LLM-based OSINT Agent with Memory, Knowledge Integration, Tool Application, and Self-Reflection

Zhijie Shen
Department of Computer Science and Technology
Tsinghua University, Beijing, China
shenzj24@mails.tsinghua.edu.cn

Qian Wu
Department of Automation
Tsinghua University, Beijing, China
q-wu24@mails.tsinghua.edu.cn

Keyu Shen
Department of Precision Instrument
Tsinghua University, Beijing, China
sky24@mails.tsinghua.edu.cn

Abstract

This paper presents an LLM-based open-source intelligence (OSINT) agent system tailored to tackle the persistent challenges posed by multi-source, dynamic intelligence cycles. The system integrates memory modules, domain-specific knowledge priors, automated tools, and self-reflection mechanisms to enable efficient data collection, processing, and analysis. Among its key features are Retrieval-Augmented Generation (RAG), which supports specialized instruction generation, and Neo4j-based knowledge graphs, which facilitate structured data representation and precise query execution. To evaluate the system's performance, we designed and performed evaluation across diverse tasks, including people profiling, organizational structure extraction, event summarization, and entity relationship mapping. The experimental results demonstrate the system's potential, achieving an accuracy of 82% in people profiling and 95% in event summarization. However, the study also identified several limitations, such as insufficient integration of multimodal data, challenges in managing conflicting information, and restricted capabilities in causal reasoning. Therefore, future work will focus on incorporating advanced self-learning mechanisms, robust fact-checking modules, and enhanced multimodal processing frameworks, ultimately enabling more effective and resilient OSINT applications in complex and evolving real-world scenarios.

1 Introduction

The global business landscape is characterized by increasing complexity and rapid change, requiring intelligent systems capable of delivering timely, accurate, and actionable insights across diverse and dynamic markets. Traditional methods for gathering market intelligence—such as embassy consultations, trade exhibitions, and manual surveys—have long served as foundational tools for international business strategies. However, these methods are increasingly constrained by static data sources, fragmented workflows, slow information retrieval, and high operational costs. As global markets evolve, these limitations hinder their ability to keep pace with the dynamic nature of modern business, particularly in multi-organizational and multi-national scenarios.

To address these inefficiencies, open-source intelligence (OSINT) has emerged as a critical alternative. OSINT systems enable large-scale, efficient information collection and analysis from publicly accessible data sources, such as web content, social media, and other open datasets, significantly raising the efficiency of the intelligence cycle [1]. Compared to traditional approaches, OSINT offers

Preprint. Under review.

broader coverage, higher operational speed, and reduced costs, making it increasingly relevant in modern business contexts. Machine learning techniques, such as support vector machine and random forest, as well as artificial neural networks, such as convolutional neural networks (CNN) and long short-term memory (LSTM) neural networks, have advanced the development of AI-driven OSINT methods, showing greater potential to reduce manual labor while achieving improved performance and efficiency.

Despite their advantages, current OSINT tools face significant challenges, including the sheer volume and variability of data, as well as the need for timely, context-sensitive analysis. While AI-driven methods have introduced promising improvements, they still exhibit notable limitations. They tend to rely heavily on single-source data, especially Twitter, which undermines their robustness and generalizability. Furthermore, these methods fail to effectively align with the entire intelligence cycle, particularly the dissemination phase, where insights must be analyzed, contextualized, and delivered to stakeholders in a timely manner [2]. As a result, current solutions often fall short of providing real-time, high-fidelity intelligence required to address the complex demands of global businesses.

In response to these challenges, we introduce an OSINT Agent that integrates cutting-edge artificial intelligence technologies with domain-specific tools and capabilities tailored for overseas marketing scenarios. At its core, the system employs a large language model (LLM) as its backbone, leveraging its advanced natural language understanding and generation capabilities to process vast amounts of unstructured data. Building on this foundation, the agent dynamically incorporates memory modules to retain and recall contextual information, enabling more consistent and insightful analyses over time. By constructing and maintaining real-time dynamic knowledge graphs, the system maps relationships among entities, events, and emerging trends, offering businesses a comprehensive understanding of global markets. To ensure broad data coverage and reduce reliance on single-source information, the OSINT Agent integrates multiple open-source intelligence tools, aggregating diverse data streams from websites, social media platforms, news outlets, and other publicly available databases. Additionally, the system employs self-reflective feedback loops, enabling iterative self-assessment and refinement of its outputs to ensure accuracy, relevance, and adaptability to evolving market demands.

