Towards Explainable Multi-Agent Systems with GraphRAG: A Guide to Explain Explanations in the Audit Domain

Emanuel Slany[1], Jonas Amling[1], Alexander Frummet[1], Moritz Lang[1] and Stephan Scheele[2]

[1] dab:GmbH, Hans-Obser-Straße 12, Deggendorf, Germany
[2] OTH Regensburg, Prüfeninger Straße 58, Regensburg, Germany

Abstract

Combining Large Language Models (LLMs) with graph-based retrieval augmented generation (GraphRAG) in multi-agent Artificial Intelligence (AI) systems promises new levels of process automation, even in highly restricted and regulated domains such as financial audits. Since trustworthiness is essential, we introduce an architecture designed with it in mind: Every agent is either inherently explainable or its decision-making mechanism is rendered transparent through post-hoc Explainable AI techniques. Procedural knowledge, agent outcomes, and their explanations are represented as nodes in a knowledge graph accessible via GraphRAG, while LLMs are confined as semantic translators, bridging graph and natural-language representations. However, when user prompts involve multiple agent subgraphs, their individual agent attribution is still opaque. We introduce an occlusion-based agent importance metric that quantifies the relative attribution of each serialized agent subgraph. Our evaluation demonstrates that the quantification of agent importance is feasible, while the presence of systematic agent interactions or narrative context effects requires further investigation.

Keywords

Explainable AI, Knowledge Graphs, GraphRAG, Multi-Agent Systems, Large Language Models, Audit

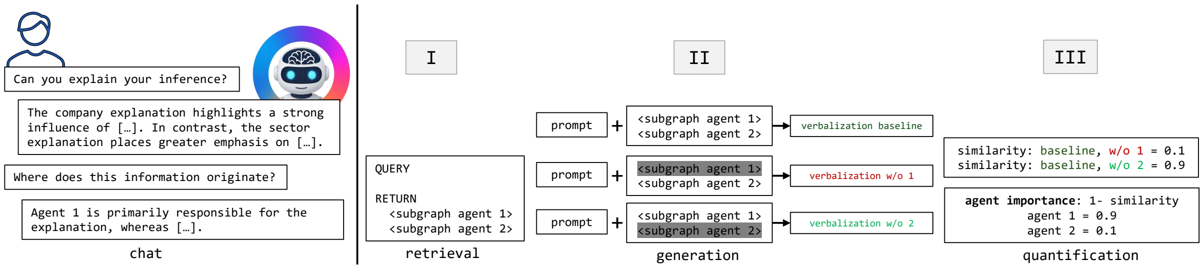
1. Introduction

Auditors routinely engage in tasks that are both domain-knowledge-intensive and highly redundant [1]. A typical scenario involves the extraction of risk indicators from annual audit reports in a specific industry, followed by a comparison with prior years to inform the auditor's examination strategy. Automation of such processes using agentic AI systems is challenging but feasible – and, if successful, can significantly enhance efficiency in financial auditing [2]. However, the validity of auditors' assessments is crucial as any inaccuracies could pose severe risks. Agentic AI systems in auditing must therefore deliver thorough transparency and interpretability in their decision making. Yet, auditing regulations also emphasize the indispensable role of human judgment, highlighting the tradeoff between automation and accountability – particularly where digital processes intersect with professional liability. We aim to advance trustworthy automation of audits with multi-agent AI systems and graph-based retrieval augmented generation (GraphRAG) [3, 4, 5]. The procedural sequencing of agents is encoded within a knowledge graph. Each agent is either intrinsically explainable or combined with Explainable AI (XAI) techniques such as feature importance values [6]. By integrating knowledge graph representations of agent behaviors and outcomes with XAI methods and Large Language Models (LLMs) [4], we enable interaction through natural language, aiming to ensure trust in agent outcomes.

Related Work. While LLMs have been used for several downstream tasks such as text generation [7], prediction [8], explanatory model finetuning [9], or explainable exploration of event data [10, 11] the complementary conjunction of LLMs and knowledge graphs under the scope of transparency has recently received greater attention [4]. Currently, LLMs are frequently discussed in relation to agentic AI systems [12]. Knowledge graphs have traditionally been leveraged for explainability [13]. The short history of graph representations for LLMs [14] has now transitioned into XAI applications [15].

