidad de las respuestas. Métricas como BLEU, ROUGE y METEOR se utilizan comúnmente para evaluar la fluidez y relevancia del texto generado. Además, se emplean métodos de evaluación especializados para medir la precisión factual y la consistencia con el grafo de conocimiento, asegurando que el contenido generado refleje con exactitud los datos subyacentes [1].

El paso de generación también implica manejar desafíos como controlar el nivel de detalle y gestionar ambigüedades en los datos recuperados. Se exploran técnicas como la generación controlada y el refinamiento de respuestas para permitir que el sistema ajuste la especificidad de la salida según las necesidades del usuario. Esto es particularmente importante en aplicaciones como la planificación financiera o los servicios de asesoramiento, donde la información precisa y exacta es primordial.

En resumen, el componente de generación de los sistemas Graph RAG es un aspecto crítico que determina la efectividad del sistema en su conjunto. Al integrar técnicas avanzadas de generación con conocimiento de grafo recuperado, los Graph RAG pueden producir respuestas que no solo son contextualmente relevantes sino también factualmente precisas. La investigación continua sigue explorando nuevos métodos y herramientas para mejorar aún más las capacidades de generación de estos sistemas, ampliando su aplicabilidad en diversas tareas de NLP."

3.4. Aplicaciones y Estudios de RAG No Financieras

Los sistemas de Generación Aumentada con Recuperación de Grafos (Graph RAG) han encontrado aplicaciones significativas más allá del dominio financiero, particularmente en campos que requieren la integración de conocimientos complejos y estructurados. Una área prominente es la salud, donde los Graph RAG ayudan en el soporte de decisiones clínicas al recuperar conocimientos médicos relevantes en respuesta a los datos de los pacientes. Permiten planes de tratamiento personalizados al considerar relaciones complejas entre síntomas, enfermedades y tratamientos. Esta aplicación requiere el manejo de datos sensibles y garantizar el cumplimiento de las regulaciones de privacidad.

Cui et al. (2024) demuestran que el marco Bailicai mejora el rendimiento de los Modelos de Lenguaje Grande (LLMs) en tareas médicas. Al integrar la inyección de conocimiento médico y la identificación de límites de auto-conocimiento, el marco mitiga eficazmente las alucinaciones, un problema común en la generación de textos médicos. Bailicai supera a modelos propietarios como GPT-3.5 y logra un rendimiento de vanguardia en varios benchmarks médicos. Los investigadores se enfocan en curar conjuntos de datos del dataset UltraMedical, asegurando que el modelo aprenda de datos médicos de alta calidad. La capacidad del marco para manejar tareas de preguntas y respuestas médicas con precisión lo convierte en un avance significativo en aplicaciones de IA médica [5].

Basándose en estos avances, **Wu et al. (2024)** proponen un marco de Generación Aumentada con Recuperación basado en grafos específicamente diseñado para aplicaciones médicas, llamado MedGraphRAG. Este marco construye un grafo jerárquico a partir de documentos médicos, libros de texto y diccionarios médicos autorizados, mejorando significativamente la capacidad de los LLMs para procesar y recuperar información médica. Al emplear una estructura basada en grafos, el sistema preserva relaciones complejas entre entidades médicas, proporcionando una precisión de recuperación superior en comparación con bases de datos planas o sistemas de recuperación simples de clave-valor. La naturaleza jerárquica del grafo permite una vinculación y recuperación más precisa, asegurando que las respuestas generadas estén basadas en evidencia y sean contextualmente precisas [11].

El enfoque principal del trabajo de Wu et al. es mejorar el rendimiento de los LLMs en el dominio médico. Su sistema MedGraphRAG aborda las necesidades específicas de los profesionales médicos al garantizar que las respuestas generadas estén respaldadas por evidencia de fuentes médicas creíbles, como libros de texto y artículos revisados por pares. La estructura jerárquica del grafo vincula documentos proporcionados por el usuario con conocimientos médicos establecidos, asegurando transparencia e interpretabilidad en diagnósticos y recomendaciones médicas. El rendimiento de MedGraphRAG en múltiples benchmarks de preguntas y respuestas médicas, incluidos PubMedQA, MedMCQA y USMLE, demuestra su efectividad, superando incluso a modelos ajustados finamente en la generación de información médica precisa y basada en evidencia [11].

Más allá de la salud, los Graph RAG mejoran el **análisis de redes sociales** al generar insights basados en la estructura y dinámica de las interacciones sociales. Pueden identificar nodos influyentes, detectar comunidades y predecir la evolución de relaciones. Esta información es valiosa para estrategias de marketing, análisis de tendencias y comprensión de fenómenos sociales. Al analizar grafos sociales complejos, las organizaciones pueden aprovechar los Graph RAG para tomar decisiones basadas en datos en entornos sociales que cambian rápidamente.

En el campo de los **Sistemas de Información Geográfica (GIS)**, los Graph RAG facilitan la recuperación de datos espaciales y relacionales para aplicaciones como la planificación de rutas, la gestión de recursos y el monitoreo ambiental. Al integrar diversos conjuntos de datos geoespaciales, los sistemas RAG pueden generar análisis comprensivos y apoyar procesos de toma de decisiones que consideran múltiples factores geográficos. El enfoque basado en grafos permite modelar relaciones y dependencias espaciales intrincadas, mejorando la precisión y utilidad de los insights generados en aplicaciones de GIS.

La versatilidad de los Graph RAG se demuestra además en aplicaciones específicas de dominio que requieren una recuperación de información estructurada y confiable. **He et al. (2024)** representan un avance significativo en esta área al crear un marco de preguntas y respuestas para un sistema Graph RAG. Al allanar el camino para chatbots específicos de dominio más estructurados y menos propensos a

alucinaciones, su trabajo aborda desafíos comunes en aplicaciones de chatbots, como mantener el contexto y asegurar la exactitud factual de las respuestas [8].