Our integrated design aims to address the limitations of existing AI-based OSINT methods, offering a scalable, real-time solution capable of delivering high-fidelity intelligence at significantly reduced operational costs. The application scenarios of our intelligent OSINT conversational agent include the analysis of individuals, organizations, events, and domains. It is capable of maintaining dynamic domain knowledge graphs and generating intelligence outputs such as country reports. By bridging gaps in the intelligence cycle and effectively synthesizing multi-source data, the OSINT Agent empowers oversea businesses to monitor global markets, identify emerging trends, and extract actionable insights with unprecedented speed and precision.

2 Related Works

2.1 Open-Source Intelligence (OSINT)

General definition: OSINT is defined as gathering information from "open sources including the Internet, traditional mass media, specialized journals, conference proceedings, think tank studies, photos, maps and commercial imagery product." [3]

Traditional methods and tools of OSINT: Varying from data collection to processing and analysis, common OSINT tools are Spiderfoot [4], Maltego [5] and Sublist3r [6]. For instance, Spiderfoot automates queries to public data sources to collect intelligence on IP addresses, domains, names, and emails. It can also scan websites for vulnerabilities such as SQL injection and cross-site scripting (XSS), making it a robust reconnaissance tool. Maltego, on the other hand, focuses on deep data analysis and visually maps relationships between entities using graph-based visualizations, enabling intuitive exploration of connections. Notably, the majority of these tools currently have not been integrated with cutting-edge AI techniques [2].

AI-based OSINT methods: Novel approaches combining artificial intelligence techniques have been employed on OSINT tasks. These AI-based OSINT methods predominantly utilize machine learning techniques such as Support Vector Machines (SVMs), Naive Bayes, Random Forests, and neural networks like Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM)

networks, as shown in Fig. 1. SVMs are commonly applied for text classification, such as fake news detection through features like TF-IDF [7]. CNNs excel in image classification tasks, as demonstrated by systems like PicHunt, which analyze social media images for law enforcement purposes [8]. Hybrid models combining CNNs and LSTMs have also shown effectiveness, such as in malicious domain detection [9] and domain categorization [10]. Less common approaches include LASSO regression for sentiment analysis preprocessing [11] and Annotated Probabilistic Temporal Logic (APT) for cyber-attack prediction [12]. Most existing research focuses on the processing and analysis phases of the intelligence cycle, while the planning and evaluation phases remain underexplored. [13].

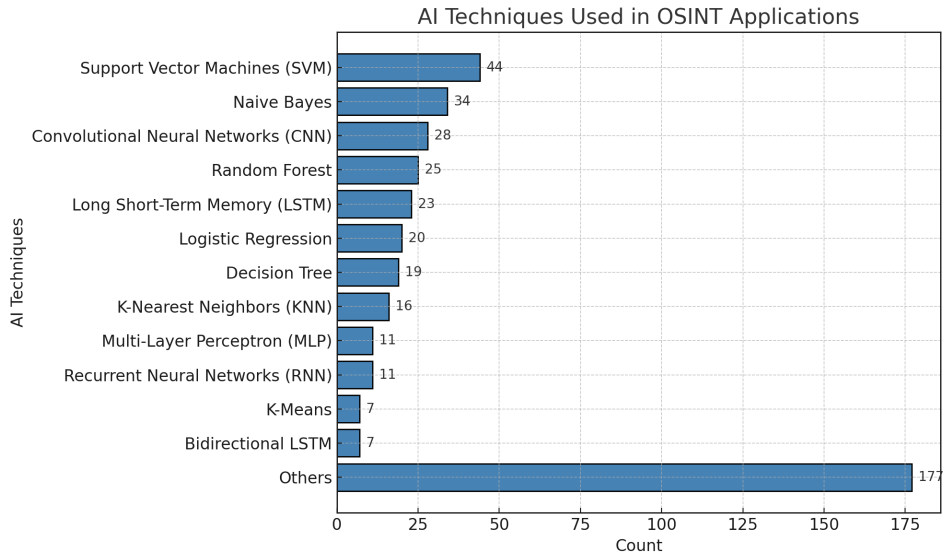

Figure 1: AI techniques used in OSINT applications. Statistics acquired from Browne et al. [2].

2.2 LLM Agent

Large Language Models (LLMs), such as GPT-4, have demonstrated remarkable capabilities in extracting insights from unstructured and multimodal data sources. Enhancements like Retrieval-Augmented Generation (RAG) further improve these models by integrating domain-specific knowledge, enabling more precise and context-aware outputs. As a result, AI agents leveraging LLMs as their backbone are emerging across both general-purpose and domain-specific applications [14]. However, the application of LLM-based Agents in OSINT remains in its early stages, where the demand for accuracy, efficiency, adaptability and domain expertise is particularly high. Current implementations often suffer from limited integration with automated workflows, real-time data collection pipelines, and iterative optimization mechanisms. These shortcomings highlight a significant opportunity to develop a unified framework that combines the adaptability of LLMs with domain-specific intelligence tools, addressing the unique challenges of OSINT and enabling more effective, real-time decision-making in real-world application.

