Erschienen in: 2025 IEEE Visual Analytics Science and Technology Challenge (VAST Challenge). Piscataway, NJ: IEEE, 2025, S. 11-12. ISBN 979-8-3315-9090-1
https://doi.org/10.1109/vast-challenge69463.2025.00010

MARINE Analytics for Radio Interception and Naval Environment

Willi Kneer* Jonathan Rentschler† Daniel A. Keim‡ Julius Rauscher§ Lucas Joos¶

University of Konstanz

Abstract

On the fictional island of *Oceanus*, journalist **Clepper Jensen** suspects that certain individuals are involved in illegal activities. To investigate his suspicion, he recorded radio conversations and, with the help of his intern, processed them into a knowledge graph. Now, he needs assistance extracting information from the knowledge graph to expose those individuals. This is where MARINE comes into play, a visual analytics tool that helps explore large-scale communication data, supported by a Retrieval-Augmented Generation (RAG) system.

Index Terms: VAST Challenge 2025, Mini-Challenge 3, Visual Analytics, Retrieval-Augmented Generation.

1 Introduction

The VAST Challenge 2025 centers on *Oceanus*, a fictional island nation experiencing rapid cultural change due to a surge in tourism sparked by a local music superstar. We present our solution to Mini-Challenge 3 (MC3). In MC3, a knowledge graph built from intercepted boat radio communications is provided. We tackle the challenge to analyze the Oceanus knowledge graph to uncover communication patterns, relationships, and potential illicit activities. To address this challenge, we developed a visual analytics tool called MARINE. The tool supports several key analytical tasks: identifying and analyzing daily communication patterns in the knowledge graph, tracking their evolution over two weeks, and determining influential entities; exploring and visualizing relationships between people and vessels to detect closely associated groups and their predominant discussion topics; detecting and linking pseudonyms to real entities to visualize these connections and assess their impact on understanding activities; and, finally, investigating whether Nadia Conti is engaged in illegal activity, while visually summarizing the evidence to confirm or refute this suspicion.

2 Data

The data was provided in the form of a knowledge graph with a highly sophisticated structure. This graph contains three node types, including entity, event, and relationship, as well as a variety of node subtypes. When two entities communicate, a communication node is created between them, containing the exchanged message. From these messages, Clepper and his intern identified and extracted additional events and relationships. Among the extracted relationships were acts of surveillance, operational involvement, professional collaboration, and personal bonds, as well as other implicit connections revealed through Clepper's message analysis.

While analysing the data, we discovered that Clepper and his intern had made some mistakes and mislabelled certain nodes in the knowledge graph. The most significant issue we identified was that some communications were incorrectly connected. To address this, we developed a script that used a pattern we had identified to detect these errors. The pattern was based on the observation that, in most messages, the first sentence first names the receiver and then the sender. Whenever the edges in the graph were connected differently, the script flagged them, and we corrected the errors manually to avoid introducing new mistakes.

*e-mail: willi.kneer@uni-konstanz.de
†e-mail: jonathan.rentschler@uni-konstanz.de
‡e-mail: keim@uni-konstanz.de
§e-mail: julius.rauscher@uni-konstanz.de
¶e-mail: lucas.joos@uni-konstanz.de

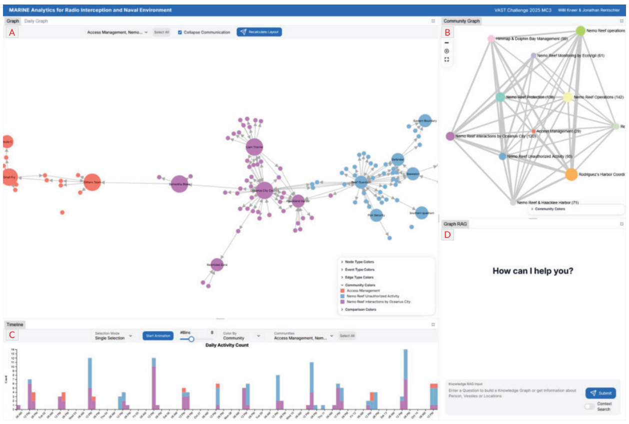

Figure 1: **Overview of the MARINE tool,** displaying (A) general graph view, (B) RAG interface, (C) community graph, and (D) timeline. The timeline and general graph display only a subset of the knowledge graph based on three selected communities.