En conclusión, los sistemas Graph RAG están haciendo contribuciones impactantes en diversos dominios no financieros. Su capacidad para integrar conocimientos complejos y estructurados en modelos de lenguaje permite salidas más precisas, ricas en contexto y confiables. Ya sea en salud, análisis de redes sociales, GIS o desarrollo de chatbots especializados, los Graph RAG proporcionan herramientas poderosas para avanzar en aplicaciones de IA y apoyar procesos de toma de decisiones sofisticados.

3.5. Aplicaciones y Estudios de RAG en Finanzas

El sector financiero ha adoptado cada vez más los sistemas de Generación Aumentada con Recuperación (RAG) para mejorar diversos servicios, incluyendo la gestión de la calidad de los datos, el asesoramiento al cliente y la planificación financiera. Estos sistemas aprovechan los Modelos de Lenguaje Grande (LLMs) y las bases de conocimiento externas para mejorar la precisión, relevancia y personalización. Aunque se ha logrado un progreso significativo, aún existe una brecha en la aplicación de sistemas **Graph RAG** específicamente para servicios de planificación o asesoramiento financiero. Esta sección revisa estudios existentes en el dominio y destaca la necesidad de una mayor investigación que integre Graph RAG en el asesoramiento financiero.

Mejorando la Gestión de la Calidad de los Datos con RAG

Raja (2024) desarrolló un marco de Generación Aumentada con Recuperación para mejorar la gestión de la calidad de los datos en el sector financiero, enfocándose en la validación de datos contextuales [13]. Los datos financieros presentan desafíos únicos debido a los estrictos requisitos regulatorios, de seguridad y de consistencia. Las herramientas tradicionales a menudo carecen de la conciencia contextual necesaria para abordar estos problemas de manera efectiva. El marco de Raja aprovecha los LLMs junto con grafos de conocimiento para mejorar la precisión de los procesos de calidad de los datos.

Los componentes clave del marco incluyen:

1. *Web Scraping para la Recolección de Datos*

- Utilizó herramientas como **Scrapy** y **BeautifulSoup** para extraer contenido financiero relevante de blogs y artículos disponibles públicamente.
- Limpió y estructuró los datos recolectados en documentos PDF para su análisis.

2. *Incrustaciones y Arquitectura del Modelo*

- Generó incrustaciones de oraciones utilizando **all-mpnet-base-v2** de Hugging Face, mapeando el texto en un espacio de 768 dimensiones para una mejor comprensión semántica.
- Empleó una tienda de índices vectoriales para una recuperación de consultas más rápida y una mejor contextualización.

3. *Arquitectura y Configuración del Sistema*

- Integró **Llama 2** con modelos preentrenados que consisten en 7 mil millones de parámetros para mejorar la calidad de las respuestas.
- Desarrolló un motor de consultas que aprovecha la tienda vectorial indexada para proporcionar respuestas conscientes del contexto a consultas relacionadas con datos financieros.

El marco fue evaluado utilizando tres casos de uso relevantes para la calidad de los datos financieros:


1. *Verificación de Oportunidad*

- **Consulta:** “¿Cómo verificar si los datos de acciones están obsoletos?”
- **Resultado:** Proporcionó recomendaciones detalladas para evaluar datos de acciones obsoletos, yendo más allá de las comprobaciones tradicionales de marca de tiempo.

2. *Verificación de Completitud*

- **Consulta:** “¿Qué estrategias debemos usar si faltan datos de ingresos de los clientes?”
- **Resultado:** Sugirió estrategias conscientes del contexto como la imputación por regresión y explicó las consecuencias de varias técnicas de imputación.

3. *Verificación de Consistencia*

- **Consulta:** “¿Son válidos estos símbolos (“\$”, “€”, “&&”, “) como monedas?”
- **Resultado:** Recomendó la validación de monedas basada en ISO, superando a los métodos basados en regex utilizados por SQL Server DQS.

El enfoque basado en RAG demostró proporcionar insights accionables no alcanzables con las herramientas estándar de calidad de datos, mostrando alta relevancia y recall en las respuestas. Sin embargo, se identificaron desafíos relacionados con la escalabilidad debido a restricciones de memoria y limitaciones en el conocimiento específico del dominio. Las mejoras futuras incluyen el entrenamiento del modelo con conjuntos de datos del mundo real provenientes de registros de Jira y entradas de Notion para mejorar el rendimiento.

Aunque el estudio de Raja muestra el potencial de los marcos RAG para mejorar los procesos de validación de datos en finanzas, se centra principalmente en la gestión de la calidad de los datos en lugar de los servicios de asesoramiento financiero.

Sistemas Multi-Agente y RAG en Servicios Financieros

Sepanosian et al. (2024) presentan un marco orientado a la industria para guiar la adopción escalable de IA en instituciones financieras [14]. Su trabajo incluye una implementación de prueba de concepto para la planificación de jubilación utilizando sistemas multi-agente y tecnologías RAG. El marco categoriza las soluciones de IA en áreas problemáticas como la recuperación de información, el análisis de datos, la generación de contenido, el soporte de decisiones y la automatización de tareas. Central en el marco está la tecnología RAG, asegurando el uso de información actualizada y precisa sin necesidad de reentrenamiento constante del modelo.

La prueba de concepto utiliza **crewAI**, un marco de orquestación multi-agente, desplegando cuatro agentes especializados:

1. *Experto en Políticas de Jubilación*

- Proporciona información sobre límites de contribución y regulaciones.

2. *Experto en la Industria*

- Ofrece insights sobre tendencias de inversión y ahorros promedio.

