*Article*

# Multi-Agent AI Framework for Automated Corporate Data Enrichment

**Sahin Uregil[1,†*], Yagiz Erdem[1,†*], Sultan Turhan[2], Günce Keziban Orman[2], and Doğa Yağmur Yılmaz[1]**

[1] Technology & Innovation, Kariyer.net, Istanbul, Turkey; sahin.uregil@kariyer.net (S.U.); yagiz.erdem@kariyer.net (Y.E.); doga.yilmaz@kariyer.net (D.Y.Y.)
[2] Department of Computer Engineering, Galatasaray University, Istanbul, Turkey; sturhan@gsu.edu.tr (S.T.); korman@gsu.edu.tr (G.K.O.)
* Correspondence: uregilsahin@gmail.com (S.U.); yagizeerdem@gmail.com (Y.E.)
† These authors contributed equally to this work.

## Abstract

The ability to rapidly acquire and process comprehensive corporate information has become critical for market analysis and competitive intelligence. Manual extraction and structuring of business data from web sources remains time-intensive and resource-demanding, limiting business intelligence scalability. We developed an automated company data enrichment system that transforms basic company identifiers into structured business intelligence profiles using multi-agent artificial intelligence (AI) frameworks and large language models (LLMs). The system implements a four-stage pipeline comprising query generation, automated information gathering, structured data synthesis, and quality validation mechanisms. The approach was evaluated on a dataset of 400 companies sourced from Kariyer.net's Turkish and global company listings. The system demonstrated varying extraction success rates across information categories: founding year (86.8%), parent company information (64.0%), subsidiary information (52.8%), funding summaries (52.0%), and founder names (22.8%). Analysis revealed significant organizational complexity within the Turkish corporate landscape, with conglomerate companies averaging 25.44 subsidiaries each compared to 6.45 for all companies. The system provides a scalable solution for enterprise-level business intelligence applications, enabling organizations to transform basic company identifiers into comprehensive corporate profiles with minimal manual intervention. The framework reduces market research timelines from weeks to hours while revealing important patterns in organizational complexity within emerging markets.

**Keywords:** artificial intelligence; business intelligence; data enrichment; web scraping; large language models; corporate information extraction; multi-agent systems; automated data processing; knowledge extraction; hierarchical databases

Received:
Revised:
Accepted:
Published:

**Citation:** Uregil, S.; Erdem, Y.; Turhan, S.; Orman, G. K.; Yılmaz, D. Y. Multi-Agent AI Framework for Automated Corporate Data Enrichment. *Journal Not Specified* **2025**, *1*, 0. https://doi.org/

## 1. Introduction

In the contemporary business landscape, the ability to rapidly acquire and process comprehensive corporate information has become a critical competitive advantage for organizations engaged in market analysis, competitive intelligence, and strategic decision-making [1,2]. The proliferation of digital information sources has created vast repositories of unstructured business data distributed across corporate websites, news articles, and public databases [1]. However, the manual extraction and structuring of this information remains a time-intensive and resource-demanding process that significantly limits the scale and efficiency of business intelligence operations [1–3].

The primary objective of this study is to develop and validate an automated company data enrichment system that transforms basic company identifiers into comprehensive, structured business intelligence profiles. This system leverages multi-agent AI frameworks [4–6] and LLMs [7–9] to systematically extract, process, and structure corporate information from web-based sources. The proposed solution aims to bridge the gap between raw company naming data and the rich, multi-dimensional corporate profiles required for advanced business analytics, competitive analysis, and market intelligence applications.

Several critical challenges emerge when attempting to automate corporate information extraction at scale. First, the heterogeneous nature of web-based corporate information presents significant data quality and consistency issues, as companies vary widely in their information disclosure practices and digital presence [1,10–12]. Second, complex organizational structures, particularly within conglomerate companies, require sophisticated parsing and relationship mapping capabilities to accurately represent subsidiary networks and parent-child relationships [13]. Third, the semantic variability in how corporate information is presented across different sources necessitates robust natural language processing (NLP) approaches capable of identifying and extracting relevant information regardless of presentation format [14]. Fourth, the need for structured output that conforms to predefined schemas while maintaining flexibility for diverse business intelligence applications presents additional technical complexity [15].

Our study addresses these challenges through the development of a comprehensive multi-stage pipeline that combines query generation, automated web information gathering, structured data synthesis, and quality validation mechanisms. By implementing this automated approach on a dataset of 400 companies sourced from Kariyer.net's[1] database of Turkish and global company listings, we demonstrate the system's capability to extract diverse corporate attributes with varying success rates, revealing important patterns in information availability and organizational complexity within the Turkish business landscape. The resulting framework provides a scalable solution for enterprise-level business intelligence applications while offering insights into the challenges and opportunities associated with automated corporate data enrichment in emerging markets.

The primary contributions of this study to the field of automated business intelligence and corporate data analysis are as follows:

- **Comprehensive Corporate Profiling Framework:** We develop an end-to-end system that transforms basic company identifiers into detailed and meaningful multi-dimensional business profiles using LLMs, enabling organizations to rapidly assess market opportunities without extensive manual research.
- **Automated Hierarchical Relationship Discovery:** Our system automatically identifies and structures complex parent-subsidiary relationships within corporate ecosystems.
- **Up-to-date Market Intelligence through Web Automation:** By leveraging automated web data collection [16], our system provides access to current market information, enabling businesses to make data-driven decisions based on up-to-date intelligence.
- **Scalable Business Intelligence Infrastructure:** The system generates machine-readable JavaScript Object Notation (JSON)[2] outputs that integrate seamlessly with existing business intelligence platforms and analytical workflows, eliminating manual data entry bottlenecks and enabling automated downstream processing for large-scale market analysis.

[1] https://www.kariyer.net

[2] https://www.json.org/json-en.html

- **Quality-Assured Information Extraction:** We implement a reflection mechanism that automatically validates extracted information quality and triggers re-processing when necessary, ensuring reliable business intelligence outputs.

## 2. Materials and Methods

### *2.1. Dataset Overview*

The dataset utilized in this study comprises a curated list of company names primarily sourced from Kariyer.net, a leading employment and recruitment platform in Turkey. The selection includes a broad range of companies operating in various sectors, with an emphasis on both Turkish enterprises and multinational corporations that have a significant presence in the Turkish market.