Langchain: The implementation of our proposed OSINT LLM agent is facilitated by LangChain [15], an open-source framework specifically designed for interaction with large language models (LLMs), such as GPT. LangChain provides a flexible and modular approach to handling natural language tasks by enabling chained processing of operations and the creation of intelligent agents. It allows the seamless integration of LLMs with external tools, APIs, and data sources, enhancing the system's ability to perform complex workflows, such as multi-step reasoning, data extraction, and decision-making. This architecture significantly improves the system's intelligence and adaptability, making it well-suited for dynamic OSINT tasks that require real-time information processing and knowledge synthesis.

3 Approach

Centered around the LLM Agent, we aim to build a comprehensive OSINT solution framework that spans frontend visualization, backend processing, as well as data collection, processing, and storage, ensuring system efficiency and scalability. The general conceptual diagram is presented in Fig. 2. In this section, we will avoid over-explaining well-established technical implementation details and instead focus on three key subsystems: data collection, knowledge graph and dynamic query mechanisms, and the LLM-based interactive OSINT agent.

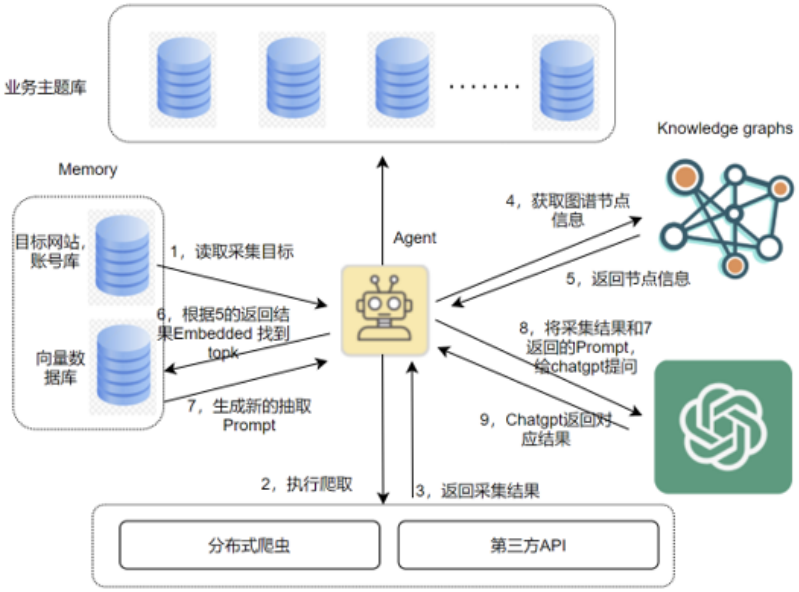

Figure 2: The general conceptual diagram.

3.1 Data Collection and Processing Subsystem

Since the performance of LLMs heavily depends on the quality and contextual relevance of the input data, the primary function of the data collection and processing subsystem is to extract relevant data from various sources, such as social media, news outlets, forums, and the dark web, and perform comprehensive preprocessing, ensuring that the LLM receives cleaned, structured, or semi-structured data. The structure of the subsystem is shown in Fig. 3.

More specifically, the data collection and processing subsystem utilizes advanced crawling technologies, including Scrapy for structured crawling tasks, Selenium and UIAutomator2 for automated interaction with dynamic content, and Fiddler-based traffic analysis for reverse-engineering API calls on social platforms. To address anti-crawling mechanisms, it employs techniques such as IP proxy rotation, dynamic header injection, and Tor-based dark web access. Data from sources like websites, social media platforms, and the deep web are gathered through a distributed architecture, leveraging containerized deployments and SOCKS5 proxies to ensure scalability and reliability. Preprocessing involves automated data cleaning, normalization, and format conversion, supported by distributed message queues for real-time data delivery to downstream systems.

