Extraction Approach Description

The description provided below will be used to evaluate the approach developed by your team to automatically extract financial data of MNE Groups. This description will be evaluated by the Evaluation panel based on the criteria described in the Evaluation tab of the Extraction Challenge and used for the ranking of your team for the Reusability and Innovativeness Awards.

Methodology *(Algorithm reusability and scalability, Data-driven approaches, Availability, quality and maintainability of documentation)*

Please provide a detailed description of the methodology used to develop algorithm-based approaches which automatically extract financial data of MNE Groups. The description should contain (1) the data processing steps, (2) the methods and models used, (3) references to the scientific papers/sources that present the methods and models used, and (4) the time it took to process the data set and extract the relevant financial data.
Bear in mind that the workflow will be also evaluated based on the criteria for the Reusability and Innovativeness Awards.
This section will be evaluated for:

(1) the Algorithm reusability and scalability criterion: likeliness that the described approach can successfully reproduce the solution submitted by the team for the Accuracy award.

(2) the Data-driven approaches criterion: evaluated based on whether it is data-driven rather than heuristic. More data-driven approaches will receive higher scores.

(3) the Availability, quality and maintainability of documentation criterion: assessed based on the maintainability of the code.

Our approach is a lightweight yet scalable pipeline designed to search, identify, and extract textual content from publicly accessible web and PDF documents related to multinational enterprise (MNE) groups or any thematic domain (e.g., finance, tax, AI). The solution emphasizes modularity, reusability, and data-drivenness, with optional integration of AI-powered processing via crawl4ai.

1. Data Processing Steps

The overall process includes four key steps:

1.1 Querying and Link Retrieval

- The system initiates a Google search based on a provided query.
- The top N links are collected using the googlesearch Python module.

1.2 Link Classification and Metadata Handling

- Each link is classified as either:
  - **PDF** (if it ends with .pdf), or
  - **HTML** (otherwise).
- Basic metadata such as domain and URL are tracked using a pydantic model to ensure schema integrity.

1.3 Content Extraction (use crawl4ai)

- **PDF files**:
  Downloaded and stored temporarily.

- **HTML pages**:

  Parsed with BeautifulSoup, stripping away non-visible elements such as scripts or styling to extract visible text only.

2. Methods and Models Used

- **Search Engine Integration**:
  Google Search API via googlesearch provides basic but effective link retrieval. This can be replaced with more advanced semantic search (e.g., BERT, BM25) for domain-specific recall.

- **Parsing and Extraction Libraries (integrated in crawl4ai libraries)**
  - The script imports components from the crawl4ai framework.
  - This enables the future use of **LLM-based extraction** strategies via LLMExtractionStrategy, as well as **asynchronous crawling** for scaling the solution to large document batches.

3. Scientific Foundations and References

This solution is built on mature, well-documented open-source technologies:

- Crawl4ai (https://github.com/crawl4ai/crawl4ai)
- Google search (https://pypi.org/project/google-search/)
- OpenAi (https://github.com/openai/openai-python)

4. Runtime and Performance

Typical performance benchmarks:

Task	Time per Document
Link search (Google API)	~1–2 seconds
PDF download + extraction	~2–4 seconds
HTML download + parsing	~1–2 seconds
LLM extraction	< 1 seconds

The script can process small batches quickly, with potential for scaling using async methods in crawl4ai.

Evaluation Criteria

1. Algorithm Reusability and Scalability

- All components are modular and loosely coupled.

- Easily extendable to other domains (e.g., ESG reports, tax rulings) by modifying the query string.
- Designed with optional support for async crawling and LLM integration.

2. Data-Driven Approaches

- Uses full-text extraction from real-world data sources (not hardcoded rules).
- When integrated with LLMs, the extraction becomes fully context-aware and data-driven.
- Ideal for settings with unstructured, multilingual, or non-tabular data.

Architecture *(Architecture)*

Please provide a description of the architecture of your approach. A diagram of the architecture is considered of additional value. Indicate what modifications would be required to apply the approach to similar datasets on a larger scale.