3. *Experto en Asesoramiento*

- Agrega información y ofrece consejos personalizados.

4. *Agente de Aseguramiento de Calidad (QA)*

- Asegura la precisión mediante la referencia cruzada de información y la señalización de inconsistencias.

Estos agentes interactúan secuencialmente, utilizando herramientas RAG y funcionalidades de LangChain para recuperar datos relevantes de manera dinámica. Las pruebas con un cliente ficticio demostraron la capacidad de los agentes para generar informes personalizados que comparan los ahorros para la jubilación del cliente con los promedios de la industria. Los agentes identificaron que las inversiones del cliente estaban desempeñándose de manera competitiva, aconsejaron sobre posibles contribuciones adicionales y proporcionaron insights regulatorios basados en actualizaciones recientes.

A pesar de los resultados valiosos, surgieron desafíos relacionados con la explicabilidad y las salidas no determinísticas inherentes a los LLMs. El agente QA a menudo requería revisión manual debido a salidas inconsistentes o ambiguas. También se notaron riesgos de seguridad, particularmente al utilizar herramientas externas como la API de OpenAI, lo que introduce preocupaciones sobre la seguridad de los datos. Los desafíos de escalabilidad incluían la sobrecarga de preprocesamiento para tipos de datos complejos y llamadas redundantes a la API que reducían el rendimiento.

Aunque este estudio demuestra el potencial de los LLMs y las herramientas RAG en aplicaciones de asesoramiento financiero personalizado, no emplea específicamente sistemas **Graph RAG**. El enfoque permanece en sistemas multi-agente y el uso de RAG para la recuperación de información, sin aprovechar las ventajas estructurales de la representación de conocimiento basada en grafos.

Integrando LLMs con Modelos de Planificación Financiera

De Zarzà et al. (2024) proponen un innovador marco de planificación financiera que combina modelos tradicionales de presupuestación con LLMs potenciados por IA para mejorar la gestión financiera tanto a nivel individual como cooperativo (hogar) [15]. El objetivo principal es democratizar la planificación financiera proporcionando recomendaciones personalizadas que optimicen el ahorro y la presupuestación a través de herramientas de IA accesibles.

El estudio se enfoca en dos modelos de planificación financiera interconectados:

1. *Modelo de Optimización de Presupuesto Individual*

- **Objetivo:** Maximizar el ahorro distribuyendo eficientemente el ingreso mensual entre categorías de gastos (por ejemplo, alquiler, comestibles, servicios).
- **Función de Utilidad:** Prioriza el ahorro mientras se cubren los gastos necesarios, definida como:

$$U(I, E) = \alpha \log(S) - \beta \sum_{o=1}^n w_o \log(E_o)$$

donde I es el ingreso, S el ahorro, E los gastos categorizados, y w los pesos de preferencias personalizadas.

- **Recomendaciones Generadas por LLM:** Sirven como una línea base orientadora para sugerir asignaciones factibles.

2. *Modelo de Presupuestación Cooperativa para Hogares*

- **Extensión del Modelo Individual:** Aborda las responsabilidades financieras compartidas entre los miembros del hogar.

- **Equidad Financiera:** Considera las prioridades individuales mientras optimiza el ahorro total:

$$TS = \sum_{o=1}^n \left(I_o - \sum_z E_{zo} \right)$$

donde TS es el ahorro total del hogar, I_o es el ingreso de cada miembro, y E_{zo} son sus respectivos gastos por categoría z .

- **Recomendaciones Informadas por LLM:** Ofrecen consejos sobre planificación presupuestaria compartida eficiente y proponen estrategias para minimizar gastos redundantes.

La integración de LLMs ofrece varias ventajas:

- **Personalización:** Adapta las recomendaciones basándose en datos específicos individuales y del hogar.
- **Adaptabilidad Dinámica:** Ajusta las recomendaciones para reflejar tendencias económicas y objetivos financieros en evolución.
- **Estrategias de Ahorro Mejoradas:** Incluye consejos sobre la creación de fondos de emergencia, ahorros para la jubilación y gestión de prioridades a corto y largo plazo.
- **Suavización del Consumo:** Utiliza principios de la teoría de juegos cooperativos para mantener estándares de vida consistentes a lo largo del tiempo.

El estudio evalúa escenarios utilizando ambos modelos, individual y cooperativo, guiados por las recomendaciones de LLM. Los insights clave incluyen:

1. *Presupuestación a Corto Plazo*

- El LLM sugiere asignaciones para fondos de emergencia y prioriza el pago de deudas.

2. *Planificación para la Jubilación*

- En un escenario de hogar, los LLMs guían a los miembros para optimizar el ahorro, equilibrando las contribuciones para la jubilación con los gastos diarios.

3. *Objetivos Financieros a Largo Plazo*

- Recomienda establecer un fondo para niños para acomodar gastos futuros como educación y atención médica.

Para modelar la interacción de los agentes financieros como un sistema adaptativo, se emplea la teoría coevolutiva extendida (EC). Los agentes ajustan iterativamente su gasto basado en las recomendaciones de LLM y su entorno financiero. Para un agente o en el tiempo t :

$$E_o(t+1) = (1 - \beta) \left(E_o(t) + \alpha \nabla U_o(E_{o(t)}, E_{-o}(t)) \right) + \beta R_{o,t}$$

donde $R_{o,t}$ es la recomendación del LLM, y α y β controlan el aprendizaje y la influencia del LLM.

Aunque la integración de LLMs produce planes de presupuesto optimizados que se alinean con las mejores prácticas en finanzas personales, la supervisión humana es esencial para validar las sugerencias del LLM y evitar posibles errores. Similar a los estudios anteriores, esta investigación no incorpora sistemas **Graph RAG**. Las recomendaciones son generadas por LLMs sin el uso explícito de bases de conocimiento estructuradas en grafos, lo que potencialmente limita la capacidad del sistema para capturar relaciones financieras complejas.