The initial dataset consisted of 400 company names sampled directly from Kariyer.net's internal listings. These include a diverse mix of small, medium, and large-scale organizations, with a subset of the entries (45 companies) identified as conglomerates. The inclusion of conglomerates provides an added dimension of structural complexity due to their typically multi-sector and multi-brand configurations [17].

The dataset was constructed in close collaboration with domain experts from Kariyer.net, ensuring that the selected companies reflect a representative sample of the broader employment landscape within Turkey. The primary data field at this stage is the *company name*, which serves as the foundational identifier for all subsequent processing and analysis.

No additional attributes or metadata (e.g., business activities, relationships, or organizational structures) were found in the raw dataset. As such, the dataset initially served as a naming index, intended to be further enhanced with detailed, meaningful and structured information in later stages of the project.

### *2.2. Problem Statement*

While the initial dataset provided a solid starting point through its comprehensive list of company names, it lacked the depth and contextual richness required for meaningful business analysis. Company names alone do not offer sufficient semantic information about a company's operations, industry focus, or organizational structure [18]. This limitation makes it difficult to derive actionable insights or perform tasks such as clustering, segmentation, or relationship mapping with any degree of accuracy or business relevance [19].

One of the primary challenges stemmed from the absence of descriptive metadata that could convey the nature of each company's products, services, or sectoral alignment. Without such information, any attempt to group or compare companies based on their business activities would be inherently limited and potentially misleading [19].

Another significant challenge involved conglomerate companies, which often operate across multiple industries through various subsidiaries or sub-brands. Treating these entities as monolithic units can introduce substantial noise into any analytical process, obscuring important intra-organizational distinctions and misrepresenting inter-company relationships. Disaggregating these complex structures requires an understanding of subsidiary relationships that was not present in the original dataset [20].

Furthermore, the lack of auxiliary information, such as founding year, founder names, or hierarchical relationships, restricted the potential for cross-company linkages or historical and structural analysis. These gaps not only hindered the development of a nuanced understanding of the corporate landscape but also limited the dataset's applicability for advanced business intelligence tasks [19].

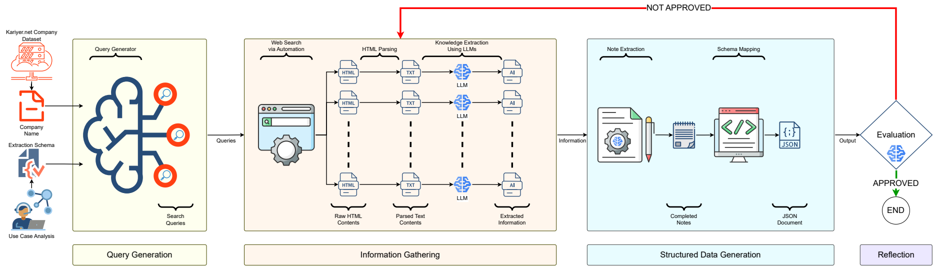

**Figure 1.** System Architecture for Automated Company Data Enrichment and Structuring

In sum, the initial dataset's narrow focus on company names posed substantial barriers to achieving the semantic and structural granularity needed for high-quality, business-driven analysis.

To mitigate these problems, a multi-agent AI system has been devised to systematically enrich and structure the dataset. This system performs detailed and meaningful enrichment of corporate metadata by automatically extracting, contextualizing, and structuring diverse business attributes from web-based sources, transforming sparse company identifiers into comprehensive organizational profiles. A flowchart outlining the overall architecture of the proposed system is presented in Figure 1.

### 2.3. Query Generation

The query generation phase constitutes the initial step of the automated information extraction pipeline. This component takes two primary inputs: the *company name*, which serves as the subject of investigation, and an *extraction schema*, which specifies the structured format and target attributes for data retrieval.

The *extraction schema* functions as a flexible blueprint guiding the downstream stages of the system. It defines the detailed structure of the desired output, allowing users to tailor the pipeline to varying business or analytical needs. The schema is represented in a JSON-like format, where each field corresponds to a particular dimension of company-related information, such as founding year, product descriptions, founders, or organizational affiliations. This standardized representation facilitates seamless data exchange and interoperability with existing enterprise systems, enabling straightforward integration into diverse business intelligence ecosystems without requiring custom data transformation processes. The modularity of the schema enables the system to dynamically adjust its behavior based on user-specified informational requirements.

For the purposes of this study, a default schema to support general business analysis tailored to our internal business use case was developed. The schema encompasses seven primary fields designed to capture essential business intelligence attributes: `company_name` (official company designation), `founding_year` (year of establishment), `founder_names` (founding team members), `product_description` (main goods or services offered), `funding_summary` (investment and funding history), `branch_dealer` (regional branches or dealer networks), and `affiliations` (organizational relationships including `subsidiaries` and `parents`).

Given these two inputs, *company name* and *extraction schema*, the system proceeds to generate a fixed number of web search queries using an LLM according to *max search queries* parameter as detailed in `QUERY_GENERATOR` row of the Table 1. These queries are semantically aligned with the schema fields and are designed to maximize the likelihood of retrieving relevant and structured information from publicly available sources.

The queries represent the information-seeking intent derived from the schema and set the stage for the subsequent phase of information gathering, where the system uses these queries to interact with web search engines and collect relevant documents.

| **Prompt Name** | **Purpose** | **Input Variables** | **Output Format** |
|---|---|---|---|
| QUERY_GENERATOR | Generate targeted search queries to gather factual company data | `{company_name}`, `{max_search_queries}`, `{extraction_schema}` | List of strings |
| PAGE_SUMMARIZER | Clean and summarize page content in response to a specific query | `{page_content}`, `{query}` | Unstructured text |
| NOTE_EXTRACTION | Extract and organize notes about a company based on raw web content | `{company_name}`, `{extraction_schema}`, `{content}` | Markdown file |
| SCHEMA_MAPPING | Convert unstructured research notes into structured data as per schema | `{extraction_schema}`, `{notes}` | Structured text (JSON-like) |
| REFLECTION | Evaluate completeness and quality of the structured information against the schema | `{schema}`, `{extraction_schema}` | Boolean value |

**Table 1.** Overview of System Prompts for Company Information Extraction Pipeline

### *2.4. Information Gathering*

Following the generation of targeted search queries, the next phase in the pipeline involves automated information gathering from the web. This step aims to collect and prepare high-quality textual data that corresponds to each of the previously formulated queries.