XAI-KRKG@ECAI25: First International ECAI Workshop on eXplainable AI, Knowledge Representation and Knowledge Graphs, October 25–30, 2025, Bologna, Italy

emanuel.slany@dab-gmbh.de (E. Slany); jonas.amling@dab-gmbh.de (J. Amling); alexander.frummet@dab-gmbh.de (A. Frummet); moritz.lang@dab-gmbh.de (M. Lang); stephan.scheele@oth-regensburg.de (S. Scheele)
0000-0001-5958-8425 (E. Slany); 0009-0000-1554-9950 (J. Amling); 0000-0002-5982-7104 (A. Frummet); 0000-0003-0787-3181 (S. Scheele)



Can you explain your inference?

The company explanation highlights a strong influence of […]. In contrast, the sector explanation places greater emphasis on […].

Where does this information originate?

Agent 1 is primarily responsible for the explanation, whereas […].

```
QUERY

RETURN
  <subgraph agent 1>
  <subgraph agent 2>
```

similarity: baseline, w/o 1 = 0.1
similarity: baseline, w/o 2 = 0.9

agent importance: 1- similarity
agent 1 = 0.9
agent 2 = 0.1

Figure 1: Agent importance. Problem statement (left) and proposed solution (right). With GraphRAG in a multi-agent AI system, the information origin becomes intractable. Our method occludes parts of the context retrieved from a knowledge graph. Each part corresponds to an agent. We assess the relevance of each agent by comparing the semantic similarity of each verbalization with occluded context to its baseline.

Problem. Despite the decision-making mechanism behind each agent has become tractable by XAI techniques, the LLM result does not disclose the attribution of agents to the verbalization (Figure 1, left). Consequently, an auditor in our running example might still be confronted with over- or under-amplified risks, undermining the organization of the entire examination strategy.

Solution. We quantify agent importance in three steps (Figure 1, right): (i) We extract the relevant subgraph sequence from the knowledge graph. (ii) Then, we generate verbalizations: a baseline representing the entire subgraph sequence and separate variants for each subgraph occlusion. (iii) Finally, we quantify the agent importance by semantic similarity metrics. The application of our approach enables auditors to understand the origin of information and evidence in multi-agent system outcomes.

Contributions. Our core contributions are twofold: First, we present a multi-agent GraphRAG architecture in a case study using open-source finance data sets that mimic the dynamics of annual audits (Section 2). Second, we formalize the proposed agent importance approach (Section 3).

Research Questions. We pose two research questions: (**R1**) Does agent importance eligibly quantify the attribution of agent contexts retrieved from knowledge graphs to LLM verbalizations? And, (**R2**) does agent importance account for biases in the narrative task specification? The research questions are preliminary evaluated by an ablation study (Section 4).

2. Case Study

To address the domain-specific requirements of financial auditing without disclosing proprietary information, we validate our architecture in a controlled scenario using open-source datasets[1]: Specifically, we model the influence of macroeconomic indicators on the relative annual return of individual company stocks and the average annual return of stocks within the same industry sector. The primary goal in this setting is to compare the explanatory factors for the company prediction with the reasons driving the sector prediction. We employ labeled property graphs for knowledge representations[2].

The agents graph (Figure 2, left) encodes the procedural task knowledge and is responsible for the task execution. Among multiple properties, agents have an API address and Pydantic models for input and output validation[3]. Natural language queries are converted to an input model and trigger API execution. The outcome of which is verified using the output model. We distinguish between two types of agents (Figure 3): static agents, which rely on deterministic computations (write) or classical probabilistic methods (predict_company, predict_sector), and LLM agents, which generate natural language responses (compare). Figure 3 depicts their functionality in detail.

[1] The following data sets are merged wrt. the date and the company symbol: economic indicators: https://www.kaggle.com/datasets/alfredkondoro/u-s-economic-indicators-1974-2024?select=cpi_data.csv, stock prices: https://www.kaggle.com/datasets/camnugent/sandp500?select=all_stocks_5yr.csv, stock information: https://www.kaggle.com/datasets/paytonfisher/sp-500-companies-with-financial-information?select=financials.csv, all 6th June 2025.

[2] https://memgraph.com/, 6th June 2025.

[3] https://docs.pydantic.dev/latest/, 6th June 2025.