3.6. La Necesidad de Graph RAG en la Planificación y Asesoramiento Financiero

Los estudios revisados ilustran el creciente interés en aplicar sistemas RAG y LLMs dentro del sector financiero, incluyendo la gestión de la calidad de los datos, la planificación de la jubilación y la presupuestación. Sin embargo, ninguno de los estudios explora específicamente el uso de sistemas **Graph RAG** para servicios de planificación o asesoramiento financiero.

Los sistemas Graph RAG aprovechan las propiedades estructurales de los grafos para representar relaciones y dependencias complejas inherentes a los datos financieros. Al recuperar e incorporar datos relacionales incrustados dentro de nodos, bordes y subgrafos, los Graph RAG pueden proporcionar recomendaciones más matizadas y ricas en contexto. Esto es particularmente valioso en servicios de planificación y asesoramiento financiero, donde comprender relaciones intrincadas entre productos financieros, tendencias de mercado, requisitos regulatorios y perfiles individuales de clientes es crucial.

Integrar sistemas Graph RAG en el asesoramiento financiero podría mejorar la personalización y precisión de las recomendaciones, permitir razonamiento de múltiples saltos sobre grafos de conocimiento financiero y mejorar la capacidad del sistema para manejar consultas complejas. Además, el uso de representaciones de conocimiento basadas en grafos puede facilitar un mejor cumplimiento de los estándares regulatorios al proporcionar insights transparentes e interpretables.

3.7. Resumen

Aunque se han logrado avances significativos en la aplicación de sistemas RAG y LLMs dentro del sector financiero, aún existe una brecha en la utilización de sistemas **Graph RAG** específicamente para servicios de planificación y asesoramiento financiero. Los beneficios potenciales de Graph RAG, incluyendo el mejor manejo de relaciones financieras complejas y la personalización mejorada, destacan la necesidad de una mayor investigación en esta área. Esta tesis tiene como objetivo abordar esta brecha investigando la integración de sistemas Graph RAG en la planificación y asesoramiento financiero, explorando su potencial para revolucionar la prestación de servicios financieros.

4. Implementación

4.1. Recolección y preprocesamiento de datos

El primer paso en la metodología implica la recolección y el preprocesamiento de datos financieros de diversas fuentes para construir una base de conocimiento completa. Estos datos abarcan varios conceptos financieros, productos, regulaciones y tendencias del mercado relevantes para el dominio del asesoramiento. Actualmente existen varios conjuntos de datos disponibles para datos financieros, como informes estructurados, fuentes de datos de mercado y documentos regulatorios. Sin embargo, para este caso de uso particular, no nos centraremos en dichos datos, ya que pueden causar ruido entre la información relevante requerida para el conjunto de datos. En su lugar, nos centraremos en la creación de un conjunto de datos personalizado que se adapte al dominio del asesoramiento financiero.

Para lograr ese objetivo, nuestra herramienta principal es el web scraping, que nos permite extraer información de sitios web financieros, blogs, foros y organismos reguladores. En este caso particular, nos centraremos en la extracción de información de blogs financieros, ya que proporcionan una rica fuente de contenido actualizado y relevante, con especial interés en el sitio web <https://www.investopedia.com/>. Este sitio web es una plataforma de educación financiera muy conocida que cubre una amplia gama de temas, incluyendo inversiones, comercio, finanzas personales y economía. Al extraer contenido de Investopedia, podemos recopilar un conjunto diverso de conceptos financieros, definiciones y explicaciones que suelen buscar los usuarios que buscan asesoramiento financiero.

Justificación: El web scraping de Investopedia garantiza que los datos sean actuales, relevantes y completos, lo cual es crucial para proporcionar un asesoramiento financiero preciso. Este enfoque también permite la creación de un conjunto de datos personalizado que se adapte específicamente a las necesidades del dominio del asesoramiento financiero.

Los datos extraídos se preprocesan para eliminar contenido irrelevante, estandarizar formatos y anotar entidades y relaciones. Este paso de preprocesamiento implica técnicas de procesamiento del lenguaje natural como la tokenización, la lematización y el reconocimiento de entidades para garantizar que los datos estén estructurados y semánticamente enriquecidos. Para este proceso, utilizaremos NLTK, que es una biblioteca de Python de PNL de uso general. El objetivo principal de este paso es crear un conjunto de datos limpio y estructurado que pueda utilizarse eficazmente para la construcción de grafos y la recuperación de información.

Justificación: El uso de NLTK para el preprocesamiento garantiza que los datos estén limpios, estructurados y semánticamente enriquecidos, lo cual es esencial para la construcción precisa de grafos y la recuperación de información.

4.2. Construcción del grafo

El núcleo de la metodología reside en la construcción de una representación del conocimiento basada en grafos que capture las intrincadas relaciones entre los conceptos financieros. Este grafo se construye utilizando los datos preprocesados, donde cada concepto financiero se representa como un nodo y las relaciones entre conceptos se representan como aristas. La estructura del grafo permite un recorrido eficiente, una recuperación sensible al contexto y una comprensión semántica del dominio financiero.

Este paso no se realizó manualmente, sino utilizando una herramienta muy útil llamada Langchain, en particular LangGraph, que es una colección de herramientas para construir y consultar grafos de conocimiento a partir de datos de texto. LangGraph utiliza modelos de procesamiento del lenguaje

natural de última generación para extraer entidades, relaciones y jerarquías de texto no estructurado, y luego construye una base de datos de grafos que puede consultarse para la recuperación de información. Al aprovechar LangGraph, podemos automatizar el proceso de construcción del grafo y garantizar la escalabilidad y la precisión del grafo de conocimiento resultante. Cabe destacar que Langchain es un conjunto de herramientas que funciona con modelos LLM de terceros, pero no los proporciona por defecto, lo que significa que el usuario debe proporcionar claves API para los modelos que desea utilizar. Personalmente, utilicé el modelo GPT-4o-mini, que es un modelo de propósito general y relativamente rápido creado por OpenAI para esta tarea.

