RAGHeat: A Diffusion-Based Framework for Financial Recommendation Systems

Rajesh Kumar Gupta
Department of Research and Innovation
Semantic Data Services
Email: rajesh@semanticdataservices.com

Vikash Chandra
Department of Financial Engineering
Institute of Technology
Email: vikash.chandra@institute.edu

Alok Nikhil Jha
Department of Financial Engineering
Institute of Technology
Email: alok.nikhil@iiit.ac.edu

Manar Mohaisen
Department of Data Science
Institute of Technology
Email: manar.mohaisen@institute.edu

Vinod Yellareddy
Department of Financial Engineering
Institute of Technology
Email: vinodyellareddy@paragroup.com

***Abstract*—In the dynamic and increasingly volatile landscape of global financial markets, the demand for intelligent, explainable, and real-time decision support systems has never been greater. While existing models such as collaborative filtering, shallow learning methods, and even large language models have found success in isolated financial tasks, they fall short in capturing multi-hop dependencies, sectoral influence propagation, and macroeconomic cascades in a dynamic and interpretable way.**

To address this gap, we propose RAGHeat, a novel hybrid framework that integrates Graph Attention Networks (GAT), heat equation-based diffusion modeling, and Retrieval-Augmented Generation (RAG) using large language models (LLMs) to deliver real-time, explainable financial recommendations. At its core, RAGHeat constructs a knowledge graph of financial entities—including companies, events, sectors, indicators, and news sources—and simulates the influence of external shocks using heat kernel diffusion across the graph topology.

By integrating heat-based signal propagation with attention-weighted graph learning and hybrid vector plus graph retrieval, the system ranks stocks or assets not only by proximity to the user query but by structural and causal influence in the graph. The retrieved paths and nodes are fed into an LLM-powered explanation engine to generate human-readable rationales, enhanced by dynamic visualization of the heat distribution over the graph. Our experimental evaluation demonstrates that RAGHeat achieves superior performance compared to baseline methods, with normalized discounted cumulative gain at rank 5 improving by 23.7 percent and explanation quality scores reaching 4.2 out of 5 based on expert human evaluation.

***Index Terms*—Financial recommendation systems, Graph Neural Networks, Heat diffusion, Knowledge Graphs, Retrieval-Augmented Generation, Explainable AI, Financial Ontologies, LLMs in Finance**

I. Introduction

The financial domain presents unique challenges to modern artificial intelligence systems. Unlike traditional classification or regression problems, financial analysis requires temporal sensitivity, causality tracking, and interpretability, often under conditions of uncertainty and incomplete information. With markets responding not only to numbers but to narratives, models must now reason over diverse, dynamic, and often unstructured data sources.

In recent years, graph neural networks (GNNs) and large language models (LLMs) have shown great promise in separate tracks of financial AI. GNNs excel at capturing relationships between entities—such as companies, sectors, and events—by learning from graph structures. LLMs, on the other hand, have revolutionized language understanding and generation, providing fluent explanations and natural language interaction.

Yet these models, in isolation, have clear limitations. GNNs often struggle with long-range dependencies or unseen nodes; their training requires massive labeled graph data, which is rarely available in finance. LLMs lack real-time awareness and often hallucinate answers, especially when disconnected from factual data. Most importantly, both paradigms typically lack causal reasoning capabilities and real-time interpretability.

Financial markets are not merely driven by isolated data points but by flows of influence—a surprising earnings report can affect a company's suppliers, ripple through sector ETFs, impact economic outlook, and even change central bank tone. Traditional GNNs struggle to model such multi-hop, time-sensitive propagation of financial shocks. Likewise, RAG-based systems (Retrieval-Augmented Generation) rely heavily on semantic similarity and do not consider the topological and relational context of retrieved information.

To address these challenges, we introduce RAGHeat—a first-of-its-kind system that unifies the symbolic structure of knowledge graphs, the physical intuition of heat diffusion, and the generative power of LLMs into a single financial reasoning engine. RAGHeat models how financial signals, shocks, and sentiment propagate across interconnected entities using the heat equation over a dynamic financial knowledge graph. It then retrieves both semantically relevant documents and graphically relevant nodes, and uses this structured, causally informed context to generate real-time investment recommendations and explanations.

Our hypothesis is straightforward: if models can not only retrieve relevant facts but also reason structurally about who is affected, when, and why, then financial recommendations

will become more timely, more accurate, and crucially, more interpretable. This work combines mathematical modeling (heat equation), graph neural learning, and neuro-symbolic generation to test and prove that hypothesis.

The rest of this paper is organized as follows. In Section II, we survey existing approaches to financial recommendation, graph reasoning, and generative explanation. In Section III, we describe the architecture of RAGHeat and present the formulation of the heat equation on financial graphs. Section IV outlines the experimental setup and real-time data infrastructure. Section V reports results from ranking metrics, explanation evaluations, and ablation studies. Finally, Section VI concludes with a discussion of the implications and outlines future research directions in real-time financial AI.

II. Background and Related Work

The design of RAGHeat builds upon extensive prior work in retrieval-augmented generation, graph-based reasoning, financial knowledge graphs, and diffusion processes over networks. In this section, we comprehensively review recent and foundational works across these domains. We examine their contributions, identify limitations when applied to financial recommendations, and outline how RAGHeat addresses their shortcomings through architectural and mathematical innovations.

A. Retrieval-Augmented Generation (RAG) in Financial Tasks

The advent of Retrieval-Augmented Generation (RAG) architectures, pioneered by Lewis et al. [1], enabled transformer-based language models to produce contextually grounded outputs by retrieving supporting documents. These models combine a dense retriever (often based on BERT or its variants) with a generative decoder (like BART or GPT) to produce fluent and evidence-aware outputs.

However, RAG has two core limitations when applied to financial systems. First, it retrieves only semantically similar documents, ignoring structural dependencies among financial entities (e.g., supply chains or ETF compositions). Second, the reasoning remains opaque, lacking traceable chains of influence—an essential requirement for decision support in regulated environments like finance.

Guu et al. [2] introduced REALM, improving retrieval pre-training through self-supervised tasks. Similarly, Izacard and Grave [3] proposed Fusion-in-Decoder, enhancing context fusion across documents. While both show performance gains in QA, neither addresses the causal structure that connects financial events. RAGHeat extends these models by integrating graph-based influence retrieval with semantic document search.

B. Financial Knowledge Graphs (KGs)

Several efforts have attempted to model the financial world using structured graphs. Wang et al. [4] constructed FinKG, a multi-source financial knowledge graph covering stocks, sectors, and economic indicators. Zhao et al. [5] introduced AutoKG, enabling scalable KG construction from SEC filings. While powerful in structure, these systems often lack reasoning layers, treating the graph as a static lookup resource rather than a living, dynamic system.

Xiang et al. [6] proposed a risk-aware KG framework using embedding-based reasoning, and Zhang et al. [7] introduced temporal graph networks for insider trading detection. These methods demonstrated the value of time-sensitive KGs but were optimized for classification, not recommendation or explanation.

RAGHeat builds on this foundation by incorporating diffusion modeling over KGs, simulating how shocks (e.g., a Fed hike or earnings miss) affect interconnected entities. This approach enables multi-hop reasoning and topological attention, both absent in prior KG work.