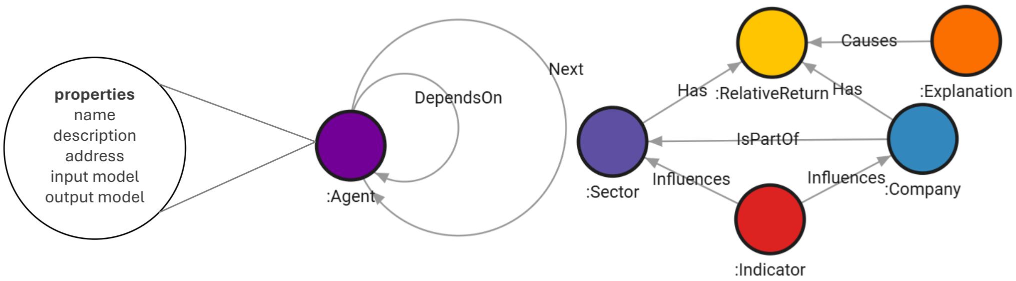
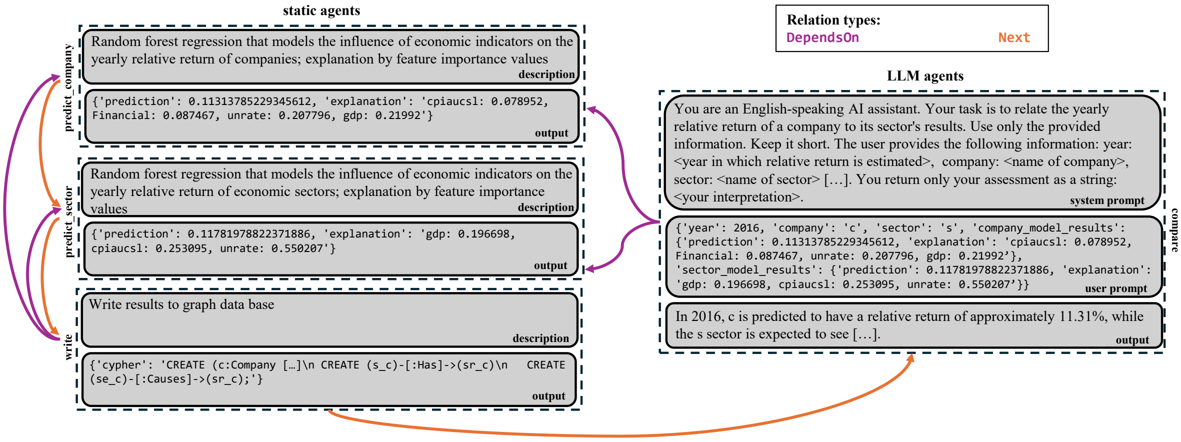


Figure 2: Knowledge graph. Labels and relations of the results (right) and the agent (left) knowledge graph. Properties are included for the latter.



Random forest regression that models the influence of economic indicators on the yearly relative return of companies; explanation by feature importance values

```
{'prediction': 0.11313785229345612, 'explanation': 'cpiaucsl: 0.078952,
Financial: 0.087467, unrate: 0.207796, gdp: 0.21992'}
```

Random forest regression that models the influence of economic indicators on the yearly relative return of economic sectors; explanation by feature importance values

```
{'prediction': 0.11781978822371886, 'explanation': 'gdp: 0.196698,
cpiaucsl: 0.253095, unrate: 0.550207'}
```

Write results to graph data base

```
{'cypher': 'CREATE (c:Company […]\n CREATE (s_c)-[:Has]->(sr_c)\n   CREATE
(se_c)-[:Causes]->(sr_c);'}
```

You are an English-speaking AI assistant. Your task is to relate the yearly relative return of a company to its sector's results. Use only the provided information. Keep it short. The user provides the following information: year: <year in which relative return is estimated>, company: <name of company>, sector: <name of sector> […]. You return only your assessment as a string: <your interpretation>.

```
{'year': 2016, 'company': 'c', 'sector': 's', 'company_model_results':
{'prediction': 0.11313785229345612, 'explanation': 'cpiaucsl: 0.078952,
Financial: 0.087467, unrate: 0.207796, gdp: 0.21992'},
'sector_model_results': {'prediction': 0.11781978822371886, 'explanation':
'gdp: 0.196698, cpiaucsl: 0.253095, unrate: 0.550207'}}
```

In 2016, c is predicted to have a relative return of approximately 11.31%, while the s sector is expected to see […].

Figure 3: Agents. Detailed description with an exemplary output of agent nodes conditioned on static models (static agents, left) and a verbalizing agent node (LLM agents, right). System and user prompts are included for the latter. The `DependsOn` (violet) and `Next` (orange) relations illustrate their dependencies.

The results graph (Figure 2, right) contains nodes generated with Cypher queries [5] yield by the write method. It encodes domain knowledge in the sense that it models dependencies between the feature and the target spaces and facilitates agent explainability.

We combine LLMs[4] and knowledge graphs as follows: Each natural language prompt is mapped to a predefined Cypher query, which either triggers agents or retrieves subgraphs. Novel queries are generated for out-of-distribution requests. The obtained subgraphs serve as additional context for the initial prompt such that the generated verbalization contains only graph-encoded information.

