POLYTECHNIC OF TURIN

Master degree course in Data science engineering

Master Degree Thesis

Information retrieval from PDF of companies

Calculation of the "intensity" metric to assess emissions

Supervisors
prof. Ernestina Menasalvas
prof. Paolo Garza

Candidate
Niccolò Revel Garrone
ID: 302302

Internship Tutor
dott. Manuel Angel Guzman Caba

Academic Year 2022-2023

To the people who have been close to me on this intense journey.

Summary

This project proposes the development of an automated system for extracting data from tables of energy companies in order to calculate the intensity metric and evaluate companies in a standardized manner. The main objective is to provide an effective methodology for assessing energy efficiency and sustainability of energy companies, enabling homogeneous comparisons among them.
In the initial phase of the project, the focus will be on extracting data from company tables. These tables may contain crucial information such as sales figures, emission values, and other relevant metrics. The system will employ advanced data extraction techniques, utilizing algorithms and machine learning models to accurately identify and extract the required data from diverse table formats and structures. Once the data has been successfully extracted, the next step is to calculate the intensity metric. The metric will be designed to quantify the relationship between energy consumption and sales for each company. By dividing the emissions by the sales figures, the intensity metric will provide a standardized measure of the energy efficiency of the company's operations. This calculation will enable a fair and comparable assessment of companies' sustainability performance.
The final objective of the project is to establish a standardized evaluation framework for energy companies based on the intensity metric. The calculated metrics will be compared across different companies within the industry, allowing for benchmarking and identification of leaders in energy efficiency. This standardized evaluation will facilitate decision-making processes for stakeholders, including investors, regulators, and consumers, by providing them with an objective and transparent method to assess the sustainability performance of energy companies.

Acknowledgements

I sincerely thank Professor Ernestina Menasalvas for her help and support during the thesis project. I also thank Professor Paolo Garza, for helping me as a co-relator for this thesis. I also thank Manuel Angel Guzman Caba, Management Solution intern, for the information and knowledge he provided to help me during my thesis work. I would also like to thank my family, friends, and girlfriend for their help, helpfulness, understanding, and for their solidarity with me during this master's program and during this time in my life.

Contents

List of Tables

List of Figures

Part I

Introduction

Chapter 1

Information retrevial

1.1 General principles

Information Retrieval[1] (IR) plays a fundamental role in our information-driven society, serving as the backbone for accessing and retrieving vast amounts of data efficiently and effectively. With the exponential growth of digital content, the ability to retrieve relevant information has become crucial in various domains, including academia, industry, and everyday life. This thesis aims to delve into the multifaceted field of Information Retrieval, unraveling its underlying principles, techniques, and advancements.
The use of IR for extracting data from PDF covers a key role since the rapid digitization of documents has resulted in a significant increase in the availability of information in PDF format. However, extracting structured data, such as tables, from PDF files remains a complex and challenging task. Tables contain valuable information that is vital for various applications, including data analysis, knowledge discovery, and decision-making processes. This master's thesis aims to explore the utilization of Information Retrieval (IR) techniques for the extraction of information contained in tables from PDF documents, addressing the underlying challenges and proposing novel solutions.
Table extraction from PDFs involves the identification and extraction of tabular structures and their corresponding content accurately. The unstructured nature of PDF files, combined with the variability in table layouts and formatting, makes this task particularly difficult. Traditional approaches relying solely on rule-based methods or template matching struggle to handle the vast diversity and complexity of tables present in real-world PDF documents. This thesis aims to leverage the principles of Information Retrieval to tackle

the table extraction problem. By treating table extraction as an IR task, we can exploit the inherent characteristics of PDF documents and leverage indexing, querying, and ranking techniques to improve the accuracy and efficiency of table extraction methods.
By focusing on the utilization of Information Retrieval techniques for table extraction from PDFs, this master's thesis aims to contribute to the advancement of information extraction methods. The findings of this research will provide valuable insights into the challenges associated with table extraction from PDF documents and offer innovative approaches for improving accuracy and efficiency in this domain. Ultimately, the results of this study can have significant implications for a wide range of applications, enabling effective utilization of tabular data locked within PDF files for decision-making, research, and data analysis purposes.

1.2 Project description

The purpose of this thesis, carried out together with Management Solution, is to be able to calculate a metric, called intensity, which is used to express the emissions intensity of a particular process or industry and help normalize emissions due to changes in organizational activity, such as the total growth of a business unit.
The importance and use of this metric is to be able to normalize and standardize the companies that are analyzed, in fact this metric returns a value that is proportional to the emissions of a certain company and the number of sales per year. The purpose of this thesis then is to assess how green a company is behaving.
To be able to calculate this metric, the most objective way is to extract from a company's sustainability report the values for scopes 1,2, and 3 of emissions, and to extract from the profit and loss account the information about the company's annual sales.
The equation 1.1 represents the intensity metric from a mathematical point of view:

$$Intesity = \frac{Scope1 + Scope2 + Scope3}{Sales} \tag{1.1}$$

In order to compute this metric, we will use data recognition and extraction techniques from tables so that we can extract the values related to scopes

1,2, and 3 and the value associated with sales.
Scopes 1, 2, and 3 in the Sustainability Reports of energy companies refer to the classification of greenhouse gas (GHG) emissions based on their origin and responsibility. This classification is defined by the GHG Protocol[2], an international standard for measuring and managing GHG emissions:

- **Scope 1: Direct greenhouse gas emissions**
  Scope 1 includes direct GHG emissions produced from sources owned or controlled directly by the energy company. These emissions may result from the combustion of fossil fuels in industrial processes, electricity generation, or internal transportation, for example. Scope 1 emissions represent the company's direct responsibility.

- **Scope 2: Indirect greenhouse gas emissions from energy**
  Scope 2 includes indirect GHG emissions resulting from the production of purchased electricity used by the energy company. These emissions are considered indirect because they are generated outside of the company's direct control but are linked to its electricity consumption. Examples include emissions from the combustion of fossil fuels in power plants that supply the company with electricity.

- **Scope 3: Other indirect greenhouse gas emissions**
  Scope 3 encompasses all other indirect GHG emissions that occur in the value chain of the energy company, including both upstream and downstream activities. These emissions are a consequence of the company's operations but occur from sources not owned or controlled by the company. Scope 3 emissions can include emissions from purchased goods and services, transportation and distribution, employee commuting, and the use of sold products. Assessing and managing Scope 3 emissions often requires collaboration with suppliers, customers, and other stakeholders.

By considering Scopes 1, 2, and 3 in their Sustainability Reports, energy companies can provide a comprehensive overview of their greenhouse gas emissions, highlighting their direct and indirect impact on climate change. This enables stakeholders to evaluate the company's efforts in reducing its carbon footprint and transitioning to a more sustainable energy future.

Chapter 2

Related work

2.1 Approaches in the literature

In the literature, the extraction of scopes 1,2 and 3 from the sustainability report tables, is not a topic analyzed and studied, being a very laborious and complex task since the different format and layout of the tables make this work very complex and time-consuming.
However, a more general approach of extracting data from tables is an area of study addressed in the literature. The challenge in this work is that tables do not have a persistent structure over time, and even within the same PDF the structures and layouts of the tables are changed as the data needs to be inserted within.
Precisely, table extraction from PDF documents is a challenging task that has garnered significant attention in the field of information extraction and document processing. Several studies have focused on developing techniques and methodologies to extract tables accurately and efficiently. The related work in this master's thesis encompasses various research efforts and approaches that have been explored in the context of table extraction from PDFs.

2.1.1 Rule-based Methods

Rule-based methods[3], in the context of table extraction from PDFs, rely on predefined rules, heuristics, and pattern matching techniques to identify and extract tables based on visual cues and document structure. These methods analyze the layout, graphical elements, and textual characteristics of the PDF documents to locate and extract tables accurately.

The rule-based approach typically involves defining a set of rules that describe the visual patterns and structural features of tables. These rules may include identifying horizontal and vertical lines, detecting cell borders, analyzing text alignments, and recognizing header rows or columns. By applying these rules to the PDF document, the algorithm can determine the presence and boundaries of tables. Template matching is a common technique used in rule-based methods, where predefined templates representing table structures are compared with the PDF document to identify matching regions. The templates can capture common table layouts, such as tables with specific numbers of rows and columns or tables with specific header formats. When a match is found, the algorithm extracts the corresponding region as a table.

Parsing algorithms[4] are also employed in rule-based methods to analyze the document's structure and hierarchy. These algorithms parse the PDF file, extracting the textual and graphical elements, and then analyze the relationships between these elements to identify table structures. For example, the algorithm may identify rows and columns based on the presence of line segments or patterns in the document. Regular expression-based methods are another approach used in rule-based table extraction. Regular expressions[5] define patterns that match specific textual or structural properties of tables, such as the presence of certain keywords or the arrangement of cell contents. By applying regular expressions to the PDF document, the algorithm can locate and extract tables based on these defined patterns.

One advantage of rule-based methods is their interpretability and control. Since the rules are predefined, it is possible to fine-tune and customize them based on specific requirements and document characteristics. However, rule-based methods may struggle with handling diverse and complex table layouts that deviate from predefined patterns. They may also be sensitive to variations in PDF document formats or lack robustness in handling noisy or imperfect documents.

In recent years, rule-based methods have often been combined with machine learning techniques or integrated into hybrid approaches to overcome their limitations. By leveraging the strengths of both rule-based and data-driven approaches, researchers aim to improve the accuracy and flexibility of table extraction from PDF documents.