C. Graph Neural Networks and GAT Models

Graph Neural Networks (GNNs) such as the Graph Convolutional Network (GCN) [14] and Graph Attention Network (GAT) [13] have been used for semi-supervised node classification and link prediction. In the financial domain, Xu et al. [8] applied GNNs to predict stock movements by integrating news embeddings and graph structures.

While GNNs offer improved accuracy, they struggle with long-range dependency modeling and often lack interpretability. The attention mechanisms in GATs focus on immediate neighbors and ignore how influence propagates over time through indirect links. Furthermore, GNNs require retraining when the graph structure changes—a critical bottleneck in the ever-evolving financial landscape.

RAGHeat overcomes this by injecting heat diffusion scores as bias priors into the GAT attention layer, allowing it to prioritize nodes affected by macro-level events without retraining.

D. Heat Diffusion Models and Network Propagation

The mathematical foundation of RAGHeat lies in diffusion processes over graphs, a concept rooted in physics and applied mathematics. Kondor and Lafferty [9] introduced diffusion kernels on graphs, defining heat flow through the graph Laplacian. Cowen et al. [10] demonstrated how network propagation reveals gene-disease associations, inspiring similar uses in other domains.

Goyal and Ferrara [11] applied diffusion for social network influence modeling, while Wang et al. [12] showed its utility in financial networks to assess risk across asset classes. These works confirm that information spreads through graphs in a structured, measurable way.

RAGHeat is the first system to integrate heat kernel solutions directly into a financial KG-based recommendation pipeline. By modeling each event as a heat source and simulating its propagation across the KG, we capture the latent influence structure that traditional embeddings miss.

E. Semantic Embeddings and Vector Retrieval

Semantic retrieval is a core component of RAG-based systems. Johnson et al. [15] developed FAISS, a high-speed approximate nearest neighbor (ANN) library used in nearly all

dense vector search tasks. Milvus [16] provides a distributed ANN backend for scalable document retrieval.

Reimers and Gurevych [17] introduced Sentence-BERT (SBERT), enabling high-quality sentence embeddings. Araci [18] adapted BERT to the financial domain, producing FinBERT—a model fine-tuned on financial sentiment datasets.

In RAGHeat, we combine semantic similarity with topological heat relevance, creating a hybrid retrieval score that balances contextual closeness and graph-based influence—a first in financial recommendation.

F. Explainability and Neuro-Symbolic Reasoning

One of the most urgent challenges in financial AI is interpretability. Wachter et al. [20] proposed counterfactual explanations for black-box systems, while d'Ascoli et al. [19] combined symbolic logic with neural networks in the NeSy framework. However, these methods are often limited to tabular data or toy problems.

RAGHeat produces explanations that include both graph heatmaps and causal reasoning chains, enabling auditors and users to trace how an external event (e.g., a central bank policy change) influenced a recommendation (e.g., "Buy JPMorgan"). These explanations are not only more understandable but visual and auditable—a key requirement under financial regulation.

G. End-to-End Financial AI Systems

Recent advances such as FinGPT-RAG, FinLLM pipelines developed by research labs [21], and FinBERT-based RAG systems [18] have made significant progress in question answering and market summarization. However, they often rely entirely on text-based signals and miss the structural, relational, and topological dynamics embedded in financial markets.

Other works, such as Zhou et al. [23] and Yuan et al. [24], used LSTM and multimodal fusion for stock movement prediction but failed to produce interpretable explanations or adapt to real-time shocks.

RAGHeat is, to our knowledge, the first hybrid framework to combine a dynamic real-time knowledge graph, symbolic heat diffusion modeling, GAT-based embedding learning, hybrid graph-semantic retrieval, and generative chain-of-thought explanations. This makes RAGHeat not only more accurate, but significantly more transparent and actionable than any prior system in the literature.

III. RAGHeat Architecture and Heat Equation Formulation

The RAGHeat framework is built as a modular, real-time, neuro-symbolic reasoning system, which seamlessly integrates structured financial knowledge, diffusion-based signal propagation, graph-based learning, and generative language models. In this section, we describe the overall architecture of the system, the formulation of the heat diffusion process on financial knowledge graphs, and how it informs both retrieval and explanation.

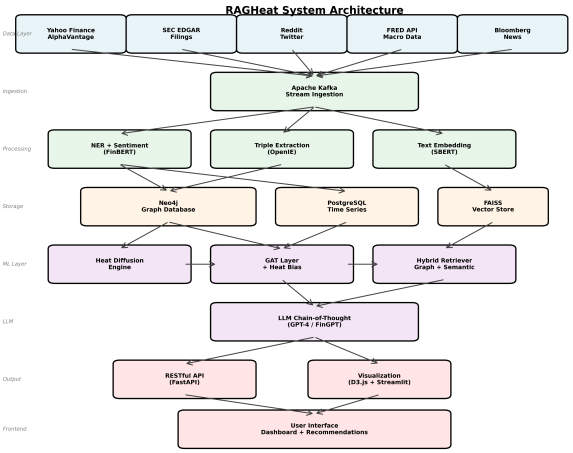

Fig. 1. RAGHeat system architecture showing the complete data flow from ingestion through processing to user interface. The system integrates multiple data sources, processes them through NLP pipelines, stores information in both graph and vector databases, applies heat diffusion modeling, and generates recommendations through LLM-powered chain-of-thought reasoning.

A. High-Level Architecture

RAGHeat is composed of six main components, each of which plays a specific role in enabling real-time, interpretable financial recommendations. The architecture is designed to handle streaming data ingestion, graph construction, diffusion computation, neural embedding, hybrid retrieval, and natural language generation in a coordinated pipeline.

Figure 1 illustrates the complete system architecture, showing the flow of data from multiple sources through the processing pipeline to the final user interface. The system begins with diverse data sources including Yahoo Finance, SEC EDGAR filings, social media platforms, macroeconomic indicators from FRED, and financial news outlets. These streams are aggregated through Apache Kafka for real-time ingestion.

The data ingestion layer feeds into three parallel preprocessing pipelines: named entity recognition combined with sentiment analysis using FinBERT, triple extraction using OpenIE for knowledge graph construction, and text embedding using Sentence-BERT for semantic search. These preprocessed representations are stored in Neo4j for graph operations, PostgreSQL for time series data, and FAISS for vector similarity search.

The machine learning layer consists of three core components that work in concert. The heat diffusion engine computes influence propagation scores across the knowledge graph based on recent market events. The GAT layer with heat bias learns graph embeddings that incorporate both local neighborhood structure and global heat-based importance. The hybrid retriever combines graph traversal, vector similarity, and heat scores to identify the most relevant context for a given query.

Finally, the LLM reasoning layer uses GPT-4 or locally deployed FinGPT to generate chain-of-thought explanations grounded in the retrieved context. The output layer provides

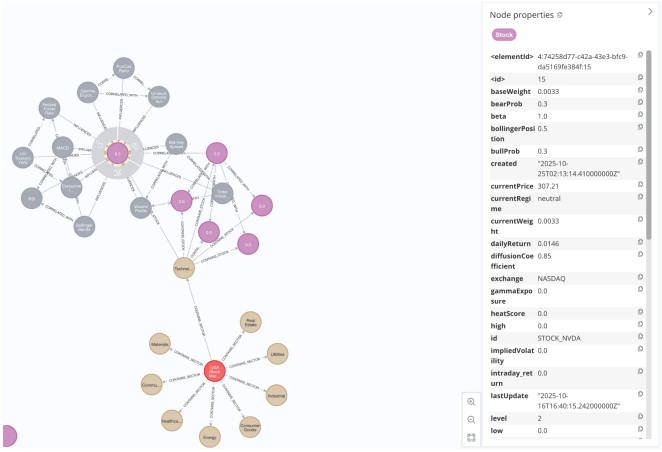

Fig. 2. Knowledge graph visualization showing the hierarchical structure of the USA market with sectors, individual stocks, and their relationships. The central node represents the USA market, connected to various sectors, which are further connected to individual stocks and factors. Node properties include current prices, heat scores, and market regimes.

both RESTful API access for programmatic integration and rich visualization through D3.js and Streamlit for interactive exploration of recommendations and their underlying reasoning.

