UNIVERSITY OF TWENTE.

Information Extraction From Sustainability Reports Using Document AI

by

Bas Vreeman

A thesis submitted to the
Faculty of Electrical Engineering, Mathematics and Computer Science (EEMCS)
in partial fulfilment of the requirements for the degree of

MSc in Business Information Technology

University of Twente

Enschede, Overijssel, The Netherlands

June 2025

ABSTRACT

Document AI aimed at extracting information from multi-page documents has advanced rapidly in recent years. Meanwhile, regulations on sustainability reporting lead to an increase in the length of sustainability reports, making information extraction from those documents is a tedious and time-consuming task. During this study, we assess the usability of the current state of Document AI for automatic information extraction from sustainability reports, to address the gap of the inability of another frequently used method for information extraction, namely LLMs, to capture visual data from the document. We do this by following the Crisp-ML(Q) and Design Science Research methodologies. First, we determine the requirements for an information extraction tool used for sustainability benchmarking together with four sustainability reporting experts. Second, we evaluate the performance of publicly available methods on sustainability reporting data, and third, we aim to adapt the best model to a sustainability reporting setting. We show how quantized low-rank adaption (QLoRA) fine-tuning and hypothetical document embeddings (HyDE) can improve Document AI models in a sustainability reporting setting, by increasing the retrieval performance of a state-of-the-art page-retrieval model, while significantly reducing the required memory. In addition, we show that an automatic fine-tuning pipeline can effectively increase the performance of this retrieval model while reducing the time needed to apply fine-tuning. Furthermore we find that when the question is aligned to the relevant passage in the document, high retrieval accuracy can be obtained, which can significantly reduce the time spent by practitioners on information extraction from those reports. Furthermore, we observe variability in the performance on different tasks. Therefore, we recommend a human-in-the-loop approach when utilizing Document AI in a sustainability benchmarking setting.

AUTHOR'S DECLARATION

I hereby declare that this thesis consists of original work of which I have authored. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize the University of Twente to lend this thesis to other institutions or individuals for the purpose of scholarly research. I further authorize University of Twente to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research. I understand that my thesis will be made electronically available to the public.

Bas Vreeman

ACKNOWLEDGEMENTS

I want to express my gratitude to the following individuals for supporting me in the process of writing this thesis.

Firstly, I want to express my appreciation to my thesis supervisors, Marcos Machado, Patricia Rogetzer, and Rob Bemthuis, for guiding me in my on a biweekly basis. Over the course of the project, they have provided me with timely and consistent feedback, which helped me efficiently finalize my thesis.

I would like to acknowledge my colleagues Taimaz Soltani, Lennart van Efferen, and Maura Willems for allowing me to join this amazing, social, but technically capable Digital Engineering team of EY. My thesis time became one of the most exciting parts of my study due to many team activities and, as the cherry on the cake, a ski trip with the entire FSO Consulting department. Also, I want to thank my counselor Erik De Wit and buddy Iris Oudejans, who gave me a smooth introduction to EY and guided me along the way.

I am also grateful to Mandy Honingh, who helped connect me to many experts in the field of sustainability reporting and invited me to a Sustainable Finance gathering already in the first weeks of my thesis. This supported me in quickly building a network in the field I conducted my thesis and in gaining an understanding of the current state of automatic information extraction in the business related to sustainable finance, as well as the processes in the company that could benefit from the tool I was going to develop.

I especially want to express my gratitude to one of the experts I was introduced to by Mandy, namely Tatiana Fetisova. Her intermediate feedback, deeper insights into sustainability reporting benchmarking, and help in setting up the expert evaluation moments were invaluable for aligning my thesis with a practical problem in the business.

Special thanks also go to my friends and family. Firstly, my parents, for supporting me during my studies, helping me move to Amsterdam, and visiting me from time to time — which I greatly valued, especially during the busier phases of writing this thesis. Furthermore, I want to thank my friend Wolf for introducing me to my temporary residence in Amsterdam. This facilitated a smooth transition to a new city that greatly enhanced the overall quality of my internship at EY.

CONTENTS

List of Figures

List of Tables

ABBREVIATIONS

ANLS Averaged Normalized Levenshtein Similarity.

API Application Programming Interface.

BTM Bi-term Topic Model.

CEM Constrained Exact Match.

CoI Contextual Late Interaction.

CRISP-DM A process model for data mining.

CRISP-ML(Q) Cross-Industry Standard Process for Machine Learning.

CSRD Corporate Sustainability Reporting Directive.

DSR Design Science Research.

ECE Expected Calibration Error.

EM Exact Match.

ESG Environmental, social, and governance issues.

ESRS European Sustainability Reporting Standards.

G-Acc Generalized Accuracy.

GPT Generative Pre-trained Transformer.

GPU Graphics Processing Unit.

GRI Global Reporting Initiative.

HyDE Hypothetical Document Embedding prompting.

IE Information Extraction.

IE Criteria Inclusion and Exclusion Criteria.

KIE Key Information Extraction.

LLM Large Language Model.

LVLMs Large Visual-Language Models.

MAE Mean Absolute Error.

MIM Masked Image Modeling.

ML Machine Learning.

MLLMs Multimodal Large Language Models.

MLM Masked Language Modeling.

MRM Masked Region Modeling.

MRR Mean Reciprocal Rank, a metric to evaluate page-retrieval performance.

NFRD Non-Financial Reporting Directive.

OCR Optical Character Recognition.

PEFT Parameter-Efficient Fine-Tuning.

QA Question Answering.

QA-pipeline Question Answering pipeline.

QLoRA A combination of quantization and LoRA.

RAG Retrieval-Augmented Generation.

RoPE Rotary Position Embedding.

SLR Systematic Literature Review.

SoTA State-of-the-art.

TCFD Task Force on Climate-Related Financial Disclosures.

VLM Vision-Language Model.

VQA Visual Question Answering.

VRDs Visually Rich Documents.

VRDU Visually Rich Document Understanding.

1

INTRODUCTION

There has been a growing recognition of the importance of sustainability for the financial success of a company. In 1994, John Elkington came up with the definition of 'The Triple Bottom Line', which consisted of three pillars, namely people, planet, and prosperity [5]. Hereafter, various other initiatives worldwide were instantiated, such as the GRI (Global Reporting Initiative), TCFD (Task Force on Climate-Related Financial Disclosures) and NFRD (Non-Financial Reporting Directive), to give guidance to the reporting of the nonexclusively financial impacts and considerations by an organization.

Recently, the European Union released the CSRD (Corporate Sustainability Reporting Directive), which applies to all companies which fall under the NFRD (Non-Financial Reporting Directive) and later also to other organizations. The CSRD regulation should solve current issues related to the comparability of sustainability disclosures by organizations falling under the NFRD. Moreover, increased disclosure to investors should motivate organizations to increase their sustainability performance [6], contributing to the European Green Deal Strategy to make Europe Carbon Neutral in 2050 [7].

The CSRD has established clear assurance requirements related to the disclosed sustainability information. This increases the reliability of this information and increases the value of the information disclosed in the sustainability reports. From 2024, companies which fell under the NFRD, among which are financial institutions (e.g., credit managers and insurance companies), are required to perform assurance. Currently, national laws implement the assurance requirements of the CSRD. From 2026, European-wide assurance laws apply [8].

An important part of the CSRD is the European Sustainability Reporting Standards (ESRS), which provide more guidance on how the full range of data points on environmental, social, and governance issues (ESG) should be reported. According to the European Union, the standards cover the full range of ESG issues, making sustainability reports more standardized and

credible, which is important for investors and supervisory agencies [9].

Research background

Especially in the financial sector, ESG has become an important factor in decision making processes [9]. Banks, supervisory institutions, and insurance companies use ESG reports to measure the level of risk of an individual organization or broader sector. This process is supported by various studies that found a link between ESG and future financial performance [10]. This led banks to include ESG in the calculation of the cost of capital of their borrowers, implying that weaker performance on ESG led to the receiving of a higher risk premium. A survey by Gaganis et al. [10], showed that more than 50% of investors evaluate the performance of ESG.

Over the years, the relationship between the E, S, or G pillars and financial performance is not considered equal. According to a survey by EY [11], environmental disclosures are the most important for credit risk assessors (banks). The risk of fines, damage costs, or large clean-up costs for companies increases the overall risk of investment in the company and thus contributes substantially to the cost of capital [10]. Also, social performance is important for credit risk assessments, as a high social performance showcases a better operating ability and good intent towards creditors. Moreover, it shows potential outside pressure that increases the likelihood of good company management [12]. Lastly, despite being considered less important than the other two pillars, the governance pillar cannot be discredited. Studies in the literature review by [10] showed that companies that are transparent (i.e., low information symmetry), show high board diversity, show strong internal control by being defended against takeovers, for example, or are located in a place with strong legal institutions receive a lower risk premium on their loans.

In addition to its importance for assessing the cost of capital of banks' borrowers, ESG also plays an important role in assessing the risk of default of a bank itself. This makes ESG also valuable for supervisory institutions. The literature has shown that, especially the social pillar, appears to have a high correlation with the risk of default of banks, which could be explained by the governance of banks being highly regulated and the environment being largely dependent on the investments itself of a bank [13]. Similarly, supervisors might be interested in the sustainability performance of insurers, considering a positive correlation with financial stability there as well [13]. Here, again, there is a distinction between the importance of the pillars. The influence of the governance pillar is again diminished. However, the environmental pillar now plays a role, which could be explained by the dependence of insurance costs on events related to climate.

Although ESG has shown to become an important factor in the operational processes of financial institutions, the decision-making process is hindered by the difficulty of assessing the ESG performance of organizations. Lack of structure in the disclosure format, lack of validation, and lengthy documents make the analysis of an organization's sustainability performance a tedious task. However, having a more standardized framework could also be problematic, as the diversity of organizations hinders the applicability of a one-size-fits-all solution and reduces the meaningfulness of disclosures [14].

Because comparing sustainability disclosures is challenging, many financial institutions utilize rating agencies to obtain an indication of a company's ESG-performance, such as Moody's [1] or Sustainalytics [2]. However, interpreting those ratings is also challenging as they diverge despite rating the same company, which can reduce their usefulness and credibility. This is mainly attributable to the difference in the measurement method [15].

To mitigate the effect of divergent ratings, companies are weighing multiple ratings or using certain ratings to measure certain characteristics, this requires extensive experimentation, after which it is still uncertain whether the ESG performance is being assessed accurately. This could have led 45 out of 50 organizations to do at least part of the assessment of ESG performance themselves, instead of fully relying on ESG rating agencies, as shown by a survey by SquareWell Partners [16]. Most of the respondents indicated using a combination of rating agencies and their proprietary analysis to assess ESG performance [17].

Thus, the recent and ongoing implementation of the CSRD should lead to more comparable and trustworthy ESG data in sustainability reports. However, ratings by agencies diverge, leading to a preference for proprietary analysis of ESG disclosures, and therefore, it is likely that most organizations at least partially do a proprietary analysis, although this will lead to tedious and time-consuming tasks [18].

Alternatively, a fruitful approach could be to leverage technology, and more specifically, Large Language Models (LLMs), which have enabled practitioners with no technical background to leverage automatic text processing, due to rapid advancements in recent years. LLMs are known for their ability to process large amounts of text and are capable of reliably extracting explicit, and sometimes even implicit, information through reasoning [19–21]. Several studies tested the ability of LLMs to extract sustainability reports. Gomes Ziegler [22] used a combination of GPT-4 and GPT-4-Vision and obtained an extraction accuracy of 77.6%.

Although Gomes Ziegler [22] obtained a promising accuracy by using an LLM only, the study also highlighted that using a combination of vision and text, a pipeline that relies only on one modality performed even better. This is due to certain shortcomings of LLMs. First, LLMs appear to have difficulties with numerical data, magnitudes [23, 24] and missing data in tables [25]. Second, considering the frequency of numerical and tabular data in sustainability reports, combined with their visual richness, LLMs alone appear to be unable to obtain an acceptable accuracy for the extraction of information from sustainability reports, when the required accuracy is greater than 99% [26, 27], indicating the need for more advanced extraction methods.

Research motivation and objectives

Visually Rich Document Understanding (VRDU), also referred to as Document AI, is an AI research direction that focuses on the understanding of and extraction from documents that contain certain difficulties that could hinder the understanding of AI documents, such as more

[1] https://events.moodys.com/esg-scores
[2] https://www.sustainalytics.com/

complex templates (e.g., two columns), changing templates, or hierarchical entities [28]. Although multimodal (i.e., image and text) methods showed a significant increase in performance, models that additionally consider layout obtain the best performance [29].

To the best of our knowledge, no previous studies assessed the current performance of state-of-the-art Document AI methods (SoTA) for IE (information extraction) from sustainability reports. The combination of: (1) the implementation of the CSRD that leads to accurate disclosures of sustainability data, and (2) the preference for proprietary sustainability report analysis makes that investigating the current state of document extraction methods could reap significant benefits for practitioners and researchers.

Previously, various Document AI methods for QA were created. Therefore, during our study, we first select the publicly available Document AI methods with the highest scores on the DocVQA [30], MP-DocVQA [31] and MM-LongBenchDoc [32] benchmarks. Second, we apply those methods to sustainability reports to find the best performing method, to, at last, maximize the performance of the best method by applying Low-Rank Adaption (LoRA) fine-tuning and hypothetical document embedding prompting (HyDE).

Therefore, this thesis has the following goals:

- Design a tool/proof-of-concept utilizing Document AI methods that adheres to business requirements for (supporting) IE from sustainability reports.
- Assess whether the current Document AI methods meet the thresholds for full or partial automatic information extraction from sustainability reports
- Enabling accessible, accurate and efficient analysis of sustainability reports

The above-mentioned research goals serve as input for the following research questions:

Research Questions

1. What is the accuracy threshold that is required for automatic extraction from sustainability reports using Document AI?
2. Which Document AI models do currently perform best on selected representative benchmarks for QA on multi-page visually rich documents (i.e., MP-DocVQA, MM-LongBenchDoc)?
3. How are Document AI models trained, built, and deployed?
4. What are they key architectural aspects of Document AI methods that obtain high retrieval and QA performance on the selected representative benchmarks?
5. How can existing Document AI methods be used or adapted so that they comply with the business requirements?

2

LITERATURE REVIEW

Among the different methods for IE from documents, recently, two main approaches stand out in the literature: (1) LLMs (2) Document AI. Those two methods are not separated, as LLMs form the basis of Document AI methods. This review of the literature investigates the applicability of these two methods for IE from sustainability reports. First, the applicability of LLMs was assessed using an SLR, to collect evidence on whether the current state of LLMs is sufficient for information extraction from sustainability reports. After concluding that LLM alone is unlikely to provide sufficient performance, a complementary literature review phase was added, consisting of a narrative literature review that assesses the current state of document AI, to select the best models for QA from sustainability reports.