In proposing an XAI method for multi-agent systems integrated with GraphRAG, we aim to address a core contradiction, which we term the explanation paradox: All agents in our architecture are either intrinsically explainable or accompanied by XAI techniques. Via GraphRAG, a LLM exclusively accesses this precomputed information. Still, recipients of a LLM verbalization are not yet aware of which agent contributed to the answer. In particular, when prompting a LLM to combine explanations, the user cannot comprehend the relevance of each incorporated agent. The agent importance method proposed in the next section is designed to resolve this paradox.

3. Methods

Agent importance quantifies the influence of a serialized subgraph of an agent on the generated LLM verbalization for a given task in a GraphRAG scenario. The agent importance approach approximates feature attributions inspired by Shapley Additive Explanations (SHAP) [16]. It systematically occludes serialized subgraphs within a GraphRAG multi-agent system, assuming a fixed verbalization task. Given a task, we retrieve and serialize the subgraphs of all addressed agents to generate a baseline verbalization. Next, we occlude one subgraph at a time and regenerate verbalizations. The intuitive idea is: The

[4] https://platform.openai.com/docs/models/gpt-4o-mini, 28th May 2025.

more similar the occluded output is to the baseline, the less the agent contributes to the verbalization. Attribution values are estimated by semantic similarity [17]. We compute the cosine similarity between the embeddings of the occluded and baseline verbalizations; agent importance is defined as the inverse of the calculated similarity.

Definition. Let s_i be a subgraph serialization of a knowledge graph, $g = (s_1, \ldots, s_i, \ldots, s_n) \in G$ be an ordered sequence of serialized subgraphs, $p \in P$ and $v \in V$ denote prompts and verbalizations, and $f : P \times G \to V$ be a LLM[5], generating a verbalization given a prompt and a subgraph sequence. Let $h : G \times G \to \mathbb{R}$ represent the cosine distance between subgraph sequence embeddings[6]. Let

$$g^{\setminus s_i} := (s_1, \ldots, s_{i-1}, \epsilon, s_{i+1}, \ldots, s_n) \text{ with } |\epsilon| = 0$$

denote the occlusion of s_i in g – practically, a substitution of s_i with an empty sequence ϵ. Finally, let

$$\boldsymbol{\phi}_{f,p}(g) := \phi_{f,p}(g, g^{\setminus s_i}) \text{ for all } i \in (1, \ldots, n) \text{ with } \phi_{f,p}(g, g^{\setminus s_i}) = 1 - h(g, g^{\setminus s_i})$$

measure the attribution of each subgraph in the subgraph sequence given a model f and prompt p. We assume that subgraph serializations are agent outcomes and thus can be mapped to their origin.

Example. Suppose an AI literature search uses three agents: one for researchers, one for publications, and one for scientific areas, with results encoded as a knowledge graph, serialized as $g = (s_1, s_2, s_3)$.

subgraph	**label**	**node**
s_1	`:Researcher`	'Ina Marie'
s_2	`:Publication`	'XAI in Multi-Agent Systems for Audit: Why Our Method is Important'
s_3	`:Area`	'Financial Audits'

Suppose the following user prompt for a GraphRAG system, which accesses the (sequence of) subgraph serializations: $p =$'Summarize the publications of Ina Marie in the field of financial audits'.

context	v	$\phi_{f,p}(g, g^{\setminus s_i})$
g	'Ina Marie has published several influential works, including XAI in Multi-Agent Systems for Audit: Why Our Method is Important, which highlights the importance of explainable AI in enhancing the efficiency and effectiveness of financial audits.'	0.0
$g^{\setminus s_1}$	'The publication XAI in Multi-Agent Systems for Audit: Why Our Method is Important explores the role of explainable AI in improving the accuracy and efficiency of financial audits.'	0.0482
$g^{\setminus s_2}$	'Ina Marie has made significant contributions to the field of financial audits through her research on innovative approaches to auditing practices.'	**0.3984**
$g^{\setminus s_3}$	'Ina Marie's work focuses on the applications of explainable AI in the auditing process.'	0.2288

The publication node subgraph frames the most important context for the verbalization.