2.1.2 Machine Learning Approaches

Machine learning approaches[6], in the context of table extraction from PDFs, involve leveraging algorithms and models to automatically learn patterns and characteristics of tables from annotated training data. These approaches aim to classify and extract table regions based on features extracted from the PDF documents.
Supervised learning algorithms[7] are commonly used in machine learning-based table extraction. These algorithms require a labeled Dataset, where each instance is annotated with the corresponding table region. Features are extracted from the PDF documents to represent the tables, and the algorithm learns to classify new instances based on these features. Support Vector Machines[8] (SVM), Random Forests[9], and Neural Networks[10] are commonly used supervised learning algorithms for table extraction.
The features used in machine learning approaches can encompass various aspects of tables, including textual attributes, structural properties, and visual characteristics. Textual features may include the presence of specific keywords or patterns within the table region. Structural properties can capture the number of rows and columns, the presence of header rows or columns, and the alignment of cells. Visual characteristics may involve features derived from image processing techniques, such as edge detection or texture analysis. To train the machine learning model, the labeled Dataset is split into a training set and a validation set. The model learns to generalize from the training examples and optimize its performance based on the validation set. The trained model can then be used to predict and extract table regions from unseen PDF documents.
Unsupervised learning methods can also be employed in table extraction to identify tables without relying on predefined rules or annotations. Clustering algorithms, such as K-means clustering[11] or DBSCAN[12], can group similar regions together, and the clusters that exhibit table-like characteristics are considered as tables. Topic modeling techniques, such as Latent Dirichlet Allocation (LDA)[13], have also been explored to identify table-like regions by modeling the latent topics within the PDF documents.
Machine learning approaches offer the advantage of adaptability and the ability to handle diverse table layouts. However, they require a substantial amount of labeled training data to achieve good performance, and the quality and representativeness of the training data can significantly impact the results. Additionally, the feature engineering process plays a crucial role in

the performance of machine learning approaches, and careful selection of relevant features is essential for accurate table extraction.
Overall, machine learning approaches have shown promise in automating table extraction from PDFs, and ongoing research focuses on improving the performance, robustness, and scalability of these techniques.

2.1.3 Natural Language Processing (NLP) Techniques

Natural Language Processing[14] (NLP) techniques, in the context of table extraction from PDFs, involve the use of computational methods to analyze and understand the textual content of the PDF documents. NLP techniques are employed to enhance the accuracy and interpretation of table headers, column names, and content.
Named Entity Recognition[15] (NER) is a common NLP technique used in table extraction. It aims to identify and classify specific entities within the text, such as names of organizations, people, dates, or locations. In the context of table extraction, NER can be utilized to recognize table headers or specific entities within the table cells. By identifying and labeling these entities, NER contributes to the accurate extraction and interpretation of table content.
Part-of-Speech[16] (POS) tagging is another NLP technique used in table extraction. It involves assigning grammatical tags to each word in a sentence, indicating its syntactic role and category (e.g., noun, verb, adjective). POS tagging can assist in identifying and understanding the context of table headers and content, allowing for more precise extraction of relevant information. Semantic analysis techniques are employed to comprehend the meaning and context of table content. Semantic analysis involves the use of linguistic patterns, semantic relationships, and ontologies to extract meaningful information from text. For example, semantic analysis can identify relationships between table headers and corresponding content, enabling more accurate extraction of structured data.
Linguistic patterns and rules can be applied to identify and extract table structures and content. These patterns capture common linguistic constructs, such as the presence of specific words or phrases, specific grammatical patterns, or numerical patterns within table cells. By leveraging linguistic patterns, NLP techniques contribute to the identification and extraction of tables with specific characteristics. Machine learning algorithms, commonly

used in NLP, can be employed for table extraction tasks. For instance, supervised learning algorithms can be trained on labeled Datasets to recognize and classify table elements based on linguistic features. Unsupervised learning techniques, such as clustering or topic modeling, can be applied to identify table-like structures or group related table elements. By integrating NLP techniques with other methods, such as rule-based or machine learning approaches, the accuracy and understanding of table extraction from PDFs can be significantly improved. NLP techniques enable the interpretation and extraction of meaningful information from the textual content of tables, enhancing the overall quality and usability of the extracted data. Ongoing research in NLP for table extraction focuses on developing advanced algorithms, improving entity recognition and semantic understanding, and exploring domain-specific language models to handle diverse types of PDF documents and optimize the extraction process.

2.1.4 Computer Vision and Image Processing

Computer vision and image processing techniques play a crucial role in table extraction from PDFs by analyzing the visual elements and structures present in the document. These techniques are employed to identify and extract table boundaries, cell structures, and other visual cues necessary for accurate table extraction.
Image segmentation[17] is a fundamental computer vision technique used in table extraction. It involves dividing an image or document into distinct regions based on visual properties such as color, texture, or intensity. In the context of PDF documents, image segmentation can be applied to identify and isolate table regions from the rest of the document. By separating tables from other content, image segmentation facilitates subsequent processing and analysis.
Edge detection[18] is another important technique used in table extraction. It aims to identify and extract the boundaries of objects or regions within an image. In the context of PDFs, edge detection can be utilized to identify the lines and borders that define table structures. By detecting the edges of table cells, rows, and columns, edge detection algorithms help in accurately extracting table content.
Contour analysis[19] is employed to analyze the shape and structure of objects within an image. In table extraction, contour analysis techniques can be applied to identify and extract the contours or outlines of tables and their constituent cells. By analyzing the hierarchical relationships and connectivity

between contours, the algorithm can accurately determine the boundaries of tables and cells. Pre-processing techniques in image processing are essential for enhancing the quality and readability of PDF documents prior to table extraction. These techniques may involve image enhancement to improve contrast, reduce noise, or sharpen edges. Pre-processing can also include deskewing, which corrects any rotation or tilt in the scanned document, ensuring that table boundaries are aligned properly.
Optical Character Recognition[20] (OCR) is a key component of image processing for table extraction. OCR algorithms analyze scanned or digital images to recognize and convert printed or handwritten text into machine-readable characters. In the context of table extraction from PDFs, OCR techniques can be employed to extract text content from table cells, enabling the extraction of structured data. Computer vision and image processing techniques are often combined with other methods, such as rule-based or machine learning approaches, to improve the accuracy and efficiency of table extraction from PDFs. By analyzing the visual elements and structures within the documents, these techniques provide valuable information for identifying and extracting tables, resulting in more reliable and precise extraction outcomes. Ongoing research in computer vision and image processing for table extraction focuses on developing robust algorithms that can handle various table layouts, handle noisy or imperfect documents, and adapt to different scanning or document capture conditions. The goal is to improve the accuracy, scalability, and automation of table extraction from PDFs through advances in computer vision and image processing techniques.

2.1.5 Hybrid Approaches

Hybrid approaches in table extraction from PDFs refer to the integration of multiple techniques and methodologies to leverage their respective strengths and overcome the limitations of individual approaches. These approaches combine different techniques, such as rule-based methods, machine learning algorithms, natural language processing (NLP) techniques, and computer vision/image processing, to achieve more accurate and robust table extraction. The main idea behind hybrid approaches is to use complementary methods in a synergistic manner to enhance the overall performance of table extraction. Here are a few examples of how different techniques can be combined within hybrid approaches:

- Rule-Based Methods with Machine Learning: In this approach, rule-based methods are combined with machine learning algorithms. The

predefined rules capture specific patterns or layout constraints, while machine learning models are trained to handle variations and adapt to different table structures. Rule-based methods can provide initial table region identification, and machine learning models can refine the extraction by learning from labeled training data.

- NLP Techniques with Computer Vision/Image Processing: Hybrid approaches combining NLP techniques and computer vision/image processing focus on improving the accuracy and interpretation of table content. NLP techniques, such as Named Entity Recognition (NER) and semantic analysis, enhance the understanding of table headers, column names, and content. Meanwhile, computer vision techniques analyze the visual cues, such as table boundaries and cell structures, to assist in accurate region identification and extraction.

- Machine Learning with Computer Vision/Image Processing: This approach combines machine learning algorithms with computer vision techniques. Machine learning models can be trained on annotated Datasets to classify and extract table regions, utilizing features extracted from the visual elements of PDF documents. Computer vision techniques, such as image segmentation or edge detection, can be employed to preprocess the PDFs, enhance the visual quality, and improve subsequent table extraction by the machine learning models.

By combining different techniques, hybrid approaches aim to overcome the limitations of individual methods and achieve higher accuracy, adaptability, and robustness in table extraction from PDFs. These approaches leverage the strengths of each technique to handle diverse table layouts, variations in document quality, and different types of PDF documents.
The selection and integration of techniques in hybrid approaches depend on the specific requirements, characteristics of the PDF documents, and the available resources. The goal is to create a cohesive framework that maximizes the benefits of different approaches and produces superior results in table extraction tasks. Ongoing research in hybrid approaches focuses on exploring new combinations of techniques, optimizing the integration process, and developing advanced algorithms to further enhance the accuracy and efficiency of table extraction from PDFs.

By examining the existing literature and related works in table extraction from PDFs, this master's thesis seeks to build upon previous advancements

and propose novel approaches that enhance the accuracy, efficiency, and scalability of table extraction methods. The review of related works will provide insights into the strengths, weaknesses, and limitations of existing techniques, identify research gaps, and lay the foundation for further exploration and innovation in the field of table extraction from PDF documents.