2.1. LLMS FOR IE FROM SUSTAINABILITY REPORTS

2.1.1. INTRODUCTION

In the context of document extraction, recent research in NLP has focused on exploring the applicability of pre-trained language models (PLMs) and large language models (LLMs). PLMs, which were deployed for the first time in 2017, are trained on a large unlabeled dataset to understand the vocabulary, semantics, and logic of a text. In the subsequent phase, the PLM is fine-tuned on a smaller dataset, which has been shown to significantly improve the performance of the model [33]. The most used PLMs are Bi-directional Encoder Representations from Transformer (BERT), or GPT-2. Later, in 2020, LLMs were first created. Contrary to PLMs, LLMs only require the pre-training phase, in which they are trained on an enormous set of data. The LLMs are designed with the goal of understanding human language. Moreover, they often outperform PLMs. Frequently used LLMs are GPT-4 and Llama [33].

Considering the need for better analysis options of sustainability reports, and the recent superior performance of LLMs, we conduct a systematic literature review (SLR) to observe the

current performance of LLMs when used for IE in various research areas, to determine their potential for IE from sustainability reports. This results in research opportunities related to automatic information extraction from sustainability reports, and in best practices, which can be used by organizations in various industries that want to start using LLMs for information extraction.

To our knowledge, no SLR evaluation of the current state of LLMs for information extraction had been carried out, after their rapid improvement in 2022. This SLR investigates the extent to which LLMs can be applied for information extraction, and thus the research question (RQ) is: "*To what extent can LLMs be used to extract information from sustainability reports?*".

In conducting this review, three themes have formed the basis for synthesizing the information. Those themes are the advantages and disadvantages of using LLMs for information extraction, requirements, and validation methods. Those themes led to the following three sub-research questions:

1. What are the advantages and disadvantages of using LLMs in extracting information from free-text reports?
2. What is required when deploying an LLM for automatic information extraction in practice?
3. How can an LLM's performance on information extraction be validated?

The first question aims to examine what an LLM can and cannot do when used for information extraction, addressing a significant part of the main research question. However, a model's disadvantages could well be mitigated by auxiliary measures. Moreover, the degree of usability of LLMs can be seen as a cost-benefit consideration, where the requirements represent the cost. Therefore, the second question explores the requirements in the literature. Lastly, when precise information extraction is required, it is important to be able to evaluate the performance of the LLM on this task. Therefore, the last question investigates the available evaluation methods. Overall, these questions help practitioners to make considerations on whether and how they want to employ an LLM for information extraction. Moreover, researchers in the area of LLMs could be incentivized to explore new solutions based on the outcomes of this SLR.

2.1.2. SLR METHODOLOGY

This SLR follows the methodology designed by Snyder [34]. By following this methodology, which describes several steps that should be followed during the literature review, we obtain a comprehensive overview of previous studies in the field. This results in an overview, potentially conclusive answers, and future research directions. The methodology entails six steps, which are (1) establishing a protocol, (2) searching literature, (3) appraisal of literature, (4) synthesis of literature, (5) analysis of literature and finally the (6) reporting of the literature.

Due to a lack of papers in the field of ESG data extraction and finance, we made an adaptation to the SLR methodology by Snyder [34], after the evaluation phase. Initially, we only extracted

papers from Scopus [1]. However, after the appraisal phase we observed a lack of papers from the desired research area of financial reports, which led us to decide to utilize other databases than Scopus to include three papers related to the applications of LLMs for ESG- or finance-data extraction.

SEARCH PHASE

The search phase encompasses the creation of search queries. The research questions form the input for this. Initially, the focus was to observe how LLMs had already been used to extract information from financial documents. However, this yielded only a few papers, and thus we broaden the scope to also include other kinds of professional reports. The practice of starting with a pilot search to optimize the search query adheres to the methodology of Snyder [34].

The queries used were:

1. *("Large Language Model*" OR "LLM*") AND ("information" OR "data") AND ("extraction" OR "retrieval" OR "filter*") AND ("technical" OR "financial") AND '("report*" OR "statement*"*
2. *("LLM*" OR "Large Language Model" OR "GPT*") AND ("deploy*" OR "implement*" OR "MLDevOps" OR "DevOps" OR "MLOps") AND ("information" OR "data" OR "knowledge") AND ("extract*" OR "retriev*" OR "filter*") AND ("requirement*" OR "prerequisite*")*
3. *"Large Language Model*" OR "LLM*" AND ("information" OR "data" OR "knowledge") AND ("extraction" OR "retrieval" OR "filter*") AND "performance metric*" OR "performance measure*" OR "performance validate*" OR "performance evaluate*"*

The queries only considered the title, abstract and keywords of the papers. The three queries were combined into one query resulting in a total of 113 papers, in the Scopus database, which only stores peer-reviewed papers and articles (See Figure 2.1). The search phase was finalized on the 27th of November 2024, thus papers published after this date are excluded. We decided to only consult Scopus, because it is one of the largest databases, covering a wide variety of research disciplines. Also, the Scopus search engine allows for advanced querying, which supports the chosen SLR approach.

APPRAISAL PHASE

During the appraisal phase, a decision is made whether papers should be included for synthesis. This is done based on the inclusion and exclusion criteria (IE criteria, hereafter) presented in Table 2.1. In total, five criteria were used for the selection of the literature. Because the goal of this literature review is to get both insight into the application of LLMs for information extraction, and into the requirements for deploying such model into a business process, the IE criteria were created as follows. When information extraction using LLMs is not part of the study, and the study is not considering requirements for deployment, the study is excluded. Although ideally, we would only include papers that describe implementation requirements for

[1] https://www.scopus.com/home.uri

the deployment of LLMs for information extraction, we only find a few papers that discuss both topics. Therefore, we make an exemption for papers which include requirements for deployments only.

Also, we filter out papers which are published before 2022, to exclude papers utilizing LLMs before their explosive increase in performance. The reasoning behind this, is that the results of those papers could be considered outdated, while the focus of this SLR is to grasp the current state of using LLMs for information extraction, so that practitioners know whether they should opt for this method of extraction, and so that researcher know what future research directions are, in this rapidly evolving field. Therefore, outdated information does not provide valuable results. In addition, we excluded collections of conference proceedings from the selected literature to maintain objectivity in obtaining the literature in the search phase, which could be reduced when we have to assume which study made the collection to be included in the results to the queries.

After creating the IE criteria, the selection of papers is performed in two rounds. The first round, which started with 113 papers, focused solely on the titles and abstracts of the papers. Here, all studies which evaluated LLMs utilized for information processing from documents are included. Thus, papers on topics such as question-answering systems or Retrieval-Augmented Generation (RAG) are included. This selection round results in 41 papers. After the first round, the goal of the research was refined to solely focus on information extraction of papers, which leads to the exclusion of another 21 papers, resulting in a final selection of 23 papers out of 113 for the synthesis phase. Part of these 23 papers, are also the three papers which are included based on the author's judgment (See Figure 2.1). In Appendix A1, the final selection of papers is listed.

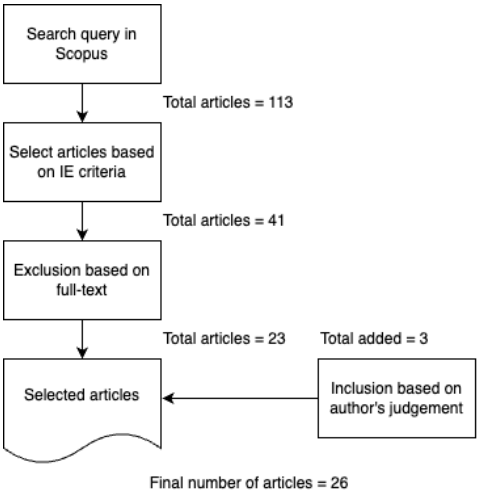

Figure 2.1: Illustration of the article selection process. IE = information extraction

Table 2.1: Summary of article selection criteria.

Criteria	Decision
Paper is about information extraction using LLM	Inclusion
Paper is about requirements of deploying in LLM in business practice	Inclusion
Paper published between 2022-2024	Inclusion
Paper's scope other than information retrieval/extraction, e.g., summarization	Exclusion
Results which are conference proceedings	Exclusion

SYNTHESIS PHASE

During the synthesis phase, all 23 papers are thoroughly summarized, while focusing on the advantages, implementation requirements, disadvantages and validation methods, Appendix A1 for the results of this phase. Moreover, inductive coding enables sense-making of shared visions across various papers, which is a common practice in the qualitative part of an SLR [34, 35]. Also, emerging themes from the literature are analyzed during this phase.

2.1.3. MAIN FINDINGS

This section describes the main findings, resulting from the synthesis of the selected literature after applying IE criteria. The results are presented following a thematical approach, as described by Varsha et al. [36]. First, this section gives an overview of the conducted meta-analysis (Section 2.1.3). Second, the results of the qualitative analysis are presented, where the themes are: kind of LLMs (Section 2.1.3), performance enhancing techniques (Section 2.1.3), applications of LLMs (Section 2.1.3), advantages and disadvantages (Section 2.1.3), implementation requirements (Section 2.1.3) and validation methods (Section 2.1.3).

META-ANALYSIS / DESCRIPTIVE ANALYSIS

We use a meta-analysis to get an insight in the year of publication of the papers, the research field, and in the number of citations. This to get an insight into how novel the research area is, the accessibility of the technique to non-computer-science related fields, and the impact of this research area. Finally, we also perform a keyword analysis to get insight in the core themes in the selected literature.

Firstly, we analyze the year of publication of the papers. This gives insight in certain trends regarding the use of LLMs for information extraction. Although we only observe publications over three years, we found a clear trend in the selected literature between 2022 and 2024. The number of papers published in 2022, 2023 and 2024 were namely 0, 6 and 20 respectively, see Figure 2.2. This trend could show that over time, more research areas start to experiment with the LLMs, which recently have become widely accessible to the wider public in 2022, with the

release of chatgpt.com[2]. Therefore, also the proportions of research areas in the selected literature are compared per year. Here, we identify no clear trend. However, we observe that the proportion of studies in the field of computer science, compared to other research areas remains stable. The majority of the papers are published in the area of computer science (39%), engineering (11%), and mathematics (10%).

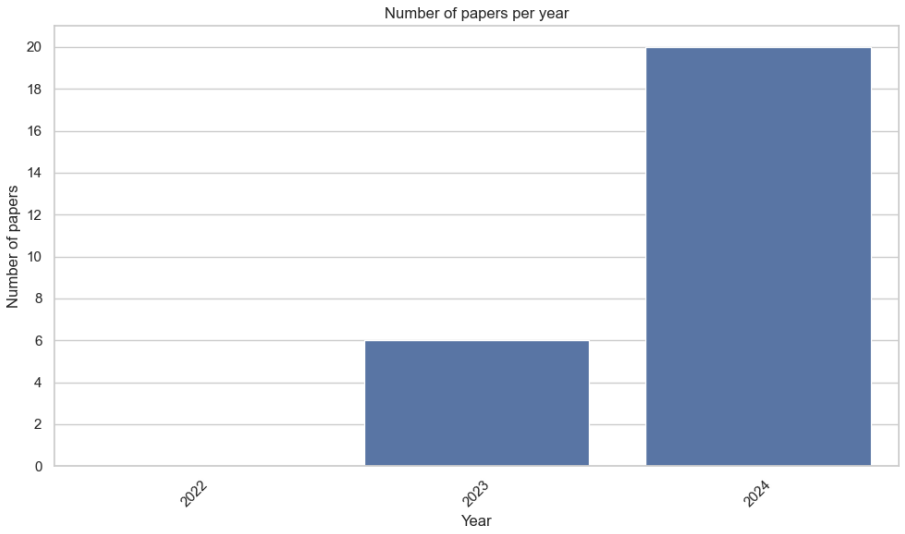

Figure 2.2: Publications per year in selected literature

Secondly, the number of citations were analyzed, as they are sometimes seen as a metric of impact of the paper [37]. Therefore, Table 2.2 provides an overview of the more and less cited papers. The paper by [20] obtains by far the most citations, namely 93. This paper discusses the application of information extraction in the medical domain. The second most cited papers obtained 10 citations, one in the medical and one in the financial domain [38, 39]. The paper by Kannan and Seki [38] used LLMs to extract ESG-evidence. The fact that this paper has already got ten citations while it was published six months ago shows significant interest in the topic of information extraction related to LLMs. Also the three papers with five citations are all applying NLP methods, including LLMs to extract sustainability information from documents. Thus, this shows that research is ongoing in this research field, considering the recent publication dates of the papers. Also, this implies that it could be valuable for practitioners to be on the lookout for breakthroughs, as new developments in this research area are to be expected. Papers with two or less citations were left out for this meta-analysis.

[2]https://chatgpt.com/

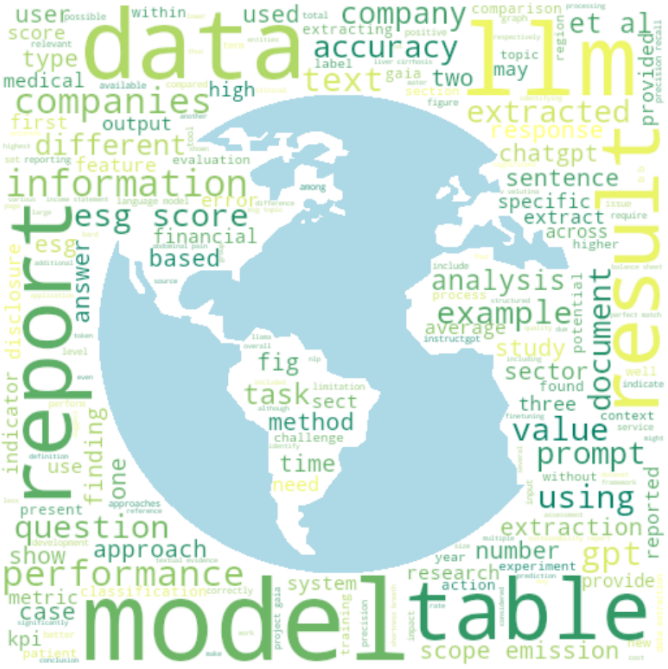

Figure 2.3: Wordcloud. The size of the words is representative for its frequency in the results, discussion and conclusion of the selected literature

Finally, we utilize a keyword analysis to get insight in the core themes. This could give insight in the most relevant topics around information extraction using LLMs. The resulting keywords are presented in Figure 2.3. The wordcloud is based on the results, discussion and conclusion sections of the papers. We observe six emerging themes from the wordcloud, namely: “model”, “table”, “result”, “LLM”, “data”, and “report”. The frequent occurrence of table is likely resulting from the fact that reports contain tables to present the extraction results. The other keywords understate that the search queries resulted in relevant studies, since those themes align with the goal here is to find studies which extract data using LLMs from reports. Interestingly, accuracy has a similar frequency in the documents as performance, showing that it is often used as a metric when evaluation performance. Also, data is mentioned frequently, which could indicate its importance for machine learning processes, both for training and evaluation.