A variety of APIs is included in data collection, each designed to interface with specific types of data sources. These APIs include endpoints for crawling social media platforms, such as Facebook, Twitter (X), Instagram, and LinkedIn, enabling the extraction of user profiles, posts, comments, followers, and geolocation data. For deep web and dark web data, the system integrates APIs capable of accessing Tor-enabled sites and performing automated data extraction using customized browser automation. News and forum data are gathered through APIs that support template-based URL generation for full-site crawling, enabling the extraction of structured information such as titles,

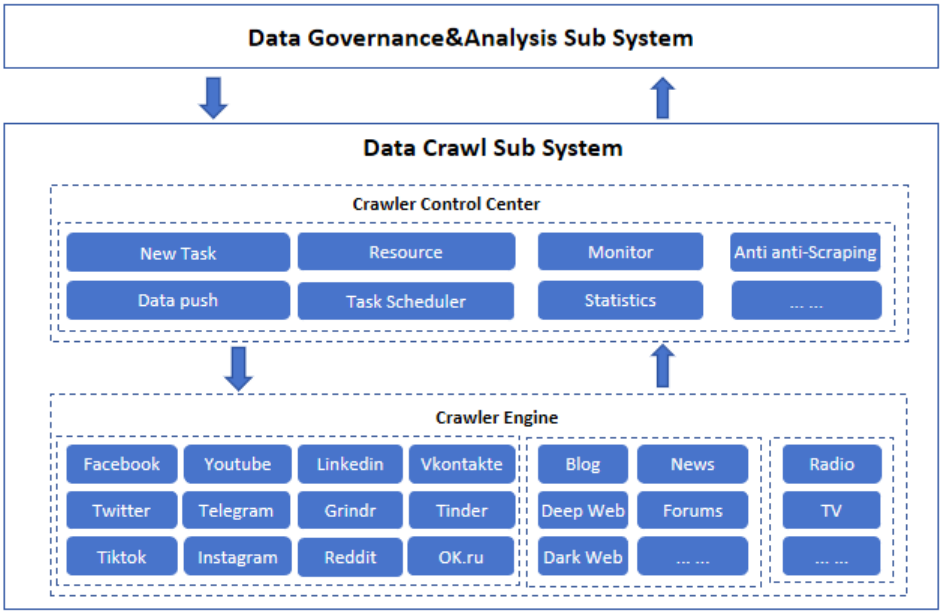

Figure 3: Structure of data collection subsystem.

content, and timestamps. Additionally, APIs for app-based data collection allow interaction with mobile applications via UIAutomator2 or similar frameworks, capturing user activity, posts, and other metadata. Each API is tailored with anti-crawling measures, such as IP rotation and session simulation, ensuring robust and adaptive access to diverse data types. These APIs collectively support comprehensive data acquisition from structured, semi-structured, and unstructured sources to meet the diverse needs of intelligence tasks.

Besides data collection, the system also integrates data preprocessing modules to optimize and standardize multi-source data. The cleaning module employs machine learning to detect issues like missing values, redundancies, and logical errors, dynamically applying solutions such as anomaly detection, imputation, and fuzzy matching. The element extraction module, based on the OPL (Object-Property-Relationship) model and NLP, parses key information and maps it to standard fields, identifying relationships and extracting valid elements. The format conversion module automatically detects and converts formats (e.g., JSON, XML, CSV) to standardize data structures. For redundancy and conflicts, the deduplication module uses similarity calculations and hashing algorithms for efficient deduplication, resolving conflicts with priority rules and intelligent merging. Finally, the aggregation module classifies and clusters data, completing property relationships and generating high-quality, standardized datasets.

3.2 Domain Knowledge Graph and Dynamic Query Mechanism

The domain knowledge graph is a core component for integrating and managing multi-source structured and unstructured data. It is constructed based on the Resource Description Framework (RDF) standard, ensuring semantic consistency and compatibility across heterogeneous data types. The backend employs the high-performance graph database Neo4j, enabling efficient storage, retrieval, and traversal of complex relationships.

The dynamic query mechanism provides flexible support for efficient interaction with the knowledge graph. Users or systems can issue queries using Neo4j's Cypher query language to retrieve complex relationship patterns or specific domain insights. Additionally, the knowledge graph supports incremental updates, continuously integrating new data sources to maintain its real-time relevance and adaptability to evolving intelligence needs.

3.3 LLM-based Interactive OSINT Agent

The OSINT Agent employs a modular architecture designed for adaptability and efficiency. The system integrates memory modules, domain-specific knowledge graphs, tool automation, and self-reflective feedback mechanisms to create a robust solution for real-time market intelligence. Figure 4 illustrates the overall workflow of the proposed OSINT Agent.