The process begins by feeding the generated queries into a web automation module that interacts with a publicly available search engine. For every individual query, the system retrieves a total of 2 × `max_search_results` URLs. The rationale behind collecting twice the target number of URLs is to maintain a buffer in case some of the pages are unreachable.

After URL collection, the HyperText Markup Language (HTML)[3] content of each page is downloaded. These raw web documents are then parsed to extract the primary text content, stripping away non-informative elements such as scripts, styles, and advertisements.

Each cleaned textual document is paired with its corresponding query and passed to an LLM. The LLM is tasked with identifying and extracting only the content that directly answers the given query, effectively acting as a semantic filter. This ensures that only relevant, concise, and contextually grounded information is carried forward to the next stage of the pipeline.

At the conclusion of this phase, the system produces up to `max_search_queries` × `max_search_results` units of cleaned and semantically filtered textual content. These documents form the raw material for the structured data generation process that follows.

[3] https://html.spec.whatwg.org/

The input and output structure of this step is given in detail in `PAGE_SUMMARIZER` row of the Table 1.

### 2.5. Structured Data Generation

Once query-specific content has been extracted and filtered, the next stage in the pipeline involves synthesizing this information into a structured and standardized format. This step bridges the gap between unstructured textual data and the structured outputs required for downstream analysis.

The process begins with a summarization task, in which an LLM processes all cleaned content collected for a single company across multiple queries. The LLM consolidates relevant information into a unified markdown-style research note, guided by the predefined *extraction schema*. This summary captures all schema-relevant insights discovered during the earlier information gathering phase, including partial or ambiguous information as explained in `NOTE_EXTRACTION` row of the Table 1. The summarization agent also flags areas of uncertainty or inconsistency across sources, supporting transparency and further human validation where necessary.

Following summarization, the research note is passed to the schema mapping agent, which converts the narrative content into a machine-readable structured format that complies with the original *extraction schema*. This final structured output is expressed in a JSON-style format as detailed in `SCHEMA_MAPPING` row of the Table 1, enabling integration with business intelligence tools or analytical pipelines.

This structured output not only enables consistency across companies but also supports automated processing, querying, and large-scale analysis. The integration of LLMs at this stage significantly reduces manual effort while maintaining interpretability and adaptability to evolving information needs.

### 2.6. Reflection

The final step in the pipeline involves a validation mechanism designed to assess the reliability and completeness of the structured output. This step, referred to as *Reflection*, employs an LLM to inspect the generated JSON document against the original *extraction schema* and *completed notes* and determine whether the information appears consistent, comprehensive, and plausible as the LLM employs meta reasoning to act as a semantic filter, identifying and extracting only the content that directly answers the given *company name* and *extraction schema* while evaluating its relevance and completeness [21].

The LLM produces a boolean flag called `is_satisfactory`. If this flag is set to `True`, the process terminates, indicating that the output is deemed acceptable. If it is set to `False`, and the parameter `max_reflection_steps` is greater than zero, the system re-initiates the information gathering phase and attempts to improve the result. This mechanism acts as a safeguard against hallucinations and incomplete outputs, improving the trustworthiness of the autonomous pipeline in a similar fashion to *chain of thought* [22–25].

## 3. Experiments and Results

### 3.1. Experimental Setup

The experimental configuration was designed to balance comprehensiveness with computational efficiency while ensuring high-quality data extraction. This section details the key parameters, schema definition, and technical specifications employed throughout the experimental process.

The automated company data enrichment system developed for this study builds upon the foundational architecture of the *company-researcher* framework from the LangChain[4] repository[5]. While this open-source framework provided the initial structural blueprint for web-based company information extraction, substantial modifications and enhancements were implemented to address the specific requirements of our research objectives and the Turkish business landscape.

### 3.1.1. Extraction Schema Configuration

The core extraction schema captures essential business intelligence attributes and encompasses seven primary fields as detailed in 2.3. The `founding_year` and `funding_summary` fields are designed to be nullable (i.e., allowed to be `null` if information is unavailable). Conversely, the `founder_names`, `branch_dealer`, `parents` (within `affiliations`), and `subsidiaries` (within `affiliations`) fields are designed to allow empty lists (`[]`) when no corresponding information is found, rather than being strictly null. This distinction allows for explicit representation of the absence of data.

Furthermore, this extraction schema is formally defined using Pydantic[6] classes within the system's codebase, which provides robust runtime validation and ensures strict adherence to the specified data types and nullability/emptiness rules for each field, mirroring the JSON schema provided to the LLM.

The complete schema definition follows a JSON schema format with explicit type constraints and validation rules as illustrated in Figure 2.

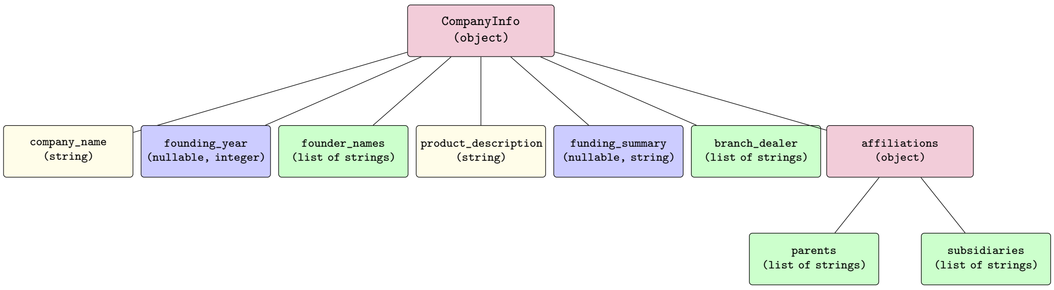

**Figure 2.** Company Information Extraction Schema Structure

### 3.1.2. System Parameters and Configuration

The experimental setup employed a calibrated set of parameters to optimize data quality and processing efficiency. The query generation phase was configured to produce eight search queries per company (`max_search_queries = 8`), providing comprehensive schema coverage while maintaining manageable computational overhead.

For each query, the system retrieved up to three search results (`max_search_results = 3`), resulting in a maximum of 24 candidate web pages per company. A character limit of 250,000 characters per web page was enforced (`max_character_count_for_one_page = 250000`), equivalent to approximately 62,500 tokens, ensuring stable LLM performance.

The system implements parallel processing through batch processing with a batch size of four (`batch_size = 4`), enabling concurrent query processing in two sequential batches. The reflection mechanism was configured with a single iteration limit (`max_reflection_steps = 1`) to balance quality assurance with computational efficiency.