Table 2.2: Citations from the Literature Review. Papers with less than 5 citations are excluded

Authors	**Citations**
[20]	93
[38], [39]	10
[40], [19], [41]	7
[42], [43], [26]	5

KINDS OF (L)LMS

Here, we summarize the methods used in the literature to extract information from text. When a study compares the performance of an LLM, they almost always compare this to another LLM. In one study by Maibaum et al. [43], the comparison also included a comparison to pre-LLM

methods. This section lays out the different models found in the selected studies. At first, the pre-generative methods are explained, secondly, the generative methods are explained.

Pre-LLM Models The study by Maibaum et al. [43] includes three pre-LLM methods, which are the methods of topic models, dictionaries, and vector embeddings. This section explains them in order from least performing to best performing. What was found in the study by Maibaum et al. [43] was that even the best-performing method performs worse at information extraction than the least-performing LLM model.

The worst-performing method at the task of information extraction was that of word embeddings. With this method, the meaning of words is deduced based on the location of their embedding as a vector in a vector space. Using those word embeddings, it is possible to get words that are related to a topic, but also a topic that is related to words. This allows for a degree of automatic topic filtering. The words that are in the vector space could form the words to analyze in a document when measuring a construct.

Another method tested is that of dictionaries. Dictionaries vary greatly in performance, which could indicate their dependency on the completeness and fit of the dictionary. Contrary to word embeddings, dictionaries are manually created, which therefore require a high degree of expert knowledge and labor. Once established, a dictionary can be used by counting the words in that dictionary to measure the degree of presence of the construct of interest [43].

Lastly, the best-performing method was that of topic models. Topic models assume that the word distribution of a text depends on the topics in a text. Topic modeling can leverage various algorithms to obtain topics from a collection of words, or a collection of words from topics. The tested algorithms in the study by Maibaum et al. [43] are the supervised latent Dirichlet allocation (sLDA) and the unsupervised bi-term topic model (BTM). The latter was the best performing of the three pre-LLM methods, which is likely attributable to the short text being preferred, when working with BTMs.

LLMs As the previous section already highlighted, LLMs outperform all the pre-LLM methods in the study by Maibaum et al. [43]. This might explain why there are no other studies in the selected literature which examine the performance of an LLM compared to pre-LLM methods. Instead, comparisons are often made between LLMs, to determine whether the proposed method is outperforming the state-of-the-art. During this studies, the models of interest are most of the time a generative pre-trained transformer (GPT) [19–21, 24, 25, 40, 41, 44–47], or a BERT [23, 24, 38, 43] variant. There were five studies where another model is studied, those models are WizardLM [18], Llama2 [24], ChatGLM2 [48], Vicuna [39] and Baichuan [49]. Note, more LLMs were analyzed in the papers, however, those were used as reference models and not as the main model of interest. This section explains and compares the characteristics of the frequently used BERT- and GPT-models used in the literature.

BERT-models are described as bidirectional, because they have the ability to analyze the con-

text of a word in a sentence all at once, instead of from left-to-right or right-to-left. This results in that BERT is better able to determine what a word means in the context. BERT was trained using a large corpus where certain words were masked, this technique is called Masked Language Modeling (MLM). By MLM, BERT has obtained an understanding of the context, which can be used to let BERT perform certain tasks, such as sentence completion, but also information extraction.

The GPTs were most often the subject of analysis in the selected studies. GPTs excel in generative tasks, where each next word is predicted based on its preceding words. Contrary to BERT, GPT is not able to process the entire context of a word at once, though the model excels at predicting the next word based on the preceding context. Despite this reduced context reading ability, GPTs have been shown to produce highly relevant results, often without requiring any fine-tuning [23, 40, 50]. Especially the larger models maintain their performance level, when fine-tuning is not performed. Several studies compare GPT and BERT models. It is not uncommon that GPT4 is able to outperform a fine-tuned BERT. Though, a fine-tuned BERT tends to outperform a smaller GPT model [41, 43].

PERFORMANCE ENHANCING TECHNIQUES

As the previous section already mentioned, the performance of a model can be improved by fine-tuning. In addition, retrieval-augmented generation (RAG) and prompt engineering techniques could improve the performance of the model. This section explains what those techniques entail.

Fine-tuning is used by various studies in the selected literature [18, 38, 42, 43, 47, 51]. Mostly, the goal of fine-tuning is to add knowledge; however, it could also be the goal to learn to format the output in a certain way [51]. To be able to fine-tune, an annotated dataset is required. This dataset consists of so-called prompt-completion pairs. The prompt is the query to the model, the completion is the desired response of the model.

Once a dataset is obtained, the set is usually subdivided into a training, validation, and testing set. All the training data is used to update the model parameters based on the prompt-completion pairs. Each time all the training data is used, a so called epoch has been fulfilled. After each epoch, the intermediate performance of the fine-tuned model is evaluated using the validation dataset. Once all epochs have passed, the test dataset is used to evaluate how the final model performs [51].

For fine-tuning, it is necessary that the model parameters can be adjusted, either directly or by using a program. In the study by Sonnenburg et al. [51], Microsoft Azure OpenAI's Services, using GPT-3, allowed for the fine-tuning of several base models. Alternatively, if the provider allows it, the model could be downloaded to fine-tune it locally. A common parameter used in the OpenAI models is the temperature, which is a number which determines the degree of randomness in the model response. To succesfully perform information extraction using an LLM, a low temperature close or equal to zero is preferred [27].

Besides fine-tuning, RAG is another technique employed to increase the performance of the model [18, 45, 48, 50]. Various studies showed that LLMs tend to provide general answers. To counteract this trait, supplying the LLM with relevant context (in free-text-format) could be utilized to increase the generation of specific, authentic, and diverse answers. This is done as follows. Firstly, the documents are subdivided into segments. To ensure to maintain continuous content, a certain degree of overlap between segments is selected. Next, the segments are embedded using an embedding model and stored in a vector database. Similarly, a user question is embedded using the same embedding model and is used to query the vector database to obtain relevant passages of text, by retrieving information which is stored nearby in the vector space. Once those passages are obtained, they are added to the user prompt as context to the LLM [48].

Lastly, this section discusses the technique of prompt engineering, which is used by various studies in the selected literature for improving accuracy [1, 19, 23, 27, 41]. This technique gained attention with the advent of GPT-models, which are often used in a chatbot setting [50]. Below, the common prompting techniques in the selected literature are discussed, namely that of: in-context-learning prompts, chain-of-thought prompts and definition prompts. The examples provided in Figure 2.4, 2.5 and 2.6 are inspired by the supplementary material by Wiest et al. [1], which presents prompts designed for extracting information about ascites from medical reports.

In-context learning implies that no previous fine-tuning was done, and the model is being used as is, with the same parameters. In-context learning can be done with a zero-shot learning approach, or a one- to few-shot learning approach. Zero-shot learning implies that no examples are given to the model on how to perform their task [1, 40]. One- or few-shot learning implies that an example is given to the model, as a prompt-completion pair, similarly to the fine-tune examples for BERT [40], however, now it is directly inserted in the prompt, and thus no technical knowledge is required for this task (See Figure 2.4).

> **Prompt:** "Is ascites present at or before patient admission?"
> **Example Response:** "{ascites: present, before_admission: true}"
> **Completion:** "{ascites: present, before_admission: true}"

Figure 2.4: One-shot learning prompt and response [1]

Chain-of-thought prompting could also benefit the performance of the model. When performing this technique, a short description is added to the prompt, which explains how the model should respond to the prompt (See Figure 2.5).

Prompt: "Is ascites present at admission? Provide an excerpt from the text, then answer the question."
Completion: "Excerpt: 'HCV cirrhosis c/b ascites.' Since the report states that the patient has cirrhosis complicated by ascites, it confirms that ascites is present at admission."

Figure 2.5: Chain-of-thought prompt with explicit reasoning [1]

Also, definition prompting could benefit the model, by providing useful definitions are added to the prompt, so that the model consists of sufficient knowledge to answer the question [1]. An example is presented in Figure 2.6. Usually, a large list of definitions are provided.

Lastly, it should be mentioned that prompts usually contain complementary information about the exact role the LLM should fulfill. For example, the prompts by Wiest et al. [1] started with something similar to: "You are programmed to be a medical assistant, you will receive reports of ..., etc.".

Questions: "Is ascites present at or before patient admission? Is abdominal pain present at or before admission?"
Definitions: "Ascites refers to the accumulation of fluid in the peritoneal cavity. Abdominal pain refers to discomfort in the abdominal area."
Completion: "Both ascites and abdominal pain were documented as present at the time of patient admission, based on medical records and patient symptoms."

Figure 2.6: Definition prompt example. Usually, many definitions can be given [1]

APPLICATIONS OF LLMS

In most of the selected literature, the tasks are named entity recognition (NER) tasks, which is a name for the task to extract certain parts of the text. Those parts can be either numerical [22, 24] or textual, and can be either from free text or from a table or a short sentence [19, 21, 23, 25, 39–41, 43, 44, 48].

Several studies take it a step further, by assigning the task of making deductions from a piece of text [1, 41]. The results vary, where Labbe et al. [41] conclude that their model (ChatGPT2) is not able to extract, Wiest et al. [1] leverages Llama 2 successfully for the task of quantitative data extraction to identify liver disease symptoms. They do this with an impressing accuracy of 90%.

In other studies, the capabilities of LLMs to handle free text are exploited to go from unstructured text to a structured database. Also in sustainability reporting there is a potential for obtaining structured data from unstructured data using LLMs, as shown by Dimmelmeier et al.

[45]. However, they observe several key challenges which require to be mitigated before this application of LLMs for information extraction can be done successfully. The most urgent challenges are the extraction of data from tables and graphs in the documents.

Methods which could be leveraged to obtain structured information from unstructured data sources is that of knowledge graphs and ontologies [18, 44]. Knowledge graphs are a presentation of data from multiple sources in a structured format, enabling semantic search, explainability, and information retrieval. Ontologies are the basis for a knowledge graph, by capturing the entities and their dependencies of a certain domain. The study by Usmanova and Usbeck [44] extends an existing ontology called OntoSustain, which was developed to capture the semantics around sustainability reporting. Using OntoSustain, the study is able to use GPT-4 to extract data from a sustainability report. The performance is dependent on the extracted topic of interest Usmanova and Usbeck [44]. Bronzini et al. [18] takes a different approach, and leverages existing ontologies to extract data in a document. Instead of using a knowledge graph which already exists, they generate a graph -including its shape- themselves using the LLM, to subsequently use it for analysis. The quality of the triple generation was not assessed in this study.

Lastly, LLMs can be used to create vector embeddings of text [43, 45]. By embedding a word or a part of text within the context of a document, or multiple documents, its semantic meaning and relation to those documents can be inferred. RAG utilizes this technique, which is discussed in further detail in Section 2.1.3.

ADVANTAGES AND DISADVANTAGES OF LLMS FOR INFORMATION EXTRACTION

The major reason for investigating the use of LLMs for information extraction from documents, is that they demonstrate to have accurate performance. The accuracy of information extraction on exclusive textual input data by GPT-4 often revolves around 95% [20, 21, 52], and is obtained by calculating the ratio of correctly extracted datapoints to the total number of datapoint disclosures (See Table 2.3 for a comparison of accuracies which were calculated in the studies for GPT-4). This high accuracy, leads to GPT-4 to outperform other models on the task of information extraction from free-text[20, 21, 40, 43]. Similar performance can be acquired by locally runnable models, such as Llama2-70b [46], and Vicuna [39].

Table 2.3: Accuracy of GPT-4, calculated as the ratio of correctly extracted data points to the total number of data point disclosures.

Study	Input Format	Accuracy
Fink et al. [20]	text only	96%
Dagli et al. [52]	text only	94.8%
Castro et al. [21]	text only	90-100%
Zou et al. [26]	multi-modal	76.9%
Hub [27]	multi-modal	79.3%
Balsiger et al. [25]	tables	83.10%

Another advantage is the absence of the need to format the input document in a certain we to

extract information from it. LLMs are capable to obtain the required information from large chunks of text, with potentially higher accuracy than humans. Since information extraction from large documents is a labor-intensive task, humans are prone to fatigue, and to make mistakes in their extractions. This makes automatic extraction methods an interesting alternative [19, 44].

Also, LLMs have been shown to store a degree of domain knowledge. GPT-4 currently tends to have the best off-the-shelf domain knowledge, and for various tasks, no fine-tuning is needed [21, 25, 40]. This leads to the fact that, even though a model with less parameters such as GPT-3.5 had been fine-tuned to better understand the domain language, it can still be outperformed by GPT-4 [43]. However, in various cases fine-tuning does show to be effective, as was shown by the study of Li et al. [47], where a fine-tuned GPT-3.5 outperformed GPT-4. Especially when working with non-generative models, fine-tuning could increase the performance of the model [43].

Various papers refer to the technique of prompt engineering as a technique where the prompt is carefully designed to obtain the best output of the LLM [1, 19, 20, 23]. The literature shows that the performance of the model can be significantly improved by several techniques, such as adding context (e.g., a piece of text), one-shot or few-shot learning. The latter two imply that within the prompt, one or more examples are given on how the LLM should perform their task [18, 39], as presented in Figure 2.4. Especially the performance of larger models seems to benefit more from prompt-engineering than from fine-tuning, although a combination of both is likely to obtain the best results [43].

Another technique which could equip LLMs with the necessary knowledge to accurately extract information is RAG [45]. This is a technique that queries data based on the prompt of the user, to accurately perform the task. Mostly, this is done using a semantic embedding of the user prompts, which is a representation in a vector space of the prompt. In this way, documents which resemble the user prompt, are close to the query in the vector space, and can be yielded and used to support the task of the model [18, 45].

Also, various studies recognize the absence of the need of retraining as a benefit [1, 20]. Contrary to previous machine learning applications, modifications to the process of the model can now be made by simply modifying the prompt. When the model demonstrates a lack of capability or knowledge, an improvement can be made by changing the prompt, including a few examples (few-shot learning), or adding an (updated) document, rather than retraining the model.

Moreover, many use cases of LLMs exploit the availability of those models via OpenAI API (Application Programming Interface), which enables programmers to add the functionality of OpenAI to their software programs. This eliminates the need to own expensive hardware [23], because the model functionality runs in the cloud. Using this cloud functionality, should be done with care however, as it is not suitable for every business operation to share their data to third-

party companies, as will be discussed later in this section. An alternative to the potentially risky API usage, is to opt for locally runnable models. There are various models which can be freely accessed for this purpose. Examples of downloadable models are, amongst others: ChatGLM [48], Llama2 [1] and BERT [42].

Lastly, both Hub [27] and Sciannameo et al. [23] find that although the LLMs were trained mostly on English datasets, it did maintain the ability to extract data from documents when they were written in other languages, such as German or Spanish.

Although the usage of LLMs for information extraction can be advantageous, as expressed in previous paragraphs, there are also disadvantages. Firstly, it seems that certain capabilities of LLMs, come with the size of the model. One of those size-dependent capabilities is the model's zero-shot learning ability. The study by Fink et al. [20] achieves reasonable performance by using GPT-4, which has a model size of around 175 billion parameters, with a zero-shot prompting strategy. However, Sciannameo et al. [23] and Van Der Elst [24] find that the zero-shot performance was lacking when using GPT-3.5 and BERT, which have model sizes of around 6 billion and 110 million parameters, respectively.