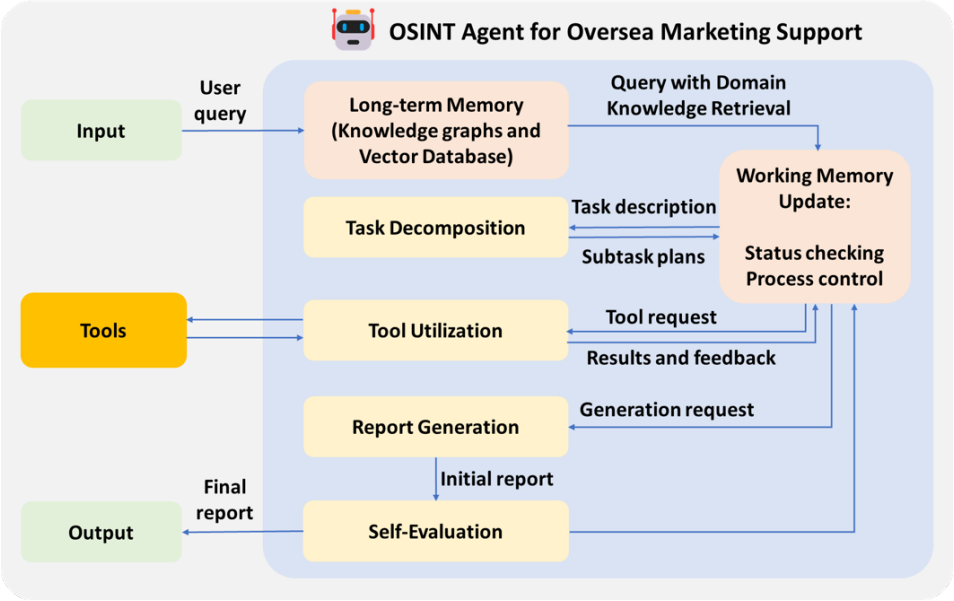

Figure 4: Workflow of the OSINT Agent.

3.3.1 Semantic Comprehension

Text-based prompt engineering strategies are leveraged to extract structured domain-specific attribute information from unstructured text. This process encompasses three key natural language processing tasks: Named Entity Recognition (NER), Relation Extraction (RE), and Event Extraction (EE):

- NER aims to identify and classify named entities in the text, such as people, organizations, and locations.
- RE focuses on identifying relationships between entities. Prompts for RE are designed to identify phrases that express relationships such as "works for," "belongs to," or "compares with."
- EE extracts event information, including event types, triggers, and participants. EE prompts focus on extracting the start position of events.

The examples for the three tasks are shown in Table 1.

While performing these tasks, LLM processes natural language prompts and unstructured text representations to generate accurate and comprehensive structured outputs. The extracted outputs can be organized and updated into the knowledge graph as nodes (entities) and edges (relationships), as well as supporting subsequent analysis.

3.3.2 Memory

The memory module maintains contextual continuity across iterative analysis tasks, enabling the system to generate insights that are consistent and contextually relevant. By leveraging LangChain's memory mechanisms, such as ConversationBufferMemory and ConversationSummaryMemory, the system efficiently retains historical context and intermediate results, which are dynamically utilized for multi-turn reasoning and report generation.

Table 1: Examples of Tasks for Structured Information Extraction

Task	Description	Example Output
NER	Identify and classify named entities, such as people, organizations, and locations.	`Person: Steve, Organization: Apple`
RE	Extract relationships between entities, focusing on phrases like "works for" or "belongs to."	`Steve` $\rightarrow$ `works for` $\rightarrow$ `Apple`
EE	Extract event information, including event type, triggers, and participants.	`Event Type: Employment, Trigger: started, Employee: Steve`

3.3.3 Knowledge

Knowledge graphs and vector databases provide professional guidance to the agent while being continuously updated based on the working memory module, supporting Retrieval-Augmented Generation (RAG), and dynamically supplying essential knowledge to the LLM during generation tasks, during which disparate data sources are transformed into actionable intelligence.

To enhance efficiency, the system employs a prompt-based attribute extraction table, maintaining precise mappings between domain types and attributes. This table supports dynamic expansion for new domains, requiring only updates to domain records, attributes, and preset prompt templates. For example, adding a "Technology" domain involves simply defining relevant attributes and prompts for seamless integration.

The system first uses the table to determine the domain type of data from diverse sources (e.g., news, think tank reports), ensuring accurate classification and template selection. Then, it generates tailored prompts to extract domain-specific attributes, such as budgets in the economic domain or organizational structures in the political domain. The extracted attributes are integrated into the knowledge graph, enriching its structure.

By combining the prompt-based table with RAG, the vector database dynamically retrieves supplementary multi-source information during attribute extraction. This improves context, accuracy, and completeness, allowing the system to handle complex datasets effectively. The integration of these mechanisms ensures precise and adaptive domain knowledge extraction while supporting system scalability and innovation.