[4] https://www.langchain.com/
[5] https://github.com/langchain-ai/company-researcher
[6] https://docs.pydantic.dev/latest/

### 3.1.3. Technical Implementation

The web automation component utilizes Selenium WebDriver[7] for browser-based interactions as it is among the most popular web automation tools [26], enabling robust handling of dynamic web content. The HTML parsing pipeline combines LXML[8] and BeautifulSoup[9] libraries for reliable content extraction across diverse web page structures. Throughout the entire pipeline, *Gemini-2.0-flash*[10] serves as the LLM for query generation, content filtering, summarization, schema mapping, and reflection as it was found to be adequately well-performing according to the existing research [27–29]. The system architecture supports both single-company processing and bulk operations, with integrated error handling and logging throughout the extraction process.

## *3.2. Data Extraction Performance Analysis*

The automated company data enrichment system demonstrated varying levels of success across different schema fields when applied to the 400-company dataset. This section presents a comprehensive analysis of feature completion rates and structural complexity patterns discovered through the extraction process.

### 3.2.1. Feature Completion Rates

The system's performance varied significantly across different information categories, as illustrated in Figure 3. The most successfully extracted field was the founding year, with 347 out of 400 companies (86.8%) having this information successfully identified and structured. This high completion rate reflects the widespread availability and standardized reporting of establishment dates across corporate websites and public records.

Corporate relationship data showed moderate extraction success, with parent company information successfully identified for 256 companies (64.0%) and subsidiary information for 211 companies (52.8%). The relatively high success rate for parent company identification suggests that ownership structures are frequently disclosed in corporate communications.

Funding-related information presented moderate extraction challenges, with funding summaries successfully compiled for 208 companies (52.0%). This completion rate reflects the heterogeneous nature of funding disclosure practices, where privately held companies often maintain limited public information about their financial backing, while venture-backed and publicly funded entities typically provide more comprehensive funding histories.

The most challenging field proved to be founder names extraction, with only 91 companies (22.8%) having complete founder information successfully identified. This low completion rate can be attributed to several factors: historical founders may not be prominently featured, founding team information is often relegated to company history sections that may be difficult to locate, and many established companies focus their public communications on current leadership rather than founding members.

These statistics, illustrated in Figure 3, are provided specifically for features that are either nullable (i.e., `founding_year`, `funding_summary`) or allowed to be empty lists (i.e., `founder_names`, `branch_dealer`, `parents`, `subsidiaries`). Other features within the schema, such as `company_name` and `product_description`, are always expected to be filled, and thus their completion rates are inherently 100%.

[7] https://www.selenium.dev/
[8] https://lxml.de/
[9] https://www.crummy.com/software/BeautifulSoup/
[10] https://gemini.google.com/app

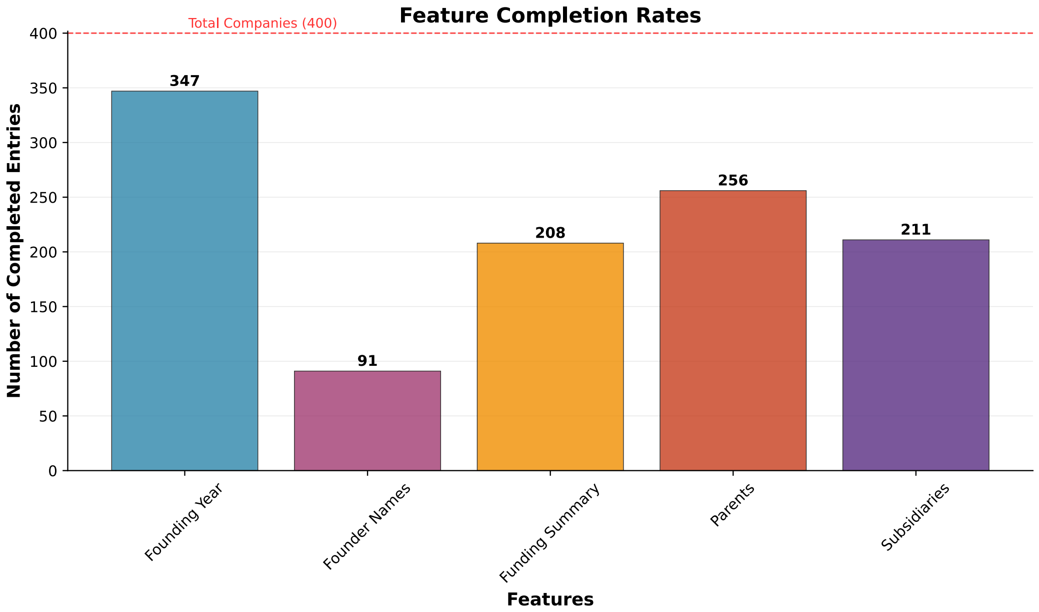

**Figure 3.** Feature Completion Rates

### 3.2.2. Subsidiary Structure Analysis

The analysis of subsidiary structures revealed significant complexity within the Turkish corporate landscape, as depicted in Figure 4. Across all 400 companies, the system identified an average of 6.45 subsidiaries per company, with substantial variation in organizational complexity.

The subsidiary distribution exhibits a pronounced right-skewed pattern, with 189 companies (47.3%) having no identified subsidiaries and 211 companies (52.8%) maintaining at least one subsidiary relationship. The maximum subsidiary count reached 89 for a single organization, demonstrating the extreme complexity possible within conglomerate structures. The standard deviation of 13.34 reflects this high variability in organizational complexity across the dataset.

Conglomerate companies, representing 45 entities within the dataset, displayed markedly different structural characteristics compared to the general population. These organizations averaged 25.44 subsidiaries each, nearly four times the overall average, confirming their role as complex multi-entity corporate structures. This finding validates the initial hypothesis that conglomerates would present unique challenges for organizational analysis and require specialized handling in business intelligence applications.

The subsidiary analysis reveals a bimodal corporate landscape in Turkey, characterized by a large number of relatively simple organizational structures alongside a smaller subset of highly complex conglomerate entities. This distribution pattern has important implications for business analysis methodologies, suggesting that different analytical approaches may be required for different organizational complexity levels within the Turkish corporate environment.