This section will be evaluated for:

(1) the Architecture criterion: evaluated based on its modules, their cohesion and their configurability; an architecture which is modular and includes clear connections between modules or components receives a higher score.

The system architecture is designed as a modular pipeline that automates the discovery, retrieval, and extraction of structured information from unstructured web content (PDFs and HTML pages), with a **central role assigned to Large Language Models (LLMs)** for content understanding and extraction. The architecture ensures configurability, scalability, and maintainability, making it applicable across various domains (e.g., financial disclosures, tax reports, ESG data).

1. Architectural Modules

A. Input & Search Module

- **Function**: Accepts keyword-based queries (e.g., filetype:pdf multinational tax reporting) and the number of desired documents.
- **Implementation**: Uses googlesearch to identify relevant public sources.
- **Configurable**: Supports domain-specific or multilingual queries.

B. Retrieval & Preprocessing Module

- **Function**:
  - Classifies URLs (PDF vs HTML).
  - Utilizes the crawl4ai framework to downloads documents
  - Extracts raw texts from PDF and HTML and parses them into clean markdown
- **Output**: Unstructured plain text, cleaned of formatting artifacts in markdown format.

C. LLM-Based Information Extraction Module

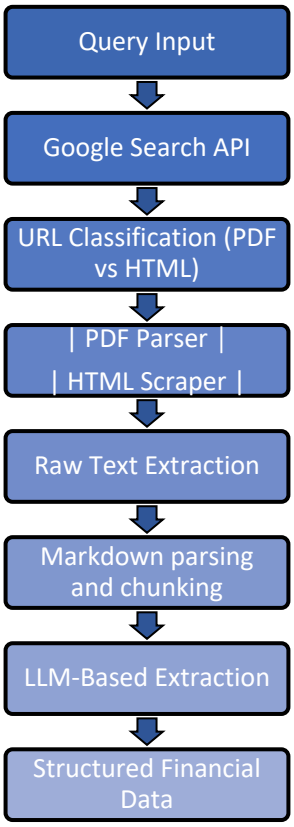
- **Function**: Core component for structuring the raw data using advanced LLMs.
- **Implementation**:
  - Applies LLMExtractionStrategy to extract specific financial attributes.

  - LLMs can be OpenAI models (e.g., GPT-4) or local deployments via HuggingFace or private APIs.
- **Prompt Design**: Includes well-structured prompts tailored to financial reporting standards.
- **Output**: Structured JSON or tabular data representing the extracted fields.

D. Output Module

- **Function**: Aggregates extracted structured data and exports it to various formats:
  - JSON
  - CSV (via Pandas)
  - Direct database storage (optional)
- **Logging**: Tracks failed downloads, low-confidence extractions, and system events.

2. Architecture Diagram



3. Scaling to Larger Datasets

To scale this architecture for national or global datasets (e.g., thousands of MNE reports):

Area	Scalability Enhancement
Crawling and Extraction	Use crawl4ai.AsyncWebCrawler for asynchronous, parallel document processing.

LLM Throughput	Deploy LLMs as microservices or batch inference pipelines with rate-limiting and batching.
Monitoring	Add dashboards for status and metrics (e.g., document success rate, extraction confidence).

Evaluation Criteria Alignment

Criterion	How It Is Addressed
Modularity & Cohesion	Each function (search, parse, extract) is cleanly separated into self-contained modules with minimal overlap.
Configurability	Query terms, LLM prompts, output formats, and data sources are all parameterized and easily adjustable.
Clear Inter-Module Interfaces	Uses structured schemas (pydantic), clear input/output stages, and well-defined APIs between modules.