3.3.4 Tool automation

Tool automation plays a vital role in the data acquisition process. By integrating advanced automation tools mentioned in 3.1, the agent collects real-time data from diverse sources, including social media, news websites, and government reports, via API interfaces and automated crawling scripts. The collected raw data have undergone cleaning and structuring using Python-based preprocessing frameworks, ensuring its suitability for downstream analysis and knowledge graph integration.

3.3.5 Self-reflection

To enhance the quality of report generation, we introduce an autonomous feedback mechanism, aiming to enable the system to iteratively optimize its performance through self-reflection. This mechanism leverages LangChain's feedback loops to evaluate outputs based on three key metrics: factual accuracy, readability, and relevance. The evaluation can be conducted via human feedback or pre-defined rules, which are systematically integrated into the workflow to ensure continuous improvement. However, due to the challenges posed by the data bias and hallucinations inherent in LLM backbone networks, which conflict with the high accuracy requirements of OSINT tasks, the formulation and effectiveness of self-evaluation metrics and system development remains a significant hurdle.

In general, by integrating state-of-the-art LLM backbones with OSINT tools, the agent dynamically retrieves and processes relevant data to enhance the depth, accuracy, and precision of its insights.

Combining the modular architecture with LangChain's advanced capabilities, the proposed OSINT Agent represents an efficient, adaptable, and intelligent solution for real-time market intelligence.

4 Experiments and Results

4.1 Objectives and Experimental Design

The primary objective of the experiments is to evaluate the functionality and performance of the proposed system, particularly focusing on its capability to collect and process data from diverse sources and the quality of the extracted information. To achieve this, we designed predefined tasks that simulate real-world OSINT scenarios, including data collection from websites and social media platforms, as well as information extraction across multiple domains such as people profiles, organizational structures, event summaries, and entity relationships.

These OSINT tasks are characterized by their high complexity and integrative nature, requiring the system to demonstrate robust data processing and analytical capabilities across diverse dimensions. Specifically:

1. **People Profile:** This task involves identifying and aggregating personal information, such as names, roles, affiliations, and key achievements. It demands precise entity recognition and relationship extraction, alongside the ability to resolve conflicting information from multiple data sources effectively.
2. **Organizational Structure:** Extracting hierarchical relationships within organizations often requires parsing both structured and unstructured data, including textual descriptions, tables, and images. Additionally, this task entails processing multi-component entities, layered organizational hierarchies, and complex structural relationships.
3. **Event Summary:** Summarizing events necessitates identifying key triggers, participants, and contextual details from diverse data streams such as news articles and social media.
4. **Entity Relationship:** Mapping and analyzing relationships between entities, such as collaborations, affiliations, or dependencies, requires the system to understand and model intricate interconnections. The task heavily relies on reasoning and abstraction capabilities to infer relationships that may not be explicitly stated but are critical for generating actionable intelligence.

By designing tasks across these dimensions, the experiments aim to evaluate the system's capability to address both the diversity and complexity inherent in OSINT scenarios. Through these tasks, we also aim to test the system's applicability and scalability in tackling real-world intelligence challenges.

4.2 Evaluation Methodology

At present, system performance is primarily assessed through human feedback, where experts manually evaluate outputs, which may require further refinement.

The human feedback evaluation was conducted using a team of five domain experts who manually reviewed the system's outputs. In the evaluation procedure, a total of 200 results were selected across the four predefined categories. To measure accuracy, the experts employed a binary scoring system, assigning a score of 1 (true) to completely correct generated content and a score of 0 (false) to content that contained even a single instance of incorrect information.

The accuracy rate of the system was calculated using the following formula:

$$\text{Accuracy Rate} = \frac{\sum_{i=1}^{200}(\text{number of } 1)}{\text{number of } 1 + \text{number of } 0}$$

Table 2 presents the accuracy rate statistics for each result type as evaluated by the human experts. The evaluation results provide insights into the system's performance across different tasks, highlighting areas of strength and identifying potential weaknesses for future improvement.

Table 2: Accuracy of Different Result Types Evaluated by Experts

Task	People Profile	Organizational Structure	Event Summary	Entity Relationship
Accuracy	82%	51%	95%	71%

4.3 Observations and Analysis

Upon further analysis of the erroneous results, major drawbacks of the current system were identified:

1. **Conflicting Information from Different Sources:** Although priority rules and intelligent merging mitigate some conflicts, the system still struggles with more tricky ones. The system relies on the most recent information when the same knowledge point (e.g., a person's profile) appears across multiple data sources such as Wikipedia, official websites, or news platforms. However, the reliance on recency can lead to inaccuracies when the latest data is unverified or conflicting, ultimately affecting the consistency and reliability of the outputs.
2. **Insufficient Multimodal Information Integration:** Critical information, such as organizational structures, is frequently represented in non-textual formats like images or embedded within web page tables. Due to the absence of optical character recognition (OCR) or image-based extraction mechanisms, the system is unable to process such formats, resulting in gaps or omissions in extracted organizational structure and relational data.
3. **Noise and Misinformation:** The system may collect fake or unreliable news articles from diverse sources, introducing noise into the dataset and distorting extracted insights such as event summaries. The lack of robust fact-checking or source validation mechanisms exacerbates the impact of misinformation, thereby compromising the overall reliability of the system's outputs.
4. **Limited Causal Reasoning Capabilities:** While the system performs well in summarization, it currently still lacks the ability to really perform causal and more abstract reasoning, which is critical for understanding and analyzing complex relationships between events and entities. For instance, identifying how one event triggers another or how entities influence each other over time requires advanced reasoning and predictive modeling.
5. **Limited Self-Evaluation and Error Correction Capabilities:** The system's ability to self-evaluate, identify its own errors, and implement corrections remains underdeveloped. Specifically, it lacks robust mechanisms for error attribution (i.e., identifying the root cause of inaccuracies), as well as strategies for dynamically revising incorrect outputs. This limitation reduces the system's ability to improve iteratively and adapt to changing or erroneous data, thus impacting its long-term reliability and scalability.

The identified limitations highlight critical areas where the current system requires improvement to achieve its full potential as a robust OSINT solution. Issues such as conflicting information from diverse sources, insufficient multimodal integration, and the presence of noise and misinformation underline the need for enhanced data validation, multimodal processing capabilities, and reliable fact-checking mechanisms. Additionally, the lack of causal reasoning, self-evaluation, and error correction capabilities limits the system's ability to generate deeper insights and iteratively improve its performance. Addressing these challenges will be key to advancing the system's adaptability, scalability, and overall intelligence, enabling it to handle increasingly complex real-world scenarios with higher accuracy and reliability.

5 Discussion

The proposed OSINT system demonstrates potential in integrating domain knowledge graphs, dynamic query mechanisms, and LLM-based capabilities for efficient multi-source data processing and intelligence extraction. Several limitations persist, highlighting opportunities for further improvement. This section delves into the broader implications of the system's design and performance, addressing its current strengths, challenges, and potential directions for future enhancement. By reflecting on

these aspects, we aim to provide a comprehensive understanding of the system's impact and identify areas where emerging technologies can be leveraged to maximize its utility and adaptability.

5.1 Multimodal Processing

The system lacks the ability to process multimodal information effectively and has not yet integrated mechanisms for handling diverse data formats such as text, images, videos, and audio. Incorporating multimodal information encoders in the future would enable the system to process and fuse heterogeneous data types, thereby enhancing its overall analytical capabilities.

5.2 More Automized Evaluation

For instance, deploying specialized evaluation agents may further reduce manual labor and raise efficiency. Challenges include developing a comprehensive and reasonable quantitative evaluation metric system, improving the accuracy and reliability of LLM-based autonomous evaluations, and ensuring consistency across diverse tasks and domains.

5.3 More Accurate Generation against Bias and Illusion

Although LLMs have achieved significant progress, they still face challenges related to bias and hallucinations in generated outputs. Future work should focus on integrating external knowledge validation mechanisms and real-time fact-checking modules to mitigate bias and enhance content reliability. Techniques such as retrieval-augmented generation (RAG), adversarial training, and multi-source cross-validation can play critical roles in addressing these issues and ensuring the factual accuracy of outputs.

5.4 More Advanced Self-Learning Capabilities

The current system has limited self-learning capabilities, heavily relying on pre-designed manual mechanisms as priors, such as carefully organized prompts. Future advancements could incorporate reinforcement learning and self-adaptive algorithms to reduce human intervention and enable the system to learn dynamically from evolving data patterns and feedback.

5.5 Multi-Agent Collaboration

Multi-agent systems (MAS) offer promising opportunities to enhance the adaptability and scalability of OSINT solutions. Future work should focus on designing agents with specialized roles, such as data collectors, processors, evaluators, and decision-makers, while enabling seamless collaboration through dynamic task allocation and communication protocols. Challenges include optimizing resource scheduling, ensuring fault tolerance, and developing robust interaction strategies that adapt to real-time changes in data sources and task complexity. The integration of MAS with LLMs can further improve system performance by leveraging collective intelligence for complex, large-scale tasks.