Another capability being affected by a reduction of model size, is the ability to extract implicit information from text. This is shown with a study by Wiest et al. [1], where they compare three variants of Llama2, each having another number of parameters, 7 billion, 13 billion or 70 billion. Their findings are that the 70 billion parameter variant is able to accurately extract implicit features, while its counterparts are not.

Secondly, LLMs might be perceived as a one-fits-all solution for information extraction from documents. However, many documents are enhanced with information rich infographics, which include all visual representations of data, and can be graphs, pictures, icons and more (See Table 2.3). LLMs have difficulties capturing this data, despite several attempts to address this issue [27]. The study by Van Der Elst [24] finds that an existing method called GRID extraction, which is an algorithm which algorithmically divides the PDF-table into a fitting grid, outperformed the LLM enabled approach, when extracting from tables. Although LLMs made significant improvements since then, recent papers do still showcase several shortcomings, such as difficulties with numerical data, magnitudes [23, 24] and missing data in tables [25]. The lack of multi-modal processing capabilities could contribute to a lower performance of GPT-4 on the lower accuracy of GPT-4 on the extraction from sustainability reports, as can be seen in the studies by Zou et al. [26] and Hub [27].

Also, similar to the improvement in zero-shot learning with a larger model size, confabulations and hallucinations seem to decline with the size of the model. Fink et al. [20] finds that GPT-4 had 12% less confabulations than its smaller predecessor, GPT-3.

Another disadvantage of using LLMs is that their complex structure makes it is almost impossible to analyze the internal operations of the model. The lack of explainability of the internal reasoning of the model, makes that LLMs are often being referred to as black boxes [23]. Guellec

et al. [39] attempts to address this the lack of explainability by asking the model in the prompt for an explanation of the reasoning. Though, what the correlation to the model output and true reasoning is, remains unclear.

Lastly, is OpenAI's GPT-4 exclusively available via the API. This poses several problems for practitioners. Firstly, this could result in significant costs, when a large number of API requests is needed [45]. Secondly, it might not be allowed to share sensitive data to a third party according to legislation in some countries, including Europe [1], excluding various organizations from application of such cloud-based LLMs. Thirdly, using an API hinders reproducibility, since the version of the model might change, which would make error analysis more difficult [19]. Those disadvantages related to the exclusive availability via API could pose a challenge for organizations that want to make use of zero- or few-shot learning capabilities of LLMs, because especially the models which have a larger model size, such as GPT-4, seem to be able to perform well with this technique.

IMPLEMENTATION REQUIREMENTS

Although LLMs could be an opportunity for organizations that would like to automate information extraction, there are challenges to consider. Those challenges form as input for requirements which might need to be adhered to when deploying an LLM in practice, depending on the use-case. This section dissects the requirements posed in the literature.

Firstly, LLMs sometimes produce incorrect results due to their probabilistic nature. Although there are ways to mitigate this behavior, such as setting the temperature to 0, or investing in precise prompt engineering, complete mitigation of randomness in the response remains to be a challenge. Therefore, setting up information architecture which checks for and moderates faulty output of the LLM might be needed to mitigate the effect might be needed. Rajan et al. [50] did this by, after generation, going over a rubric to assess the models output on factual accuracy, groundedness and relatedness to previous answers. Hub [27] used another solution, namely to prompt the model to label hesitant cases, so that post-processing could be applied here.

Secondly, handling sensitive data is required for various use cases. One way to cope with sensitive data, could be to use a technique called data scrubbing [53]. This implies that all potentially sensitive data is removed by a smaller model which can be ran locally, before making use of an external API.

Another requirement of using LLMs is that sufficient hardware capacity must be available when choosing to run the model locally [21]. When there is also sensitive data handling, it is likely that organizations require to use a local model. However, organizations often do not possess such hardware, and therefore need to limit the size of the LLM they utilize. The model performance, including its ability to collect implicit information, is to some extent dependent on the model size, which might reduce the applicability of LLMs for organizations handling sensitive data and having limited hardware capabilities available. Though, Wiest et al. [1] shows, that even

though Llama2 is a smaller, locally runnable model, with the right prompting techniques it is still possible to let the model make the right deductions. Moreover, they found a pattern that the smaller the model, the more relevant prompt engineering is for maintaining high performance.

Various studies opt to use a locally runnable model instead of the API. In the study by Yang et al. [48], a technique called quantization is applied, which has the aim to reduce the model size so that it could be ran locally. Quantization implies that the weights of a pre-trained model are converted to consist of a lower number of bits. Although this positively influences the model size, it could also makes the model slower. The study by Yang et al. [48] showed that the model where quantization was applied maintained similar performance to the initial model, while even performing better on the ROUGE-L[3] score, which measures the largest common subsentence between the initial text and its summarization. It seemed that the quantized model has a higher likelihood of outputting the exact information than its non-quantized counterparts.

Another challenge when applying LLMs for the extraction of information, is that they are sometimes inaccurate when extracting numerical data from tables. As the study Balsiger et al. [25] shows, the highest performance was obtained by GPT-4, which extracted 83% of the values correctly, BARD was only able to do so for 63% of the tasks. Balsiger et al. [25] suggests investigating integrating the process of the LLM with some sort of calculating module, to cope with this effect, which would be a requirement when a low error rate is required on this task, as the average error rate in the study was 25%.

Also Dimmelmeier et al. [45] poses a lack of understanding from infographics as a major challenge. As Section 2.1.3 describes, the current state of LLMs is not suitable to extract from those media. Gomes Ziegler [22] shows that a multi-modal method for information extraction could mitigate this issue. In the study, they cross-validate results between the textual and the image model. The visual model was GPT-4 Vision. The results were significantly better, than using those the solely the text-module or the image-module.

Depending on the size of the model, prompt engineering may also require a significant amount of time to be invested. The study by Labbe et al. [41] compares two prompts, one prompt has consistently a better performance of around 7% on all metrics (accuracy, recall, precision). The dependency on the prompt might not be the same for each model however, as Wiest et al. [1] found that the larger the model, the less the performance was affected by prompt engineering.

Lastly, when an ML model is deployed in the operations of an organization and reasoning is involved, it is important to be able to detect the internal reasoning of the model, to ensure ethical correctness. Although the importance of explainability of LLMs in information extraction is recognized, only the study by Guellec et al. [39] attempts to obtain an explanation of the behavior of the model. They do this by asking the model to provide an explanation for its reasoning.

To summarize, the following requirements could apply when using an LLM for information extraction:

[3]https://aclanthology.org/W04-1013.pdf

- Handling faulty or inconsistent output due to the probabilistic nature of LLMs.
- Sensitive data handling when using cloud functionality.
- Hardware performance requirements when running locally.
- Solving numerical data problem using external solutions.
- Application of size reduction techniques when having limited resources.
- Solving absence of graphical data analysis capability of LLM.
- Investing in prompt optimization.
- Depending on the use case, explainability

VALIDATION METHODS

Methods to perform validation include direct methods, such as measuring whether the output aligns with the desired output of the model, and indirect methods, where the output is used to obtain another metric, which is then compared to a golden standard. This section focuses only on metrics that validate information extraction. Metrics mentioned in the selected literature which are measuring something else, are excluded for this review.

Indirect validation methods Assessing the performance of a student on an exam cannot be done without having the answers, or knowing the answers yourself. Likewise, to assess the performance of an LLM, a labeled dataset is often required. If this dataset is not available, expert knowledge might be needed, which is not always available, for example due to insufficient money or time. Thus, in those cases, it might be preferred to opt for an indirect measure of the performance of the LLM, making use of already present resources, and thereby saving time and money. Moreover, when LLMs are used to improve a certain task which is quantitatively measurable, it might be preferred to use a process performance indicator directly, since this is the variable of interest anyways [42, 46].

Two studies applied this method of indirect validation. Jose et al. [46] used the Mean Absolute Error (MAE) between the true estimated state of a machine, and the predicted state, which was expressed in a number. Here, they made a comparison between a fine-tuned version of the LLM and the pre-trained version. Equation 2.1 below describes how the MAE is calculated:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{2.1}$$

where y_i represents the true state values, $\hat{y}_i$ represents the predicted state values by the model, and n is the number of observations. Similarly, Kim et al. [42] compared a base model which was not fine-tuned, against a fine-tuned variant by looking at the return on investment, when using the output of the model as input for investment decisions.

Direct validation methods When an expert opinion can be obtained, or an answer sheet is present, all sorts of metrics can be employed. This section discusses the metrics, and the motivation of including them in the evaluation of a model, after expanding on how to obtain expert opinion for evaluation.

When comparing the output of a machine learning model against the experts results on the same task, it is common practice to analyze the Inter-Annotator's Agreement (IAA) agreement between the different annotators, although not every study does this [19, 21].

García-Barragán et al. [40] used for this the F1-score, which is explained below with Equation 2.2:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2.2}$$

Here the precision is the number of true positive predictions, divided by the total predicted positives. Recall is the number of true positive predictions, divided by the actual positive predictions. In the F1-score, often, the predictions are compared against the baseline, determined by expert annotators, however, in this case, annotator A, is compared against annotator B. The study showed an F1-score of 90% which indicated sufficient agreement between the two.

Another method employed to obtain a ground-truth to compare against by Wiest et al. [1], was to first let experts individually label the cases, and when disagreement was present, allow a discussion. Lastly, a common evaluation method of the IAA is the Cohen's Kappa metric, which was used by Usmanova and Usbeck [44]. The formula for Cohen's Kappa is given in Equation 2.3:

$$\kappa = \frac{P_o - P_e}{1 - P_e} \tag{2.3}$$

where:

$$P_o = \frac{\text{Number of agreements}}{\text{Total number of annotations}}$$

$$P_e = \sum_i \left(\frac{\text{Total annotations by annotator } i}{\text{Total number of annotations}} \right)^2$$

When the Cohen's Kappa metric is -1, this indicates complete disagreement and 1 indicates complete agreement. Values of Cohen's Kappa above 0.5 indicate a moderate to strong agreement.

Once the labeled dataset has been obtained, a comparison between the output of the model and the ground-truth values from the dataset can be executed. For this, the most used metrics were the F1-score, precision, accuracy, and recall, which are explained below. For more information on which study used which validation methods, consult Appendix A1.

Less common metrics which could be applied for information extraction are the receiver-operating characteristic curve (ROC) [43], and the negative predictive value (NPV) [1, 21]. The ROC is a method to determine which model is best by mapping the true positive rate (TPR) on one axis,

against the false positive rate (FPR) on the other axis. Here, the best model is the model for which the area under the curve is the largest. Based on preference, other metrics could be put on the axes as well, so did the study in Maibaum et al. [43] also have a ROC with precision and recall. The NPV could be seen as the opposite of the precision. It measures how often the outcome is truly negative, when the model predicts negative.

Contrary to the aforementioned methods, qualitative assessment of the performance is also possible [18, 51]. Bronzini et al. [18] did a qualitative analysis of the output by querying the analyzed model once more, this time to assign it to evaluating its own output. The assignment of the model was to label the output with a number on a scale, based on the quality of the output. In the study by Sonnenburg et al. [51], the qualitative part of the assessment was twofold. Firstly, experts assessed the scientific quality of the responses. Secondly, the resulting output of the model on the task at hand was compared to the results of experts performing the same task.

Lastly, an evaluation approach when combining RAG with information extraction was employed by Zou et al. [26], where they made the distinction between the accuracy of disclosure, and the extraction accuracy. Here, the former indicates whether the model was able to distinguish whether certain information was present in document, and the latter indicates whether it extracted the correct sustainability information.

DISCUSSION

This SLR analyses the potential of LLMs to extract information from sustainability reports. Due to the diversity and freedom in those reports, manual analysis is a tedious task. Furthermore, comparison between several sustainability reports might be hindered due to a lack of structure. Recent advancements of LLMs, have caused LLMs to become an interesting option to consider for information extraction. Several studies across several domains have already examined their suitability for this task. Previously, no SLR focused on the potential of LLMs for information extraction from sustainability reports had been conducted. Therefore, this study contributes to the knowledge base by giving an overview of the meta-information of the studies, sorts of LLMs applied, their benefits and disadvantages, requirements for implementing LLMs and validation methods. Also, does this SLR indicate future research directions. Furthermore, does this study provide an overview of best practices for practitioners who are opting to use LLMs for information extraction from free-text-documents, which is discussed in Section 2.1.4.

Despite its contributions, this study has several limitations. Firstly, only one database had been used to obtain the initial selection of papers. This might have unintentionally led to the exclusion of valuable papers. This might also have led to the fact that a limited number of studies were found which discussed the extraction of data from financial- or sustainability reports. Moreover, the discussion of the extraction of quantitative data, was limited.

This leads to the second limitation of this study, which is that we did not completely adhere to the SLR methodology by Snyder [34]. By adding three papers in the search phase, we made

an adaptation to the SLR methodology, because we deemed there to be an insufficient number of papers in the selected literature discussing information extraction from financial or sustainability reports. However, we still guarantee reproducibility of this study.

Also, the limitations of this SLR did not prevent the identification of three future research opportunities. Firstly, the selected literature showed that extracting information from tabular and numerical data is a challenge for LLMs. LLMs employed for information extraction could make mistakes by misguessing the magnitude or hallucinating numbers, which could be a problem when analyzing sustainability reports to support decision making, where inaccuracy is not tolerable. This yields the request for more research on how to instruct LLMs to not make those mistakes, or alternatively, how to repress the symptoms of this behavior, so that they can be used reliably in an organizations context.

Secondly, all selected studies, in addition to the one by Gomes Ziegler [22], excluded infographics and graphs for the information extraction, because the LLM cannot use input other than text. Especially when working with sustainability reports, infographics capture a large portion of the information. Excluding those could significantly impact the likelihood of successfully distilling the relevant information from the report. Thus, methods are also required to capture information from infographics and graphs.

Lastly, Section 2.1.3, explains that the current most advanced models are only available via an API (OpenAI). It is quite likely that future models with similar capabilities will also only be available like this, since the majority of organizations lack the resources to run such enormous models. Thus, ongoing research is needed to explore how handling sensitive data can be done responsibly, which is particularly interesting for organizations in industries such as finance and health, where vast amounts of personal data are processed.

To summarize, the research gaps resulting from this SLR are:

- Tabular and numerical data
- Infographics and graphs
- Sensitive data handling

2.1.4. MANAGERIAL PERSPECTIVE

Section 2.1.3 discussed the main findings of the synthesis of the selected literature. These findings are not only useful for researchers, but could also have implications for practitioners. This section discusses those implications.

This SLR highlighted the capabilities of LLMs in extracting information from free text, where GPT-4, the state-of-the-art model, often extracted information with an accuracy of 95%. This performance is comparable to that of humans performing information extraction, which can reduce the need for manual labor in this task [19]. Additionally, utilizing LLMs for information extraction could lower the error rate. However, it should be noted that algorithmic errors by an

algorithm/machine are often perceived with greater aversion than human mistakes [54].