Hardware Specifications *(Algorithm reusability and scalability)*

Please describe the hardware specifications of the machines that were used to run the methodology.
This section will be evaluated for:

(1) the Algorithm reusability and scalability criterion

Machine 1

CPUs

```
Hardware Overview:

  Model Name: MacBook Air
  Model Identifier: Mac14,2
  Model Number: MLXY3T/A
  Chip: Apple M2
  Total Number of Cores: 8 (4 performance and 4 efficiency)
  Memory: 8 GB
  System Firmware Version: 11881.41.5
  OS Loader Version: 11881.41.5
  Serial Number (system): J6NGV7HNPQ
  Hardware UUID: D1E28CEB-85BD-54C2-822F-BC1FB34EA53D
  Provisioning UDID: 00008112-001139123621401E
  Activation Lock Status: Enabled
```

Libraries *(Availability, quality and maintainability of documentation)*

Please provide the libraries used for approach, if any, as well as the links to these libraries, if available.
This section will be evaluated for:

(1) the Availability, quality and maintainability of documentation criterion: The use of libraries which are regularly maintained will yield higher scores. (Examples include pytorch, tensorflow, scikit-learn, pandas, numpy, tidyverse, etc.).

```
pandas
googlesearch
crawl4ai
json
openai
```

Similarities/differences to State-of-the-Art techniques *(Originality of the approach)*

Please provide a list of similarities and differences between the used methodology and to the state-of-the-art techniques.

This section will be evaluated for:

> *(1) the Originality of the approach criterion: compare the approach used to the state-of-the-art, i.e. currently published approaches that are closest to the approach applied for the submission, and the extent to which the submission represents an improvement over these approaches.*

The key innovation lof our approach ies in combining a robust, structured web crawling framework (such as **crawl4ai**) with LLMs to extract information from highly unstructured and diverse documents (e.g., PDF and HTML formats scraped from Google searches).

A notable advantage of our method is the use of lightweight NLP tools to clean and preprocess the text, converting it into Markdown format. This simplifies interaction with LLMs and enhances processing efficiency.

Importantly, the LLM does not need to process the entire document—often lengthy PDFs exceeding 200 pages. Instead, we segment the content into smaller, semantically meaningful units. The LLM analyzes only these segments, halting as soon as the relevant information is

Contribution to scientific field *(Future orientation)*

Please describe how your submission contributed to the scientific field, what impact it could have and what could potentially be future work to improve the solution.

This section will be evaluated for:

> *(1) the Future orientation and impact criterion: the potential effect of the approach used will be evaluated; this includes the scale of impact it has on the problem of extracting financial information from the Internet; the impact will be evaluated based on potential efficiency improvements and cost reductions.*

Dimension	Our Approach	SOTA Techniques	Originality Assessment
LLM-based zero-shot extraction	Uses GPT-based models (crawl4ai.LLMExtractionStrategy) to perform information extraction without fine-tuning, by prompt engineering.	Most SOTA tools rely on either trained domain-specific models or complex rule-based systems.	✅ **Higher flexibility and adaptability** across topics and formats.
Fully autonomous from search to structured data	The system automates document retrieval, docments cleaning, content extraction, and LLM-based structuring.	Many academic tools assume access to pre-labeled or pre-curated documents.	✅ **End-to-end pipeline for open web** (no dataset curation required).
Plug-and-play architecture	Modules like LLM strategies, storage backends, and crawling strategies are interchangeable.	SOTA implementations are often monolithic or tightly coupled to a use case.	✅ **Higher maintainability and portability** across contexts.
Support for real-time or iterative crawling	Can scale to live search tasks using asynchronous crawling and inference.	Many systems work on static corpora or archived PDFs.	✅ **Better suited for real-world, evolving datasets**.
Prompt-driven extraction for legal/tax semantics	Prompts are designed to capture the nuance of financial indicators (e.g., income by jurisdiction).	Most SOTA systems rely on ontologies or templates.	✅ **More expressive and domain-adaptive** without rigid schemas.