6 Conclusion

In this work, we proposed a modular OSINT Agent framework that integrates a domain knowledge graph, dynamic query mechanisms, and advanced LLM-based capabilities to address the challenges of real-time, multi-source data intelligence. The system leverages the RDF standard and Neo4j to construct and manage a flexible, semantically enriched knowledge graph, ensuring compatibility and efficient data retrieval. By incorporating prompt-based attribute extraction and domain-specific templates, the system dynamically adapts to new domains and attributes with minimal manual intervention. Additionally, the integration of Retrieval-Augmented Generation (RAG) enhances the accuracy and contextual relevance of extracted information by dynamically retrieving supplementary data from vector databases. Together, these components form a cohesive pipeline for efficient data cleaning, domain classification, information extraction, analysis and standardization.

Future work will focus on further automating evaluation mechanisms and optimizing the system's performance through enhanced multi-agent collaboration and advanced feedback loops. By combin-

ing cutting-edge technologies, this framework provides a robust foundation for long-term OSINT innovation and application in diverse scenarios.

References

[1] Mohammed Ogab Mohammed ALharthi. The role of digital knowledge tools (osint) in raising the efficiency of organizations. *International Multilingual Academic Journal*, 1(1), 2024.

[2] Thomas Oakley Browne, Mohammad Abedin, and Mohammad Jabed Morshed Chowdhury. A systematic review on research utilising artificial intelligence for open source intelligence (osint) applications. *International Journal of Information Security*, pages 1–28, 2024.

[3] Central Intelligence Agency. INTelligence: Open Source Intelligence. Historical Document, 2010. Archived at: `https://web.archive.org/web/20200303002208/https://www.cia.gov/news-information/featured-story-archive/2010-featured-story-archive/open-source-intelligence.html`.

[4] Steve Micallef. Spiderfoot. `https://github.com/smicallef/spiderfoot/releases`, 2021. Accessed: 2024-06-17.

[5] Maltego. Maltego. `https://www.maltego.com/`, 2022. Accessed: 2024-06-17.

[6] Ahmed Aboul-Ela. Sublist3r. `https://github.com/aboul3la/Sublist3r`, 2020. Accessed: 2024-06-17.

[7] Hany Ahmed, Issa Traore, and Sherif Saad. Detection of online fake news using n-gram analysis and machine learning techniques. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 10618 of *LNCS*, pages 127–138. Springer, December 2017. doi: 10.1007/978-3-319-69155-8_9.

[8] Shivangi Goel, Niharika Sachdeva, Ponnurangam Kumaraguru, A. V. Subramanyam, and Deepti Gupta. Pichunt: Social media image retrieval for improved law enforcement. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 10046 of *LNCS*, pages 206–223. Springer, 2016. doi: 10.1007/978-3-319-47880-7_13. arXiv:1608.00905.

[9] Chhavi Choudhary, Ranjith Sivaguru, Miguel Pereira, Bin Yu, André C. Nascimento, and Martine De Cock. Algorithmically generated domain detection and malware family classification. In *International Symposium on Security in Computing and Communication (SSCC 2018): Security in Computing and Communications*, pages 640–655. Springer, 2019. doi: 10.1007/978-981-13-5826-5_50.

[10] Wen Yang and Kwok-Yan Lam. Automated cyber threat intelligence reports classification for early warning of cyber attacks in next generation soc. In *Lecture Notes in Computer Science (LNCS)*, volume 11999, pages 145–164. Springer, 2020. doi: 10.1007/978-3-030-41579-2_9.

[11] C. S. PavanKumar and L. D. DhineshBabu. Novel text preprocessing framework for sentiment analysis. In *Smart Innovation, Systems and Technologies*, volume 105, pages 309–317. Springer, 2019. doi: 10.1007/978-981-13-1927-3_33.

[12] Eric Marin, Mohammed Almukaynizi, and Paulo Shakarian. Inductive and deductive reasoning to assist in cyber-attack prediction. In *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 262–268. IEEE, 2020. doi: 10.1109/CCWC47524.2020.9031154.

[13] N. Ekwunife and N. Ekwunife. National security intelligence through social network data mining. In *Proceedings of the 2020 IEEE International Conference on Big Data (Big Data 2020)*, pages 2270–2273. IEEE, 2020. doi: 10.1109/BigData50022.2020.9377940.

[14] Yuheng Cheng, Ceyao Zhang, Zhengwen Zhang, Xiangrui Meng, Sirui Hong, Wenhao Li, Zihao Wang, Zekai Wang, Feng Yin, Junhua Zhao, et al. Exploring large language model based intelligent agents: Definitions, methods, and prospects. *arXiv preprint arXiv:2401.03428*, 2024.

[15] H. Chase. Langchain, 2022. URL `https://github.com/hwchase17/langchain`.