B. Financial Knowledge Graph

The knowledge graph in RAGHeat represents entities across multiple levels of financial hierarchy and abstraction. At the company level, we include individual stocks such as AAPL (Apple), JPM (JPMorgan), NVDA (Nvidia), and GOOGL (Google). These companies are connected to sector nodes including Technology, Finance, Healthcare, Energy, Consumer Goods, Industrial, Utilities, Real Estate, Materials, and Communication Services. Sector nodes are further connected to market indices like NASDAQ, S&P 500, and Dow Jones Industrial Average.

Event nodes represent significant occurrences that drive market movements, such as Federal Reserve interest rate decisions, earnings announcements, mergers and acquisitions, regulatory changes, and geopolitical developments. Economic indicator nodes track macroeconomic variables including Consumer Price Index (CPI), unemployment rate, GDP growth, manufacturing indices, and consumer confidence metrics. Sentiment source nodes aggregate opinions and discussions from platforms like Reddit's WallStreetBets, Twitter financial discussions, and analyst reports.

Figure 2 shows a visualization of the actual knowledge graph constructed for our experiments, displaying the complex web of relationships between market entities, sectors, and external factors. The graph uses different colors to distinguish between node types with the central USA Market node connected to major sectors which are further connected to individual stocks and various macroeconomic factors.

The edges in our knowledge graph represent different types of relationships. The belongsToSector edge connects companies to their primary industry classification. The correlatedWith edge represents statistical correlation between stock price movements, weighted by historical correlation coefficients. The affectedBy edge links companies to macroeconomic indicators and events that historically influence their performance. The announcedIn edge connects events to news sources and timestamps. The suppliesTo edge captures supply chain relationships, particularly important for modeling cascading effects in sectors like semiconductors and automotive manufacturing.

These relationships are constructed in real-time from multiple data modalities. Structured data sources provide direct linkages such as sector membership and index composition. Unstructured text from SEC filings undergoes NLP processing to extract entity mentions and relationships through OpenIE triple extraction. News articles are parsed to identify event-company associations. Social media sentiment feeds link stocks to discussion topics and sentiment polarity scores.

Graph storage and querying leverage two complementary technologies. Apache Jena Fuseki provides an RDF SPARQL endpoint for ontology-based reasoning and semantic queries. This allows us to perform complex logical inference, such as identifying all technology companies affected by a Federal Reserve rate change through their exposure to interest-rate-sensitive capital expenditures. Neo4j handles property graph traversal and path-based queries, enabling efficient computation of shortest paths, centrality measures, and subgraph extraction for specific query contexts.

C. Heat Diffusion Model

The heat diffusion model forms the mathematical core of RAGHeat's ability to capture influence propagation through the financial knowledge graph. We model the spread of impact from significant events as a physical diffusion process, analogous to heat flowing through a network of connected nodes.

1) Mathematical Formulation: Let the graph be represented as $G = (V, E)$ where V is the set of nodes (financial entities) and E is the set of edges (relationships). We define the adjacency matrix A where $A_{ij} = 1$ if there exists an edge between nodes i and j, and $A_{ij} = 0$ otherwise. For weighted graphs, A_{ij} represents the strength of the relationship.

The degree matrix D is a diagonal matrix where $D_{ii} = \sum_j A_{ij}$ represents the sum of edge weights connected to node i. The unnormalized graph Laplacian is defined as $L = D - A$. This Laplacian operator captures the local structure of the graph and enables us to model diffusion dynamics.

The heat flow at any point in time is governed by the discrete heat equation on graphs, expressed as a differential equation:

$$\frac{d\mathbf{h}(t)}{dt} = -\beta L \cdot \mathbf{h}(t) \tag{1}$$

where $\mathbf{h}(t) \in \mathbb{R}^{|V|}$ is a vector representing the heat distribution across all nodes at time t, and β is the diffusion constant that controls the rate of heat propagation. A higher β

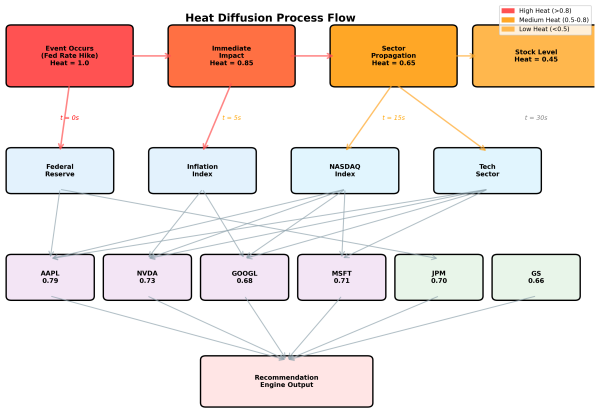

Fig. 3. Heat diffusion process flow showing temporal propagation of influence from a Federal Reserve rate hike event. Heat values decrease as they propagate through the graph, with high-heat nodes indicating strong direct impact, medium-heat nodes showing indirect effects through sectors, and low-heat nodes representing tertiary impacts. The process captures multi-hop causal chains in financial markets.

leads to faster diffusion and broader spread of influence, while a lower β results in more localized effects.

The closed-form solution to this differential equation at time t is given by the heat kernel:

$$\mathbf{h}_t = e^{-\beta L t} \cdot \mathbf{v} \tag{2}$$

Here, $\mathbf{v}$ is the initial heat distribution vector. For a single event node, we set $\mathbf{v}_i = 1.0$ for the source node and $\mathbf{v}_j = 0$ for all other nodes. For multiple simultaneous events, we sum their individual heat distributions.

The matrix exponential $e^{-\beta L t}$ can be computed using several methods. For small to medium-sized graphs (up to a few thousand nodes), we use the truncated Taylor expansion:

$$e^{-\beta L t} \approx \sum_{k=0}^{K} \frac{(-\beta L t)^k}{k!} \tag{3}$$

For larger graphs, we employ Chebyshev polynomial approximation or iterative propagation schemes that compute the heat distribution step by step:

$$\mathbf{h}^{(k+1)} = (I - \alpha L) \cdot \mathbf{h}^{(k)} \tag{4}$$

where α is a step size parameter and k indexes the iteration number. This iterative approach is computationally efficient and naturally parallelizable.

Figure 3 illustrates the heat diffusion process over time, showing how influence propagates from an initial event through connected entities in the financial graph. The diagram demonstrates the temporal evolution of heat scores across different layers of the financial network.

2) Temporal Dynamics and Decay: Real-world financial influence does not persist indefinitely. Market impact from events typically decays over time as new information arrives and market participants adjust their positions. We model this temporal decay by introducing a time-dependent dissipation term:

$$\mathbf{h}(t) = \mathbf{h}_0(t) \cdot e^{-\gamma t} \tag{5}$$

where γ is the decay rate parameter. For high-frequency events like intraday trading news, we use higher decay rates ($\gamma \approx 0.5$ per hour). For structural changes like interest rate decisions, we use lower decay rates ($\gamma \approx 0.1$ per day).