Moreover, when extracting data from tabular formats, the performance of LLMs is insufficient. Similarly, the presence of infographics in the input document reduces the quality of information extraction of the LLM. Both shortfalls of LLMs require alternative methods to be employed when dealing with documents which contain important information in table, chart, or visual formats.

Practitioners should also consider that running LLMs locally requires expensive hardware. Alternatively, cloud-based LLMs can be accessed via an API, but this approach may pose regulatory risks when processing sensitive or private data, as it involves transmitting information to a third-party provider. Furthermore, relying on a cloud-based solution can lead to significant costs when LLM usage scales. Therefore, selecting the optimal approach depends on the specific use case and should be carefully evaluated.

In general, LLMs show impressive ability to extract information from documents. However, a 100% appears to be infeasible due to the inability to extract visual information.

2.2. DOCUMENT AI FOR INFORMATION EXTRACTION FROM SUSTAINABILITY REPORTS

After researching the potential of LLMs for accurate extraction from sustainability reports, we concluded that the current extraction performance of LLMs for information extraction from sustainability reports is insufficient. This is partly attributable to their inability to grasp visual and spatial elements in the document, which are lost when the text in the document is entered as a one-dimensional string. Document AI is a research domain which uses multi-modal LLMs (MLLMs), to better grasp the contents of a document. Enabling better comprehension compared to solely relying on textual input. We explore the potential of Document AI by doing this second literature review.

2.2.1. INTRODUCTION

The layout and visual information of a document provides important information for accurate question answering and information extraction. Semantic elements, such as paragraphs, lists, and captions, and visual elements, such as tables and figures, are such components that give meaning to the textual relations of a document. Business documents are typically visually rich documents (VRDs) Huang et al. [55]. Visually Rich Document Understanding, to which we refer by Document AI in the remainder of this study, is a research domain that employs various AI techniques to better understand the contents of a document [28]. Contrary to LLMs, Document AI not only considers textual content but also layout and visual information.

Document AI architectures are built to support various downstream tasks, such as visual question answering (VQA), key information extraction (KIE), and information grounding. VQA refers to questions that are answered based on a VRD. KIE is the extraction of either a standard set of document components, such as the title or the footnote, or more specific information such as the customer name in a receipt. Therefore, KIE focuses more on direct extractions, while VQA aims more at answering a question through reasoning. However, there could be overlap in the tasks of VQA and KIE, such as finding the Scope 1 emission in the document, which can be both a VQA or a KIE task, as this can be both the answer to a question, but also the topic of interest for more specific KIE Ding et al. [56]. In this paper, we use IE and QA interchangeably to refer to the task we require from the Document AI system, since the downstream task will include both more direct extractions (extractions) and questions where reasoning is used (abstractions) .

Recently, transformers have enabled rapid improvements of Document AI methods. Transformers are an evolutionary method which uses attention to consult an entire sequence of tokens (could be words or numbers) at once. Attention is a mechanism where it is learned what the relations between various tokens are. To be able to capture these relations, the transformer is trained using unsupervised pre-training Douzon et al. [57]. In the pre-training process, a proportion of the tokens in a piece of information is masked, which serve as placeholders for the predictions of the model. By letting the model predict what masked tokens should have been, and calculating a loss when the model is wrong, transformer-based models like BERT ef-

fectively learn contextual relationships in text. This makes them useful for various information processing tasks, including text and document understanding Devlin et al. [58].

Previously, extraction from VRDs required substantial manual input to annotate data; however, unsupervised learning methods applicable to transformers, such as Masked Language Modeling (MLM), Masked Image Modeling (MIM), and Masked Region Modeling (MRM), allow models to leverage large amounts of unlabeled data, which benefits the performance of transformers. Two components in the transformer architecture are the encoder and the decoder. The encoder embeds the input to the model, which could be text, vision, or layout data. The decoder returns embedded information in human-readable language.

Previous studies have shown that layout and image information is important for accurate IE and VQA from a document. LayoutLM was the first study to apply text and image learning at the same time when training a transformer Huang et al. [55]. Their positional encoding was absolute and therefore not robust for out-of-domain data. Later, various innovations have been added to encode positional information in the transformers, such as relative positional encoding and rotary position embedding (RoPE), which uses a rotation of the attention score to give an indication to the model about the relative positions of the tokens [59]. Later, methods that rely fully on computer vision to process the document become more popular, having a simpler approach by not requiring to integrate the information from several modalities, which can be complex, and lead to overlap of information, resulting in an unnecessarily large context window. Those models are often referred to as large visual language models (LVLMs) Wang et al. [2], Dong et al. [60].

2.2.2. GOAL / RESEARCH QUESTIONS

Given that sustainability reports are VRDs and that the use of LLMs alone has been shown to produce insufficient performance for IE from sustainability reports (Section 2.1), it is interesting to evaluate to what extent Document AI can perform better on this task. The goal of this literature review section is to explore how Document AI can help to automatically extract information from sustainability reports. To do this, we select the best models by comparing performance on four selected benchmarks. Then, we study the techniques and architectures of the selected models, to get an understanding of what techniques contribute to their good performances, and whether those can be applied for information extraction from sustainability reports.

Therefore, the research questions for this narrative literature review are as follows:

- Which document AI models are currently the most accurate?
- How can document AI models be trained, built, and deployed? (while adhering to business requirements)
- What contributes to a high model performance of document AI models on the selected benchmarks?

2.2.3. METHODOLOGY

To obtain an overview of the current state of Document AI, we performed a narrative literature review, which allows for a degree of flexibility and author's judgment [61, 62]. This helps since we have a unique approach to collecting the literature, which was as follows. Firstly, we collected and compared the performance of the models used in recent studies on Document AI (see Table 2.4, to get an overview of the best performing methods. Secondly, we selected the papers that yielded the best performing methods to further analyze them.

During this study, we first select the best-performing ready-to-use methods on sustainability reports. Subsequently, the selected model architectures are utilized and potentially improved to develop a pipeline that meets the requirements for deployment in a business setting. To identify these best-performing models, we distinguish between their performances on the following benchmarks: DocVQA Mathew et al. [30], MP-DocVQA Tito et al. [31], MP-DocVQA (Unlimited) Kang et al. [63], and MM-LongBenchDoc Ma et al. [32]. DocVQA contains 50000 questions on 12000+ images from document pages from the UCSF Industry Documents Library. Most documents come from the tobacco, food, drug, fossil fuel and chemical industry, as they contain various images. All questions are limited to extraction from one page, which is not aligned with a business scenario, where information often needs to come from a document consisting of multiple pages. For this reason, the MP-DocVQA was created by Tito et al. [31], which is a dataset which used the DocVQA (DocVQA paper) dataset as a basis, but removed the restriction of one page and instead used a limit of maximum 20 pages. Thus, this dataset consists of QA pairs for documents consisting of 1 to 20 pages, reflecting a more realistic business scenario [31].

However, sustainability reports often exceed 50 pages. The benchmarks more representative of this page number are MM-LongBenchDoc [32] or an unlimited variant of MP-DocVQA [63]. MM-LongBenchDoc is aggregated from various previously existing benchmarks, such as DUDE, SlideVQA and FinanceBench, contains on average 47.5 pages and has more than 30 documents which are longer than 70 pages, on a total of 135 documents. This benchmark probably aligns best with QA on sustainability reports. As alternative, one study analyzed the unrestricted variant of MP-DocVQA, which also leads to documents ranging up to 800 pages [63].

Performance in benchmarks measuring Document AI performance is often measured using averaged normalized Levenshtein similarity (ANLS) [31, 64], or Generalized Accuracy (G-Acc) [64]. ANLS is a measure where an algorithm is used to determine the similarity between the golden standard answer (the correct answer, evaluated by annotators of the data) and the output of the model. Alternatively, G-Acc uses GPT as an evaluator to determine whether the answer aligns with the golden standard answer or not. Both metrics have a minimum of 0 and a maximum of 1. However, what distinguishes the ANLS from the G-Acc, is that the ANLS handles a threshold for the similarity score, below which a score of 0 will be assigned. In most papers, the score is multiplied by 100, to obtain a number between 0 and 100.

During the first phase of this study, we make a selection of previous studies that developed a

document AI method or model. We do this by selecting the three best models for which the model weights are available, so that we can assess the performance of those methods within the given timeframe of our study. It is possible that better methods exist of which the weights or code is not available. Those studies will also be included in the analysis. In this way, we are testing the models we can, while not disregarding potentially better techniques from recent literature. In addition to performance on the benchmarks directly, page-retrieval performances on MP-DocVQA and MM-LongBenchDoc are also analyzed.

To select the literature for the first phase, we primarily relied on the snowballing method [65]. This implies that we explore studies by targeted search, or by evaluating citations and references of studies, to obtain useful material for our study. For the page-retrieval task, an additional targeted search was conducted; however, it did not yield any papers that had not already been identified through snowballing. Google Scholar [4] was used as search engine, as this covers a wide selection of databases, and, most importantly, also includes Arxiv, which is a database of preprint papers. This is valuable, since the research domain is developing rapidly, and papers which are published a year ago, are often already outdated, as can be seen by the model performance of our selected literature. During the second phase of our study, we synthesize the information according to the research questions, using inductive coding [66].

2.2.4. MAIN FINDINGS FROM THE LITERATURE

The resulting studies from the selection process as discussed in Section 2.2.3, led the selection of studies as presented in Table 2.4. For all results, we refer to Appendix F.

For selection of page-retrieval methods, the selection process differed because there are a limited number of studies that considered the page-prediction ability of the model. Also, the datasets for those studies differ, which hinders effective comparison of the page prediction results. Therefore, every study where page-prediction is a measurable sub-task of the model, is considered and analyzed. This led to the selection of four studies, which are RM-t5[NA] [67], SelfAttnVQA [63], ColPali [68], and SV-RAG [64].

[4] https://scholar.google.com/

Table 2.4: Comparison of models on four benchmarks: DocVQA, MP-DocVQA, MP-DocVQA-U, and MM-LongBenchDoc. ANLS = Average Normalized Levenshtein Similarity, G-Acc = Generation Accuracy. Par = Parameter size. "[NA]" indicates that the model weights or code was not available, and that this model is not tested in the first phase of this study, see Section 2.2.3. Note: MM-LongBenchDoc measures performance in G-Acc, not in ANLS, which is indicated by a double midrule in the table.

Benchmark	**ANLS**	**Par**
DocVQA		
Qwen2-VL-72B	96.5	72B
Qwen2-VL7B	94.5	7B
DocVLM [NA]	92.8	7B
Arctic-TiLT [NA]	90.2	800M
InternLM-XComposer2-4KHD	90.0	8B
DocFormerV2	87.84	750M
MP-DocVQA		
DocVLM [NA]	84.5	7B
M3DocRAG	84.4	10B
Arctic-TiLT [NA]	81.2	800M
GRAM [NA]	80.3	859M
Wukong [NA]	76.9	8.5B
DocFormerV2 [NA]	76.4	750M
SV-RAG-internVL2	71.0	4B
mPlug-DocOwl2	69.42	8B
MP-DocVQA-U		
SelfAttnVQA	0.54	273M
Benchmark	**G-Acc**	**Par**
MM-LongBenchDoc		
SV-RAG-InternVL2	34.0	4B
M3DocRAG	21.0	10B
Arctic-TILT [NA]	25.8	800M

As we mentioned in Section 2.2.3, the long-context benchmarks MP-DocVQA-U and MM-LongBenchDoc, best reflect the scenario where Document AI is employed to extract information from sustainability reports. However, only SelfAttnVQA had been tested on the unlimited variant of MP-DocVQA. Similarly, most papers did not measure their performance on MM-LongBenchDoc. The papers that did, obtained a much lower score than the currently SoTA (state-of-the-art) model GPT-4o [5]. Those papers obtain G-Acc scores between 21 [69] and 34 [64], while GPT-4o obtains a G-Acc of 42.8, see Table 2.4. This indicates that there is room for improvement, although it is worth mentioning that a human also has a challenge answering the questions in the dataset, obtaining a score of 65.8.

Since few papers assessed performance on MM-LongBenchDoc and MP-DocVQA-U, we also explore model performances on middle-long documents, for which we use MP-DocVQA [31] benchmark. This dataset used DocVQA [30] as a basis, but removed the page restriction of documents from one to maximum 20. The better performing (above 80 on MP-DocVQA) multi-page models which have been selected for this study, were extensions of single-page models (DocVLM, M3DocRAG, Arctic-TILT, GRAM). The major challenge when dealing with large documents is the limited “attention span” of the current Document AI methods, which is the result

[5]https://openai.com/index/hello-gpt-4o/

of the way the attention matrix works, namely by learning the influence of each token, to each other token, leading transformer models to scale quadratically with an increase of the input [70]. This implies that as soon as the context window grows by n , the attention matrix, which stores the potential relationships, grows by $n \times n$.

For single-page documents, the size of the attention matrix is within capabilities of the Document AI models, however, when multi-page document documents need to be processed, it is likely that the attention matrix size exceeds the input size limit. To address this limitation, solutions typically focus on either compressing data to fit within the context window or directly fetching relevant document sections or pages for processing. Compression techniques aim at condensing the information without losing significant comprehension capabilities [55, 71, 72]. For example, DocVLM [71] compresses the OCR content using 64 learnable queries, substantially reducing the context window by approximately 80%, while still maintaining a similar comprehension ability of the model. This technique notably improves Qwen2-VL by 2.4% and InternVL2 by 3.7%, achieving state-of-the-art performance on MP-DocVQA.

Similarly, DocFormerV2 [72] uses a downsampling layer for compression, together with a simple linear projection, to reduce the context window coming from the visual data. Although this model was not specifically built for multi-page document parsing, it performs quite well. Considering that no further special techniques were applied to reduce the context window, the way that the image is handled by DocFormerV2 might be a useful technique for other multi-page Document AI models. LayoutLM3 [55] took it a step further and directly inputted the image-patches into an encoder by directly applying linear projection to the image patches. Thereby, both DocFormerV2 and LayoutLM3 replaced previously computationally expensive CNN transformers, while reducing context size at the same time.

Paired with employing compression, also chunking helps to maintain a manageable attention matrix size. Arctic-TILT [73] applied chunking when making a modification to TILT [74], which initially considered the complete input document at once. Instead, Arctic-TILT considers each page apart when creating an encoded version of the document. Then the entire pipeline is trained end-to-end, including the decoder, so that it can obtain the right information from the encoded chunks.

Also, the model of GRAM [70] is implemented with a chunking technique to mitigate the quadratic scaling problem. On top of that, they create a page token, next to the page embedding, for each page. Then, GRAM utilizes several global-local encoders, which allow for exchange of information on which document parts are important, so that more attention can be given to those parts. This leads to better oversight, and therefore improved question-answering. Since GRAM still saw a linear scaling problem after their innovation, they also employ another compression mechanism based on the query of the end user.