Justificación: Automatizar el proceso de construcción del grafo con Langchain garantiza la escalabilidad y la precisión, mientras que el uso del modelo GPT-4o-mini proporciona una extracción de entidades y una asignación de relaciones eficiente y precisa.

La forma en que funciona es:

Proporcionar un conjunto limpio de documentos (en este caso, artículos de blogs financieros de Investopedia después del preprocesamiento).

Pasar esos documentos a través del pipeline de LangGraph, utilizando un enfoque de Herramienta o de Prompt. El enfoque de herramienta utiliza el enfoque de salida estructurada subyacente del modelo, haciendo que la respuesta se produzca en formato json con los nodos y aristas del grafo. El enfoque de prompt es semánticamente el mismo, pero a veces difiere en la salida, en lugar de utilizar la salida estructurada, utiliza un prompt para modelar la respuesta del sistema, un prompt que puede ser personalizado por el usuario.

Después de ese procesamiento, el sistema recopila todas las entidades encontradas en todos los documentos, así como las entidades entre ellas. Esto es particularmente útil, ya que permite al sistema crear relaciones entre entidades que no están directamente relacionadas en el mismo documento, y también utiliza el poder de un modelo de lenguaje para crear un grafo más preciso y completo.

Luego podemos construir el grafo utilizando una base de datos de grafos, en este caso utilizaremos Neo4j, que es una base de datos de grafos popular que permite el almacenamiento, la consulta y la visualización eficientes de datos de grafos. Las entidades se almacenan como nodos en el grafo, y las relaciones se almacenan como aristas, formando una representación conectada y estructurada de la base de conocimiento financiero.

Justificación: El uso de Neo4j para el almacenamiento y la consulta de grafos garantiza una gestión eficiente y escalable del grafo de conocimiento, mientras que el enfoque de salida estructurada de LangGraph garantiza una construcción precisa y completa del grafo.

El grafo resultante es un grafo textual, lo que significa que las relaciones y los nodos son texto puro, no se les añaden propiedades adicionales.

4.3. Incrustación de vectores

Para permitir la recuperación y generación eficiente de asesoramiento financiero, la metodología incorpora técnicas de incrustación de vectores para representar datos textuales en un espacio vectorial continuo. Este paso es crucial para calcular similitudes semánticas entre las consultas de los usuarios y los nodos del grafo, así como para generar respuestas contextualmente relevantes utilizando modelos de lenguaje.

Para esta tarea, utilizaremos un modelo de Hugging Face en forma de Sentence Transformer, que es un modelo basado en transformadores que mapea oraciones a vectores de alta dimensión en un espacio semántico. Esto nos permite codificar tanto las consultas de los usuarios como los nodos del grafo como vectores, lo que permite cálculos de similitud rápidos y precisos.

Justificación: El uso de Sentence Transformers de Hugging Face garantiza una alta precisión y rendimiento en los cálculos de similitud semántica, lo cual es esencial para la recuperación precisa de información y la generación de respuestas.

El proceso de incrustación de vectores implica la codificación de cada nodo y arista del grafo como una representación vectorial, que luego se almacena en una base de datos vectorial para su recuperación eficiente (MongoDB), cada uno en su propia colección (arista y nodo). Cuando se recibe una consulta de usuario, también se codifica como un vector utilizando el mismo modelo Sentence Transformer, y se recuperan los nodos y aristas más similares del grafo basándose en la similitud del coseno.

Justificación: Almacenar incrustaciones de vectores en MongoDB garantiza una recuperación eficiente y escalabilidad, mientras que el uso de la similitud del coseno para la coincidencia garantiza una recuperación de información precisa y relevante.

Para fines de prueba, también realizaremos una incrustación de vectores para cada documento en su conjunto.

4.4. Responder a una consulta

El mecanismo de recuperación y generación del sistema está diseñado para recuperar eficientemente conceptos y relaciones financieras relevantes basándose en las consultas de los usuarios. Este proceso consta de varios pasos clave: procesamiento de consultas, cálculo de similitud vectorial, extracción de subgrafos y generación de respuestas.

Como se mencionó anteriormente, la consulta se limpia de la misma manera que los documentos, y luego se pasa a través del modelo Sentence Transformer para obtener una representación vectorial de la consulta. Luego, el sistema calcula la similitud del coseno entre el vector de consulta y todos los nodos y aristas del grafo. Se seleccionan los k nodos y aristas más similares para formar un subgrafo que contenga la información más relevante para la consulta del usuario.

Justificación: El uso de la similitud del coseno para la coincidencia garantiza que se seleccionen los nodos y aristas más relevantes, mientras que la formación de un subgrafo garantiza que la información recuperada sea contextualmente relevante y completa.

El subgrafo resultante de la recuperación es un término amplio, pero hemos utilizado dos enfoques diferentes para crearlo, cada uno con sus propias ventajas e inconvenientes:

Básico: este enfoque es extremadamente simple, ya que el subgrafo resultante solo incluirá los nodos y aristas recuperados y sus vecinos directos. Este enfoque es rápido y eficiente, pero puede que no capture todo el contexto de la consulta. Sin embargo, los resultados siguen siendo muy buenos, como veremos más adelante.