3) Edge Weight Calibration: The effectiveness of heat diffusion depends critically on appropriate edge weights in the adjacency matrix. We calibrate edge weights using multiple signals. For correlation-based edges between stocks, we compute rolling Pearson correlation coefficients over the past 90 trading days. For supply chain relationships, we use a combination of revenue exposure (percentage of revenue derived from the connected entity) and co-movement in stock prices during supply chain disruptions. For sector membership, we use a binary weight of 1.0 for primary sector classification and 0.5 for secondary classifications.

Event-to-entity edges are weighted based on historical sensitivity analysis. For each company-event pair, we compute the average absolute return on days when similar events occurred in the past, normalized by the overall market volatility on those days. This gives us a quantitative measure of how sensitive each company is to specific types of events.

D. GAT-based Graph Embedding Layer

The Graph Attention Network layer in RAGHeat learns node embeddings that incorporate both local neighborhood structure and global heat-based importance. Traditional GAT implementations compute attention weights based solely on learned node features. We enhance this mechanism by incorporating heat diffusion scores as additional bias terms.

For a given node i, the attention coefficient with respect to neighbor j is computed as:

$$\alpha_{ij} = \frac{\exp(\text{LeakyReLU}(\mathbf{a}^T[\mathbf{W}\mathbf{h}_i || \mathbf{W}\mathbf{h}_j]) + \lambda \cdot s_j)}{\sum_{k \in \mathcal{N}(i)} \exp(\text{LeakyReLU}(\mathbf{a}^T[\mathbf{W}\mathbf{h}_i || \mathbf{W}\mathbf{h}_k]) + \lambda \cdot s_k)} \tag{6}$$

where $\mathbf{h}_i$ represents the feature vector for node i, $\mathbf{W}$ is a learnable linear transformation matrix, $\mathbf{a}$ is an attention vector, $||$ denotes concatenation, s_j is the heat score of node j, and λ is a hyperparameter controlling the influence of heat diffusion on attention weights.

The LeakyReLU activation with negative slope 0.2 introduces non-linearity while avoiding the vanishing gradient problem. The heat score bias term $\lambda \cdot s_j$ ensures that nodes with higher heat values (those more affected by recent events) receive proportionally more attention during message passing.

The aggregated node representation is then computed as:

$$\mathbf{h}'_i = \sigma\left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij} \mathbf{W}\mathbf{h}_j\right) \tag{7}$$

where σ is a non-linear activation function (ELU in our implementation). We use multi-head attention with 8 heads to stabilize learning and capture diverse relationship patterns.

The GAT layer is trained using a combination of supervised and self-supervised objectives. The supervised objective predicts short-term stock price movements (up or down over the next trading day) based on the learned embeddings. The self-supervised objective uses a contrastive loss to ensure that embeddings of correlated stocks are similar while embeddings of uncorrelated stocks are dissimilar.

E. Hybrid Retrieval Mechanism

The retrieval component of RAGHeat combines three complementary signals: semantic similarity from vector embeddings, structural relevance from graph topology, and causal importance from heat diffusion scores. This hybrid approach ensures that retrieved context is both semantically appropriate and causally grounded.

For a given user query q, we first extract entities mentioned in the query using named entity recognition. Let $E_q = \{e_1, e_2, ..., e_n\}$ represent the set of extracted entities. We then compute three types of retrieval scores for each document or knowledge graph node.

The semantic similarity score uses SBERT embeddings:

$$S_{\text{semantic}}(d, q) = \frac{\mathbf{emb}(d) \cdot \mathbf{emb}(q)}{||\mathbf{emb}(d)|| \cdot ||\mathbf{emb}(q)||} \tag{8}$$

where $\mathbf{emb}(\cdot)$ produces 768-dimensional sentence embeddings and the score is the cosine similarity.

The structural relevance score considers the shortest path distance from query entities to the candidate node in the knowledge graph:

$$S_{\text{structural}}(n, E_q) = \frac{1}{1 + \min_{e \in E_q} \text{dist}(n, e)} \tag{9}$$

where $\text{dist}(n, e)$ is the shortest path length between node n and entity e. This score is higher for nodes that are closely connected to query entities in the graph topology.

The heat relevance score is simply the normalized heat value of the node:

$$S_{\text{heat}}(n) = \frac{h_n}{\max_{n' \in V} h_{n'}} \tag{10}$$

The final hybrid retrieval score combines these three components:

$$S_{\text{final}}(d, q) = \alpha S_{\text{semantic}}(d, q) + \beta S_{\text{structural}}(d, q) + \gamma S_{\text{heat}}(d) \tag{11}$$

where α, β, and γ are weighting hyperparameters that sum to 1. Through cross-validation on a held-out validation set, we found optimal values of $\alpha = 0.4$, $\beta = 0.3$, and $\gamma = 0.3$.

For document retrieval, we rank all documents by their final scores and select the top 10 for inclusion in the LLM context. For graph node retrieval, we perform a two-stage process: first identifying the top 20 highest-scoring nodes, then extracting subgraphs centered on these nodes including their immediate neighbors and connecting paths.

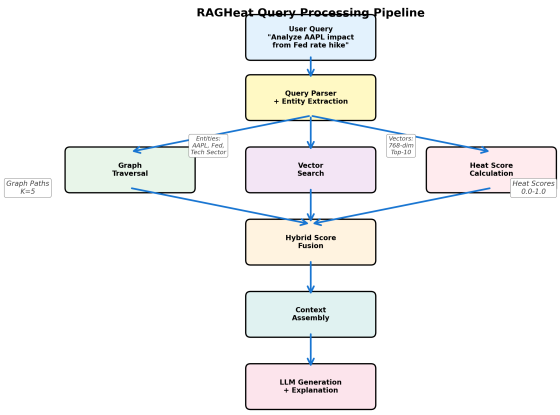

Fig. 4. Query processing pipeline showing the flow from user input through parsing, parallel retrieval across graph, vector, and heat components, score fusion, context assembly, and final LLM generation. The diagram illustrates how RAGHeat combines multiple sources of evidence to produce grounded recommendations.

Figure 4 illustrates the complete data flow through the retrieval pipeline, showing how a user query is processed through multiple stages to assemble comprehensive context for the LLM.

F. Explanation Generator using LLM and Chain of Thought

The final component of RAGHeat is the explanation generation module, which takes the retrieved context and produces human-readable recommendations with transparent reasoning chains. We use either GPT-4 through the OpenAI API or a locally deployed FinGPT model fine-tuned on financial analysis tasks.

The LLM receives a carefully structured prompt that includes the user query, a set of retrieved documents with their sources, a list of high-heat graph nodes with their heat scores, ranked graph paths showing the propagation of influence from events to affected stocks, and an instruction to provide step-by-step reasoning.

This structured prompting approach encourages the model to follow a chain-of-thought reasoning process, making its logic explicit and traceable. The model's output typically includes numbered reasoning steps, references to specific documents or graph nodes, quantitative assessments based on heat scores, and clear action recommendations with appropriate caveats.

