

# BUSINESS SCRAPING PIPELINE

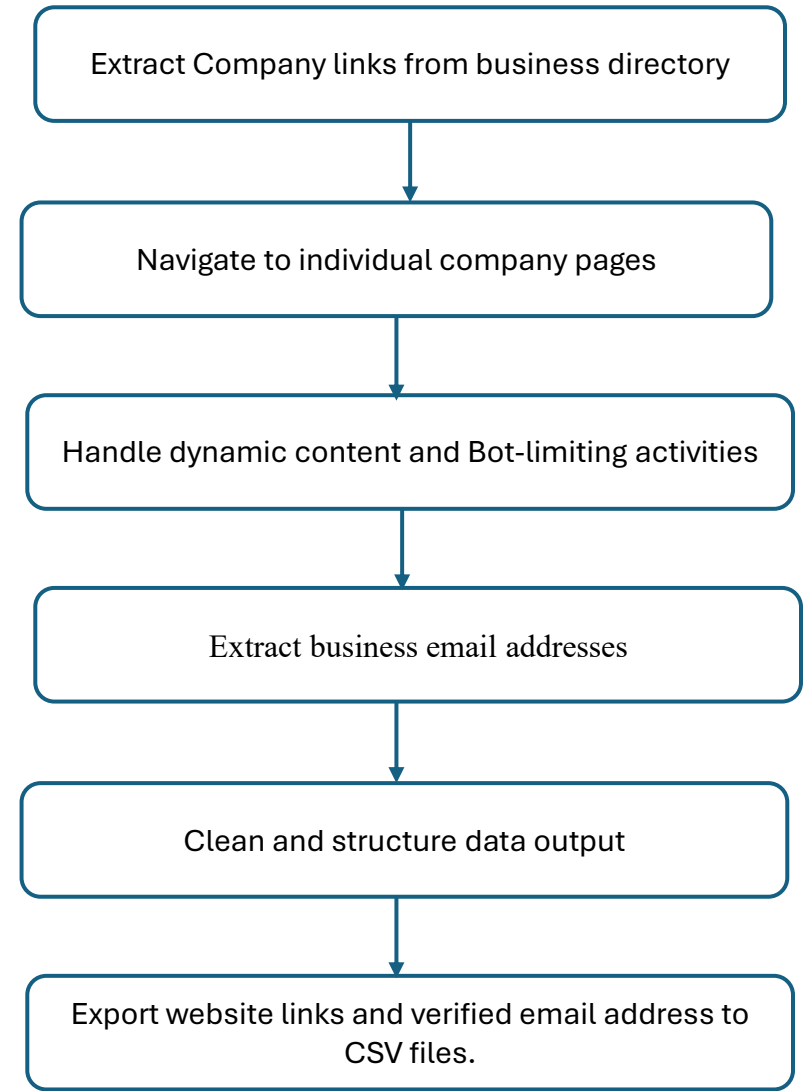
Extracting Verified Business Contacts from Public Directories

**Modular • Scalable • Multi-Industry Support**

Presented by : Sai Pavan Kumar Yedduri  
E-Mail: [saiyedduri97@gmail.com](mailto:saiyedduri97@gmail.com)








# Bigger Picture: Challenge

- Built a modular scalable pipeline that collects the business information(company name, company, company website URL and email address) across various sector pages of Europages business directory using Python language.
- The Business Pipeline is designed to take a sector specific HTML page as input, extract the business email addresses and finally export email addresses as CSV files.
- Exported CSV files follow the format:
  - links\_<sector>.csv:  
Company Name, Country, Europages URL, Company Website URL
  - emails\_<sector>.csv:  
Company Name, Country, Company Website URL, Email address
- Extracted email addresses of 20 companies each over 10 sectors.



**Fig1 : Process steps for collecting contact details from public directories**

# Implemented Modular Architecture

-  **WebScrapingEngine.py** → Dual-mode (Requests + Selenium), handles static/dynamic sites.
-  **DirectoryParser.py** → Extracts company links, handles pagination
-  **ContactExtractor.py** → Finds emails across multi-page company sites
-  **ContactValidator.py** → Filters valid business emails
-  **Data Processor.py** → Cleans, deduplicates, standardizes data
-  **CSVExporter.py** → Structured export for analytics/CRM
-  **CompanyInfo.py** → Unified storage of company data

# Extraction of company website links

- Start from Europage directory URL → Directory\_parser.py
- Loop up to max\_pages:
  - Fetches for company tags on sector page based on CSS link selector. 'a[data-test="company-name"]'
  - For each <a> tag:
    - Get href → e.g., BULUT-CNC-MACHINING/00000005488957-001.html
    - Convert to full URL → e.g., <https://www.europages.co.uk/BULUT-CNC-MACHINING/00000005488957-001.html>
    - Extract company name from the europage company link(BULUT-CNC-MACHINING).
    - Append CompanyInfo(name,europage\_url) to results.
- Pagination CSS selector 'a[aria-label="Next page"]' searches for europage of companies for next page.