3 Application

At the core of MARINE is the main knowledge graph view (Figure 1A). Users can filter the graph by community clusters, interact through tooltips, or collapse communication nodes into simple edges to reduce clutter. For better readability, node positions can be recalculated on demand. To see when communications take place, the timeline view comes in (see Figure 1C). The timeline allows users to filter and explore events over time. With the help of brushing, we have linked it to the graph view (Figure 2A and B). The color encodings can be changed to differentiate by event type or community. There's also an animated playback mode to cycle through events step by step, revealing how the situation evolves.

MARINE provides a community-level view (Figure 1B) to show the result of the Leiden algorithm that we used to detect and visualize groups of entities that interact frequently. Each node size reflects the community's size, while edges between communities represent inter-community connections. In the Graph RAG window (Figure 1D), users can query a large language model (LLM), with relevant parts of the knowledge graph automatically retrieved as context to produce informed answers. There is also an option to visualize the subgraph that the system used to answer the question, allowing users to trace the reasoning behind each response. We can see this in (C) and (D) of Figure 2.

To help answer complex questions, we have implemented an RAG system. We use a large language model (LLM), in our case a local instance of Phi-4 [1], to extract named relevant entities from

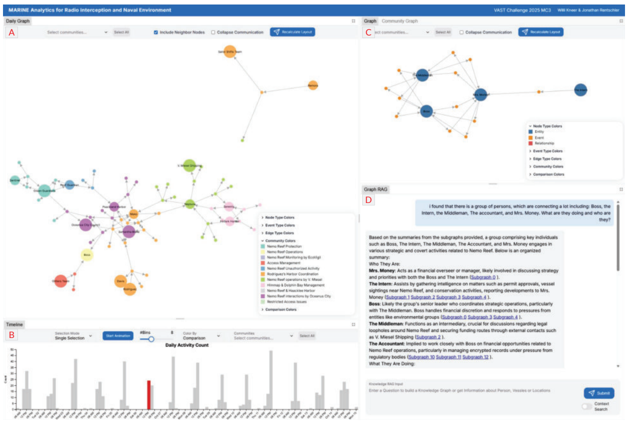

Figure 2: **Example interaction with the MARINE tool,** illustrating (A) graph view of a temporal subset brushed in the (B) timeline component and (C) subgraph provided as evidence by the (D) RAG query.

the user query and answer the question based on the retrieved subgraph. We started by implementing the Graph RAG pipeline [2]. However, we found that this implementation was not suitable for our use with a local LLM. We noticed that a single query could take up to 15 minutes to answer. The small LLM limited our context window, which led to many hallucinations in the Graph RAG pipeline. Furthermore, we wanted a feature to display the subset of the knowledge graph that was used to generate the answer. Therefore, we implemented two pipelines, which we call Conversation Search and Context Search (see Figure 3 and Figure 4).

Conversation Search begins with a user query. We first extract all named entities with the help of an LLM. Next, we query our graph database to retrieve all communications between those entities, chunk and sort them by time, and build subgraphs from the resulting nodes. The LLM then summarizes each subgraph. Finally, we combine these summaries into a single answer, prompting the LLM to cite the corresponding subgraphs so that they can be visualized. The final answer is then returned to the user. Conversation Search is designed for fast answers when the relevant entities are already known. Figure 3 illustrates the process step by step.

Context Search adds an indexing step. Here, we detect communities within the graph using the Leiden algorithm and summarize each community with the LLM. During the query step, we again start with the user query, then use the summaries together with the question to extract all relevant entities. Afterwards, the process is the same as in Conversation Search. Context Search is intended to assist at the start of an investigation, helping to discover and explore relevant areas of the graph. The steps are illustrated in Figure 4.