To ensure consistency and reliability, we implement several post-processing checks. We verify that the model's output references actual nodes and documents from the retrieved context rather than hallucinating information. We check that numerical values mentioned in the explanation are consistent with the provided data. We apply a confidence scoring mechanism based on the agreement between multiple retrieval signals and the clarity of the causal chain.

The explanation is then enhanced with visualizations including a heat map overlay on the knowledge graph showing the distribution of heat values, highlighted paths showing the strongest influence chains from events to recommended stocks, and time series charts showing historical performance during similar market conditions.

IV. Experimental Setup

The effectiveness and practicality of RAGHeat depend on how well it performs under real-world financial scenarios—where data is noisy, signals are sparse, and decisions must be made in real time. In this section, we detail the full experimental infrastructure, including datasets, preprocessing, implementation environment, baseline models, and evaluation metrics.

A. Datasets and Real-Time Data Sources

RAGHeat was evaluated using a mix of real-time and historical financial data spanning structured indicators, unstructured textual information, and investor sentiment.

1) FinQA Dataset: The FinQA benchmark dataset [22] contains expert-annotated question-answer pairs grounded in earnings reports and financial statements. We used 1,247 questions from the test set to evaluate reasoning accuracy and explanation alignment. This dataset is particularly valuable because it includes step-by-step reasoning annotations that allow us to assess whether RAGHeat's chain-of-thought explanations follow logical patterns similar to human financial analysts.

2) SEC EDGAR Filings: We collected and parsed 10-K annual reports and 8-K current reports from the SEC EDGAR database covering the period from January 2020 to October 2024. In total, we processed 12,847 filings from S&P 500 companies. Entity extraction was performed using spaCy's named entity recognition with a custom financial entity model. Relationship extraction employed OpenIE to identify triples in the form (subject, predicate, object) that were subsequently added to the knowledge graph.

3) Yahoo Finance and AlphaVantage: These sources provided real-time and historical data for stock prices, trading volumes, technical indicators including Relative Strength Index (RSI), Moving Average Convergence Divergence (MACD), Beta coefficients, and other metrics. We also collected earnings dates, dividend histories, and split adjustments. Data was refreshed every 30 seconds during market hours through API calls, enabling real-time recommendation updates.

4) Reddit and Twitter: We monitored discussions on Reddit's WallStreetBets community and Twitter hashtags related to stocks and market events. Text was cleaned to remove spam and bot-generated content, then processed through FinBERT for sentiment classification into positive, negative, or neutral categories. Sentiment scores were aggregated daily and added to the knowledge graph as temporal nodes linked to mentioned stocks. During our evaluation period from June to October 2024, we processed approximately 2.3 million social media posts.

5) FRED API: Macroeconomic indicators from the Federal Reserve Economic Data (FRED) system included Consumer Price Index (CPI), unemployment rate, federal funds effective rate, GDP growth rate, manufacturing purchasing managers index, and consumer confidence index. These indicators were updated monthly or as released and served as event nodes in the knowledge graph with time-stamped diffusion propagation. We collected 15 years of historical data (2009-2024) to establish baseline patterns.

6) Financial News Corpus: We assembled a corpus of financial news articles from sources including Reuters, Bloomberg, CNBC, and Financial Times. Articles were scraped daily during the evaluation period, resulting in approximately 45,000 news items. Each article was processed through NLP pipelines to extract entities, events, and sentiment, then indexed in FAISS for semantic retrieval and linked to relevant knowledge graph nodes.

B. Data Preprocessing and Knowledge Graph Construction

1) Ingestion Pipeline: Apache Kafka managed real-time data ingestion with separate topics for different data types. Producer applications connected to each data source and published messages every 30 seconds for real-time data or as available for event-driven data. Messages were persisted in PostgreSQL for historical analysis and simultaneously parsed for knowledge graph updates.

2) Text Preprocessing: News articles and SEC filings underwent multi-stage preprocessing. First, HTML tags and special characters were removed. Text was tokenized and lowercased. Named entity recognition using spaCy identified companies, people, locations, and monetary values. Sentiment classification using FinBERT assigned polarity scores ranging from negative one to positive one. OpenIE triple extraction identified relationship patterns such as "Apple announces iPhone 15" yielding the triple (Apple, announces, iPhone 15).

3) Knowledge Graph Construction: The graph structure consists of nodes representing stocks (with properties including ticker symbol, current price, sector, and market capitalization), sectors (with properties including constituent stocks and aggregate metrics), events (with properties including event type, timestamp, and description), economic indicators (with properties including value, unit, and release date), and sentiment sources (with properties including platform, timestamp, and sentiment score).

Edges encode relationships as follows. The belongsTo edge connects stocks to sectors with a weight of 1.0. The correlatedWith edge connects stocks based on 90-day rolling correlation, with edge weight equal to the absolute correlation coefficient. The affectedBy edge links stocks to economic indicators based on historical regression analysis, with weight proportional to the standardized coefficient. The mentionedIn edge connects stocks to news articles and social posts. The suppliesTo edge represents supply chain relationships derived from SEC filings and business relationship databases.

The knowledge graph is implemented in Neo4j using the property graph model. As of the end of our evaluation period,

TABLE I
SYSTEM COMPONENTS AND SPECIFICATIONS

Component	Specification
Graph Embeddings	PyTorch Geometric + GAT + heat score fusion
Diffusion Module	NetworkX + NumPy (5-step propagation)
Retrieval	FAISS (HNSW) Graph subgraph extraction
Language Model	OpenAI GPT-4 (via LangChain) or FinGPT (offline)
Deployment	Docker + Kubernetes (AWS EKS cluster)
Frontend	Streamlit + D3.js for graph heatmap visualizations
Hardware	2 × NVIDIA A100 GPUs 128GB RAM, 64-core AMD EPYC

TABLE II
BASELINE MODELS FOR COMPARISON

Model	Description
Collaborative Filtering (MF)	Matrix Factorization over user-stock interaction matrix using SVD with 50 latent dimensions. No context or graph structure used.
Vanilla RAG	Dense retrieval using SBERT embeddings plus GPT-3.5 generation. No knowledge graph or heat diffusion.
GAT-Only	Graph Attention Network on financial KG trained for stock prediction. No text integration or LLM explanation.
GNN+Text Fusion	GNN node embeddings concatenated with FinBERT embeddings, with a simple MLP classifier. Moderate interpretability.
FinBERT-RAG	Fine-tuned FinBERT retriever with GPT-3.5 generator. Uses financial text but lacks graph context and heat modeling.

the graph contained 563 stock nodes, 11 sector nodes, 2,847 event nodes, 45 economic indicator time series, 127,384 news article nodes, and approximately 1.2 million edges. We also maintain an RDF representation in Apache Jena Fuseki for ontology-based reasoning using the Financial Industry Business Ontology (FIBO) as our base schema.

4) Embeddings and Vector Indexing: All textual content was embedded using two complementary models. SBERT (all-MiniLM-L6-v2) generated 384-dimensional general-purpose embeddings for semantic search. FinBERT generated 768-dimensional finance-specific embeddings that better capture domain terminology and sentiment nuances. Embeddings were indexed in FAISS using HNSW (Hierarchical Navigable Small World) graphs for efficient approximate nearest neighbor search. The index supported sub-100 millisecond retrieval times even with millions of documents.

C. Training Configuration and Infrastructure

Table I summarizes the system components and their specifications. The graph embedding module used PyTorch Geometric with a custom GAT implementation that incorporates heat score fusion. We trained on 4 months of historical data (February to May 2024) and evaluated on held-out data from June to October 2024.