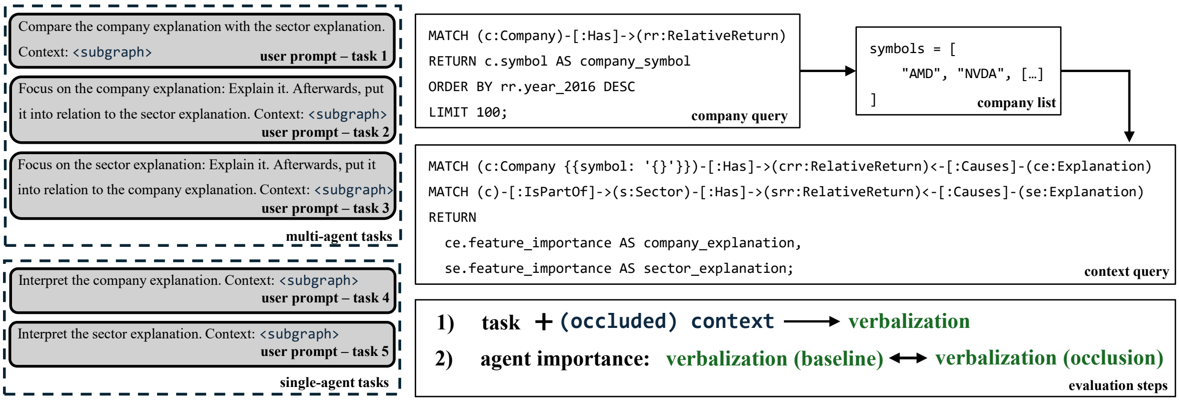
The original SHAP framework is characterized by theoretical properties – local accuracy, faithfulness, missingness, and consistency – which have been formally established over the course of its development [18, 16, 19]. Furthermore, its computational feasibility has been systematically analyzed [20]. Despite proposing our method in a mathematically sound style, we leave a formal assessment for future work, yet highlight the importance of which. In contrast, the experiments of the subsequent section provide quantitative evidence through an ablation study that evaluates the importance of agent subgraph serializations in diverse tasks.

4. Experiments

Suppose two single-agent – explanation interpretations – and three multi-agent tasks – explanation comparisons, either with or without narrative biases – in the domain of Section 2. We select the 100 companies with the highest relative return in 2016 and retrieve the subgraphs for the company and the corresponding sector explanation. We exploit the following evaluation strategy (Figure 4): (i) obtain occluded and baseline verbalizations, and (ii) estimate the agent importance (Definition 3).

[5] https://platform.openai.com/docs/models/gpt-4o-mini, 28th May 2025.

[6] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2, 04th June 2025.



Compare the company explanation with the sector explanation. Context: <subgraph>

Focus on the company explanation: Explain it. Afterwards, put it into relation to the sector explanation. Context: <subgraph>

Focus on the sector explanation: Explain it. Afterwards, put it into relation to the company explanation. Context: <subgraph>

Interpret the company explanation. Context: <subgraph>

Interpret the sector explanation. Context: <subgraph>

```
MATCH (c:Company)-[:Has]->(rr:RelativeReturn)
RETURN c.symbol AS company_symbol
ORDER BY rr.year_2016 DESC
LIMIT 100;
```

```
symbols = [
    "AMD", "NVDA", […]
]
```

```
MATCH (c:Company {{symbol: '{}'}})-[:Has]->(crr:RelativeReturn)<-[:Causes]-(ce:Explanation)
MATCH (c)-[:IsPartOf]->(s:Sector)-[:Has]->(srr:RelativeReturn)<-[:Causes]-(se:Explanation)
RETURN
  ce.feature_importance AS company_explanation,
  se.feature_importance AS sector_explanation;
```

context query
1) task + (occluded) context → verbalization
2) agent importance: verbalization (baseline) ↔ verbalization (occlusion)
evaluation steps

Figure 4: Experimental setup. For each of the five experimental (single- and multi-agent) tasks and for each of the 100 companies obtained from the results graph (company query), obtain subgraphs containing an explanation for the company and the sector prediction (context query). Then, execute two steps: 1) Obtain verbalizations for the entire context and possible occlusions. 2) Estimate the agents' importance values by semantic comparison.