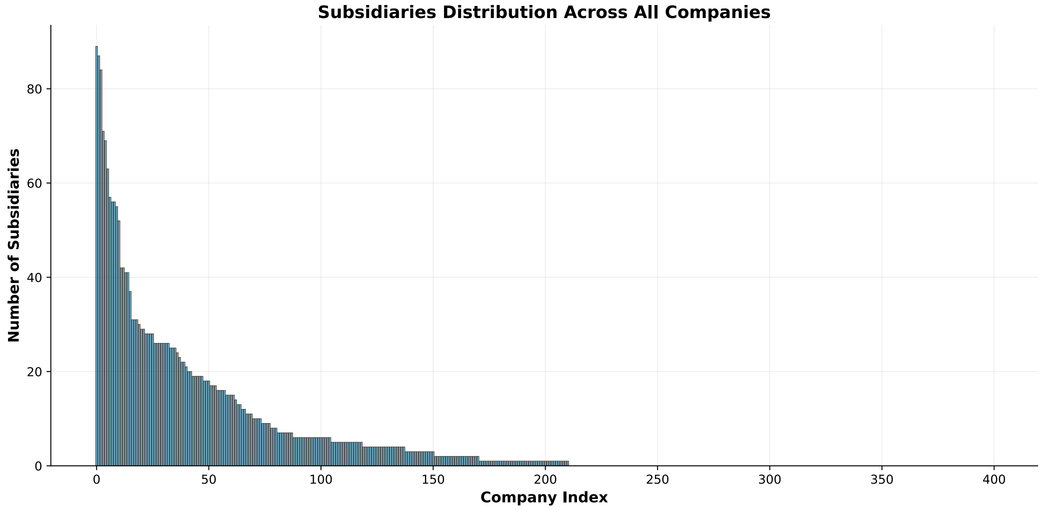

**Figure 4.** Subsidiaries Distribution

### *3.3. Running Example*

To better illustrate the workings of this process throughout the explained steps, we follow a concrete running example: *Kariyer.net A.Ş.*, a prominent employment and recruitment company in Turkey. The following subsections outline each phase of the pipeline with their respective inputs and outputs.

**Query Generation**

*Inputs:* The system takes the following inputs:

- `company_name: Kariyer.net A.Ş.`
- `max_search_queries: 8`
- `extraction_schema: Schema with user defined fields`

*Outputs:* The system produces 8 semantically-aligned search queries:

- `Kariyer.net A.Ş. founding year`
- `Kariyer.net A.Ş. founder names`
- `Kariyer.net A.Ş. product description`
- `Kariyer.net A.Ş. subsidiaries`
- `Kariyer.net A.Ş. parent company`
- `Kariyer.net A.Ş. company profile`
- `Kariyer.net A.Ş. branch dealers`
- `Kariyer.net A.Ş. about`

**Information Gathering**

*Inputs:* For each of the 8 queries generated previously, the system performs a web search with:

- `query: (e.g., Kariyer.net A.Ş. about)`
- `max_search_results: 3`

*Outputs:* Up to 3 unstructured textual contents per query are collected via web automation and filtered by an LLM, resulting in a maximum of 24 cleaned page contents. Below is one such example for the query *"Kariyer.net A.Ş. about"*:

```
Kariyer.net is Turkey’s largest employment platform, operating since 1999.  It
aims to connect job seekers and employers, offering job postings and career
guidance.  It also provides resources for university selection and career planning.
Kariyer.net offers various filters for job searching, enabling users to find
opportunities based on city, sector, position level, department, work style, and
education level.  The platform provides guidance at every stage of the career
journey and provides career advice on resume preparation, job searching,
interviews, and work life.  Kariyer.net also offers career resources such as the
Position Guide, University Guide, and Career Guide.  The HTML content also
includes information about Kariyer.net being a part of a group of companies.  The
```

listed companies are: arabam.com, chemOrbis, cimri.com, emlakjet, hangikredi.com, neredekal.com, sigortam.net, and steelOrbis.

### Structured Data Generation

*Note Extraction Phase*

*Inputs:*

- company_name: Kariyer.net A.Ş.
- extraction_schema: Extraction schema used in the previous step
- content: Combined content of the 24 cleaned documents from previous step

*Outputs:* A markdown-style research note is generated:

Kariyer.net A.Ş. Research Notes

- Company Name: Kariyer.net A.Ş.
- Founding Year: 1999
- Founder Names: Not found in provided sources.
- Product Description: Turkey's largest online employment platform, connecting job seekers and employers. Offers job postings, career guidance, resume database access, and tools for both job seekers and employers across multiple industries. Provides resources for university selection and career planning.
- Branch Dealers: Kariyer.net has offices in Istanbul Ümraniye and Yeşilköy, Adana, Ankara, Antalya, Bursa, Denizli, Elazığ, Eskişehir, Gaziantep, İzmir, Kayseri, Kocaeli, Konya, Mersin, Sakarya, and Samsun.
- Affiliations:
    - Parents: iLab Holding (established in 2000 by Mustafa Say through Access Turkey Capital Group); Kariyer.net became part of iLab in 2006 (acquired on March 8, 2006).
    - Subsidiaries:
        * İşin Olsun (Launched in 2017)
        * İşkolig (Added in 2018)
        * Techcareer.net (Launched in 2021)
        * WeBrand (Launched in 2021)
- Investments Made By Kariyer.net: PeopleBox (Later Stage VC, 29-Apr-2022), Coensio (Corporate, 07-Sep-2021), İşkolig (Buyout/LBO, 27-Jun-2018)
- Companies within the same group as Kariyer.net: arabam.com, chemOrbis, cimri.com, emlakjet, hangikredi.com, neredekal.com, sigortam.net, steelOrbis
- Investors: iLab Ventures (Venture Capital investor holding a majority stake), AccessTurkey Capital Group
- Competitors: LinkedIn, Talent.com, Job&Talent, BountyJobs, Bamble, Jobtome, InfoJobs, Moovijob, Chiletrabajos, Jobtic, VDAB
- Headquarters: Saray Mahallesi Site Yolu Sokak Anel İş Merkezi No: 5 Kat: 3 Istanbul, 34768, Turkey.
- R&D Center Registration Date: January 1, 2014
- Membership Date (ARGEMİP): October 18, 2019

*Schema Mapping Phase*

*Inputs:*

- extraction_schema: Extraction schema used in the previous step
- notes: Research note generated in previous phase

*Outputs:* A structured JSON-style company profile is created:

```
{
    "company_name": "Kariyer.net A.Ş.",
    "founding_year": 1999,
    "founder_names": [],
    "product_description": "Turkey's largest online employment platform, connecting job
    seekers and employers. Offers job postings, career guidance, resume database access,
    and tools for both job seekers and employers across multiple industries. Provides
    resources for university selection and career planning.",
    "branch_dealer": [],
    "affiliations": {
        "parents": [
            "iLab Holding"
        ],
        "subsidiaries": [
            "İşin Olsun",
            "İşkolig",
            "Techcareer.net",
            "WeBrand"
        ]
    }
}
```

**Reflection** In the case of our running example, the returned `is_satisfactory` flag from the reflection step was *True*, indicating that the extracted information met the completeness and quality criteria defined by the schema. As a result, the pipeline halted further iterations and finalized the structured output without requiring re-processing.