Another way to reduce the context window is to employ RAG-like retrieval generation (RAG) methods [64, 68, 75]. The idea behind this method is that relevant passages are fetched, so

that thereby irrelevant passages are filtered out, reducing the context window, and directing the focus of a QA component on the relevant data. Two papers (M3DocRAG [69], SV-RAG [64]) use a previous model architecture called ColPali [68], which uses the concept of Contextual Late Interaction (Col), which implies that information not initially stored as a single vector, instead, the model has a more rich token-level representations, which are used by the model during inference. Then, during training, the goal is to obtain relevant information to the query. For this, similarity matching is applied. This is a technique where the embedding of the query is compared to the embeddings of the passages in the document. M3DocRAG outperformed SV-RAG on the MP-DocVQA benchmark, likely due to the larger MLLM used in M3DocRAG.

Alternatively, Wukong [75] created their own retrieval module, which they refer to as the 'sparse sampler'. In contrast to ColPali [68], it returns passages of the document, such as a figure or a paragraph. This offers one lower layer of granularity compared to fetching entire pages. The performance of Wukong is almost unaffected by an increase of the input document size. Unique to Wukong, is that they first use a PDF parser, which combines digestion of, on the one hand the visual information of the document, such as charts, tables, and images, and on the other hand the textual information. In this way, visual context is only leveraged when needed, due to which the context window is efficiently utilized [75].

All methods relying on the retrieval of relevant passages [64, 69, 75], found that relying on the top-5 best matching results yielded a higher quality of answers, than when they rely on the top-1 result. Since the retrieval accuracies of the measured top-1 are between 79 and 90 percent on MP-DocVQA, the top-1 frequently misses the relevant page, and when the number of pages is increased, the probability that the required page is present increases [64]. This could lead to the expectation that using a large selection of pages would lead to a high accuracy, however, expanding to much more passages than five does not lead to a large performance increase, while the context window grows. Thus, fetching the top-5 seems to be the optimum with the current passage-retrieval quality.

PAGE-RETRIEVAL

The approach used by M3DocRAG [69], where they used one pre-trained page-retriever, and one pre-trained MLLM for QA, is more adaptable than techniques such as GRAM [70] or Arctic-TILT [73], which requires end-to-end training. Instead, M3DocRAG combines a retrieval model and an MLLM model, without any training, making it an attractive option in a business setting, as new and better models can be 'plugged in' to increase pipeline performance [69].

Inspired by the architecture of M3DocRAG, we also analyze single page methods and page-retrieval methods, to explore potential improvements to this pipeline by replacing the page-retrieval, or QA-module. The only study that assessed the page-retrieval performance on MM-LongBenchDoc was SV-RAG. SV-RAG employed the same MLLM for retrieval as for answer generation. What they found, is that the retrieval accuracy increases with the number of parameters used in the base-model, see Table 2.5.

The retrieval accuracy on the MP-DocVQA dataset was higher, likely because of the lower number of pages in the MP-DocVQA dataset. The evaluated methods differ substantially. Hi-VT5 [31] uses a classification module designated to make a prediction of the page based on the contents. M3DocRAG [69] uses similarity matching to obtain the most relevant page. In the study on self-attention between page extractions [63], the self-attention module is an embedding of a single-page encoder to determine which page is relevant to the question, after which the page is used once more as context for decoding. Lastly, RM-t5 outperforms the others with an accuracy of 88.3% being more than 6.5% higher than the others. RM-t5 employs a recurrent memory token, which transfers information from the previous pages to the next one. This closely aligns with human comprehension, where previous context helps comprehending the context of the current page. Considered that the parameters are similar to the other methods [31, 63], and much lower than M3DocRAG's ColPali, this method proves to be effective.

Furthermore, SelfAttn gives an idea on how the model performance on MP-DocVQA (which is restricted to 20 pages), could translate to an unrestricted scenario. The SelfAttn scoring module had one of the best performances on page-prediction on the MP-DocVQA benchmark. However, this performance decreased significantly from 81. 55% to 60. 45% when the average number of pages in the dataset increased from 5.1 to 38.5. This is something to take into regard when employing page-retrieval modules in a similar way as Cho et al. [69].

Table 2.5: Comparison of R@1 page-retrieval performances on MM-LongBenchDoc and MP-DocVQA. * indicates that the model is built using the SV-RAG method. [NA] indicates that the code is not publicly available

MM-LongBenchDoc			MP-DocVQA		
Model	R1	#Par	Model	R1	#Par
Col-PaliGemma*	60.7	3B	M3DocRAG	81.05	3B
Col-InternVL2*	63.2	4B	Hi-VT5 [NA]	79.23	316M
Col-Phi3-vision*	65.1	4.2B	RM-t5 [NA]	88.32	312M
			SelfAtt	81.55	273M

TRAINING DOCUMENT AI METHODS FOR MULTI-PAGE

Because we want to explore opportunities for improving existing pipelines to adapt them to sustainability reports for an optimal performance, we observe also how multi-page document AI models are trained. In general, we see a distinction of three methods, namely end-to-end training, the training of a compression module and no training, see Table 2.6 for an overview, including the training data used in the methods.

The transformer architecture allows for a new kind of training, namely that of self-supervised pre-training. One advantage of this is that no human labels need to be added to the data [55]. However, it also has the disadvantage that the transformer models need a vast collection of data to be trained, require computing resources with sufficient RAM, and might take days to weeks to train [73].

Because the number of parameters in more advanced models increases, often containing be-

tween 0.8B and 7B, fine-tuning, or pre-training them from the bottom up is challenging and requires a large investment in time and resources. To address this, various studies explored the reuse of an (M)LLM, while largely [64, 70], or completely [69, 75] takes over their parameters.

SV-RAG [64] and GRAM [70] are two models that reuse model parameters, although still require updates to the model architecture. For SV-RAG, this is the case because to utilize the capabilities of the model, they use the LoRA technique by Hu et al. [76], to create adapters to the model. LoRA, which stands for Low-Rank Adaption, is a technique where instead of fine-tuning a complete Large Language Model, only the parameters of two smaller matrices that approximate the larger matrix are fine-tuned, while maintaining similar performance to those implementations where the full model is fine-tuned. For the page-retrieval adapter, they used the same datasets as those used in ColPali. For the QA adapter, only SlideVQA was used. As part of fine-tuning, the technique called contrastive learning is applied. In contrastive learning, the dataset consists of queries linked to a positive example and a negative example. Which, using a reward and a loss, help in more efficiently fine-tuning the adapter [64, 75]. Chen et al. [64] applied LoRA in such a way, that they reuse the same MLLM as a basis for both retrieval and QA, using their so-called dual-adapter architecture. This saves memory, as it is not required to store two models, one for retrieval and one for QA.

GRAM applied another approach [70], using the pre-trained weights of DocFormerv2 [72], Though, for training the global-local encoder-decoders modules, which are used for finding the relevant information in the document, their architecture needs to be trained in an end-to-end manner, which means that the full pipeline is updated. However, the number of datasets used for this was much smaller than that typically used for training an MLLM. Only the MP-DocVQA, DUDE and DocVQA datasets were used for this. Showing the effective reuse of model parameters of DocFormerV2. Here, pre-training is still omitted, making this a more accessible option.

In contrast to GRAM and SV-RAG that require some level of adaptation of the base-model, DocVLM [71], M3DocRAG [69] and Wukong [75] leave the reused model's parameters as is, see Table 2.7. DocVLM reuses as base model Qwen2-VL-7B [2]. During training, the parameters of Qwen2-VL-7B remain frozen, and only the 64 learnable queries, the OCR-image alignment and the OCR encode were trained. With the goal of allowing the output of those modules to be processable by Qwen2-VL-7B. Their compression method reduced the input size, while maintaining most of the performance.

Similarly, Wukong [75] reused weights for the vision encoder and the LLM, which come from IXC2-VL-4KHD. All which is trained is the sparse sampler, which is essentially a similar retrieval method to ColPali and the system used in SV-RAG, only now it takes as input OCR inputs and extracted figures from the PDF Parser. Training the sparse sampler is done using supervised-fine-tuning and contrastive learning, as was explained earlier in this section.

Lastly, there are also models that were fully trained from start to finish, namely Arctic-TILT [73] and DocFormerV2 [72]. Arctic-TILT reuses and modifies the architecture of a previous method

called TILT, for the reason of reducing the context size and the resource requirements of a multi-page model. To obtain this goal, they make several modifications to the architecture of DocFormerV2 which require it to be retrained, such as a modified text-vision fusion algorithm and chunked processing using a sparse attention matrix. An advantage of Arctic-TILT's end-to-end training, is that their pipeline adjustments made it possible to train the pipeline end-to-end on a single GPU of 24GB RAM [73].

DocFormerV2 is trained end-to-end as well, though is an odd one out, as Appalaraju et al. [72] did not specifically design this model for multi-page document understanding. The model was trained using self-supervised pre-training, where the encoder was given the task of token-to-grid prediction and token-to-line prediction. On the end of the pipeline, the decoder was trained using masked language modeling, to generate the right text [72].

Table 2.6: Training Approaches and Datasets for Various Document AI Models

Study	**Training**	**Training Data**
Arctic-TILT	End-to-end	Pre-training: CCPdf, OCR-DL, fine-tuning: Kleister Charity, Kleister NDA, CHARTInfographics, DeepForm, DocVQA, DUDE, FUNSD, InfographicVQA, SQuAD 2.0, TAT-DQA, VQA-CD, VQAonBD
DocFormerV2	End-to-end	Industrial Document Library (ICL)
GRAM	End-to-end	MP-DocVQA, DUDE, DocVQA
DocVLM	Training of compression module	OCR: DocVQA, InfoVQA, ST-VQA, TextVQA, OCR-VQA, ChartQA, TextCaps, TAT-DQA; Vision: COCO Caption, VQA-V2
Wukong	Training of compression module	PaperPDF (Source Wukong), DocVQA, ChartQA, InfoVQA, MP-DocVQA, DUDE
SV-RAG	Training of two LoRA adapters	Retrieval adapter: DocVQA, InfoVQA, TATDQA, arXivQA, synthetic data; QA adapter: SlideVQA
M3DocRAG	None	Not further trained

Table 2.7: Single-page Models Used in Multi-page Document AI Systems

Model	**Single-page Models**
DocVLM	DocFormerV2 + Qwen2-VL-7B
GRAM	DocFormerV2
M3DocRAG	Qwen2-VL-7B
Wukong	Intern2-VL-4KHD
DocFormerV2	Self-built
SV-RAG	Intern-VL2
Arctic-TILT	TILT

OUTPUT VALIDATION

Since the ANLS on MP-DocVQA, and the G-Acc on MM-LongBenchDoc do not adhere to the business requirements, and cannot be trusted on blindly, insights into the quality of the model output is required. In the selected literature, grounding Wang et al. [2], Expected Calibration

Error (ECE) Borchmann et al. [73] and Area Under the Risk-Coverage Curve (AURC) Borchmann et al. [73] were applied for quality assessment. Here, grounding implies that the model can locate where the information came from. Ideally, the model specifies the precise location in the document. However, the only model which has this trait is Qwen2-VL, which returns the answer including a bounding box, see Figure 2.7.

Referring Grounding

```
<vision_start>Picture1.jpg<vision_end>
<object_ref_start>the eyes on a giraffe<object_ref_end>
<box_start>(176,106),(232,160)<box_end>
```

Figure 2.7: Example of referring grounding where the model identifies the coordinates of a referenced object ("the eyes on a giraffe") within the given image. Based on Qwen2-VL [2]

Other methods do yield certain pages and passages; however, they yield five of those for optimal performance, which makes it challenging to quickly trace back where the information came from [64, 69, 75]. Here, Wukong has a somewhat better interpretability, because it is not fetching entire pages, but instead retrieves passages or images from the document.

Lastly, ECE was applied by Arctic-TILT [73], which is a computation of the confidence of the answer. ECE is calibrated by having a validation-set, and can later be used to get an indication of whether the model is outputting truthful information.

SUMMARY OF NARRATIVE LITERATURE REVIEW

With this literature review, our aim is to get an understanding of what document AI models are the best for sustainability reports and why, so that we can use this knowledge to make improvements on existing models, if needed when QA performance is shown to be insufficient after the first round of experiments that are performed in the overarching study of which this literature review is part.

When working with sustainability reports, especially multi-page understanding is interesting. Here, the models that efficiently make use of the input context perform best. Therefore, various methods are utilized to reduce the input size, which can be distinguished as retrieval-like methods and compression methods. Here, the retrieval methods follow a pattern of, based on Col-like similarity matching, finding the most relevant pages or passages to filter out the irrelevant ones.

In contrast, compression methods used a more complex approach. Here, DocVLM trains 64 queries, to compress OCR information. GRAM uses multiple global-local encoders, to learn which information is important to the query, and Arctic-TILT effectively reduces context size using a sparse attention matrix combined with a special text-vision fusion model.

When analyzing performance on multi-page, both the compression and the retrieval techniques obtain good results. Also, it can be seen that the models that outperform are in some

way utilizing a vision-based model, regardless of whether the approach is compression or retrieval. On MP-DocVQA, this was Qwen2-VL-7B in both DocVLM and M3DocRAG, and on MM-LongBenchDoc this was InternVL2, in SV-RAG.

Despite obtaining good performance on MP-DocVQA, the performance appears to decrease when the length of input documents increases. Wukong was the only pipeline that was not affected by an increase of the document size [75]. In the selected literature. Thus, for IE from sustainability reports, which are documents often exceeding 100 pages, which is substantially more than the average number of pages used in MM-LongBenchDoc.

On the other hand, training can be used to update the retrieval or QA models. Whether training is useful depends on whether the retrieval and QA models are able to understand the terminology in the field of sustainability reporting. If this understanding is lacking, the retrieval model might create worse embeddings and the QA will generate worse answers. Using pretraining, or fine-tuning on the required tasks and documents, this understanding can be improved. Since those vision models are computationally heavy, having more than 7B parameters, training them requires substantial computational resources. Therefore, a commonly used technique is LoRA, which freezes the weights of the base-model so that only two smaller matrices need to be trained.

Moreover, there is potential for improvement without training. This can be done with modular architectures, such as that of DocVLM, M3DocRAG and Wukong, by replacing their current QA models with the best available models to date, potentially via API, allowing maximum model size. For example, the retriever of SelfAttn [63] could be combined with the best VLM available at the date of the experiment. Or, in M3DOCRAG [69] ColQwen [6] can be used instead of ColPali.

Lastly, the selected studies focused mainly on building and testing a new Document AI model. The studies did not consider pipeline improvements, which could be combined with already existing methods. The most simple idea could be to use iterations, combined with a confidence estimation such as ECE [73]. Another idea could be to combine with a second (M)LLM prompt, to check if the information is truly present in the page. A perfect model might not be necessary in this way and research into this area could also be fruitful.

[6] https://huggingface.co/vidore/colqwen2-v0.1

2.3. CONCLUSION

By performing an extensive literature review, our aim was to obtain information on the most promising direction for automatic information extraction from sustainability reports. This literature review consists of two phases: (1) SLR: Information Extraction using LLMs and (2) narrative literature review: Information extraction using Document AI. The literature review led to the following conclusions.