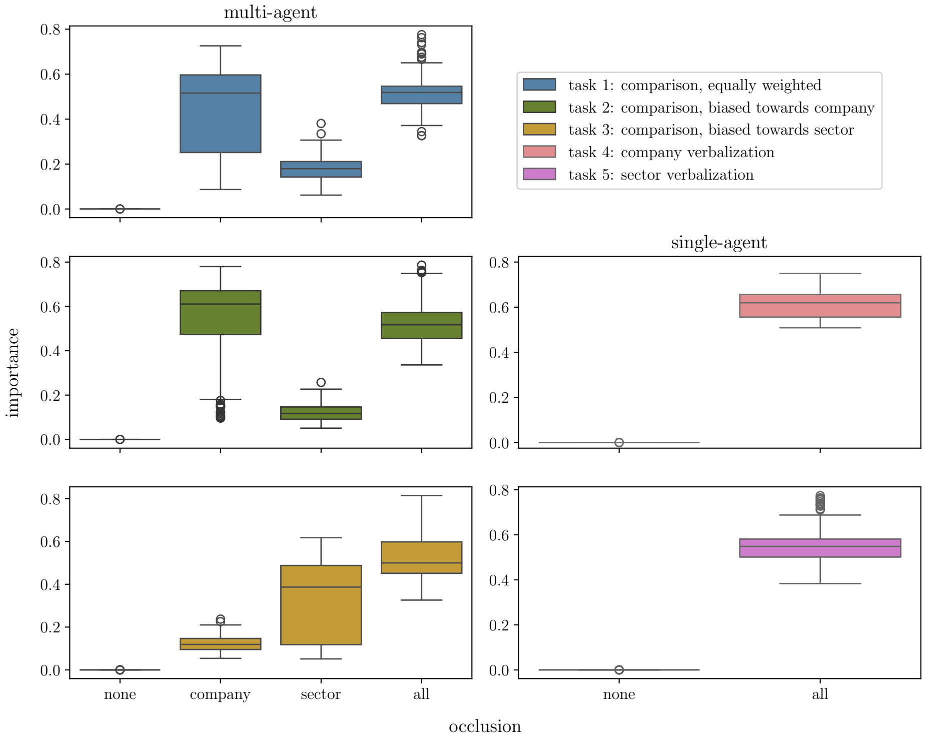


Figure 5: Results. Comparison of the impact of agent context occlusion on agent importance values for multi-agent (left, tasks 1-3) and single-agent (right, tasks 4 and 5) tasks. Each subplot contains boxplots that estimate importance values by occlusion of one or all agent contexts in relation to the baseline (none).

Figure 5 presents one subplot per task, with each containing a boxplot corresponding to an agent importance evaluation. Three observations can be drawn from the results: (i) Occluding agents presumed to be relevant increases their attribution scores. (ii) Narrative task context effects are only partially reflected, as the sector-level agent generally receives lower attribution. (iii) In multi-agent configurations, the ablation study (the additional occlusion of none or all subgraphs) demonstrates that agent importance scores are not strictly additive.

Within a narrow experimental scope, the research questions can be answered as follows: (**R1**) Agent importance eligibly quantifies the attribution of agent contexts retrieved from knowledge graphs. Context effects undermine desirable mathematical properties such as additivity. (**R2**) Agent importance tends to account for biases in the task specification. The experiments reveal narrative specificities of the domain such as an elevated importance of the term *company* compared to the term *sector*.

5. Discussion

The integration of structured knowledge, XAI, and LLMs enables the automation of redundant tasks even in domains requiring substantial domain expertise, such as financial auditing. The use of multi-agent systems in combination with GraphRAG offers a promising architecture – provided that each agent discloses its decision making. Determining the extent to which a LLM relies on the outputs of individual agents remains intractable. This challenge is encapsulated in what we term the explanation paradox: Even if every system component is individually explainable, the attribution of each component remains opaque. The agent importance method addresses this gap. It estimates the relative contribution of each agent by computing the inverse of the semantic similarity between the LLM's baseline verbalization and the verbalization generated after occluding the respective agent.

Main Findings. Three findings emerge from our contribution: (i) Quantifying agent importance addresses a critical gap on the way towards trustworthy multi-agent systems in high-stake domains; yet is feasible as evidenced by our preliminary results. (ii) While our method is mathematically grounded, it remains theoretically incomplete. Assumptions derived from supervised learning models, e.g., the ones from SHAP [16], are not seamlessly transferable to generative models. (iii) Our findings hint at two key challenges: (a) Narrative context effects introduced by task prompts may bias the relevance estimates. And, (b) correlations between agent subgraphs might confound the measured importance values.

Related Results. Our main findings can be situated into an area of methods, which aims to overcome semantic limitations [21] or *hallucinations* [22] in LLMs. What has started as a systematic combination of supplementary information with LLMs [23], has transitioned into a structured representation of domain knowledge with researchers questioning their attribution [15]. Attribution methods have a rich tradition in XAI [6], of which some of them obtain their importance estimate by occlusion [16]. Closely related to our approach are [24], [25], and [26], who study the alignment of multiple information sources in traditional LLM settings and classic or graph-enhanced RAG architectures, respectively.