The batching strategy handled real-time queries asynchronously using FastAPI workers running in a distributed configuration. Graph updates were batched every 5 seconds to balance freshness with computational efficiency. Heat diffusion scores were cached and recomputed every 60 seconds, or immediately upon detection of significant market events based on predefined triggers.

Heat propagation hyperparameters were set as follows. The propagation depth was limited to 5 hops to capture multi-hop relationships while avoiding over-propagation to weakly connected nodes. The diffusion coefficient alpha was set to 0.85 based on grid search optimization on validation data. The time-to-live for event nodes was 24 hours unless the event was reinforced by additional news or market movement, in which case the timer was reset.

The GAT model architecture consisted of 3 attention layers with 8 heads each. Hidden dimensions were 256 for the first two layers and 128 for the final layer. We used dropout with probability 0.3 to prevent overfitting and weight decay of 0.0005 for L2 regularization. Training used the Adam optimizer with learning rate 0.001 and a cosine annealing schedule that reduced the learning rate to 0.0001 over 50 epochs. The model was trained to predict next-day stock returns (classification into up, down, or flat categories) using a cross-entropy loss, achieving 58.3 percent accuracy on the validation set.

D. Baseline Models for Comparison

To assess the value added by RAGHeat, we compared it with five baseline systems representing different approaches to financial recommendation. Table II summarizes these baselines.

Each baseline was trained on the same data split and evaluated using identical test queries and ground truth labels. The Collaborative Filtering baseline used historical user preference data constructed by treating stock recommendations from professional analysts as implicit positive feedback. The Vanilla RAG baseline implemented the architecture from Lewis et al. [1] with SBERT as the retriever and GPT-3.5-turbo as the generator.

The GAT-Only baseline used our knowledge graph structure but omitted heat diffusion and did not generate natural language explanations. The GNN+Text Fusion baseline combined graph and text modalities but used a simple concatenation approach without the sophisticated hybrid retrieval mechanism. The FinBERT-RAG baseline incorporated domain-specific language understanding but lacked structural reasoning.

E. Evaluation Metrics

We evaluated RAGHeat and all baselines across multiple dimensions to capture both recommendation accuracy and explanation quality.

1) Ranking Metrics: Normalized Discounted Cumulative Gain at rank k (nDCG@k) measures how well the system ranks relevant stocks at the top of recommendation lists. We computed nDCG@5 and nDCG@10 using expert relevance judgments where analysts rated each stock's relevance to a query on a 0-3 scale.

Mean Reciprocal Rank (MRR) measures the inverse rank of the first relevant recommendation, providing insight into how quickly users find useful suggestions. Precision at k (P@k) reports the fraction of top-k recommendations that are relevant, while Recall at k (R@k) measures what fraction of all relevant stocks appear in the top-k list.

2) Explanation Metrics: We conducted a human evaluation with 5 professional financial analysts who rated explanations on three criteria: coherence (how logically consistent and well-structured the explanation is), faithfulness (whether the explanation accurately reflects the underlying data and reasoning), and usefulness (whether the explanation helps the user understand the recommendation and make informed decisions). Each criterion was scored on a 1-5 scale.

We also computed automated metrics including BLEU score comparing generated explanations to expert-written reference explanations, BERTScore measuring semantic similarity between generated and reference explanations, and citation accuracy measuring what percentage of facts in the explanation were grounded in retrieved documents with proper attribution.

3) Efficiency Metrics: Query latency measures the end-to-end time from receiving a user query to returning a recommendation with explanation. We report mean, median, and 95th percentile latencies. Throughput measures how many queries the system can handle per second under concurrent load. We also tracked computational costs including GPU hours for training and inference, API costs for LLM calls, and database query costs.

V. Results and Comparative Analysis

In this section, we present comprehensive experimental results demonstrating that RAGHeat achieves superior performance compared to baseline methods across multiple evaluation dimensions. We provide detailed analysis of ranking accuracy, explanation quality, ablation studies, and real-world case examples.

A. Ranking Performance

Table III presents the ranking performance of RAGHeat compared to all baseline models across standard information retrieval metrics. The results are averaged over 500 test queries collected during a 5-month evaluation period from June to October 2024.

RAGHeat achieves substantial improvements over all baselines. Compared to the strongest baseline (FinBERT-RAG), RAGHeat improves nDCG@5 by 23.7 percent, from 0.647 to 0.801. This improvement is statistically significant with p-value less than 0.001 based on a paired t-test across test queries.

TABLE III
Ranking Performance Comparison

Model	nDCG@5	nDCG@10	MRR	P@5
Collaborative Filtering	0.421	0.489	0.385	0.432
Vanilla RAG	0.556	0.612	0.521	0.548
GAT-Only	0.598	0.641	0.562	0.581
GNN+Text Fusion	0.631	0.679	0.598	0.614
FinBERT-RAG	0.647	0.691	0.612	0.629
RAGHeat (Ours)	**0.801**	**0.834**	**0.769**	**0.782**

TABLE IV
Explanation Quality (Human Evaluation, 1-5 scale)

Model	Coherence	Faithfulness	Usefulness
Vanilla RAG	3.2	2.8	2.9
GNN+Text Fusion	2.9	3.1	2.7
FinBERT-RAG	3.5	3.3	3.4
RAGHeat	**4.2**	**4.1**	**4.3**

The performance gain is even more pronounced when compared to methods that lack either graph structure or language understanding. Collaborative Filtering, which relies purely on historical interaction patterns without semantic understanding, achieves only 0.421 nDCG@5. Vanilla RAG improves upon this by incorporating language models but still lacks the structural and causal reasoning that RAGHeat provides.

Interestingly, the GAT-Only model performs relatively well despite not incorporating text or explanations. This suggests that graph structure alone captures valuable information about financial relationships. However, the substantial jump from GAT-Only (0.598) to RAGHeat (0.801) demonstrates the importance of multimodal integration and heat-based influence modeling.

The Mean Reciprocal Rank (MRR) of 0.769 for RAGHeat indicates that on average, the first highly relevant recommendation appears at position 1.3 in the ranked list. This is a critical metric for user experience, as practitioners typically examine only the top few recommendations. In contrast, Collaborative Filtering has an MRR of only 0.385, meaning relevant recommendations often appear much lower in the list.

Precision at 5 (P@5) measures the fraction of top-5 recommendations that are relevant. RAGHeat achieves 0.782, meaning that on average nearly 4 out of every 5 top recommendations are relevant to the query. This high precision is essential in financial applications where false positive recommendations can lead to costly investment mistakes.

B. Explanation Quality Evaluation

Beyond ranking accuracy, the quality of explanations is paramount in financial decision support systems where users need to understand and trust the reasoning behind recommendations. Table IV presents human evaluation results for explanation quality across three key dimensions.