2.2 Description of algorithms used

In this project, a hybrid approach was used, based on the use of different techniques and libraries to achieve a maximized result in terms of accuracy and time. A hybrid approach is well-suited for extracting tables of unknown structure that can be structured, semi-structured, or unstructured. This is because a hybrid approach combines the strengths of different techniques to handle varying scenarios. When dealing with structured tables, the structured data extraction methods of the hybrid approach can accurately identify and extract tabular data based on known patterns and defined structures. For semi-structured tables, the combination of techniques like computer vision enables the system to analyze the visual cues and textual content to identify and extract the tabular information. In the case of unstructured tables, the hybrid approach using rule-based can help by looking for recognizable patterns that can serve to recognize previously observed structures. By combining these different approaches, a hybrid method can provide a versatile and effective solution for extracting tables from documents with unknown structures, as reported in the figure 2.1, regardless of whether they are structured, semi-structured, or unstructured.
Given that we therefore have tables whose format and layout are unknown, a hybrid approach was adopted in this project, which by combining two libraries, Tabula[21] and Camelot[22], and with the use of regex succeeds in having good accuracy in the number of tables recognized and in the accuracy of the contents within the tables. The combination of Camelot, regex, and Tabula proves to be an efficient solution for extracting tables from PDF files. Let's delve into the detailed reasons why this combination is effective:

- **Camelot** is a Python library that facilitates table extraction from PDF files using computer vision techniques. It offers an intuitive and user-friendly interface to analyze the visual components of PDF documents and extract tabular data. Camelot leverages the power of image processing algorithms to detect and recognize table structures within PDFs.
  One of the key features of Camelot is its ability to handle a wide range

of table layouts, including those with complex structures or irregular formats. By analyzing the visual cues present in the PDF, such as lines, borders, and cell boundaries, Camelot can accurately identify table boundaries. Camelot supports various table extraction methods, including both lattice-based and stream-based approaches. The lattice-based approach works well for tables with clearly defined grid lines, while the stream-based approach is effective for tables without visible grid lines. This flexibility allows Camelot to adapt to different types of table layouts, ensuring accurate extraction regardless of the structure.
Another notable feature of Camelot is its support for multiple output formats. Extracted tables can be saved in various formats, such as CSV, Excel, HTML, or JSON, making it easy to integrate the extracted data into different workflows or applications. Camelot also provides options for specifying custom table areas to extract, handling headers and footers, and dealing with rotated or skewed tables. These features enhance the versatility and adaptability of Camelot for extracting tables from PDFs with diverse characteristics. Moreover, Camelot offers integration with other popular Python libraries, such as Pandas[23], Numpy[24], and Matplotlib[25], facilitating seamless data analysis and visualization workflows after table extraction.
Overall, Camelot simplifies the process of extracting tabular data from PDF files by leveraging computer vision techniques. Its ability to handle various table structures, support multiple output formats, and provide customization options makes it a powerful tool for extracting valuable

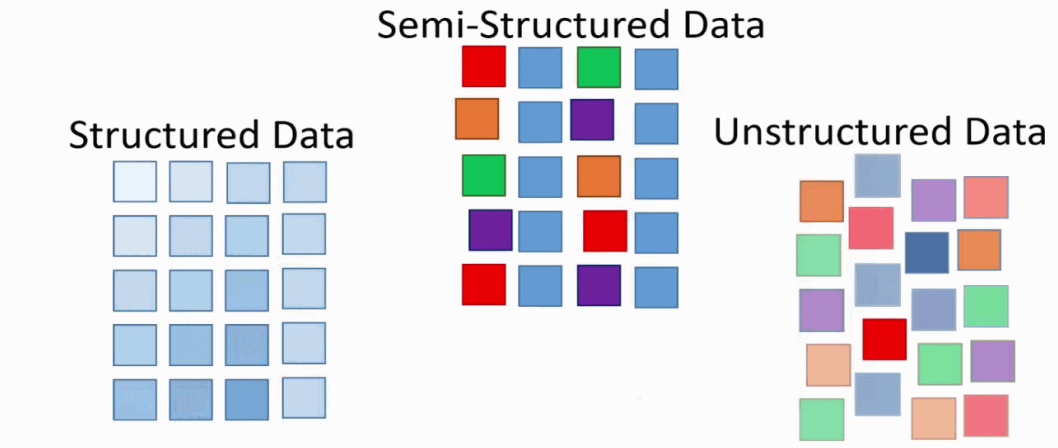

Figure 2.1. Different type of data

information from PDF documents.

- **Tabula** is an open-source Java library and command-line tool that specializes in extracting tabular data from PDF documents. It provides a user-friendly interface and a range of features that facilitate the extraction process.
  One of the key features of Tabula is its ability to handle a wide variety of PDF layouts and table structures. It utilizes advanced document analysis algorithms to analyze the PDF files and identify tables within them. Tabula can detect lines and text within the tables, allowing for accurate extraction of tabular data.
  Tabula supports both interactive and batch extraction modes. In the interactive mode, users can manually select the tables to extract by drawing bounding boxes around them. This enables precise control over the extraction process, especially in cases where there are multiple tables or complex layouts within a PDF. In the batch mode, Tabula can automatically detect and extract tables from a set of PDF files, making it suitable for large-scale data extraction tasks.
  The extracted tabular data can be saved in various formats, including CSV, JSON, and Excel. Tabula also provides options for customizing the output, such as specifying the delimiter for CSV files or selecting specific sheets from multi-sheet PDFs. This flexibility ensures that the extracted data is presented in a format that is convenient for further analysis or integration into other systems. Tabula offers a range of command-line options, allowing users to customize the extraction process based on their specific requirements. It provides features like specifying the area to extract, handling complex table structures, and skipping headers or footers. These options enhance the accuracy and efficiency of the extraction process, especially when dealing with PDFs that have varying layouts. Furthermore, Tabula has a friendly and active community of developers who provide support, contribute to the improvement of the library, and address user inquiries and issues.
  Overall, Tabula is a powerful and versatile tool for extracting tabular data from PDF files. Its ability to handle different table structures, support various output formats, and provide customization options makes it a valuable resource for data extraction and analysis tasks

- **Regular expression**, commonly referred to as regex, are a powerful tool for pattern matching and manipulation of text data. They are especially useful for extracting tables due to their versatile matching capabilities.

Regex allows for the definition of specific patterns or rules that match a particular structure or format within the text. For table extraction, regex patterns can be created to match the characteristics of table structures, such as row and column patterns, cell delimiters, or header/footer markers.
Regex provides a flexible and expressive way to define patterns. It allows you to specify precise rules for matching elements within the text, such as row and column separators or cell content patterns. By crafting appropriate regex patterns, you can accurately identify and extract table components.
Tables can have various structures, including different numbers of rows and columns, varying cell arrangements, and diverse header/footer formats. Regex allows for dynamic adaptation to these variations. By constructing flexible regex patterns, you can accommodate different table layouts and extract data reliably across various documents.
In addition to matching patterns, regex offers powerful text manipulation capabilities. You can use regex to clean and preprocess the text before extracting the table. This can involve removing unwanted characters, handling line breaks, or replacing specific patterns with appropriate separators. By manipulating the text with regex, you can enhance the quality and accuracy of the extracted table data.
Regex can be automated to process large numbers of documents in a scalable manner. Once you have defined the regex patterns for table extraction, you can apply them to a batch of PDFs or text files. This automation saves time and effort, especially when dealing with a large Dataset or recurring extraction tasks.
In summary, regex is valuable for table extraction because of its pattern matching capabilities, dynamic adaptability to diverse table structures, text manipulation features, scalability, and integration with programming languages. By harnessing the power of regex, you can effectively extract tabular data from documents, including PDFs, and further analyze and utilize the extracted information.

Part II

Problem specification

Chapter 3

Solution description

3.1 Problem description

3.1.1 General description

In today's business landscape, sustainable practices and financial performance go hand in hand. To understand and measure the impact of a company's operations, it is crucial to extract scope values and revenue values from sustainability reports and profit and loss accounts in PDF format. Sustainability reports provide comprehensive insights into a company's environmental performance, including direct and indirect emissions categorized into different scopes. Extracting scope values allows for the calculation of emission intensity, which measures the environmental impact per unit of revenue or other relevant metrics.
On the other hand, profit and loss accounts offer a financial snapshot of a company's revenues, expenses, and profitability. Extracting revenue values from these accounts is essential for assessing a company's financial performance and understanding its revenue-generating capabilities. Revenue values are key in calculating financial metrics such as revenue intensity, which evaluates the revenue generated per unit of environmental impact or other relevant metrics.
The process of extracting scope values and revenue values from PDF documents poses several challenges. PDFs are not inherently structured data formats, making it necessary to employ various techniques to extract the desired information accurately.
Once the scope values and revenue values are successfully extracted and validated, they can be integrated into a unified Dataset for further analysis.

This integration may involve combining data from multiple pages or sources to create a comprehensive picture of a company's environmental and financial performance. With the integrated Dataset, calculations can be performed to calculate metrics like emission intensity and revenue intensity, providing insights into the relationship between a company's environmental impact and its financial success.
In conclusion, the extraction of scope values and revenue values from sustainability reports and profit and loss accounts in PDF format is a critical step in assessing a company's environmental sustainability and financial performance. By accurately extracting and analyzing this data, organizations can make informed decisions, identify areas for improvement, and develop sustainable strategies for a greener and financially sound future.

3.1.2 Description of the implemented solution

The goal of this thesis, as explained earlier, is to calculate intensity metrics so that a company can be evaluated with standardized values based on revenue. In fact, a large company will have a higher number of emissions than a smaller company, but that does not make the small company behave more green than a large company. Therefore, using this metric helps to analyze different companies in an objective and standardized way.
In order to calculate the metrics then, two information are needed: emissions and sales. Information regarding emission scopes are found in sustainability reports, while information regarding sales, are usually within the profit and loss account. The purpose of the thesis then is to be able to extract the last year's numerical values associated with the three different scopes and to be able to extract the numerical value associated with last year's sales. The complexity of this project is due to the fact that within the sustainability reports, there are both values for the different scopes associated with past years, in order to compare them with the values obtained in the current year, and target values for the future. Similar situation arises for the value associated with sales. The other difficulty in developing this project is due to the high variability and difference between tables, even within the same report.
As the figures 3.1, 3.2 and 3.3 below show, taking ENDESA's sustainability report as an example, within the PDFs are the values for scope 1 in three different tabular formats.