Employing LLMs for the information extraction from sustainability reports has several advantages. Firstly, it can handle large quantities of free-text data. Secondly, LLMs to date show impressive zero-shot capabilities, meaning that they display a large level of domain knowledge, without any specific training. Moreover, it requires minimal knowledge to setup the LLM, and modify its functioning, which can be done using prompt engineering. Lastly, when a model shows to lack understanding of certain topics, the supply of context documents allows for easily equipping the model with the right knowledge, to enhance its performance to the task at hand.

On the other hand, LLMs have the shortcoming that they can only process textual data, due to which the frequently presented figures, charts and images in sustainability reports cannot be processed. Moreover, this data is also ingested as one string of text, due to which layout information (partly) gets lost, causing problems to capture information precisely from tables and identifying dependencies in a context. This leads to a loss of information processed by the model, which is problematic when the goal is to maximize the extraction accuracy.

Therefore, the method of VRDU or Document AI seems a promising direction, where varying methods are used to ingest layout and visual information in an LLM. At their core, Document AI methods use an LLM, combined with a visual encoder, to process the layout and infographics of a document. Recent best-performing methods on DocVQA fully rely on visual input, instead of combining text and vision, obtaining a maximum ANLS of 96.5 on the benchmark, while the models are substantially smaller than proprietary models such as GPT-4o.

Although fully relying on visual input, it also results in a large context size when the number of pages increases. Without access to computing resource with a large RAM, processing of more than 20 pages at once is often infeasible due to out of memory errors. To solve this problem, Document AI methods apply compression and retrieval to remove irrelevant information from the query.

We used the benchmarks MP-DocVQA and MM-LongBenchDoc, to obtain understanding of the suitability of the SoTA Document AI methods on sustainability. The average number of pages of MP-DocVQA is 8.27 pages, and the best Recall@1 is 88.32. The average number of pages of MM-LongBenchDoc is 49.4 and the highest Recall@1 is 65.1. Considered that sustainability reports often count more than 120 pages, it is likely that further improvements are needed to the existing architectures. This is underscored once more by the QA scores on the given benchmarks, where the highest ANLS on MP-DocVQA is 84.5, and the highest G-Acc on

the MM-LongBenchDoc is 34.0. Thus, improving the models seems to be required when employing Document AI for IE from sustainability reports. LoRA fine-tuning could be a valuable approach here. Potentially, leveraging the modularity of architectures such as M3DocRAG, and replacing the retrieval or question-answering modules could reap improvements without requiring any training.

3

METHODOLOGY

3.1. STUDY DESIGN

3.1.1. CRISP-ML(Q)

CRISP-ML(Q) [3] is a process model specifically created for ML projects, fulfilling the need for guidance that practitioners and project organizations have. CRISP-ML expands on CRISP-DM, which is a process model for data mining. Although most phases of CRISP-ML and CRISP-DM overlap, CRISP-DM fails to address machine learning specific tasks. CRISP-ML gives guidance to those tasks, and, on top of that, provides a quality assurance methodology to help practitioners with challenges that typically raise during machine-learning projects.

CRISP-ML describes six phases [3]:

1. **Business and Data Understanding:** This phase consists of defining the business objectives and translating it this to ML objectives. Success criteria should be measurable, and created on business, ML and economical level. Moreover, feasibility of the project should be assessed, to prevent premature failures due to false expectations.

   By allocating time and costs on collecting , and verifying the quality of data in this phase, the feasibility of the project can be assessed. Moreover, the quality of data shall be assessed. Output of this phase is a scope for development, the success criteria of the application, and a data quality verification report.

2. **Data Preparation:** During this phase, the dataset is created, to serve as input for the subsequent modeling phase. Often, alternations take place between the modeling and data preparation phase. The steps described are selecting data, cleaning data and constructing data

3. **Modeling:** This phase depends on the business objectives defined earlier, and lead to certain properties of the model. Six essential properties to evaluate are: performance

metric, robustness, explainability, scalability, resource demand and model complexity. During this phase, one or multiple models are created. Potentially, existing literature on similar applications can be consulted. Depending on the use case, this phase also consists of the training of the model. Lastly, reproducibility is considered during this phase, where a distinction between method reproducibility and result reproducibility is considered.

4. **Evaluation:** Describes how to validate the performance, including best practices around test-set construction. Here, the framework also emphasizes the importance of evaluating the robustness and explainability of the model and explains how this can be done. Robustness is important to guarantee that the model still functions when data is perturbed. Explainability helps end-user and practitioner in finding errors and could potentially enable further model improvement.

5. **Deployment:** During this phase, the model is employed in a practical setting. The first step here, is to define the inference hardware. Also, it is important to monitor the model performance in a practical setting, where data might be new or deviating from what the model has been trained on. May it be, that the model under-performs, a fallback plan can mitigate adverse effects. Lastly, if all runs well, it may still be that usage is underwhelming, this can be prevented by starting with a PoC before creating the final product.

6. **Monitoring and Maintenance:** Machine learning applications are used over a long period and therefore have a life cycle which has to be managed. Over time, the model performance might degrade, due to an increasing deviation of input data from the data the model had been trained on. By monitoring whether the input data remains between expected thresholds, degradation of the model can be detected early. Retraining or fine-tuning the model can help in regaining the expected model performance.

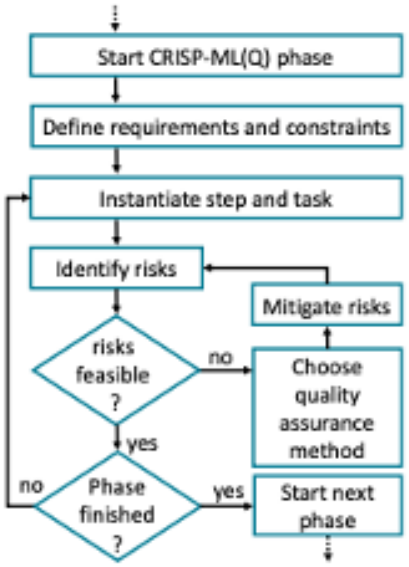

Figure 3.1: Publications per year in selected literature. Source: Studer et al. [3]

This framework prescribes to, for every phase, define requirements, constraints and identify risk and challenges related to the phase, see Fig 3.1. CRISP-ML emphasizes that the phases are iterative, and back- and forth movement between phases is essential in machine-learning projects. Besides describing six phases, CRISP-ML(Q) created a quality-assurance framework.

Lastly, CRISP-ML prescribes several traits of a model which, next to the performance of the model, need to be assessed, namely the robustness, scalability, explainability, model complexity and the resource demand. Here, the robustness means that the model performance is resilient to varying inputs. Scalability implies that the model can handle increasing volumes of data. Explainability is the interpretability of the reasoning of the model, using post hoc methods or directly, e.g. by looking at the parameters of the model. The complexity of the models should be suitable for the complexity of the input data. Lastly, it should be evaluated whether the resource demand matches the availability in the business.

3.1.2. DESIGN SCIENCE RESEARCH

In core component in Design Science Research, is the creation of an artifact with the goal of solving an important business problem which has not been solved before. The artifact should be created based on existing theories and knowledge. Finally, the artifact should be extensively evaluated Hevner et al. [77]. To help researchers execute DSR, [78] created a mental model using consensus building, based on previous research in the field. The mental model consists of six steps:

1. **Problem identification and motivation**
   In this step, the problem and the value of a solution are defined. Here, it is also vital to show the importance of the research to motivate stakeholders along the road of creating

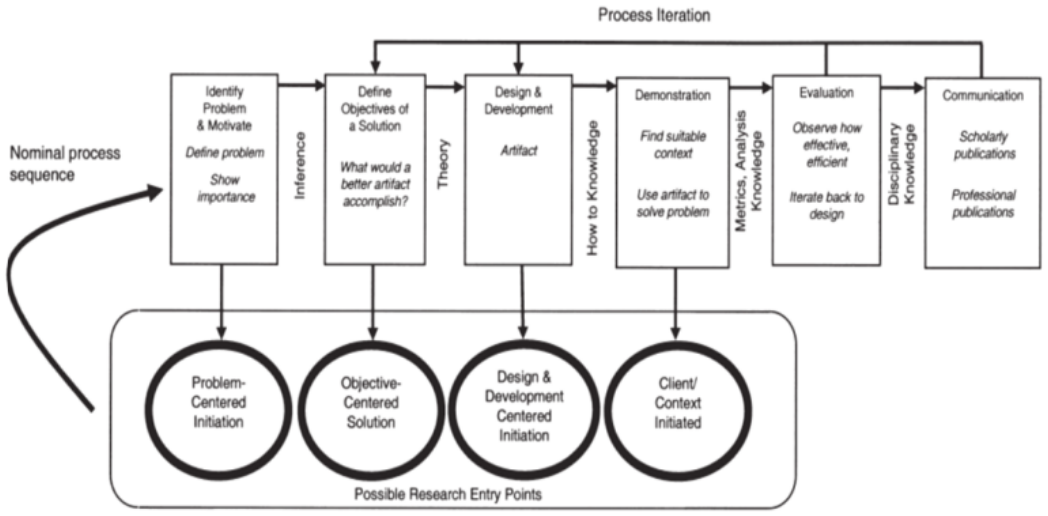

Figure 3.2: Design Science Research Methodology. Source: Vom Brocke et al. [4]

the artifact.

2. **Define the objectives for a solution**
   This step realizes objectives which should be obtained with creating the artifact. The problem definition serves as input for the objectives.

3. **Design and development**
   In this step, the artifact is designed and created. Artifacts can be anything in which a research contribution is implemented in the design, such as constructs, models, methods or new properties of certain resources.

4. **Demonstration**
   In the demonstration step, the artifact should be employed to solve the problem defined in step 1. Several methods can be used for this, such as experimentation, simulation or proof.

5. **Evaluation**
   By comparing the objectives to the results of employing the artifact in the demonstration, the suitability of the artifact for solving the problem is evaluated.

6. **Communication**
   Finally, the importance of the problem, and the suitability of the solution, should be communicated to the relevant audiences, in the relevant format.

The DSR process does not necessarily need to start in the first step. Instead, various types of study are characterized by starting at another step Peffers et al. [78], as shown in Figure 3.2.

3.2. ANALYTICAL METHODS

Various studies make use of benchmarks to assess the capabilities of the Document AI models. Frequently used benchmarks are among others: SlideVQA, DocVQA and DUDE, which aim to assess the models understanding of documents containing slides, images and charts. There are also benchmarks that aim to assess more specific capabilities of the model, such as the Kleister Charity benchmarks, which consists of 2788 financial reports, testing the model's financial understanding.

3.2.1. MEASURING PERFORMANCE

Various benchmarks apply automatic evaluation methods. Reasons for choosing an automatic evaluation metric, are an increased consistency and a reduced need for human resources. How this works, is that the benchmark consists of a set of questions and the desired answer, and uses the automatic evaluation metrics to assess the correctness of the answer [79].

Automatic analysis can be hindered by the generative nature of LLMs, which are the core of the majority of current state-of-the-art Document AI methods. The generated answers may give a correct answer in many variations, making the automatic evaluation more difficult [31, 68, 73]. Hardcoded rules, such as `model_answer == golden_answer`, also referred to as Exact Match (EM), result often in an overly sensitive evaluation metric for analyzing LLM outputs. Although various studies still apply this metric, others choose to soften the rules for evaluation by applying a Constrained Exact Match (CEM). The CEM allows the predicted answer to be a substring of the golden answer. For example, 'Messi' would be an exact restricted match when the golden answer is 'Lionel Messi'. The EM and CEM metrics work when answers are shorter; however, when answers are longer—which is often the case in generative methods—they are too rigid [80]. Therefore, instead of EM and CM, various studies used the Average Normalized Levenshtein Similarity (ANLS), which does not award zero points for a slight difference between the model response and the golden standard Mathew et al. [30]. The ANLS provides a similarity score if the similarity is above a certain threshold, and zero if it is below the threshold.

However, also the ANLS can be a suboptimal evaluator for certain use cases, since the ANLS especially works for answers one two maximum a few sentences, though, when a longer response is required, the ANLS is not a good fit. Peer et al. [81] found for example, that sentences that were semantically similar but rephrased were obtaining a low similarity score, while essentially meaning the same. Various other formulaic methods were tested, such as BLEU, ROUGE and METEOR, but all performed poorly [82].

Thus, more advanced methods are required. Other studies employed the BERT-score for comparing human annotations and model outputs, as it was shown to better align with human judgment [83]. Utilizing the current zero-shot capabilities on various downstream tasks, Deng et al. [84] created the Generalized Accuracy (G-Acc) and Generalized F1-Score (G-F1) metrics, which uses ChatGPT to evaluate the answer compared to the golden standard, based on predefined rules. This LLM-based evaluation method is better able to grasp semantic similarities,

while also being able to detect fine-grained differences between the response and golden standard, leading to better alignment with human judgment compared to the previous methods such as the BERT-score and ROUGE [85].

However, using the proprietary method of ChatGPT has several disadvantages. Firstly, proprietary models have a lack of transparency, while being for-profit. This is undesired, since the evaluation methods play an important role for the development of models and science. Widespread use of proprietary methods for evaluation might lead to undesired power given to the owners of those models, while it is unclear how they influence the results of models [86]. Furthermore, there is little to no version control in proprietary methods, which hinders the reproducibility of studies. Lastly, when evaluation is also required for training, the pay-per-use nature of proprietary methods can quickly lead to high costs [86].

To solve the problems related to the evaluation based on LLM, Prometheus (2) was created. Prometheus is open-source, and therefore enables research to do version control. Furthermore, among all open-source and baseline models, Prometheus obtains the highest alignment with human judgment, and is therefore highly suitable for automatic evaluation.

Prometheus can be downloaded via Huggingface. The model preferably is controlled by a formatted prompt, containing an instruction, response(s), reference answer (optional) and a user-defined evaluation criterion. As a response, the model returns an integer from 1 to 5 and an explanation for the choice, where the integer is based on the criteria defined in the input prompt.

4

EXPERIMENTAL SET-UP

The goal of this study is to build a document AI pipeline that enables practitioners to extract information from sustainability reports quickly and accurately. Following the approach of CRISP-ML(Q) and DSR (see Chapter 3), this study goes through six phases (see Fig 4.1). In detail, our approach is to assess the performance of existing methods on sustainability reports, by comparing existing Document AI architectures, to then select the best performing model and, based on previous literature on Document AI, propose an improvement to mitigate the shortcomings of previous literature. Finally, the goal is to create a tool that adheres to the business requirements for using document AI for QA on sustainability reports in the context of benchmarking. Here, benchmarking implies that organizations are compared, based on their disclosed information in their sustainability reports. For this, accurate extraction is required, since a benchmarking study can be the input of major business decisions, such a decision of which organization should win a tender.