Avanzado: este enfoque es más complejo y costoso desde el punto de vista computacional. También conocido como el Subgrafo de Expansión Mínima, es esencialmente un problema de optimización en el que nos gustaría encontrar el subgrafo conectado que minimice la cantidad de aristas y nodos involucrados, al tiempo que se asegura de que todos los nodos y aristas relevantes estén presentes. Si no

existe tal subgrafo, entonces se divide el grafo en componentes conectados y se devuelve un Subgrafo de Expansión Mínima para cada componente, por supuesto trabajando con los nodos y aristas más relevantes que están presentes en el componente respectivo.

Justificación: El uso de enfoques básicos y avanzados para la extracción de subgrafos garantiza la flexibilidad y la precisión en la captura del contexto completo de la consulta, al tiempo que se equilibra la eficiencia computacional, dependiendo de cuál se elija utilizar.

Ahora, dado el subgrafo, utilizamos un modelo de lenguaje para generar una respuesta que sea contextualmente relevante para la consulta del usuario. En este caso utilizaremos el modelo Gemini-2.0-exp, un modelo de vanguardia desarrollado por el equipo de Google AI, experto en tareas textuales. El prompt se construye concatenando la consulta del usuario con la representación textual del subgrafo, lo que proporciona al modelo el contexto necesario para generar respuestas informativas y coherentes.

Justificación: El uso del modelo Gemini-2.0-exp garantiza una generación de respuestas de alta calidad y contextualmente relevante, mientras que la construcción del prompt con el subgrafo garantiza que el modelo tenga el contexto necesario para generar respuestas precisas.

4.5. Utilizando un ejército de expertos

Lo que hemos visto hasta ahora es un sistema capaz de generar respuestas basadas en una consulta del usuario, pero ¿qué ocurre si la consulta del usuario no es lo suficientemente clara o el sistema no es capaz de recuperar la información necesaria? Aquí es donde entra en juego el ejército de expertos.

El ejército de expertos es una colección de expertos en el dominio contruidos de la misma manera que el sistema, pero con la diferencia de que cada uno es experto en un campo diferente del dominio financiero. Cada uno de los llamados “expertos” idealmente tendrá un conjunto diferente de documentos y un prompt diferente para generar la respuesta. Haciendo que la respuesta que un experto genera, aunque también relativamente similar a la de otro, tenga un enfoque completamente diferente. Por ejemplo, en este trabajo vamos a utilizar 3 expertos diferentes: finanzas personales, economía e inversión. Cada uno tendrá una base de datos designada a partir de artículos y blogs de su campo.

Justificación: El uso de un ejército de expertos garantiza que el sistema pueda manejar una amplia gama de consultas financieras con alta precisión y profundidad, al tiempo que aprovecha la experiencia específica del dominio para obtener respuestas más precisas y completas.

Luego pediremos a cada experto que genere una respuesta basada en la consulta proporcionada y tendremos un agente orquestador que recopilará esas respuestas y, en otra llamada LLM, las combinará en una única respuesta para el usuario. Esto da como resultado una respuesta multifacética, que depende en gran medida de los expertos que elijamos para la tarea y de cómo compongamos el conjunto de datos para cada uno de ellos.

Justificación: La combinación de respuestas de múltiples expertos garantiza que la respuesta final sea multifacética y completa, aprovechando los puntos fuertes de cada experto específico del dominio para obtener una respuesta más precisa e informativa.

5. Evaluación

5.1. Descripción general de la evaluación

Para evaluar nuestro sistema, utilizaremos 20 consultas diferentes con diversos grados de dificultad para que el sistema genere resúmenes. Luego compararemos los resúmenes generados con resúmenes predefinidos manualmente para ver el rendimiento del sistema. Además, hay 5 consultas comodín que no están incluidas en las 20 consultas; son consultas intencionalmente complejas o formuladas incorrectamente para ver cómo se comporta el sistema en tales casos, si hay una respuesta incoherente o ninguna respuesta.

5.2. Métrica ROUGE para la evaluación

La métrica ROUGE (Recall-Oriented Understudy for Gisting Evaluation) se utiliza ampliamente para evaluar la calidad del resumen de texto y otras tareas de procesamiento del lenguaje natural. Mide la superposición entre un texto generado (por ejemplo, un resumen) y un texto de referencia. ROUGE se centra principalmente en la exhaustividad (recall), pero también se puede configurar para evaluar la precisión y la puntuación F1.

Estas son sus principales variantes:

1. ROUGE-N: Mide la superposición de n-gramas (por ejemplo, secuencias de n palabras) entre el texto generado y el texto de referencia.
 - ROUGE-1: Superposición de unigramas (palabras individuales).
 - ROUGE-2: Superposición de bigramas (secuencias de dos palabras).
2. ROUGE-L: Evalúa la subsecuencia común más larga (LCS) entre el texto generado y el de referencia, considerando la estructura de la secuencia.
3. ROUGE-W: Una variante ponderada de ROUGE-L que asigna más importancia a las coincidencias consecutivas.
4. ROUGE-S: Mide la superposición de skip-bigramas, lo que permite que las palabras coincidan incluso si no son adyacentes.

ROUGE es particularmente eficaz para tareas donde la salida generada debe parecerse mucho al texto escrito por humanos. Al incluirla en su sección de evaluación, usted enfatiza un enfoque cuantitativo y estandarizado para evaluar la calidad del texto generado.

En este caso particular, analizaremos las puntuaciones ROUGE-1 y ROUGE-L para los resúmenes generados, en particular la puntuación de exhaustividad (recall).

5.3. Métricas de evaluación

Clasificaremos 3 sistemas:

Nuestro sistema con una generación de subgrafos básica - lo llamaremos GRAPH-BASIC.

Nuestro sistema con una generación de subgrafos avanzada - lo llamaremos GRAPH-ADVANCED.