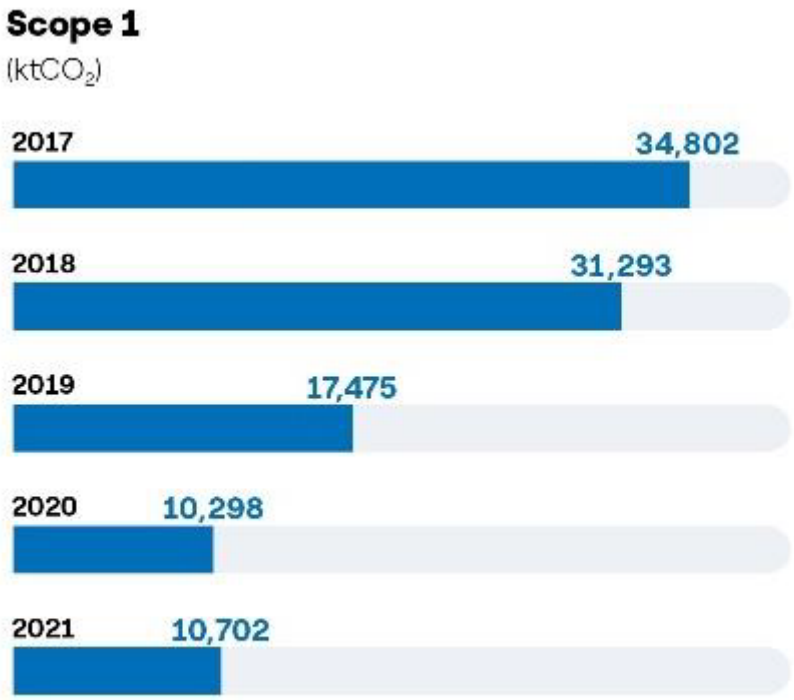

Figure 3.1. Example number 1 table for scope 1 from ENDESA

CO_2eq EMISSIONS (t)*

Scope Type	2019	2020	2021
CO_2eq (t) Scope 1	17,474,762	10,298,310	**10,702,129**
CO_2eq (t) Scope 2 (location based)[12]	460,890	457,184	**470,773**
CO_2eq (t) Scope 3	25,359,022	21,213,651	**21,737,472**
Total	**43,294,674**	**31,969,145**	**32,910,373**

Figure 3.2. Example number 2 table for scope 1 from ENDESA

Pillar	Matters	21 core WEF KPIs	Representative KPIs for ENDESA	2021	2020	Sustainability and EINF Reference 2021
Principles of governance	Governing purpose	Setting the purpose		Open Power strategic positioning		page 13
	Quality of governing body	Board composition	Women on the Board of Directors (%)	**36.4%**	30.8%	page 241
	Stakeholder engagement	Material issues that affect stakeholders		Priorities for the company and stakeholders		page 28
	Ethical behaviour	Anti-corruption	Workers who have received training on the anti-corruption policies and procedures (number)	**3,678**	2,035	page 235
			Total number of incidents of conflicts of interest/corruption confirmed (number)	**1**	3	page 295
		Protected ethics advice and reporting mechanisms	Complaints of breaches received through the ethics channel and other means (number)	**7**	4	page 294
	Risks and opportunity oversight	Integrating risk and opportunity into business processes				page 44
Planet	Climate change	Greenhouse gas (GHG) emissions	Scope 1 GHG emissions (t eq)	**10,702,129**	10,298,310	page 119
			Scope 2 GHG emissions (t eq) – *location based*	**470,773**	457,184	page 119
			Scope 2 GHG emissions - Acquisition of energy from the grid (Tn.)	**5,516**	No figures available	page 120
			Scope 2 GHG emissions - Losses from the distribution grid (Tn.)	**465,257**	No figures available	page 120
			Scope 3 GHG emissions (t eq)	**21,737,472**	21,213,651	page 120
		Implementation of TCFD		Qualitative		page 90
	Nature loss	Land use and ecological sensitivity	Protected areas affected (Km^2)	**789**	874	page 215
	Fresh water availability	Fresh water consumption in water stressed areas	Water withdrawal (hm^3)	**4,861.5**	5,215.3	page 204
			Water withdrawal in stressed areas (%)	**18%**	14%	Page 206

Figure 3.3. Example number 3 table for scope 1 from ENDESA

As for extracting sales values, it is easier, because the tables are in a more standard format. In this case, however, the complexity lies in the keyword search, in fact unlike scopes, which by convention are always written in the same way, with regard to sales, they can be written as sales, turnover or revenue.

Here are some additional details regarding the challenges and techniques involved in extracting scope values and revenue values from sustainability reports and profit and loss accounts in PDF format:

- **Text Extraction**: PDF files store text as a collection of characters positioned at specific coordinates on the page. Extracting text from a PDF requires utilizing libraries or tools that can interpret the underlying text structure. PyPDF2[26] is a commonly used library for extracting text from PDF documents. However, text extraction may not always capture the desired formatting or structure, making it necessary to further process and analyze the extracted text.
- **Table Extraction**: Sustainability reports and profit and loss accounts often contain tabular data, where scope values and revenue values may be organized. Extracting data from tables in PDFs can be challenging due to variations in table formatting, merged cells, and complex layouts. Tools like tabula and Camelot offer functionalities for extracting tables from PDFs. Tabula uses a technique called "extraction by area" to locate and extract tables, while Camelot employs algorithms to identify table structures. After table extraction, the data can be processed to identify and extract the relevant values.
- **Pattern Matching**: Once the text or tables are extracted, pattern matching techniques can be used to identify the desired values. Regular expressions (regex) are powerful tools for pattern matching and can help identify specific keywords or patterns associated with scope values (e.g., "scope 1," "scope 2," "scope 3") and revenue values (e.g., "revenue," "revenues," "sales"). By searching for these patterns within the extracted text or table data, the corresponding values can be identified and extracted.
- **Data Validation and Cleaning**: Extracted data may contain noise, inconsistencies, or errors. It is essential to validate and clean the extracted values to ensure accuracy. This process may involve removing unnecessary characters, converting data types, handling missing values, and addressing data inconsistencies.

- **Integration and Calculation**: Once the scope values and revenue values are extracted and validated, they can be integrated into a unified Dataset for further analysis. Integration may involve combining data from multiple pages or sources. With the integrated Dataset, calculations can be performed to calculate intensity metrics such as emission intensity or revenue intensity. These calculations help evaluate the relationship between environmental impact and financial performance.

As explained before, the combination of Camelot, regex, and Tabula offers a comprehensive and effective approach for extracting tables from PDF files. Camelot provides visual analysis of images, regex identifies table structures within the text, and Tabula extracts the data in a structured format. This combination successfully handles different table structures present in PDFs, ensuring accurate extraction of tabular data. The synergy between these three components provides a powerful method for efficiently and reliably extracting and manipulating tables from PDF files.
The figure 3.4 below shows a schematic of how the three approaches used work in parallel, and then are combined for preprocessing and later Dataframe creation. The algorithm 1 shown on the next page, on the other hand, shows at the pseudocode level the steps followed to extract the tables; starting with the Pdf and parsing the Pdf and using the three different approaches.

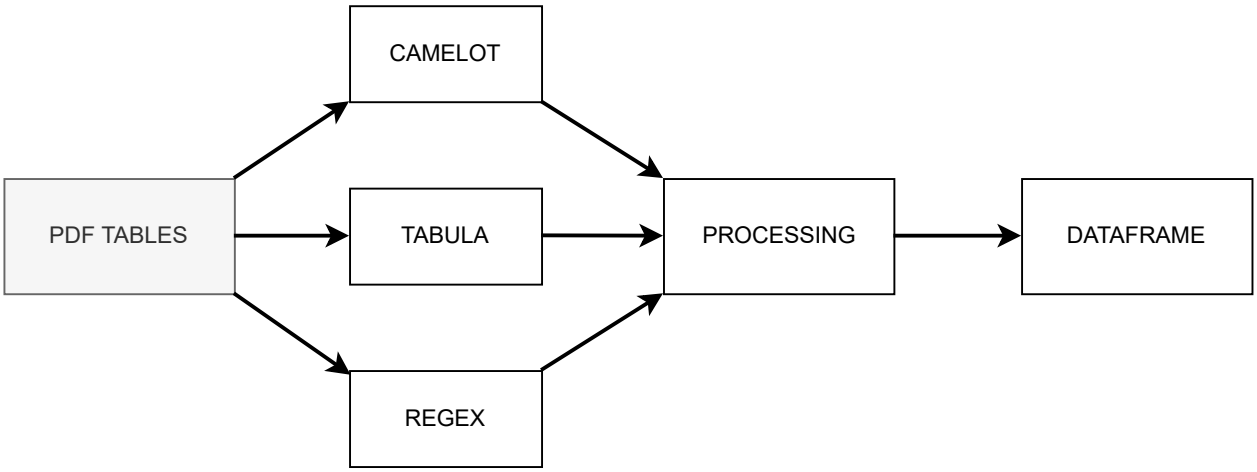

Figure 3.4. Schema of how Camelot, Tabula and regex are used

Algorithm 1 Table Extraction

Input: PDF file path
Output: Extracted tables as a DataFrame
Procedure `ExtractTables`(pdf_file)
  $extracted\_tables \leftarrow \{\}$ $pdf\_text \leftarrow ConvertPDFtotextusingPDFextractor$
  `// Table extraction using one of the three methods`
  $method\_tables \leftarrow Tablextractor(pdf\_file)$ **foreach** $table$ ***in*** $method\_tables$ **do**
    $extracted\_tables \leftarrow AddTableToDictionary(extracted\_tables, table)$
  end
  return $extracted\_tables$
Function `AddTableToDictionary`($dict, table$)
  $df \leftarrow ConvertTableToDataFrame(table)$ $dict \leftarrow AddToDictionary(dict, df)$ **return** $dict$