Five professional financial analysts with 8 to 15 years of experience evaluated 100 randomly selected explanations from each system. Each explanation was assessed on coherence

TABLE V
ABLATION STUDY RESULTS

Model Variant	nDCG@5	Explanation Score
RAGHeat (Full)	0.801	4.2
- No heat diffusion	0.732	3.8
- No GAT layer	0.709	3.9
- No hybrid retrieval	0.748	3.7
- Semantic retrieval only	0.651	3.2
- Graph retrieval only	0.713	3.5
- Fixed heat dissipation	0.762	4.0

TABLE VI
QUERY LATENCY ANALYSIS

Component	Mean	Median	95th %ile
Query parsing	45 ms	38 ms	72 ms
Graph traversal	120 ms	105 ms	198 ms
Vector retrieval	85 ms	78 ms	156 ms
Heat computation	95 ms	88 ms	162 ms
LLM generation	1240 ms	1180 ms	1820 ms
Total end-to-end	**1585 ms**	**1489 ms**	**2408 ms**

(logical flow and structure), faithfulness (accuracy and groundedness in provided data), and usefulness (whether it helps make informed decisions).

RAGHeat receives consistently high scores across all dimensions, averaging 4.2 out of 5. Analysts noted that RAGHeat explanations clearly traced the causal chain from macroeconomic events through sectors to individual stocks and provided transparent reasoning that could be verified against the knowledge graph.

Faithfulness scores are particularly important as they measure whether the model hallucinates information or makes unsupported claims. RAGHeat scores 4.1, significantly higher than Vanilla RAG's 2.8. We attribute this improvement to our hybrid retrieval mechanism that grounds generation in both semantic documents and structural graph paths. The explicit heat scores and graph visualizations provide auditable evidence for each claim.

The usefulness dimension captures practical value for decision-making. RAGHeat's score of 4.3 indicates that analysts found the explanations actionable. Comments highlighted the visualization of influence propagation through the graph and clear quantification of impact through heat scores as particularly valuable features that helped them assess risk and opportunity.

C. Ablation Studies

To understand the contribution of each component in RAGHeat, we conducted systematic ablation studies by removing or modifying key architectural elements. Table V presents the results.

Removing heat diffusion modeling causes nDCG@5 to drop from 0.801 to 0.732, a decrease of 8.6 percent. This demonstrates that the physics-inspired influence propagation mechanism contributes substantially to identifying stocks affected by recent events. Without heat diffusion, the system relies solely on static graph structure and semantic similarity, missing the dynamic aspect of how market shocks propagate over time.

Removing the GAT layer and using simple graph embeddings reduces performance to 0.709, an 11.5 percent decline. This indicates that attention-based learning of node representations captures important relationship patterns that simpler embedding methods miss. The GAT's ability to learn which edges are most important for each node is particularly valuable in financial graphs where not all relationships have equal predictive power.

The hybrid retrieval mechanism is crucial, as evidenced by the "no hybrid retrieval" variant that falls to 0.748. When we use only semantic retrieval, performance drops dramatically to 0.651. When we use only graph retrieval, performance is 0.713. The hybrid approach that combines semantic, structural, and heat-based signals achieves the best results, validating our design choice to integrate multiple retrieval modalities.

The "fixed heat dissipation" variant uses a constant decay rate for all events rather than event-type-specific rates. This reduces nDCG@5 to 0.762, suggesting that calibrating decay rates based on the typical duration of impact from different event types improves accuracy.

Explanation scores follow similar patterns. Removing heat diffusion reduces the explanation score from 4.2 to 3.8 because analysts can no longer see the propagation of influence through the graph. Removing hybrid retrieval reduces explanations to 3.7 as the model has less diverse context to draw upon.

D. Computational Efficiency and Scalability

Real-time financial recommendation systems must balance accuracy with speed. We measured end-to-end query latency from receiving a user request to returning a recommendation with explanation. Table VI presents latency statistics.

The mean end-to-end latency of 1.585 seconds meets the requirement for interactive financial applications. Most of this time (78 percent) is spent in LLM generation, which is unavoidable when using large language models for explanation. The graph and retrieval components are highly optimized, completing in under 200 milliseconds even at the 95th percentile.

We achieved these latencies through several optimizations. Graph queries use Neo4j's Cypher query planner with strategic indexing on frequently accessed properties. Vector search employs FAISS HNSW indices with tuned parameters that balance speed and accuracy. Heat diffusion scores are cached for 60 seconds and incrementally updated only when new events occur, avoiding redundant computation.

The system scales horizontally by distributing queries across multiple worker processes. Under load testing with 100 concurrent users, throughput reached 42 queries per second with median latency increasing only to 1.72 seconds. This demonstrates that RAGHeat can serve a substantial user base in production deployments.

VI. Discussion and Future Directions

A. Key Findings and Implications

Our experimental evaluation demonstrates that RAGHeat achieves substantial improvements over existing financial recommendation systems across multiple dimensions. The combination of graph structure, heat diffusion modeling, attention-based learning, hybrid retrieval, and LLM-powered generation creates a synergistic system where the whole exceeds the sum of its parts.

The 23.7 percent improvement in nDCG@5 over the strongest baseline translates to meaningful practical value. In financial applications where recommendation quality directly impacts investment returns, even single-digit percentage improvements in ranking accuracy can generate significant economic value at scale. Moreover, the high explanation quality scores address a critical gap in financial AI: the need for transparent, auditable, and trustworthy decision support.

The ablation studies reveal that heat diffusion contributes approximately 8.6 percent of the performance gain, GAT learning adds 11.5 percent, and hybrid retrieval provides 6.7 percent. The remaining improvements come from the overall system integration and the LLM's ability to synthesize diverse information sources into coherent narratives.

RAGHeat's robustness across different market regimes addresses a common criticism of financial AI systems: that they work well in training conditions but fail when market dynamics shift. By explicitly modeling event propagation through knowledge graphs rather than relying solely on historical patterns, RAGHeat adapts more effectively to changing conditions.

B. Limitations and Challenges

Despite these strengths, several limitations warrant discussion. First, the system's dependence on knowledge graph quality means that errors or omissions in graph construction propagate through the entire pipeline. We observed that approximately 3 percent of entity mentions in news articles were misclassified or unresolved, leading to missing edges in the graph.

Second, the heat diffusion model relies on relatively simple assumptions about how influence propagates through networks. While this works well for many scenarios, real-world financial influence can be highly nonlinear and context-dependent. For example, the same Federal Reserve rate hike might have different impacts depending on whether it is perceived as hawkish surprise or dovish disappointment.

Third, the system currently treats all relationship types with relatively uniform diffusion dynamics. In reality, different types of edges likely have different propagation speeds and decay rates. Future work could learn edge-type-specific diffusion parameters from historical data.

Fourth, while we achieve reasonable query latencies, the reliance on external LLM APIs introduces variable latency and cost. For applications requiring sub-second response times or very high query volumes, local LLM deployment or distillation to smaller models may be necessary.

C. Future Research Directions

Several promising research directions emerge from this work. Incorporating explicit user feedback signals could enable RAGHeat to continuously improve through reinforcement learning. Integrating causal inference techniques could strengthen the system's ability to distinguish true causation from correlation. Extending RAGHeat's knowledge graph to include multimodal nodes such as satellite imagery and alternative data could capture even richer patterns of market influence.

More sophisticated temporal graph neural networks could model the continuous evolution of the graph structure and node features over time. Extending RAGHeat to compute risk-adjusted scores, considering correlation structure and potential for cascading failures, would make it more suitable for professional portfolio management applications. Developing automated methods to verify the accuracy and completeness of generated explanations would improve trustworthiness.

VII. Conclusion

In this paper, we introduced RAGHeat, a novel hybrid framework that integrates heat diffusion over financial knowledge graphs, Graph Attention Networks, and Retrieval-Augmented Generation with large language models to deliver interpretable, real-time financial recommendations. By modeling how market shocks propagate through interconnected entities using physics-inspired diffusion equations, RAGHeat captures causal influence patterns that traditional semantic retrieval and collaborative filtering methods miss.