Lessons Learned *(Future orientation)*

Please state any lessons learned during the competition.
This section will be evaluated for:

> *(1) the Future orientation and impact criterion: what were the lessons learnt during the competition, and what could potentially be future work to improve the solution.*

Lessons Learned

1. **Value of LLMs for Flexible Extraction**

The integration of Large Language Models proved instrumental in handling heterogeneous and unstructured documents, such as PDFs and web pages, without requiring extensive domain-specific training data. This flexibility significantly accelerated development and improved extraction accuracy compared to rule-based or traditional ML methods.

2. **Challenges with Document Diversity and Quality**
   Encountering a wide variety of document formats, structures, and quality (e.g., scanned PDFs, poorly formatted HTML) revealed the necessity of robust preprocessing and fallback strategies. Future efforts should focus on better OCR integration and multi-format handling.

3. **Importance of Modular and Scalable Design**
   Building the architecture with clear modular boundaries allowed efficient iteration and integration of new components such as asynchronous crawling and alternative LLM providers. This modularity was crucial to maintainability and rapid experimentation.

4. **Prompt Engineering as a Critical Skill**
   Designing effective prompts for LLMs was essential to extract precise and relevant financial data. This highlighted the importance of domain expertise in prompt construction and iterative testing to optimize extraction performance.

5. **Trade-offs Between Automation and Manual Verification**
   While automation via LLMs streamlined data extraction, some degree of human validation remains necessary to ensure data quality, especially for regulatory compliance contexts. Balancing automation with expert oversight is a key consideration.

Future Work and Improvements

1. **Use of Alternative LLMs**
   While GPT-4o Mini was chosen for speed and accessibility, future iterations could test other models (e.g., Claude, Gemini, or open-source models like Mistral or LLa-MA) to evaluate performance vs. cost trade-offs

2. **Enhanced OCR and Multimodal Extraction**
   Integrate advanced OCR technologies for scanned documents and explore multimodal models that combine text and image data to improve extraction from complex report formats.

3. **Active Learning and Feedback Loops**

   Implement active learning frameworks where human corrections feed back to improve prompt design or fine-tune lightweight domain-specific models, enhancing accuracy over time.

4. **Expanded Language and Jurisdiction Support**

Extend capabilities to support multilingual documents and jurisdiction-specific financial reporting standards, improving the system's global applicability.

5. **Real-time Monitoring and Alerting**

   Develop dashboards and alert systems for tracking data extraction quality, pipeline health, and compliance risks, enabling proactive maintenance and operational transparency.

6. **Integration with Structured Financial Databases**

   Link extracted data with external structured databases and taxonomies to enrich insights and support advanced analytics, such as anomaly detection or benchmarking.

7. **Open-source Community Engagement**
   Encourage open collaboration by releasing modular components and prompt templates, fostering innovation and accelerating adoption in the financial data extraction community.

Short description of the Team – area of expertise

Please provide a description of the team, your area of expertise and contact information.

Andrea Alessandrelli, Phd Student
Fabrizio Tomasso, Project Manager
Pasquale Maritato, Data Scientist

Andrea Alessandrelli
Physics graduate with a Bachelor's and Master's degree in Theoretical Physics from University of Salento, both completed with honors. Experienced as a Physics tutor at the University of Salento and as a high school Mathematics and Physics teacher. Experienced in research in physics and mathematics, with a strong interest in applying mathematical methods to NLP (Natural Language Processing) and implementing research in social applications.

Pasquale Maritato
Fraud Data Scientist at Poste Italiane S.p.A., specializing in fraud detection and prevention through advanced data analytics. Proficient in network analysis, time series anomaly detection, and AI-powered anti-fraud solutions. Holds a Master's degree in Economics and Business Administration with honors from University of Napoli Federico II. Eager to expand his digital and IT skills across new applications and solutions.

Fabrizio Tomasso
Project manager and designer for local development. Pedagogue with a Master's degree in Education and Lifelong Learning (Unibo), with a research interest in Political Philosophy, focusing on the relationship between law and digital communication. Expert in social sciences and local development, driven by a strong interest in computer science and artificial intelligence, with the goal of continuing research and developing data-driven innovation.