3.2 Designed solution

In this section, the solution implemented at the logic and sequence level is explained in detail.
First, the project is divided into two separate parts, one for extracting information from the sustainability report, and one for extracting data from the profit and loss account. Since we do not know the exact location of the important data within PDFs, we adopted as a strategy to extract all the text within the PDF using a library already implemented in python called PyPDF2.
PyPDF2 is a Python library that provides functionalities for working with PDF documents. It allows developers to extract text, manipulate pages, merge and split PDFs, encrypt and decrypt files, and perform various other operations on PDFs. The library is built on top of the original PyPDF library but offers additional features and improvements. One of the main features of PyPDF2 is text extraction. It enables users to extract text content from PDF documents, including both plain text and formatted text. This functionality is particularly useful when analyzing PDFs for specific information or performing text-based searches within the documents. PyPDF2 also allows for the manipulation of PDF pages. Developers can merge multiple PDF files into a single document, extract specific pages or ranges of pages,

rotate pages, and reorder them as needed. These operations provide flexibility in handling PDF files and customizing their structure. The library supports encryption and decryption of PDF files, enabling users to protect sensitive information within the documents. It provides methods for setting passwords and permissions, as well as decrypting encrypted PDFs with the appropriate credentials.
While PyPDF2 is a powerful library for basic PDF manipulation and extraction tasks, it has some limitations. It does not support all PDF features, such as interactive forms or advanced encryption methods. Additionally, PyPDF2 is not designed for direct content editing within the PDF documents, as it focuses primarily on extraction and manipulation of existing content.
Overall, PyPDF2 provides a convenient and user-friendly interface for working with PDF documents in Python. It simplifies common PDF operations and enables developers to extract text, manipulate pages, encrypt files, and perform various tasks required for PDF processing in a straightforward manner.

3.2.1 Table extraction from PDFs

So, through the use of this library we are able to extract the textual content of each page of the PDF. Having, for each PDF, a very variable number of pages, which depends on the fact of how much information is reported within the report, we use regular expressions, called regex, which search within the page if keywords appear.
In the case of searching for numerical values within the sustainability report, the words "scope 1" are used as keywords , "scope 2" and "scope 3". On the other hand, in the case of searching for values within the profit and loss account, the words "revenues", "turnover" and "sales" are searched. Using this method, and taking into consideration that thanks to the PyPDF2 library even tables are decomposed into text, one is able to filter only those pages where the information sought appears, without having to apply the table extraction methods even on pages where the desired information is not present. This implementation greatly helps to decrease program execution time, as it dramatically reduces the number of times the table extraction functions are called. Once only the pages containing the keywords have been extracted, the three table extraction methods that were previously explained from a

theoretical point of view are applied; for each page all three extraction methods, Camelot, Tabula, and a function constructed by regex, are applied. If at least one of the three methods extracts at least one table, a new dictionary is created within a list in which we find as the first key the text that is on the current page of the PDF, and then a new key is created for each method that finds at least one table within the page. Once all pages have been extracted from the PDF, this first program section returns a list of dictionaries, in which the number of items within the list represents the number of pages in the pdf in which the keywords appear. This algorithm is applied to the sustainability report, to extract the tables containing purposes, and to the profit and loss account, to extract the tables containing sales-related values.

Algorithm 2 Extracting tables from PDFs

```
Input  : file  - PDF name
         pattern  - Pattern's regex
Output: filtered_page_contents  - Pages containing tables

file ← "Filename"  pattern ← Regex pattern
page_contents ← []  filtered_page_contents ← []

// Open the PDF in a read mode
with open(file) as pdf_file:
    // Iterate on all pages of the PDF file
    for page_num ← 0 to len(pdf.pages) do
        // extract the text from the page
        text ← pdf_page.extract_text()
        // Checks whether the text contains the pattern
        if pattern.search(text) then
            // extract the table using one of the three method
            method_tables ← extract_tables_method(text)
            // Checks whether at least one of the methods has
               extracted a table
            if any(pattern(table.values) in (method_tables) then
                filtered_page_contents.append({'page_num':     page_num+1,
                 'text': text, 'tables_method': method_tables})
```

The algorithm 2 shows the pseudo code used to implement the first part,

which returns as output a list of dictionaries, where each dictionary within the list represents a page of the original pdf in which at least one table was found by at least one of the three methods.
In case tables from all methods are extracted from a page, the result is reported as follows :

filtered_page_contents[{
'page_num': page_num + 1, 'text': text,
'tables_tabula': tabula_tables, 'tables_camelot': camelot_tables,
'tables_regex': regex_tables }]

3.2.2 Creation of a regex function

At this point, we have two different dictionary lists: one for tables extracted from the sustainability report, and one for tables extracted from the profit and loss account. To get to this, as explained earlier, the three different approaches are used. The first two, already previously exposed, use libraries implemented in python, which makes their use easier and faster. Looking initially at the behavior of these two libraries, it can be seen that just using them does not lead to optimal results, since anyway being the pdf tables in different formats, it can happen that both methods are not able to extract the tables correctly. The idea of implementing a hand-developed function comes from the need to have a triple check to get a better result. By having a triple control, the accuracy and precision of table extraction increases considerably, making this project more efficient in terms of accuracy.
In order to develop a function that uses regexes to extract tables from PDFs, we need to find a pattern that identifies the header of the table, a pattern that identifies which rows are inside the table, and a pattern that, to improve the accuracy of table extraction, is able to remove rows that do not contain the desired information. In developing the function that is used for header recognition, use is made of the fact that the tables studied previously to develop this project, being tables that present numerical values for each key involved, present in the header the years to which the values refer. Therefore, using the fact that the tables are converted to text format via PyPDF2, we can search within each row, via regex, if a sequence of characters appears that can represent a sequence of consecutive numbers, which theoretically represent the years present in the header. Through the use of regex, the row and its location within the PDF page is saved, so that we can then later start from that point to parse and format the rows present within the table.

Once the header is saved, it is formatted and saved in a list, so that each year represents a value within the list, so that subsequently each row within the table is formatted in the same way.
Once the header is formatted, the main function starts extracting text from each line, which is transformed and formatted in the same way as the header. During line formatting, some malfunctions may occur: It may happen that some rows have fewer values than the header, or it may pick up that some rows have special characters that are not recognized during extraction.
To overcome this problem, each row is inserted within a list, and the values are inserted within the table starting at the end of the list, where there are normally numeric values associated with the years. Once the numeric values are finished, that is, once the same number of numbers are found as are in the header in the list, a check is made via regex in which a search is made to see if at least one of the keywords is contained within the row. If at least one of the keywords is contained within the row, the only information saved besides the numeric values is the keyword. Once this process is completed for each row within the table, a final check function is applied, which removes each row from the table if it does not contain one of the keywords.
This last control is used to decrease the number of rows within each table, making the reading and analysis process faster and more efficient. The figure 3.5 shows how the function works.

The use of this algorithm, leads to excellent results at the level of table extraction, as it is very fast in terms of time and very accurate in terms of the accuracy of the data extracted from the table. One of the problems that may arise while using this algorithm is that the tables within the PDF have a strange layout that is extracted differently from the classical type on which the created regex function is based. However, having a very good number of tables extracted coin this method and, considering the fact that this method is added to the other two existing ones so as to have a triple check, it represents a very good solution.
The image 3.6 and table 3.1 below represent an example of how a table contained within a PDF page is extracted through the use of the regex function just explained. As can be seen, all rows that, in this case we are dealing with the sustainability report and therefore we are considering scopes 1,2 and 3 in the search, do not contain one of the keywords or do not pass the imposed filters, are saved and transformed within the Dataframe.