To further ensure the reliability and correctness of the reflection mechanism, we conducted a thorough evaluation of the system's outputs in collaboration with domain experts from Kariyer.net. Specifically, we inspected both the structured outputs and their corresponding reflection results for both this running example and the entire dataset of 400 companies. This joint review process confirmed that the reflection step effectively identified incomplete or inconsistent information and successfully triggered re-processing when necessary. These observations validate the reflection mechanism's role in enhancing output quality and reinforce the overall system's trustworthiness for enterprise-grade data enrichment tasks.

### *3.4. Business Impact and Operational Value Creation*

To evaluate the computational efficiency of our system, we conducted timing analyses using free-tier resources as shown in Table 2. *Gemini-2.0-flash*'s free tier served as the primary bottleneck, necessitating the implementation of deliberate sleep mechanisms to comply with rate limiting constraints, ultimately achieving an average processing time of 169.78 seconds per company across the complete dataset of 400 companies, which would normally take weeks if performed by humans. This performance characteristic reflects the system's ability to efficiently manage API constraints while maintaining consistent throughput that only increases linearly at scale.

| Dataset Size | Total Time (s) | Average Time (s) |
|---|---|---|
| 25 | 4,435.37 | 177.41 |
| 50 | 8,730.46 | 174.61 |
| 100 | 17,411.34 | 174.11 |
| 200 | 34,156.47 | 170.78 |
| 400 | 67,911.52 | 169.78 |

**Table 2.** System Processing Time Analysis Across Different Dataset Sizes

The automated company data enrichment system delivers substantial business value through its ability to transform basic company identifiers into comprehensive, structured business intelligence. This transformation addresses critical operational challenges faced by organizations operating in the Turkish business ecosystem, particularly in the context of market analysis and strategic decision-making.

The system's ability to systematically extract and structure company information enables organizations to develop comprehensive market intelligence with significantly reduced manual effort. Traditional market research approaches requiring weeks of manual data collection and analysis can now be accomplished in hours through automated processing.

The automated extraction of product descriptions and service offerings facilitates rapid competitive benchmarking and market positioning analysis. Organizations can now efficiently identify direct competitors, and assess competitive threats without extensive manual research.

The automated nature of the data enrichment process enables scalable business development activities that were previously constrained by manual research limitations. Sales and business development teams can now rapidly assess and prioritize potential clients based on comprehensive company profiles, founding characteristics, and organizational complexity metrics.

The system's ability to process large sets of companies efficiently, demonstrates its utility for large-scale market analysis. Organizations can now systematically evaluate entire market segments, identify high-potential prospects based on specific criteria (such as founding year, organizational complexity, or parent company relationships), and optimize resource allocation for business development activities.

## 4. Discussion

The system's performance characteristics would fundamentally change if *Gemini-2.0-flash*'s paid tier were utilized, primarily due to the elimination of rate limiting constraints. As demonstrated in our timing analysis in Table 2, our current implementation processes companies at an average of 169.78 seconds per company when constrained by free-tier limitations. In a paid-tier scenario, our architecture would enable parallel processing of companies, resulting in considerably shorter processing times. Given that our current implementation incorporates two 60-second sleep mechanisms to accommodate free-tier limitations, the actual computational overhead per company amounts to merely 49.78 seconds. This analysis suggests that with paid-tier access, our system could achieve substantially improved performance, processing entire datasets within a fixed time window rather than scaling linearly with dataset size.

The key contributions of this study, ranging from comprehensive corporate profiling to automated relationship discovery, were not just theoretical aspirations but were actively demonstrated through practical experiments. Each component of the system was subjected to real-world evaluation using a 400-company dataset, ensuring that the proposed solutions translated into measurable outcomes. For instance, the generation of detailed, machine-readable company profiles from only company names (as demonstrated in Section 3.3) validates our end-to-end enrichment capability. Similarly, the automated extraction of hierarchical relationships (Figure 4) empirically demonstrates the system's ability to model complex corporate ecosystems.

Further experimental validation was provided through live web automation (*Information Gathering* phase of Section 3.3), where the system consistently acquired current and relevant data across diverse sources. The outputs were automatically structured into standardized JSON format (*Structured Data Generation* phase of Section 3.3), reinforcing the system's utility for scalable integration. Lastly, the reflection mechanism (*Reflection* phase of Section 3.3) enabled quality assurance by re-triggering incomplete or inconsistent pipelines, another functionality validated through real-time processing of company entries. These experiments collectively anchor our contributions in applied results, reinforcing the system's readiness for business-oriented applications.

The automated company data enrichment system presented in this study offers a novel approach to transforming unstructured web-based information into actionable business intelligence for the Turkish corporate environment. Our findings demonstrate the system's efficacy in extracting diverse attributes, including company name, founding year, product descriptions, and affiliations, and highlight the varying success rates depending on the attribute's prominence and consistency across public web sources. The analysis of subsidiary structures further underscores the system's capability to untangle complex corporate hierarchies, particularly within conglomerates.

This work aligns with recent advancements in leveraging LLMs and agent-based frameworks for enterprise information processing. Similar to EICopilot [30], a solution enhancing search and exploration over large-scale knowledge graphs, our system employs LLM-driven agents to automate information extraction and summarization from natural language queries. Both systems recognize the inherent complexity of enterprise data and strive to improve efficiency and accuracy compared to manual methods. Furthermore, the incorporation of a "reflection" mechanism in our pipeline, designed to assess the reliability and completeness of the structured output and trigger re-iterations if unsatisfactory, mirrors the iterative refinement processes seen in EICopilot and the annual report analysis method, highlighting a shared best practice in building robust LLM-driven data extraction systems.