Limitations. Four major limitations can be identified in our work: (i) Domain: In general, our approach is domain-agnostic. However, although it is motivated by the audit domain, we abstract the case study to the broader finance domain. Due to domain specificities, our method may not fully generalize to the intended context. (ii) Applicability: The motivating example centers on the aggregation of agent explanations. While we emphasize that agent importance is applicable to any task prompt in a multi-agent system employing GraphRAG, our experiments are limited to explanation interpretations or comparisons. (iii) Experiments: The experimental design is sparse, and the presented results are preliminary. More comprehensive empirical validation and a baseline comparison are necessary to draw robust conclusions. (iv) Theoretical contradictions: The findings expose unresolved theoretical challenges, e.g., the impact of prompt phrasing, contextual influence in natural language, and correlations among subgraphs. Also, in contrast to many predictive models, LLM outcomes are not deterministic.

Future Work. First, we will enhance agent execution from natural language prompts by enabling the parallelization of multiple agent calls. Second, we aim to improve and publicly release a user interface to facilitate more intuitive and accessible interactions with the system. With respect to the proposed attribution method, we will formally derive and prove desirable attribution properties within the context of generative AI systems. Lastly, we will extend our experimental evaluation by defining a range of diverse tasks across various multi-agent data sets and comparing attribution results across LLMs.

Reproducibility Statement. We used open-source data and provided each model, its corresponding instruction, and graph query. An extended evaluation is in progress, and we plan to publish the code in the future.

Ethical Considerations. There are no specific ethical concerns to declare. Our overarching goal is to develop trustworthy multi-agent systems. We believe transparency in the decision-making process is essential.

Declaration on Generative AI. During the preparation of this work, we used `gpt-4o` and Grammarly for spell-checking and grammar correction, and `gpt-4o-mini` for experimental purposes[7].

Acknowledgments. This article is part of the project BayVFP Data Tales (# DIK-2407-00007 // DIK0660/01).

[7] `gpt-4o`: https://platform.openai.com/docs/models/gpt-4o, Grammarly: https://www.grammarly.com/, `gpt-4o-mini`: https://platform.openai.com/docs/models/gpt-4o-mini, all 28th May 2025.

References

[1] J. Kokina, S. Blanchette, T. H. Davenport, D. Pachamanova, Challenges and opportunities for artificial intelligence in auditing: Evidence from the field, International Journal of Accounting Information Systems 56 (2025) 100734. doi:10.1016/j.accinf.2025.100734.

[2] R. Samiolo, C. Spence, D. Toh, Auditor judgment in the fourth industrial revolution, Contemporary Accounting Research 41 (2023) 498–528. doi:10.1111/1911-3846.12901.

[3] T. Guo, X. Chen, Y. Wang, R. Chang, S. Pei, N. V. Chawla, O. Wiest, X. Zhang, Large Language Model Based Multi-agents: A Survey of Progress and Challenges, in: Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024, Jeju, South Korea, August 3-9, 2024, ijcai.org, 2024, pp. 8048–8057. URL: https://www.ijcai.org/proceedings/2024/890.

[4] S. Pan, L. Luo, Y. Wang, C. Chen, J. Wang, X. Wu, Unifying Large Language Models and Knowledge Graphs: A Roadmap, IEEE Trans. Knowl. Data Eng. 36 (2024) 3580–3599. doi:10.1109/TKDE.2024.3352100.

[5] N. Francis, A. Green, P. Guagliardo, J. Holland, P. Llewellyn, P. Selmer, T. Taylor, P. Wood, Cypher: An Evolving Query Language for Property Graphs, in: Proceedings of the 2018 International Conference on Management of Data (SIGMOD), ACM, 2018, pp. 1433–1445. doi:10.1145/3183713.3190657.

[6] G. Schwalbe, B. Finzel, A comprehensive taxonomy for explainable artificial intelligence: a systematic survey of surveys on methods and concepts, Data Mining and Knowledge Discovery (2023). doi:10.1007/s10618-022-00867-8.

[7] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language Models are Few-Shot Learners, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual, 2020. URL: https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

[8] H. Tang, C. Zhang, M. Jin, Q. Yu, Z. Wang, X. Jin, Y. Zhang, M. Du, Time Series Forecasting with LLMs: Understanding and Enhancing Model Capabilities, SIGKDD Explor. Newsl. 26 (2025) 109–118. doi:10.1145/3715073.3715083.

[9] E. Slany, S. Scheele, U. Schmid, Explanatory Interactive Machine Learning with Counterexamples from Constrained Large Language Models, in: A. Hotho, S. Rudolph (Eds.), KI 2024: Advances in Artificial Intelligence, Springer Nature Switzerland, Cham, 2024, pp. 324–331. doi:10.1007/978-3-031-70893-0_26.