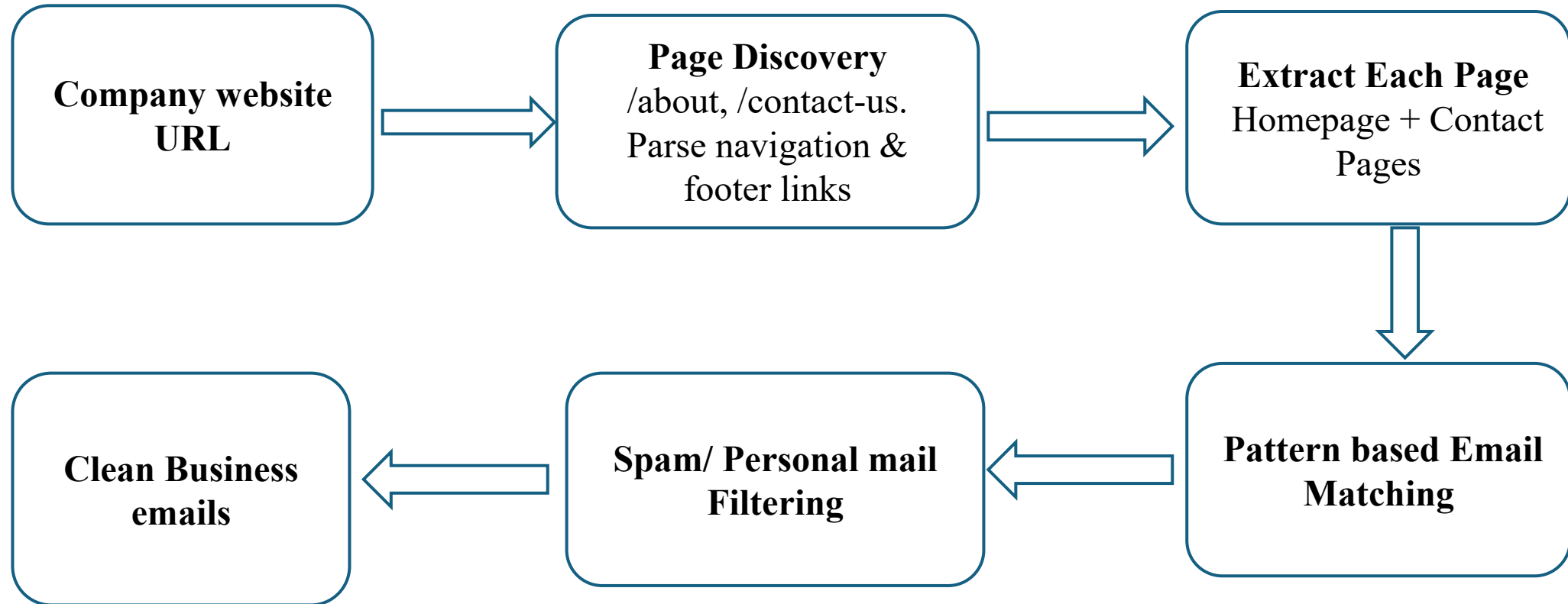
- Extract the company website links from Europages profile → ContactExtractor.py
- Fetches and parses the HTML of the company profile page through website button on europage company page.

self.engine.get\_page(profile\_url, wait\_for\_element='.website-button')

Ex: <https://bulutcncmakina.com/portfoyl/>

- Filter-out the URL to ensure it is a real company website (not Dropbox, Hubspot, Google drive. etc.) based on their domain.

# Process of email extraction



# Pattern based email extraction

## 1. Mail to Link Extraction

- Highest reliability - parses structured HTML mailto attributes

Ex: href="mailto:geral@agsmachining.com?subject=..."  
→ Extracts: geral@agsmachining.com

## 2. Advanced Regex Pattern

Comprehensive text pattern matching for visible emails

`r'\b[A-Za-z](?:[A-Za-z0-9_-]*[A-Za-z0-9])?@[A-Za-z0-9](?:[A-Za-z0-9_-]*[A-Za-z])?\.[A-Za-z]{2,6}\b'`

- Username starts with a letter.
- Username can have letters, digits, underscores, hyphens, but must end with a letter or digit.
- Domain starts with a letter/digit, can contain letters/digits/hyphens/underscores, must end with a letter
- Surrounded by word boundaries.

## 3. Fetch in specific contact ares

Extracts the email addresses by searching specific contact areas of the page such as '.contact, .contact-info, .contacto, .contatti, '.email, .mail, .footer.

# Encountered Challenges

- Extracting email text is difficult as it includes concatenated texts to the email.

Cause: Due to bad design of webpages

Ex:  [alain.renner@kalt-ag.ch](mailto:alain.renner@kalt-ag.ch) ([alain.renner@kalt-ag.ch](mailto:alain.renner@kalt-ag.ch) )

- In some websites, no email addresses are provided. Only Contact fields are provided.

Ex: <https://www.burgondie.info/>

- Some websites contain generic place holder emails such as:
  - email@email.com
  - your@company.com
  - mail@business.com

# Results: Emails extracted per sector

Sector	Total Companies	Total Emails	Avg Emails per Company
Children Clothing	17	32	1.9
CNC Machinery	16	33	2.1
Dairy Products	16	59	3.7
Energy Storage	18	50	2.8
Food Additives	14	72	5.1
Gym Equipment	14	58	4.1
Mobile Phones	14	77	5.5
Pharmaceuticals	15	46	3.1



# Key Areas for ML/LLM Enhancement

## **Intelligent Data Validation:**

- While filtering/cleaning emails, ML models would be able to predict/identify correct emails on the trained supervised emails.
- It learns common domains (gmail.com, yahoo.com, kalt-ag.com) and learns typical typos(.con).

## **LLM Email Extraction:**

- LLMs excel at understanding context by understanding emails like:  
"contact [at] company [dot] com“

## **Adaptive Scraping Strategies :**

- Using LLM, system learns from successful and failed extractions to automatically adjust scraping approaches for different website architectures and anti-bot measures.

THANK YOU