However, a key distinction lies in the primary focus. While EICopilot is engineered to navigate and query existing, pre-structured knowledge graphs, leveraging technologies like Gremlin scripts[11] for precise data retrieval, our system adopts a more foundational and streamlined methodology that addresses the challenge of initial data enrichment and structuring through transforming raw, unstructured web content (including natural language descriptions and HTML content) into organized, machine-processable company profiles. This 'structuring from raw data via LLM' approach eliminates the need for any pre-existing knowledge graph or structured schemas, enabling the direct generation of meaningful data from the ground up, starting solely with company names. This positions our system as a vital tool for direct conversion of web-native, unstructured information into a usable structured format.

The demonstrated capability to identify and structure complex subsidiary relationships, especially for conglomerate companies (averaging 25.44 subsidiaries each), further validates the critical role of such automated systems. This level of detail in organizational complexity is challenging to ascertain manually and is crucial for comprehensive business intelligence, aligning with the complex relationship mapping endeavors highlighted in both EICopilot's graph-based exploration and the general need for semantic understanding in enterprise data analysis.

Compared to existing open-source frameworks, such as the LangChain AI *company-researcher*[12], which served as our foundational architecture, our system introduces substantial modifications. The most significant of these is the complete redesign of the information gathering mechanism. Rather than relying on prebuilt third-party solutions such as Tavily[13] or other search APIs, we implemented a fully custom web automation layer to extract domain-specific data with greater control and accuracy.

In addition to these infrastructural changes, we introduced targeted improvements in prompt engineering, a critical component in shaping the behavior and performance of language model-based agents [31–34]. While agent prompts were initially sourced from the LangChain repository, they were customized to align with our specific use case requirements. Additional enhancements include the integration of Pydantic-based schema validation, scalable bulk processing workflows, and a modular architecture designed for production-grade deployment. These improvements collectively enable our system to address the demands of enterprise-level applications, particularly within the Turkish-language market.

[11] https://tinkerpop.apache.org/
[12] https://github.com/langchain-ai/company-researcher
[13] https://www.tavily.com

While our study demonstrates the technical feasibility of automated company data enrichment, it does not quantitatively assess the downstream business impact of the enriched dataset on organizational decision-making processes. Future work should incorporate impact assessment methodologies to evaluate how enriched company profiles influence key business processes. Controlled studies comparing business outcomes using manually researched data versus automatically enriched profiles would provide valuable insights into the system's practical utility and identify areas where information quality most significantly influences business performance.

In addition, future research could explore integrating specialized extraction modules, similar to the annual report analysis method, to deepen the information extracted for specific document types (e.g., financial statements, legal filings) for companies where such structured reports are available. Additionally, enhancing the system's capabilities for automatic knowledge graph construction or populating existing graphs with the enriched data could further bridge the gap with knowledge graph-focused systems like EICopilot. Lastly, continued optimization of prompt engineering and LLM configurations, particularly for attributes with lower completion rates such as founder names, will be vital for maximizing the accuracy and completeness of the extracted business intelligence.

**Author Contributions:** Conceptualization, S.U., Y.E., S.T. and G.K.O.; methodology, S.U. and Y.E.; software, S.U. and Y.E.; formal analysis, S.U. and Y.E.; investigation, S.U. and Y.E.; resources, D.Y.Y.; data curation, S.U. and Y.E.; writing—original draft preparation, S.U. and Y.E.; writing—review and editing, S.U., Y.E., S.T. and G.K.O.; visualization, S.U. and Y.E.; supervision, S.T. and G.K.O.; project administration, D.Y.Y.; funding acquisition, D.Y.Y. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** The datasets presented in this article are not readily available because they include proprietary information sourced from private databases owned by Kariyer.net. Although the data were enriched using publicly available web sources through web automation, the core dataset remains confidential and subject to corporate privacy restrictions.

**Acknowledgments:** The authors would like to thank İlknur Köseoğlu Sarı for her role in project management. We acknowledge the use of LangChain's *company-researcher* repository as a foundational component of this study and appreciate LangGraph's flexibility, which enabled the creation of fully customizable agents used to build and operate the system. Python was utilized throughout the development of the system, supported by libraries such as Selenium for web automation, BeautifulSoup and `lxml` for HTML parsing, and Pydantic for structured data validation. We also used Google's *gemini-2.0-flash* for LLM-based parts of the system. During the preparation of this manuscript, the authors used ChatGPT (OpenAI, GPT-4o, 2025) for assistance with editing and word choice. The authors have reviewed and edited the output and take full responsibility for the content of this publication.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

| | |
|---|---|
| MDPI | Multidisciplinary Digital Publishing Institute |
| AI | Artificial Intelligence |
| LLM | Large Language Model |
| NLP | Natural Language Processing |
| HTML | HyperText Markup Language |
| JSON | JavaScript Object Notation |
| API | Application Programming Interfarce |

