Revisión de la 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.

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 noved

osas de manera efectiva. Al integrar incrustaciones semánticas, los sistemas RAG se vuelven más aptos para manejar las complejidades del lenguaje natural y las intricacias 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.

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.

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."

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.

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:

     text{“TS”} = sum*(o=1)^n (I_o - sum_z E*(zo))

     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:

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

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.

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.

Conclusión

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.

Bibliography

[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 Papasarantopoulos, 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.