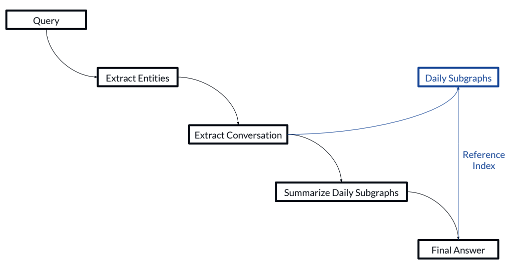

Figure 3: **This diagram illustrates the Conversational Search Pipeline,** which processes a user's query step by step. First, key entities are extracted from the input. Next, relevant data is retrieved as a subgraph. The information is then summarized for clarity before the system generates a precise final answer.

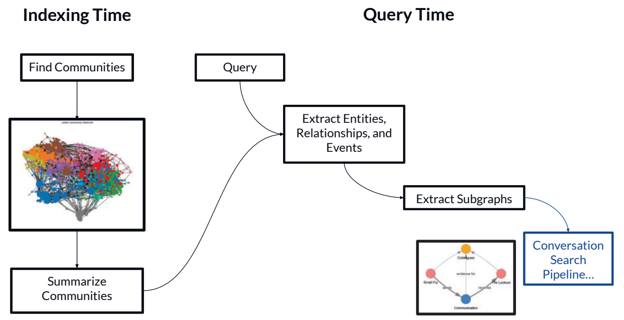

Figure 4: **This diagram depicts the Context Search Pipeline.** First, an indexing step partitions the knowledge graph into communities and generates summaries for each. During entity extraction, these precomputed summaries are leveraged to efficiently identify and prioritize relevant information.

4 Findings

Using the timeline component with the community coloring, we identified daily peaks in communication, typically between 10:00 AM and 4:00 PM, with a consistent high between 10:00 AM and 12:00 PM. The *Rodriguez Harbor Coordination* community was active every day, while *Nemo Reef Operations* showed midday activity with minor exceptions. Activity was noticeably lower during the first weekend than during the second.

Entity-level analysis using the GraphRAG pipeline revealed influence relationships, for example, the vessel *Defender* supervising *Seawatch*. Community detection revealed eleven distinct communities, many centered on *Nemo Reef*. GraphRAG summaries allowed quick contextualization of each group. Certain communities, such as *Nemo Reef Operations* by V. Miesel and *Rodriguez Harbor Coordination*, showed strong interconnections, suggesting collaboration or shared interests. Topic patterns indicated recurring themes of environmental management, operational coordination, and restricted access control.

Suspicious aliases could be mapped to real individuals: *The Middleman* was identified as **Liam Thorne**, *Mrs. Money* as **Elise**, *The Intern* as **Sam**, *The Lookout* as **Kelly**, *Small Fry* as **Rodriguez**, and *Boss* as **Nadia Conti**. *The Accountant* was likely **Davis**.

RAG-assisted queries and subgraph visualizations further connected *Boss* (Nadia Conti) to interactions with other pseudonym holders, notably *The Middleman* and *The Accountant*, in contexts suggesting coordination of undisclosed or potentially illicit meetings. These patterns support Clepper's suspicion of her continued involvement in questionable activities.

5 Conclusion

MARINE enables Clepper to visually explore and analyze the Oceanus knowledge graph to uncover communication patterns, relationships, and potential illicit activities. It supports temporal analysis to detect messaging trends and shifts over the two-week observation period. Furthermore, it reveals interaction clusters between people and vessels, and provides summaries of topic areas that are predominant for each group. The tool also helps Clepper to uncover aliases of individuals and provide evidence to assess and justify suspicions of illegal activity.

References

[1] M. Abdin, J. Aneja, H. Behl, S. Bubeck, R. Eldan, S. Gunasekar, et al. Phi-4 Technical Report, Dec. 2024. doi: 10.48550/arXiv.2412.08905 1
[2] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, et al. From Local to Global: A Graph RAG Approach to Query-Focused Summarization, Feb. 2025. doi: 10.48550/arXiv.2404.16130 2