## References

1. Grigalis, T.; Cenys, A. State-of-the-art web data extraction systems for online business intelligence. *Informacijos mokslai* **2013**, *64*, 145–155. https://doi.org/10.15388/Im.2013.0.1595.
2. Arslan, M.; Cruz, C. Business-RAG: Information Extraction for Business Insights. In Proceedings of the Proceedings of the 21st International Conference on Smart Business Technologies - Volume 1: ICSBT. INSTICC, SciTePress, 2024, pp. 88–94. https://doi.org/10.5220/0012812800003764.
3. Feuerriegel, S.; Hartmann, J.; Janiesch, C.; Zschech, P. Generative AI. *Business Information Systems Engineering* **2024**, *66*, 111–126. https://doi.org/10.1007/s12599-023-00834-7.
4. Park, J.S.; O'Brien, J.C.; Cai, C.J.; Morris, M.R.; Liang, P.; Bernstein, M.S. Generative Agents: Interactive Simulacra of Human Behavior, 2023, [arXiv:cs.HC/2304.03442].
5. Hong, S.; Zhuge, M.; Chen, J.; Zheng, X.; Cheng, Y.; Zhang, C.; Wang, J.; Wang, Z.; Yau, S.K.S.; Lin, Z.; et al. MetaGPT: Meta Programming for A Multi-Agent Collaborative Framework, 2024, [arXiv:cs.AI/2308.00352].
6. Wu, Q.; Bansal, G.; Zhang, J.; Wu, Y.; Li, B.; Zhu, E.; Jiang, L.; Zhang, X.; Zhang, S.; Liu, J.; et al. AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation, 2023, [arXiv:cs.AI/2308.08155].
7. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need, 2023, [arXiv:cs.CL/1706.03762].
8. Radford, A.; Narasimhan, K. Improving Language Understanding by Generative Pre-Training. 2018.
9. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners, 2020, [arXiv:cs.CL/2005.14165].
10. John, B. Challenges and Solutions in Data Integration for Heterogeneous Systems **2025**.
11. Naveed, H.; Khan, A.U.; Qiu, S.; Saqib, M.; Anwar, S.; Usman, M.; Akhtar, N.; Barnes, N.; Mian, A. A Comprehensive Overview of Large Language Models, 2024, [arXiv:cs.CL/2307.06435].
12. Zhao, W.X.; Zhou, K.; Li, J.; Tang, T.; Wang, X.; Hou, Y.; Min, Y.; Zhang, B.; Zhang, J.; Dong, Z.; et al. A Survey of Large Language Models, 2025, [arXiv:cs.CL/2303.18223].
13. Jen, C.H. Exploring Construction of a Company Domain-Specific Knowledge Graph from Financial Texts Using Hybrid Information Extraction. Dissertation, Stockholm, Sweden, 2021.
14. Adnan, K.; Akbar, R. An analytical study of information extraction from unstructured and multidimensional big data. *Journal of Big Data* **2019**, *6*, 91. https://doi.org/10.1186/s40537-019-0254-8.
15. Dagdelen, J.; Dunn, A.; Lee, S.; Walker, N.; Rosen, A.S.; Ceder, G.; Persson, K.A.; Jain, A. Structured information extraction from scientific text with large language models. *Nature Communications* **2024**, *15*, 1418. https://doi.org/10.1038/s41467-024-45563-x.
16. Ning, L.; Liang, Z.; Jiang, Z.; Qu, H.; Ding, Y.; Fan, W.; yong Wei, X.; Lin, S.; Liu, H.; Yu, P.S.; et al. A Survey of WebAgents: Towards Next-Generation AI Agents for Web Automation with Large Foundation Models, 2025, [arXiv:cs.AI/2503.23350].
17. Snihur, Y.; Tarzijan, J. Managing complexity in a multi-business-model organization. *Long Range Planning* **2018**, *51*, 50–63. https://doi.org/https://doi.org/10.1016/j.lrp.2017.06.010.
18. Yerva, S.; Miklos, Z.; Aberer, K. It was easy, when apples and blackberries were only fruits **2010**. *1176*.
19. Datta, S.; Bhattacharjee, S.; Das, S. Clustering with missing features: a penalized dissimilarity measure based approach. *Machine Learning* **2018**, *107*, 1987–2025. https://doi.org/10.1007/s10994-018-5722-4.
20. Lv, Zhepeng. Business-oriented data management in conglomerates: insights from government data sharing. *E3S Web Conf.* **2021**, *290*, 02031. https://doi.org/10.1051/e3sconf/202129002031.
21. Zhou, Y.; Muresanu, A.I.; Han, Z.; Paster, K.; Pitis, S.; Chan, H.; Ba, J. Large Language Models Are Human-Level Prompt Engineers, 2023, [arXiv:cs.LG/2211.01910].
22. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E.; Le, Q.; Zhou, D. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, 2023, [arXiv:cs.CL/2201.11903].
23. Kojima, T.; Gu, S.S.; Reid, M.; Matsuo, Y.; Iwasawa, Y. Large Language Models are Zero-Shot Reasoners, 2023, [arXiv:cs.CL/2205.11916].
24. Zhou, D.; Schärli, N.; Hou, L.; Wei, J.; Scales, N.; Wang, X.; Schuurmans, D.; Cui, C.; Bousquet, O.; Le, Q.; et al. Least-to-Most Prompting Enables Complex Reasoning in Large Language Models, 2023, [arXiv:cs.AI/2205.10625].
25. Wang, B.; Min, S.; Deng, X.; Shen, J.; Wu, Y.; Zettlemoyer, L.; Sun, H. Towards Understanding Chain-of-Thought Prompting: An Empirical Study of What Matters, 2023, [arXiv:cs.CL/2212.10001].
26. Kuutila, M.; Mäntylä, M.; Raulamo-Jurvanen, P. Benchmarking Web-testing - Selenium versus Watir and the Choice of Programming Language and Browser, 2016, [arXiv:cs.SE/1611.00578].
27. Buscemi, A.; Proverbio, D. ChatGPT vs Gemini vs LLaMA on Multilingual Sentiment Analysis, 2024, [arXiv:cs.CL/2402.01715].
28. Cai, H.; Cai, X.; Chang, J.; Li, S.; Yao, L.; Wang, C.; Gao, Z.; Wang, H.; Li, Y.; Lin, M.; et al. SciAssess: Benchmarking LLM Proficiency in Scientific Literature Analysis, 2024, [arXiv:cs.CL/2403.01976].

29. Viegas, C.; Gheyi, R.; Ribeiro, M. Assessing the Capability of LLMs in Solving POSCOMP Questions, 2025, [arXiv:cs.CL/2505.20338].
30. Yun, Y.; Ye, H.; Li, X.; Li, R.; Deng, J.; Li, L.; Xiong, H. EICopilot: Search and Explore Enterprise Information over Large-scale Knowledge Graphs with LLM-driven Agents, 2025, [arXiv:cs.IR/2501.13746].
31. Sahoo, P.; Singh, A.K.; Saha, S.; Jain, V.; Mondal, S.; Chadha, A. A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications, 2025, [arXiv:cs.AI/2402.07927].
32. Amatriain, X. Prompt Design and Engineering: Introduction and Advanced Methods, 2024, [arXiv:cs.SE/2401.14423].
33. Schulhoff, S.; Ilie, M.; Balepur, N.; Kahadze, K.; Liu, A.; Si, C.; Li, Y.; Gupta, A.; Han, H.; Schulhoff, S.; et al. The Prompt Report: A Systematic Survey of Prompt Engineering Techniques, 2025, [arXiv:cs.CL/2406.06608].
34. White, J.; Fu, Q.; Hays, S.; Sandborn, M.; Olea, C.; Gilbert, H.; Elnashar, A.; Spencer-Smith, J.; Schmidt, D.C. A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT, 2023, [arXiv:cs.SE/2302.11382].

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.