[10] J. Amling, E. Slany, C. Dormagen, M. Kretschmann, S. Scheele, Bridging the Interpretability Gap in Process Mining: A Comprehensive Approach Combining Explainable Clustering and Generative AI, in: Explainable Artificial Intelligence - 3rd World Conference, xAI 2025, Istanbul, Turkey, July 09-11, 2025, Springer, to appear.

[11] C. Dormagen, J. Amling, S. Scheele, U. Schmid, Explaining Process Behavior: A Declarative Framework for Interpretable Event Data, in: 3rd World Conference, xAI 2025, Istanbul, Turkey, July 09-11, 2025, Late-breaking Work, Demos and Doctoral Consortium, CEUR-WS, to appear.

[12] S. Hosseini, H. Seilani, The role of agentic AI in shaping a smart future: A systematic review, Array 26 (2025) 100399. doi:https://doi.org/10.1016/j.array.2025.100399.

[13] E. Rajabi, K. Etminani, Knowledge-graph-based explainable AI: A systematic review, Journal of Information Science 50 (2024) 1019–1029. doi:10.1177/01655515221112844.

[14] J. Sun, C. Xu, L. Tang, S. Wang, C. Lin, Y. Gong, L. M. Ni, H.-Y. Shum, J. Guo, Think-on-Graph: Deep and Responsible Reasoning of Large Language Model on Knowledge Graph, 2024. arXiv:2307.07697.

[15] R. Wu, P. Cai, J. Mei, L. Wen, T. Hu, X. Yang, D. Fu, B. Shi, KG-TRACES: Enhancing Large Language Models with Knowledge Graph-constrained Trajectory Reasoning and Attribution Supervision,

2025. arXiv:2506.00783.

[16] S. M. Lundberg, S. Lee, A Unified Approach to Interpreting Model Predictions, in: I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, 2017, pp. 4765–4774. URL: https://proceedings.neurips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html.

[17] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi, BERTScore: Evaluating Text Generation with BERT, in: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020, 2020. URL: https://openreview.net/forum?id=SkeHuCVFDr.

[18] H. P. Young, Monotonic solutions of cooperative games, International Journal of Game Theory 14 (1985) 65–72. doi:10.1007/BF01769885.

[19] L. Heidrich, E. Slany, S. Scheele, U. Schmid, FairCaipi: A Combination of Explanatory Interactive and Fair Machine Learning for Human and Machine Bias Reduction, Machine Learning and Knowledge Extraction 5 (2023) 1519–1538. doi:10.3390/make5040076.

[20] M. Arenas, P. Barcelo, L. Bertossi, M. Monet, On the Complexity of SHAP-Score-Based Explanations: Tractability via Knowledge Compilation and Non-Approximability Results, Journal of Machine Learning Research 24 (2023) 1–58. URL: http://jmlr.org/papers/v24/21-0389.html.

[21] K. J. Hammond, D. B. Leake, Large Language Models Need Symbolic AI, in: A. S. d'Avila Garcez, T. R. Besold, M. Gori, E. Jiménez-Ruiz (Eds.), Proceedings of the 17th International Workshop on Neural-Symbolic Learning and Reasoning, La Certosa di Pontignano, Siena, Italy, July 3-5, 2023, volume 3432 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 204–209. URL: https://ceur-ws.org/Vol-3432/paper17.pdf.

[22] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, T. Liu, A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions (2023). doi:10.48550/ARXIV.2311.05232.

[23] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. tau Yih, T. Rocktäschel, S. Riedel, D. Kiela, Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, 2021. arXiv:2005.11401.

[24] X. Yue, B. Wang, Z. Chen, K. Zhang, Y. Su, H. Sun, Automatic Evaluation of Attribution by Large Language Models, in: H. Bouamor, J. Pino, K. Bali (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2023, Association for Computational Linguistics, Singapore, 2023, pp. 4615–4635. doi:10.18653/v1/2023.findings-emnlp.307.

[25] A. Abolghasemi, L. Azzopardi, S. H. Hashemi, M. de Rijke, S. Verberne, Evaluation of Attribution Bias in Retrieval-Augmented Large Language Models, 2024. arXiv:2410.12380.

[26] J. Gao, X. Zou, Y. Ai, D. Li, Y. Niu, B. Qi, J. Liu, Graph Counselor: Adaptive Graph Exploration via Multi-Agent Synergy to Enhance LLM Reasoning, 2025. arXiv:2506.03939.