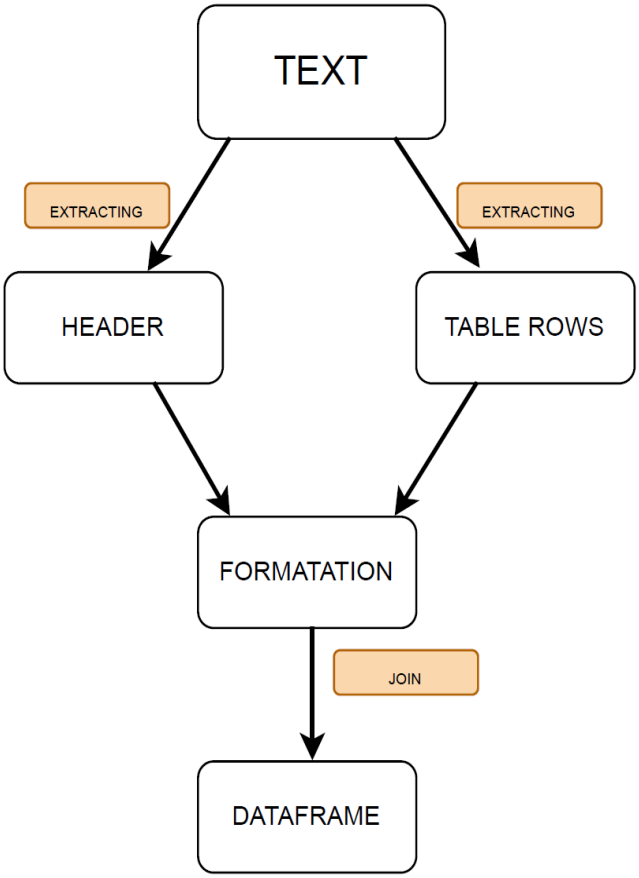

Figure 3.5. Schema of how the regex function works

Environmental indicators	2018	2019	2020
Total GHG emissions - Scope 1 (mt CO_2 eq)	57.2	46.2	**38.6**
of which emissions from energy production	54.7	43.7	**36.4**
of which CH_4 emissions	1.8	1.7	**1.5**
Total GHG emissions - Scope 2 (mt CO_2 eq)	2.9	2.5	**2.3**
Total GHG Emissions - Scope 3 (mt CO_2 eq)	139.2	133.6	**134.0**
of which use of products sold	62.0	60.9	**61.5**
CO_2 emission ratio - Energy generation - Scope 1 (kg CO_2 MWheq)	284.1	220.0	**212.5**
Reduction of CO_2 emission ratio for electricity production compared with 2012 (%)	-29	-44	✔**-52**
NOx emissions (kt)	60.4	52.8	**47.5**
SO_2 emissions (kt)	118.3	124.3	**119.6**
Fine particle emissions (kt)	4.8	4.7	**4.4**
Primary energy consumption - Total (excluding own consumption) (TWh)	330	343	**285**
Total consumption - freshwater and non-freshwater (mm^3)	85.1	94.5	**76.8**
Rate of reduction in the water consumption of industrial activities compared with 2019 (%)	NA	0	**-19**
Reduction in ratio of freshwater withdrawals/MWh of electricity production compared with 2012 (%)	-39	-36	✔**-46**
Environmental risk prevention plan (% of relevant revenues)	87.6	81.2	**82.7**
Environmental expenditure (€ m)	406	466	**553**
Environment-related complaints (no.)	24	10	**6**
Environment-related convictions (no.)	0	1	**2**
Amount of compensation (€ k)	0	13	**14**
Non-hazardous waste recovery rate (%)	85	68	**76**
Hazardous waste recovery rate (%)	30	30	**30**
Certified environmental management system (% of relevant revenues)	80.3	72.4	**75.7**

Figure 3.6. Example of a possible table, extracted from Engie Sustainability report

	2018	2019	2020
Scope 1	57.2	46.2	38.6
Scope 2	2.9	2.5	2.3
Scope 3	139.2	133.6	134.0

Table 3.1. Example of a dataframe extracted from a table

3.2.3 Data cleaning and data filtering

Once the tables have been extracted from the pages that contain at least one of the keywords, the tables obtained from the three different methods are transformed into Dataframes, through the use of Pandas, which allows the extracted tables to be transformed in list format into Dataframe objects.
After having the tables in a Dataframe format, the process of filtering and formatting the tables begins. As mentioned earlier, the tables are extracted from the pages that contain at least one of the keywords, this implies that some of the extracted tables may not contain the important information and related to this or project. The filtering process involves eliminating tables in which numeric values do not appear in the header, since normally as explained earlier in the last chapter,having to search for numeric values associated with years, the header must be formed by the years to which the values refer.
Another process of table transformation occurs because it may happen that the extraction of tables through functions already implemented, thus through the use of tabula and camelot, the formatting of the header is incorrect. In fact, it may happen that new rows are added to the beginning of the table containing null rows. By eliminating them, we can then have better precision in extracting values once we have only the tables we need.
Once the tables have been filtered and transformed, we obtain a list of Dataframes that will consist of the tables that are important for extracting the numerical values of interest.

3.2.4 Extracting values from tables

Once the tables are ready, we proceed with the search and extraction of the values of interest. For this part, the searches for purposes and for sales work in parallel, keeping the same method but changing only a few details, such as the keywords to be searched and the units of measurement.
The main idea for this part of the project is to analyze each Dataset and

create a dictionary of dictionaries, where each principal dictionary represents one of the analyzed PDFs, which is composed of a dictionary where each key represents one of the searched keywords. If that word was found within the Dataframe and a numeric value associated with the last desired year was found, the numeric value is entered into the dictionary created. In this part of the project we filter one last time on the content of the rows of the dictionary; in fact, it may happen that more information related to a specific purpose is reported within a table, such as information related to CO2 emissions, or or information related to the emission ratio. Since the purpose of this project is the extraction of the values of the scopes, some keywords are sought that can help in the extraction of only the desired and necessary values. Having then extracted the desired values, all that remains is to save them in the correct unit of measurement; In the case of values for emissions, we create a function that searches within the row for one of the keywords associated with that unit of measurement, i.e., searching for words such as "thousands" or"millions," and successively, based on the word found, we multiply the value obtained by the reference value for the unit of measurement keyword. In the case associated with extracting values for sales, the keywords in this case are "million" and "billion." Again, once the unit of measurement is extracted, the numerical value is multiplied by the associated value.
At this point, having two dictionaries of dictionaries, only one value per keyword needs to be extracted. It may happen that, in some cases, non-numeric values or numeric values not related to our project are extracted from Dataframes that have passed filtering but are not important. To extract only the necessary values, we use a majority method, in which the more times a numeric value is found within the dictionary, the more important it is. To give greater weight to theoretically correct values, we give greater weight to the Dataframes from which we extract the values associated with all keywords. In fact, as can be imagined, it is quite normal to think of connected values referenced to similar topics as being included in the same table within the PDF. At this point, a final dictionary is created in which are the keywords and the numerical value found repeatedly related to it. At this point, you are able to calculate the metric of interest, intensity, in the following way:

$$Intesity = \frac{Scope1 + Scope2 + Scope3}{Sales} \tag{3.1}$$

Chapter 4

Experimental evaluation

This section brings and analyzes the results obtained from this project. Since the goal of this project is the extraction of information found within the tables of PDFs, classical metrics for evaluating the functionality of a project cannot be used, since the goal is to extract a relatively small number of tables, but that must be very precise. For the creation of this project, a large number of PDFs were not used, given the high difficulty in being able to find a large number of companies' sustainability and profit and loss account PDFs.
Since it was not possible to find a large number of PDFs, the focus was on correctly extracting emission and sales information for each company.
Some examples of the results obtained are given in this section to make the process carried out clearer. As explained earlier, as this is a very complicated project, the accuracy in extraction is not absolute, in fact the type of formatting of the tables, the arrangement in space, color, shading and gradation can be disturbing elements in extracting information from the tables.
Having premised that the difficulty of the task is considerable, given the complicated structure of the tables, another problem encountered in this task is that of the execution time to get to the solution of the problem. In fact, as you can imagine, the execution time of the project, depends solely on two factors:

- **length of PDF**: the length of the PDF is a very important factor to consider if you want to study the time it takes the program to analyze and extract the PDF. In fact, taking into consideration 5 different sustainability reports from companies we can observe that the extraction time is proportional to the length of the PDF; in fact, increasing the number of pages in the PDF, the extraction time of it also increases.

As can be seen from the table 4.1 below, when the number of pages increases, the table extraction time also increases. And proportional, since usually the greater the number of pages, the longer it takes to extract the information, but not directly, since the time depends not only on the number of pages, but also on another factor that we will analyze in the second point.

	Trials	N of pages	Time mean	Time std
Engie	5	60	70.12 s	12.35s
Holmen	5	100	134.22 s	29.23 s
Eni	5	452	1092 s	121.78 s
Endesa	5	354	396.34 s	43.77 s
Chevron	5	84	124.14 s	22.10 s

Table 4.1. Example of some pdfs and their extraction times in relation to the number of pages

- **PDF complexity**: Another factor to consider regarding the time taken to extract information in tables within PDFs concerns the structure and complexity of the PDF. In fact, as explained earlier, tables are extracted only from pages where the regex function finds at least one keyword within the page. If no table appears within that page that contains the information regarding purposes or issues, the table will still be extracted and only then, before inserting it within the list of useful tables will a check be made on the contents of the table. Therefore, the number of tables within the pages and the structure and complexity of the tables are also an important factor regarding the execution time. It may happen that some tables have strange layout, nuances or complex structures, which increase the extraction time.

4.1 Results

In this section, examples of the results obtained will be brought so that we can better visualize the purpose of the project and be able to understand its use. Four different companies for which the algorithm was applied will be brought as examples, and the results obtained will also be displayed.

4.1.1 EDF

EDF group greenhouse gas report ($MtCO_2e$)	2019	2020	2021
Scope 1 emissions	33	28	27
Scope 2 emissions	0.3	0.3	0.3
Scope 3 emissions	119	107	102

Figure 4.1. Emissions values from EDF sustainability report (2021)

(in millions of euros)	2021	2020	Variation	Variation (%)	Organic variation (%)
Sales	84,461	69,031	15,430	22.4	21.6
Operating profit before depreciation and amortisation (EBITDA)	18,005	16,174	1,831	11.3	11.3
Operating profit (EBIT)	5,225	3,875	1,350	34.8	35.9
Income before taxes of consolidated companies	5,585	1,293	4,292	331.9	334.3
EDF net income	5,113	650	4,463	686.6	719.1
Net income excluding non-recurring items (1)	4,717	1,969	2,748	139.6	150.3
Net income excluding non-recurring items, adjusted for the remuneration of hybrid bonds	4,170	1,468	2,702	184	n.a
Group cash flow (2)	(1,525)	(2,660)	1,135	42.7	n.a
Net indebtedness (3)	42,988	42,290	698	1.6	n.a

Figure 4.2. Sales value from EDF Management report (2021)

	2019	2020	2021
Scope 1	33	28	27
Scope 2	0.3	0.3	0.3
Scope 3	119	107	102
Sales(m)	\	69031	84461
Intensity	\	0.00195	0.0053

Table 4.2. Values extracted from the EDF PDFs and the intensity metric calculated with them

	n	N of pages	Time mean	Time std
Sustainability report	5	572	122.84 s	24.43 s
Management report	5	33	36.64	4.76 s