Un sistema RAG simple con incrustación vectorial de texto completo (los artículos mismos) - lo llamaremos RAG-TEXT.

Utilizaremos el modelo GPT-o1 de vanguardia como referencia.

En cuanto a las métricas que utilizaremos, son las siguientes:

Precisión: qué tan bien se desempeña el sistema en las 20 consultas. Esto tendrá 2 partes: las puntuaciones ROUGE-1 y ROUGE-L para los resúmenes generados; y una puntuación autogenerada sobre 5 del modelo de referencia, es decir, se le pedirá al modelo GPT-o1 que genere una puntuación sobre 5 a partir de los resúmenes generados, la puntuación promedio se mostrará como Puntuación LLM.

Robustez: qué tan bien se desempeña el sistema en las consultas comodín. Una consulta bien manejada añade uno a la puntuación, lo que significa que una puntuación perfecta es 5.

Rendimiento: tiempo promedio necesario para generar la consulta.

5.4. Consultas de ejemplo

Algunas consultas de ejemplo que se utilizarán para la evaluación son:

P: “¿Cómo puedo calcular el interés compuesto de mis ahorros manualmente?”

R: “El interés compuesto se calcula utilizando la fórmula $A = P(1 + r/n)^{(nt)}$, donde A es el valor futuro, P es el capital, r es la tasa de interés anual, n es el número de veces que se compone el interés por año y t es el tiempo en años”.

P: “¿Cómo funciona el sistema de tramos impositivos para alguien que gana \$60,000 al año?”

R: “Para un contribuyente soltero, sus ingresos se gravarán progresivamente. Para 2024, pagará el 10% sobre los ingresos hasta \$11,000, el 12% sobre los ingresos de \$11,001 a \$44,725 y el 22% sobre los ingresos superiores a esa cantidad”.

Ahora, para las consultas comodín, para probar si el sistema alucina o proporciona una respuesta incoherente a una pregunta ya incoherente:

P: “¿Cuánto ahorraré mensualmente si gano \$4,000 pero gasto \$6,000 cada mes?”

R: Aquí la respuesta correcta es señalar que si el usuario gasta más de lo que gana, entonces no estaría ahorrando dinero en absoluto.

P: “Si ahorro \$100 semanales durante 2 años a una tasa de interés del 10%, ¿qué tendré en 5 años?”

R: Aquí el sistema debería señalar que faltan variables; si el usuario declara que ahorra \$100 semanales durante 2 años, entonces el sistema no tiene forma de saber cuánto ahorra semanalmente durante los siguientes 3 años.

5.5. Resultados

5.5.1. Precisión

Los valores se aproximan al tercer decimal

	ROUGE-1		ROUGE-L		
Modelo	Exhaustividad	F1	Exhaustividad	F1	Puntuación LLM
GRAPH-BASIC	0.655	0.212	0.473	0.153	4.48

	ROUGE-1		ROUGE-L		
GRAPH-ADVANCED	0.891	0.083	0.744	0.069	4.86
RAG-TEXT	0.501	0.189	0.362	0.131	4.22

5.5.2. Rendimiento

Modelo	Tiempo promedio empleado (en s)
GRAPH-BASIC	15.407271986007691
GRAPH-ADVANCED	25.086044921875
RAG-TEXT	15.621680946350098

5.5.3. Robustez

Modelo	Respuestas comodín correctas (de 5)
GRAPH-BASIC	4
GRAPH-ADVANCED	5
RAG-TEXT	5

5.6. Resumen

Como podemos observar en los resultados, en términos de precisión, el modelo GRAPH-ADVANCED supera a los otros modelos tanto en las puntuaciones ROUGE como en la puntuación LLM. Esto es comprensible debido a la forma en que genera un subgrafo con mucho contexto y, por lo tanto, la respuesta generada necesita llenar menos lagunas en el conocimiento. El segundo mejor en ese sentido es el modelo GRAPH-BASIC, que aún supera al modelo RAG regular para esta tarea. Podemos ver que la puntuación F1 es la más alta tanto en la métrica ROUGE-1 como en la ROUGE-L, lo que significa que los resúmenes generados son más precisos y concisos que los de sus competidores.

Esto también se refleja en la robustez de los modelos, donde el modelo GRAPH-ADVANCED y el modelo RAG-TEXT obtienen una puntuación perfecta de 5 sobre 5, mientras que el modelo GRAPH-BASIC obtiene una puntuación de 4 sobre 5. Esto se debe a que el modelo GRAPH-BASIC no proporciona tanta información a la parte de generación como los otros modelos y, por lo tanto, es más probable que falle en las consultas comodín donde algunas variables no se proporcionan claramente.

En términos de rendimiento, el modelo GRAPH-BASIC es el más rápido debido a que tiene la ventana de tokens de contexto más pequeña (tanto en entrada como en salida) de los tres modelos. Y, evidentemente, el modelo GRAPH-ADVANCED es el más lento debido a la complejidad del algoritmo de generación de subgrafos.

6. Conclusiones

Esta tesis exploró el desarrollo de un sistema de preguntas y respuestas para la planificación financiera impulsado por un sistema de generación aumentada por recuperación (RAG) basado en grafos. Las etapas de investigación e implementación destacaron el potencial del sistema para transformar la planificación financiera al combinar técnicas avanzadas de recuperación con la generación de lenguaje natural. Se pueden extraer las siguientes conclusiones de este estudio:

El uso de sistemas RAG basados en grafos demostró ser eficaz para gestionar las complejidades de los datos financieros. Las estructuras de grafos permitieron la representación de relaciones matizadas entre conceptos financieros, mientras que la recuperación basada en incrustaciones garantizó la flexibilidad semántica en el manejo de diversas consultas de los usuarios. Este enfoque híbrido abordó desafíos específicos del dominio, como la relevancia contextual, la representación precisa de las dependencias financieras y la comprensión de la intención del usuario.