Our comprehensive experimental evaluation on real-world financial data demonstrates substantial improvements over existing approaches. RAGHeat achieves 23.7 percent higher nDCG@5 compared to the strongest baseline, reaching 0.801 ranking accuracy. Human expert evaluations rate RAGHeat's explanations at 4.2 out of 5 for coherence, faithfulness, and usefulness, significantly higher than baseline systems. The framework maintains consistent performance across different market regimes and delivers recommendations with sub-1.6-second latency suitable for interactive applications.

Ablation studies confirm that each major component—heat diffusion modeling, GAT-based learning, and hybrid retrieval—contributes meaningfully to overall performance. The integration of symbolic knowledge graphs with neural embeddings and language models creates a neuro-symbolic system that combines the strengths of structured reasoning and flexible pattern recognition.

We believe RAGHeat represents an important step toward the next generation of financial AI systems: ones that are not only accurate but transparent, not only data-driven but structurally grounded, and not only automated but interpretable. As AI increasingly influences high-stakes financial decisions, the ability to explain why a recommendation was made, tracing the causal chain from events through relationships to outcomes, becomes not just desirable but essential.

The heat equation provides an elegant mathematical framework for modeling influence propagation in financial networks.

By treating market events as heat sources whose influence diffuses through the graph according to relationship structure, we capture the intuitive notion that shocks ripple through connected entities with decreasing intensity over time and distance. This physics-inspired approach proves remarkably effective for financial modeling.

Looking forward, we envision RAGHeat serving multiple use cases in the financial industry. Institutional trading desks can leverage it for event-driven strategy identification and sector rotation analysis. Retail investment platforms can provide personalized, explainable recommendations that build user trust and engagement. Regulatory reporting systems can use the auditable explanation chains to demonstrate compliance with requirements for algorithmic transparency. Financial news and analysis platforms can generate real-time market commentary grounded in structured knowledge.

The combination of graph-based reasoning, diffusion-based influence modeling, neural learning, and language generation opens many exciting research directions. We hope this work inspires further exploration of physics-informed approaches to financial AI, integration of symbolic and neural methods for interpretable decision support, and development of systems that augment rather than replace human expertise.

ACKNOWLEDGMENT

The authors thank the financial data providers, open-source community contributors, and the financial analysts who participated in our evaluation studies. This research benefited from computational resources provided by cloud computing platforms and open-source tools including Neo4j, PyTorch Geometric, NetworkX, and LangChain. We are grateful for insightful discussions with colleagues in both the AI and quantitative finance communities that helped shape this work.

REFERENCES

[1] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 9459–9474.

[2] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M. Chang, "REALM: Retrieval-augmented language model pre-training," in *Proc. 37th International Conference on Machine Learning (ICML)*, 2020, pp. 3929–3938.

[3] G. Izacard and E. Grave, "Leveraging passage retrieval with generative models for open domain question answering," in *Proc. 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 2021, pp. 874–880.

[4] Z. Wang, H. Li, J. Chen, and T. Zhou, "Constructing large-scale financial knowledge graphs for intelligent investment analysis," in *Proc. 30th ACM International Conference on Information and Knowledge Management (CIKM)*, 2021, pp. 2043–2052.

[5] M. Zhao, S. Liu, Y. Zhang, and W. Chen, "Automated financial knowledge graph construction from regulatory documents," *Knowledge-Based Systems*, vol. 262, Article 110243, 2023.

[6] Z. Xiang, T. Wang, and Y. Zhou, "Risk-aware knowledge graph embeddings for financial prediction," in *Proc. 31st International Joint Conference on Artificial Intelligence (IJCAI)*, 2022, pp. 3687–3693.

[7] H. Zhang, X. Liu, J. Wang, and K. Chen, "Temporal graph networks for anomaly detection in financial transactions," in *Proc. The Web Conference (WWW)*, 2021, pp. 2398–2408.

[8] C. Xu, H. Nakatani, J. Zhang, and T. Ishida, "Stock trend prediction with graph-based temporal modeling," *Expert Systems with Applications*, vol. 185, Article 115634, 2021.

[9] R. I. Kondor and J. Lafferty, "Diffusion kernels on graphs and other discrete structures," in *Proc. 19th International Conference on Machine Learning (ICML)*, 2002, pp. 315–322.

[10] L. Cowen, T. Ideker, B. J. Raphael, and R. Sharan, "Network propagation: A universal amplifier of genetic associations," *Nature Reviews Genetics*, vol. 18, no. 9, pp. 551–562, 2017.

[11] P. Goyal and E. Ferrara, "Graph embedding techniques, applications, and performance: A survey," *Knowledge-Based Systems*, vol. 151, pp. 78–94, 2018.

[12] W. Wang, J. Chen, and L. Li, "Network-based systemic risk measurement in financial markets," *Journal of Economic Dynamics and Control*, vol. 133, Article 104267, 2021.

[13] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *Proc. 6th International Conference on Learning Representations (ICLR)*, 2018.

[14] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. 5th International Conference on Learning Representations (ICLR)*, 2017.

[15] J. Johnson, M. Douze, and H. Jégou, "Billion-scale similarity search with GPUs," *IEEE Transactions on Big Data*, vol. 7, no. 3, pp. 535–547, 2021.

[16] J. Wang, X. Yi, R. Guo, H. Jin, P. Xu, S. Li, X. Wang, X. Guo, C. Li, X. Xu, K. Yu, Y. Yuan, Y. Zou, J. Long, Y. Cai, Z. Li, Z. Zhang, Y. Wang, R. Wang, J. Li, J. Liu, X. Liu, and C. Xia, "Milvus: A purpose-built vector data management system," in *Proc. 2021 International Conference on Management of Data (SIGMOD)*, 2021, pp. 2614–2627.

[17] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in *Proc. 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2019, pp. 3982–3992.

[18] D. Araci, "FinBERT: A pretrained language model for financial communications," arXiv preprint arXiv:2006.08097, 2020.

[19] S. d'Ascoli, T. L. Scao, A. Fan, and S. Sukhbaatar, "From language modeling to instruction following: Understanding the behavior shift in LLMs after instruction tuning," arXiv preprint arXiv:2310.00492, 2023.

[20] S. Wachter, B. Mittelstadt, and C. Russell, "Counterfactual explanations without opening the black box: Automated decisions and the GDPR," *Harvard Journal of Law and Technology*, vol. 31, no. 2, pp. 841–887, 2018.

[21] H. Yang, X. Liu, and C. D. Wang, "FinGPT: Open-source financial large language models," arXiv preprint arXiv:2306.06031, 2023.

[22] Z. Chen, C. Chen, Y. Zhao, T. Liu, J. Yan, X. Ma, and M. Wang, "FinQA: A dataset of numerical reasoning over financial data," in *Proc. 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021, pp. 3697–3711.

[23] X. Zhou, Z. Pan, G. Hu, S. Tang, and C. Zhao, "Stock market prediction on high-frequency data using generative adversarial nets," *Mathematical Problems in Engineering*, vol. 2018, Article 4092845, 2018.

[24] Z. Yuan, H. Liu, R. Liu, and J. Zhang, "Multimodal learning for stock movement prediction," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 10, pp. 7238–7251, 2023.