Table 4.3. Mean time and std time for extracting tables from EDF PDFs

4.1.2 EDP

Quarterly P&L (€ million)	1Q20	2Q20	3Q20	4Q20	1Q21	2Q21	3Q21	4Q21	Δ YoY %	Δ QoQ %	2020	2021	Δ %
Revenues from energy sales and services and other	3 502	2 681	2 876	3 389	3 088	2 995	3 917	4 982	47%	27%	12 448	14 983	20%
Cost of energy sales and other	2 027	1 499	1 757	2 074	1 780	1 888	2 699	3 781	82%	40%	7 356	10 148	38%
Gross Profit	**1 475**	**1 182**	**1 119**	**1 315**	**1 308**	**1 108**	**1 218**	**1 201**	**-9%**	**-1%**	**5 092**	**4 835**	**-5%**
Supplies and services	201	201	207	248	195	213	207	274	10%	32%	857	889	4%
Personnel costs and Employee Benefits	165	157	143	203	162	171	159	175	-14%	10%	667	666	0%
Other operating costs (net)	128	(60)	13	(460)	100	(85)	47	(398)	-13%	-	(379)	(335)	11%
Operating costs	**494**	**297**	**363**	**(9)**	**457**	**300**	**413**	**50**	**-**	**-88%**	**1 145**	**1 220**	**7%**
Joint Ventures and Associates	(1)	6	(2)	0	13	20	10	65	-	-	3	108	-
EBITDA	**980**	**891**	**754**	**1 325**	**864**	**828**	**815**	**1 216**	**-8%**	**49%**	**3 950**	**3 723**	**-6%**
Provisions	16	35	78	(17)	12	(9)	50	7	-	-	112	61	-46%
Amortisation and impairment (1)	367	401	340	524	356	366	376	634	21%	69%	1 632	1 732	6%
EBIT	**597**	**455**	**336**	**818**	**496**	**470**	**389**	**575**	**-30%**	**48%**	**2 206**	**1 931**	**-12%**
Financial Results	(206)	(162)	(137)	(166)	(123)	(131)	(102)	(155)	-7%	52%	(671)	(511)	24%
Profit before income tax and CESE	**391**	**293**	**199**	**652**	**373**	**339**	**287**	**421**	**-36%**	**46%**	**1 535**	**1 420**	**-8%**
Income taxes	92	42	39	136	63	100	74	25	-82%	-67%	309	262	-15%
Extraordinary contribution for the energy sector	63	(0)	3	-	51	0	0	2	-	-	65	53	-18%
Net Profit for the period	236	252	157	517	259	239	213	394	-24%	85%	1 161	1 105	-5%
Attrib. to EDP Shareholders	**146**	**169**	**108**	**378**	**180**	**164**	**167**	**146**	**-61%**	**-13%**	**801**	**657**	**-18%**
Attrib. to Non-controlling Interests	90	83	49	138	79	75	46	248	80%	-	361	448	24%

Figure 4.3. Sale values from EDP result report (2021)

	2019	**2020**	**2021**
Scope 1	14.363	9.304	9.805
Scope 2	0.846	0.593	0.792
Scope 3	11.730	11.572	10.304
Sales(m)	\	12448	14983
Intensity	\	0.00173	0.00140

Table 4.4. Values extracted from the EDP PDFs and the intensity metric calculated with them

CLIMATE CHANGE	UN	2021	2020	2019	2018
HYDROELECTRIC PRODUCTIVITY INDEX					
Portugal	#	0.93	0.97	0.81	1.05
Spain	#	0.91	1.03	0.90	1.28
EMISSIONS					
Specific CO_2 emissions[1]					
Global	g/kWh	164	146	216	257
Thermal	g/kWh	673	567	649	768
CO_2 equivalent emissions					
Scope 1	**ktCO₂eq**	**9 805**	**9 304**	**14 363**	**18 429**
Stationary combustion	ktCO₂eq	9 781	9 273	14 338	18 404
SF_6 Emissions	ktCO₂eq	11	17	9	10
Company fleet	ktCO₂eq	14	13	15	15
Natural gas consumption	ktCO₂eq	0	0	0	0
Scope 2 (location-based[2])[4]	**ktCO₂eq**	**792**	**594**	**846**	**602**
Electricity consumption in office buildings	ktCO₂eq	2	1	1	2
Electricity losses in distribution	ktCO₂eq	766	568	824	577
Renewable plants self-consumption	ktCO₂eq	24	25	21	23
Scope 2 (market-based[3])[4]	**ktCO₂eq**	**773**	**574**	**829**	**585**
Electricity consumption in office buildings	ktCO₂eq	0	0	0	0
Electricity losses in distribution	ktCO₂eq	766	568	824	577
Renewable plants self-consumption	ktCO₂eq	7	6	5	8
Scope 3[5]	**ktCO₂eq**	**10 304**	**11 572**	**11 730**	**11 334**
Purchased goods and services (C01)	ktCO₂eq	721	18	28	49
Capital goods (C02)	ktCO₂eq	2 610	335	349	330
Fuel and energy related activities (C03)	ktCO₂eq	5 185	6 807	6 784	6 399
Upstream transportation and distribution (C04)	ktCO₂eq	66	933	611	675
Waste generated in operations (C05)	ktCO₂eq	18	n.a.	n.a.	n.a.
Business travels (C06)	ktCO₂eq	3	2	7	10
Commuting (C07)	ktCO₂eq	12	n.a.	n.a.	n.a.
Use of sold products (C11)	ktCO₂eq	1 688	3 478	3 951	3 871
SF_6	kg	399	724	394	440
Portugal	kg	195	206	194	246
Spain	kg	45	298	54	100
South America	kg	159	217	140	92
North America	kg	0	0	6	0
Rest of the Europe	kg	0	3	0	3
APAC	kg	0	0	0	0

Figure 4.4. Emissions values from EDP sustainability report (2021)

	n	N of pages	Time mean	Time std
Sustainability report	5	331	471.34 s	47.21 s
Profit and Loss report	5	29	34.45 s	3,87 s

Table 4.5. Mean time and std time for extracting tables from EDP PDFs

4.1.3 Holmen

Greenhouse gas emissions Scope 1–3, '000 tonnes CO_2e	2021	2020
Scope 1: Direct greenhouse gas emissions[8)]	97	79
Scope 2: Indirect greenhouse gas emissions from purchased electrical energy[9)]	60	38
Scope 3: Emissions in the value chain	550	460
of which category 1: Purchased goods and services[10)]	*136*	*100*
of which category 2: Capital goods[11)]	*120*	*80*
of which category 3: Fuel and energy-related activities[12)]	*38*	*36*
of which category 4: Upstream transport[13)]	*56*	*56*
of which categories 6 & 7: Travel	*4*	*4*
of which category 9: Downstream transport[14)]	*196*	*184*
Total emissions*	707	578

Figure 4.5. Emissions values from Holmen sustainability report (2021)

	Quarter			Full year	
SEKm	**4-21**	3-21	4-20	**2021**	2020
Net sales	**4 770**	4 877	4 249	**19 479**	16 327
Operating profit excl. item affecting comparability	**1 185**	1 129	595	**4 061**	2 479
Operating profit	**1 006**	978	595	**3 731**	2 479
Profit after tax	**868**	763	512	**3 004**	1 979
Earnings per share, SEK	**5.4**	4.7	3.2	**18.5**	12.2
Operating margin, %*	**25**	23	14	**21**	15
Book value, forest assets	**47 080**	43 693	43 202	**47 080**	43 202
Cash flow before investments and change in working capital	**704**	980	369	**3 375**	2 411
Debt/equity ratio, %	**9**	10	10	**9**	10

Figure 4.6. Sale values from Holmen year-end report (2021)

	2019	2020	2021
Scope 1	\	0.79	0.97
Scope 2	\	0.38	0.60
Scope 3	\	4.60	5.50
Sales(m)	\	16327	19479
Intensity	\	0.0004	0.0005

Table 4.6. Values extracted from the Holmen PDFs and the intensity metric calculated with them

	n	N of pages	Time mean	Time std
Sustainability report	5	100	28.34 s	6.65 s
Profit and Loss report	5	20	24.56 s	3.26 s

Table 4.7. Mean time and std time for extracting tables from Holmen PDFs

4.1.4 ENI

		2022	2021	2020
TRIR (Total Recordable Injury Rate)	(total recordable injuries/worked hours) x 1,000,000	0.41	0.34	0.36
employees		*0.29*	*0.40*	*0.37*
contractors		*0.47*	*0.32*	*0.35*
Direct GHG emissions (Scope 1)	(mmtonnes CO_2eq.)	39.39	40.08	37.76
Indirect GHG emissions (Scope 2)		0.79	0.81	0.73
Indirect GHG emissions (Scope 3) other than those due to purchases from other companies[b]		164	176	185
Net GHG Lifecycle Emissions (Scope 1+2+3)[c]		419	456	439
Net Carbon Intensity (Scope 1+2+3)[c]	(gCO_2eq./MJ)	66	67	68
Net carbon footprint upstream (Scope 1+2)[c]	(mmtonnes CO_2eq.)	9.9	11.0	11.4
Net carbon footprint Eni (Scope 1+2)[c]		29.9	33.6	33.0
Direct GHG emissions (Scope 1)/operated hydrocarbon gross production (upstream)	(tonnes CO_2eq./kboe)	20.64	20.19	19.98
Carbon efficiency index Group		32.67	31.95	31.64
Direct methane emissions (Scope 1)	(ktonnes CH_4)	49.6	54.5	55.9
Volumes of hydrocarbon sent to routine flaring (upstream)	(billion Sm^3)	1.1	1.2	1.0
Total volume of oil spills (> 1 barrel)	(barrels)	6.139	4,408	6,824
of which: due to sabotage		*5,253*	*3,053*	*5,866*
operational		*886*	*1,355*	*958*
Freshwater withdrawals	(million m^3)	131	125	113
Re-injected production water	(%)	59	58	53

Figure 4.7. Emissions values from ENI sustainability report (2022)

	2020	2021	2022
Scope 1	37.76	40.08	39.39
Scope 2	0.73	0.81	0.79
Scope 3	185	176	164
Sales(m)	43987	76575	132512
Intensity	0.005	0.003	0.002