La integración de Neo4j para el almacenamiento de grafos y MongoDB para la recuperación basada en incrustaciones permitió una consulta y una gestión del conocimiento eficientes. La capacidad de extraer subgrafos relevantes para las consultas de los usuarios demostró la capacidad del sistema para ofrecer respuestas precisas y contextualmente apropiadas, lo que lo hace particularmente adecuado para tareas de planificación financiera personal.

La adopción de la ingeniería de prompts zero-shot a través de la API de Gemini permitió al sistema generar respuestas en lenguaje natural de alta calidad sin un entrenamiento extenso en datos específicos del dominio. Este enfoque no solo redujo los costos de desarrollo, sino que también proporcionó adaptabilidad a la evolución del conocimiento financiero y las necesidades de los usuarios.

El rendimiento del sistema, medido por la precisión, la corrección fáctica, la eficiencia y la robustez, validó la viabilidad de los sistemas RAG basados en grafos para aplicaciones fintech. Las pruebas destacaron áreas de mejora, particularmente en el manejo de consultas ambiguas y la optimización de estrategias de recuperación, allanando el camino para futuras mejoras.

Este estudio contribuye al creciente cuerpo de investigación sobre soluciones fintech inteligentes al presentar un marco modular, impulsado por grafos, para el desarrollo de sistemas de preguntas y respuestas. Su metodología, que enfatiza la escalabilidad, la riqueza semántica y la rentabilidad, puede servir como base para una mayor innovación en este dominio.

6.1. Limitaciones y trabajo futuro

A pesar de sus prometedores resultados, este estudio enfrentó limitaciones que presentan oportunidades para una mayor exploración:

- Desafíos de escalabilidad: A medida que crece el grafo de conocimiento, la eficiencia computacional y la velocidad de recuperación pueden necesitar optimización.
- Adaptación al dominio: Si bien es eficaz en la planificación financiera, la adaptabilidad del sistema a otros dominios aún debe probarse.
- Métricas de evaluación mejoradas: La incorporación de la satisfacción del usuario y el compromiso a largo plazo en el marco de evaluación proporcionaría información más profunda sobre la aplicabilidad en el mundo real.

La investigación futura podría centrarse en refinar las técnicas de extracción de subgrafos, explorar la integración de datos multimodales y mejorar la personalización del asesoramiento financiero al

proporcionar información sensible al contexto del usuario. Además, extender el sistema para manejar datos y transacciones en tiempo real aumentaría su utilidad práctica.

Al integrar técnicas avanzadas de recuperación y generación, esta tesis demuestra cómo un sistema RAG basado en grafos puede brindar asesoramiento financiero personalizado, ofreciendo un paso prometedor en la evolución de las aplicaciones fintech inteligentes.

Bibliografía

- [1] BOCI PENG *et al.*, “Graph Retrieval-Augmented Generation: A Survey,” vol. 37, no. 4.
- [2] Lisa Ehrlinger and Wolfram Wöß, “Towards a Definition of Knowledge Graphs.”
- [3] Paulheim H., “Knowledge Graph Refinement: A Survey of Approaches and Evaluation Methods.”
- [4] Yuntong Hu, Zhihan Lei, Zheng Zhang, Bo Pan, Chen Ling, and Liang Zhao, “GRAG: Graph Retrieval-Augmented Generation.”
- [5] Cui Long, Yongbin Liu, Chunping Ouyang, and Ying Yu, “Bailicai: A Domain-Optimized Retrieval-Augmented Generation Framework for Medical Applications.”
- [6] Chao Jin *et al.*, “RAGCache: Efficient Knowledge Caching for Retrieval-Augmented Generation.”
- [7] Rong-Ching Chang and Jiawei Zhang, “CommunityKG-RAG: Leveraging Community Structures in Knowledge Graphs for Advanced Retrieval-Augmented Generation in Fact-Checking.”
- [8] Xiaoxin He *et al.*, “G-Retriever: Retrieval-Augmented Generation for Textual Graph Understanding and Question Answering.”
- [9] Zetian Hu, Weijian Xie, Xuefeng Liang, Yuhui Liu, Kaihua Ni, and Hong Cheng, “WeKnow-RAG: An Adaptive Approach for Retrieval-Augmented Generation Integrating Web Search and Knowledge Graphs.”
- [10] Jinyuan Fang, Zaiqiao Meng, and Craig Macdonald, “TRACE the Evidence: Constructing Knowledge-Grounded Reasoning Chains for Retrieval-Augmented Generation.”
- [11] Junde Wu, Jiayuan Zhu, and Yunli Qi, “Medical Graph RAG: Towards Safe Medical Large Language Model via Graph Retrieval-Augmented Generation.”
- [12] Wenyu Huang, Mirella Lapata, Pavlos Vougiouklis, Nikos Papasarrantopoulos, and Jeff Z. Pan, “Retrieval Augmented Generation with Rich Answer Encoding.”
- [13] Komal Azram Raja, “Domain Specific Data Quality Framework.”
- [14] Thomas Sepanosian, Zoran Milosevic, and Andrew Blair, “Scaling AI adoption in finance: modellingframework and implementation study.”
- [15] I. de Zarzà, J. de Curtò, Gemma Roig, and Carlos T. Calafate, “Optimized Financial Planning: Integrating Individual and Cooperative Budgeting Models with LLM Recommendations.”
- [16] Jinyuan Fang, Zaiqiao Meng, and Craig Macdonald, “REANO: Optimising Retrieval-Augmented Reader Models through Knowledge Graph Generation,” vol. 1.