Table 4.8. Values extracted from the ENI PDFs and the intensity metric calculated with them

		2022	2021	2020
Sales from operations	(€ million)	132,512	76,575	43,987
Operating profit (loss)		17,510	12,341	(3,275)
Adjusted operating profit (loss)[a]		20,386	9,664	1,898
Exploration & Production		*16,411*	*9,293*	*1,547*
Global Gas & LNG Portfolio		*2,063*	*580*	*326*
Refining & Marketing and Chemicals		*1,929*	*152*	*6*
Plenitude & Power		*615*	*476*	*465*
Adjusted net profit (loss)[a][b]		13,301	4,330	(758)
Net profit (loss)[b]		13,887	5,821	(8,635)

Figure 4.8. Sale values from ENI year-end report (2022)

	n	N of pages	Time mean	Time std
Sustainability report	5	452	1092 s	121.78 s
Annual report	5	76	74.34 s	18.36 s

Table 4.9. Mean time and std time for extracting tables from ENI PDFs

The figures 4.1 , 4.3 ,4.5 and 4.7 show the values of scopes inside the Sustainability report within three possible energy companies, EDF, EDP , Holmen and ENI.
Instead, the figures 4.2, 4.4 ,4.6 and 4.8 show sales values within the same companies, but in the Profit and Loss account pdfs.
The tables of the different pdfs are very different from each other and how a cross-solution that can use different types of algorithms is the best solution. Different shades, different setting of columns and information present for each row pose a problem for a solution that follows a linear trend. The developed approach allows the development of different implementation methodologies, which greatly improve the results obtained from the different approaches used explained above.
It must be remembered, however, that the tables in the pdfs, although from different companies, still have a constant basic structure, namely the name of the purpose in the first column, the years as column names, and numerical values representing the values of the scopes.
The tables 4.2 , 4.4 ,4.6 and 4.8 show the values extracted from the tables for the three purposes and for sales. As can be seen, the extracted values refer to the number of year in the columns, tables with less year will extract fewer values from the tables. In addition, the tables also show the values for the intensity metric, calculated later using the formula explained earlier.
Instead, the tables 4.3, 4.5 ,4.7 and 4.9 show the number of attempts, number of pages of pdfs , mean and standard deviation to analyze the pages and extract the contents from the tables.
As can be seen and as analyzed above, the number of pages affects the average time to extract the information. In addition, the complexity of the pages and tables also affects the difficulty and timing of analysis and extraction.

Part III

Conclusion and future work

Chapter 5

Final consideration

After showing some examples of how it is possible to use the created algorithm, we can therefore conclude that the project has many positive and interesting aspects, however, as could be seen, it also has some weaknesses. The program's adaptability in extracting tables of different types, with very differently structured data, is one of the strengths of this project; moreover, the ability to extract the desired tables is due to a good accuracy in the extraction. The total accuracy, which consists of the number of information extracted correctly from the PDFs is 79%.
The value obtained represents an excellent goal considering the difficulty in the work done, given by the difficulty of finding a pattern that was usable on several different types of tables.
The use of this work therefore can be useful in searching for key values within tables, which is still to date a very time-consuming task. This project may represent an initial solution for a more advanced project, through the use of computer vision algorithms adapted to the needs and perhaps in a future use of transformers.
Having considered the good points of this project, let us now consider the weaknesses that were observed during the construction and testing of the project. One of the most critical aspects is that related to the table extraction time, which depends on the length and complexity of the PDFs. On average, the longer a PDF is, the longer the extraction term, so using this work may not be optimal for large PDFs, for which the extraction work may even take up to an hour. An initial optimization was applied to it, however, by extracting only the tables from pages that contain at least one keyword, instead of parsing every table within the PDF. This optimization reduced the extraction time dramatically, reducing it to 1/5 of the time.

The last problem encountered is that of the inability to extract from tables that differ too much from the structure of the recognition method or the automatic methods adopted in this project. As mentioned earlier, the accuracy is not complete, as the extraction of tables is not 100% guaranteed, but still remains high given the use of three different methods to optimize the extraction. Having these strengths and some weaker points, it represents a good starting point for the development of a method that can recognize the highest number of layouts and data structures in table format.
In conclusion, the use of this thesis may be more successful within the next two years, as companies are trying to structure the data and information in the tables in a standard way that is congruent with the established guidelines. The new law will come into effect at the beginning of 2024, so the use of this algorithm can definitely increase, and its accuracy in extraction will also increase significantly, also reducing the extraction time due to the fact of the complexity of the tables.

Chapter 6

Future work

In this last section, I would like to discuss some possible future work related to information extraction from tabular data. This area of research holds tremendous potential for unlocking valuable insights and enabling data-driven decision-making. As we look ahead, there are several exciting avenues to explore and enhance the extraction process.

Firstly, we can focus on improving the extraction of relationships within tables. Given the diverse formats and structures of data, developing more sophisticated algorithms and techniques to accurately identify and extract relationships between table columns would significantly enhance the quality of extracted data.

Secondly, we can delve into the management of table semantics. Tables often contain implicit information and semantic relationships that can be challenging to extract using pattern recognition algorithms alone. By integrating semantic understanding into information extraction techniques, we can achieve a deeper interpretation of table context and extract more meaningful insights.

Additionally, expanding information extraction to support multiple languages is an important endeavor. With the globalization of data applications, the ability to handle multilingual data becomes crucial. Developing algorithms and models that can extract information from tables in different languages would help tackle the challenge of language diversity.

Complex tables present another intriguing area for future work. Tables can become intricate, with merged cells, nested structures, and complex information within individual cells. Developing algorithms capable of handling such complexity and extracting information from these intricate structures would be a significant research direction.

In conclusion, the future of information extraction from tables holds immense potential for advancements. By focusing on improving relationship extraction, incorporating semantic understanding, handling multilingual data, tackling complex table structures, establishing evaluation standards, and considering unstructured text, we can push the boundaries of information extraction and unlock deeper insights from tabular data.

Bibliography

[1] Baeza-Yates, Ricardo and Ribeiro-Neto, Berthier and others, "Modern information retrieval", vol 463,1999.

[2] Protocol, Greenhouse Gas. "Greenhouse gas protocol." Sector Toolsets for Iron and Steel-Guidance Document (2011).

[3] Shigarov, Alexey, Vasiliy Khristyuk, and Andrey Mikhailov. "TabbyXL: Software platform for rule-based spreadsheet data extraction and transformation." SoftwareX 10 (2019): 100270.

[4] Goodman, Joshua. "Parsing algorithms and metrics." arXiv preprint cmp-lg/9605036 (1996).

[5] Friedl, Jeffrey EF. "Mastering regular expressions. " O'Reilly Media, Inc.", 2006.

[6] Freitag, Dayne. "Machine learning for information extraction in informal domains." Machine learning 39 (2000): 169-202.

[7] Caruana, Rich, and Alexandru Niculescu-Mizil. "An empirical comparison of supervised learning algorithms." Proceedings of the 23rd international conference on Machine learning. 2006.

[8] Suthaharan, Shan, and Shan Suthaharan. "Support vector machine." Machine learning models and algorithms for big data classification: thinking with examples for effective learning (2016): 207-235.

[9] Rigatti, Steven J. "Random forest." Journal of Insurance Medicine 47.1 (2017): 31-39.

[10] Wang, Sun-Chong, and Sun-Chong Wang. "Artificial neural network." Interdisciplinary computing in java programming (2003): 81-100.

[11] Likas, Aristidis, Nikos Vlassis, and Jakob J. Verbeek. "The global k-means clustering algorithm." Pattern recognition 36.2 (2003): 451-461.

[12] Khan, Kamran, et al. "DBSCAN: Past, present and future." The fifth international conference on the applications of digital information and web technologies (ICADIWT 2014). IEEE, 2014.

[13] Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet

allocation." Journal of machine Learning research 3.Jan (2003): 993-1022.
[14] Nadkarni, Prakash M., Lucila Ohno-Machado, and Wendy W. Chapman. "Natural language processing: an introduction." Journal of the American Medical Informatics Association 18.5 (2011): 544-551.
[15] Mansouri, Alireza, Lilly Suriani Affendey, and Ali Mamat. "Named entity recognition approaches." International Journal of Computer Science and Network Security 8.2 (2008): 339-344.
[16] Voutilainen, Atro. "Part-of-speech tagging." Vol. 219. The Oxford handbook of computational linguistics, 2003.
[17] Haralick, Robert M., and Linda G. Shapiro. "Image segmentation techniques." Computer vision, graphics, and image processing 29.1 (1985): 100-132.
[18] Ziou, Djemel, and Salvatore Tabbone. "Edge detection techniques-an overview." Pattern Recognition and Image Analysis C/C of Raspoznavaniye Obrazov I Analiz Izobrazhenii 8 (1998): 537-559.
[19] Arbelaez, Pablo, et al. "Contour detection and hierarchical image segmentation." IEEE transactions on pattern analysis and machine intelligence 33.5 (2010): 898-916.
[20] Singh, Sukhpreet. "Optical character recognition techniques: a survey." Journal of emerging Trends in Computing and information Sciences 4.6 (2013).
[21] Ershov, N. M. "Tabula language for description of structured data." Moscow University Computational Mathematics and Cybernetics 33 (2009): 214-218.
[22] "https://camelot-py.readthedocs.io/en/master/"
[23] McKinney, Wes. "pandas: a foundational Python library for data analysis and statistics." Python for high performance and scientific computing 14.9 (2011): 1-9.
[24] Oliphant, Travis E. "Guide to numpy." Vol. 1. USA: Trelgol Publishing, 2006.
[25] Hunter, John D. "Matplotlib: A 2D graphics environment." Computing in science engineering 9.03 (2007): 90-95.
[26] "https://pypi.org/project/PyPDF2/"