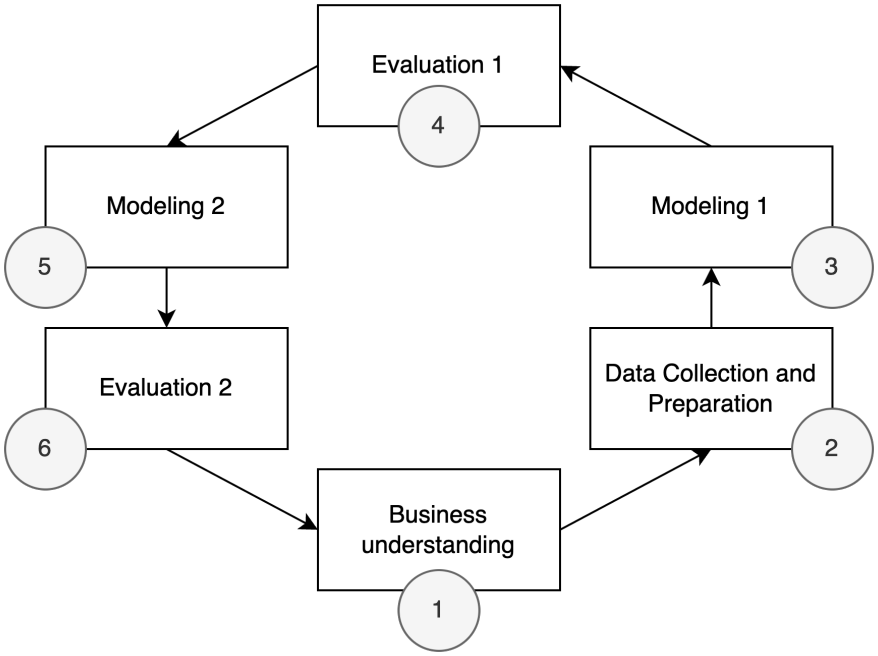

Figure 4.1: The order of phases during this study. The numbers in the circles indicate the phase numbers.

4.1. PHASE 1: BUSINESS UNDERSTANDING

As a primary step of this study, we interview four professionals in the field of sustainability reporting to determine success criteria for a Document AI method for information extraction from sustainability reports, from a business, a ML and economic perspective, following the Crisp-ML (Q) methodology by Studer et al. [3]. The experts involved in the study are all occupied with the new CSRD regulations or sustainability reporting / benchmarking, though come from different backgrounds, namely: risk, business consulting, climate, and accounting.

Our discussions with experts resulted in quantified success criteria (see Section 5.1) and the following requirements: (1) to directly extract information, (2) to assess whether certain information is disclosed in the report, and (3) to realize a narrative or summary about the processes of the subject of the report related to ESG.

4.2. PHASE 2: DATA COLLECTION AND PREPARATION

To analyze the performance on the required tasks specified in Section 4.1 we initially searched for existing annotated data on sustainability reports related to those tasks. For finding and selecting a dataset, the requirements are that the subject report is multipage, that the annotated data is about information extraction, classification, or summarizing, and, lastly, that the location of the evidence is annotated. The datasets used in our study are summarized in Table 4.1.

Once we found the datasets, we transformed them so that they are all formatted in a standardized format, which means that they have the same columns and data types. This enables us to easily create datasets and analyze the results collectively. The columns are:

- Document: This is the document name. Depending on the model, this refers to a PDF, or a folder of images. `M3DOCRAG`, for example, expects a PDF, while `SV-RAG` expects images of the pages.
- Question: This is the complete query to the model, including potential prompt engineering. However, the system prompt, which comes from the initial model pipelines, is not included here.
- Retrieval_response: In this field, the model stores its top-k retrieval indexes and related scores, where k is indicated by the user.
- QA_response: The response of the model to the user query.
- Page: This consists of one or more golden standard answers.
- Golden: This is the golden answer.

As evaluation data, we used two publicly available datasets and two datasets created by EY. The publicly available datasets we refer to as `Scope123` [1] and ClimRetrieve [2]. `Scope123` is a dataset that contains manually annotated scope emissions in sustainability reports. This dataset is a composition of sustainability reports from a diverse range of locations, to ensure generalizability.

To prepare this dataset, we select the columns `emission_year`, `scope_1`, `scope_2_market`, `scope_2_location`, `url` (which we use as a unique reference) and the pages where those emissions can be found. Initially, this dataset did not contain queries aimed at extracting the topics. Therefore, we created the query dynamically, using a rule-based Python function, as can be seen in Figure 4.2. To guide the model in providing the correct responses, we apply a one-shot in-context learning approach. Lastly, we did not use all rows in the dataset and only selected the first 25.

[1] https://huggingface.co/datasets/nopperl/corporate-emission-reports/discussions
[2] https://github.com/tobischimanski/climretrieve

```python
def create_prompt(description, emission_year, target):
    message= f"""
    {description}

    Your task is to extract from the report the {target} emissions for the year {emission_year}.

    Respond by only giving the value in metric tonnes CO2eq.

    So for example if the report says:
    "In 2021, our scope 1 emissions were 1000 metric tonnes CO2eq."
    You should respond with:
    1000
    If the report says:
    "In 2021, our scope 1 emissions were 5 kilotonnes of CO2eq."
    You should respond with:
    5000

    The {target} emissions for the year {emission_year} are:
    """
    return message
```

Figure 4.2: Prompt for generating the prompts

The second dataset, called `ClimRetrieve`, consists of 'Yes' or 'No' questions on whether certain information is disclosed in the document. An example question from this dataset is: 'Does the company identify any impacts of its business activities on the environment?' So, the questions employ the model for a binary classification task, which is to determine whether certain information related to the `TCFD` is present in the report. Instead of focusing on `ESG`, this dataset focuses on questions related to the `TCFD`. This is a different topic; however, we assume that it indicates the potential of the model in the task of answering questions in sustainability reports, since there is a broad overlap between the `TCFD` and the environmental pillar of the `CSRD`.

The objective of the model is to binary classify whether or not the information is present. The studies were selected so that the number of golden answers that were 'No' is roughly equal to the golden answers that were 'Yes'. This reduces the likelihood that the model coincidentally obtains a good result by always predicting the same result. By randomly dropping entries from the overrepresented "yes" column, our final ClimRetrieve dataset resulted in having 58 entries, of which, in 31 cases, the information was present in the document.

Furthermore, the `ClimRetrieve` dataset consists of questions, answers, an indication of the page where information can be found, and also the information itself based on which the conclusive answer to the question can be given. Therefore, the dataset frequently contains the same question multiple times, albeit with a different piece of indicative text. For evaluation of our model, this is not desired, since the model is going to predict the same thing for all the different instances. Therefore, we merged those cases so that per document, there is only one question-answer pair. The pages are concatenated so that there is a list of pages on which the indicative text can be found. This transformation might benefit certain metrics, such as the first relevant page index. However, the Recall@k should be largely unaffected (see Section 4.4).

This dataset consisted solely of question-answer pairs. Our literature review highlights that prompt engineering could benefit the capabilities of the model (see 2.1.3). Despite M3DOCRAG not explicitly mentioning the application of prompt engineering, we tested various prompts to obtain a maximum performance on each dataset. Again, the system prompt as given by the available code of the methods remains unchanged.

The `Scope123` and `ClimRetrieve` datasets are the only publicly accessible datasets that fit our study. Unfortunately, both datasets fall short in the sense that only a part of the topics about which sustainability experts consult the document are covered by the datasets. `Scope123` focuses on emissions from Scopes 1, 2, and 3, but does not cover topics such as sustainable assets under management, the gender pay gap, and the percentage of women in top positions. Also, ClimRetrieve aligns well with the environment pillar in CSRD, however, the social and governance pillars are not represented in these data. Thus, to ensure sufficient coverage of the domain tasks which the model should be able to perform, more data is needed.

For this, we obtained a dataset from a sustainability-related department within EY, focused on non-financial accounting. The dataset consists of manually extracted data from 15 European banks. This data extraction was performed with high precision, where all the information extracted is cross-checked by various evaluators. From this dataset, we create two subsets. The first one, we refer to as EY Classifications, has the goal of evaluating model performance on classifying whether the topic the question asks about is present in the text or not. The other dataset, we call EY Quantifications, is aimed at IE, and thus the model task is to return a piece of information in the document, based on the question.

The classification dataset came mainly from a section that asks about disclosures related to ESRS in the EY dataset. An example question is: "Does the institution disclose information on its own workforce (related to S1)?" For the information extraction (EY Quantification) dataset, we created the prompts ourselves based on the initial topic specifications. This resulted in prompts of the following format: "What were the total CO_2 emissions in 2021, 2022 and 2023, if disclosed? Answer the question as in the following example: '2021: not disclosed, 2022: 34 ktons CO_2, 2023: 10 Mt CO_2, use the unit which is also used in the document.'. Here, we ask the model to return what is indicated for 2021, 2022 and 2023, which resembles the way the manual extractions in this dataset were done, namely: what was reported was extracted and added to the dataset.

Initially, the EY datasets were mostly formatted as topic-answer pairs. Therefore, to create a suitable input for the model, we manually created queries for the dataset. In addition, we again made a sub-selection of the data. Initially, the dataset consisted of extractions from 15 reports and more than 60 questions per report. Testing on all those questions would require a too long evaluation time; therefore, we selected the first five reports in the excel sheet (see Appendix E for an example prompt per dataset).

Table 4.1: Overview of task datasets and corresponding datapoint estimations. Q = Questions, R = Reports

Dataset	**What**	Pages	**Task description**	**Datapoints (Q×R)**
Scope123	Information Extraction	67 (Sustainability report)	Extract the scope emissions.	4×25 = 100
ClimRetrieve	Themes classification (disclosure of...)	141 (Sustainability report)	Yes/no answering.	12 × 30 = 360
EY Quantitatives	Information Extraction	426 (Full annual report)	Answering varying questions about all kinds of topic, related to E, S and G pillors of Sustainability Reporting	10×5 = 50
EY Classifications	Themes classification (disclosure of...)	426 (Full annual report)	Answering questions about whether certain information related, or about the ESRS is disclosed in the report. (or in the pages)	10×5 = 50

4.3. PHASE 3: MODELING 1

During this phase, we test existing models of which the code is publicly available from the previous literature on sustainability reports. We start this phase by making a selection of the models that could be built in the limited time frame of this study. A prerequisite for this is that the model code and weights are available, to maintain a manageable project in the given time-span.

The models we test are:

- M3DocRAG [69]
- SV-RAG-InternVL2 [64]
- mPlugDocOwl2 [87]
- SelfAttnVQA - scoring module only [63]

All tests were performed on the `NVIDIA a40` GPU on a High-Performance Computing `SLURM` cluster. For evaluation, we use a csv file consisting of questions, the right document, and the retrieval and answer fields that must be filled in by the model. For evaluation, the `csv` file was downloaded to a local computer to analyze it using a `Jupyter Notebook` in `Visual Studio Code`.

For `M3DOCRAG`, we use an unofficial implementation by Omar Alsaabi [3]. This, because at the time of selecting the models, the code of `M3DOCRAG` was, to our knowledge, not publicly available. In contrast to Alsaabi, who uses the quantized variant for question answering, we use the original `Qwen2-VL-7B` model for question answering. `M3DOCRAG` has a modular design, in which the retrieval module and the question answering module can be replaced. We follow their implementation of the paper, using `Colpali` as retriever and `Qwen2-VL-Instruct-GPT-Int4`, as a QA module.

For both `SV-RAG-InternVL2` and `MPlugDocOwl`, the original model code and weights were used. In contrast, for `SelfAttn-VQA`, we rewrote/added code to fetch the top-k indexes, which

[3] https://github.com/Omaralsaabi/M3DOCRAG

was not outputted by any of their shared scripts out-of-the-box. Most of the input for our program comes from their shared script called `train.py`.

4.4. PHASE 4: EVALUATION 1

During this first evaluation phase, the goal is to obtain the best performing model in terms of answer accuracy and page predictions to, in the second evaluation phase (Section 4.6), use this pipeline to create a model which suits the application of document AI for VQA on sustainability reports.

4.4.1. PAGE-RETRIEVAL

For all QA tasks, we evaluate the accuracy of the page retrieval. This is done in two ways: (1) by observing the proportion of relevant pages retrieved, also known as `Recall@k`, and (2) by analyzing the index of the first relevant page. The calculation of `Recall@k` is formally defined in Equation 4.1.

- Q: the set of queries (questions)
- For each query $q \in Q$:
  - G_q: the set of relevant documents (gold standard) for query q
  - R_q^k: the set of top-k retrieved documents by the model for query q

$$\text{Recall@k} = \frac{1}{|Q|} \sum_{q \in Q} \frac{|G_q \cap R_q^k|}{|G_q|} \tag{4.1}$$

We analyze the `recall@k` for $k = 5, 10, 20$ and 30. Here, we increase the k to 30, as we expect this to be a feasible size for the current state of language models to extract information from documents, including sustainability reports [88] .

In addition, we analyze the first relevant page index, which is the main component of another commonly used metric to evaluate page-retrieval performance called the `Mean Reciprocal Rank` (`MRR`) [89]. Using the first relevant page index, we calculate how many pages need to be retrieved, so that in `x`% of the cases, the right page is added. If the business requires an accuracy of 97%, this means, for example, that the number of pages fetched should be equal to or more than the 97% quantile. For this calculation, we disregard the option that the model can also incorrectly answer based on the retrieved pages.

4.4.2. QUESTION ANSWERING CAPABILITY ANALYSIS

We fully automate model output evaluations to ensure finalization of the study within the limited time frame. Following previous studies, we use LLMs for grading the model output in the EY quantifications and the `Scope123` datasets, which aim to evaluate the IE performance of the models. Since `GPT-4o` outperforms `Prometheus 2` [86], we choose to use `GPT-4o` to ob-

tain the G-Acc of our models [90, 91]. For the binary classification datasets ClimRetrieve and EY Classifications, we use a rule-based algorithm to automatically evaluate the model responses, by only focusing on the 'yes' or 'no' in the model response given when answering questions in the datasets.

For the EY Quantifications and Scope123 dataset, the goal is mainly to extract certain pieces of information directly from the text. However, the model can make mistakes here, such as partially answering, answering in the wrong format, or answering in the wrong units. As prescribed in the previous literature on the use of LLMs for the evaluation of model outputs, we provide clear instructions on the criteria for a certain score (see Figure 4.3) [91]. We use a Python script to extract the scores returned by the evaluator and to calculate the G-Acc (see 3.2.1). In the EY Quantifications dataset, the model response often consists of three parts, each focusing on 2021, 2022 or 2023. Therefore, to calculate the final G-Acc, the evaluator model returns a score for each year in the golden answer. Then, the average G-Acc score is taken. Lastly, the idea of the G-Acc score is to allow more flexibility in the response of the model. However, an answer which is wrong should not be rewarded. Thus, we assign 0 points for a score below 4, and 1 point for a score equal to or greater than 4, allowing for some flexibility.

Annotation Instructions

Your evaluating the output of a GenAI model and comparing it to a golden standard answer.

It is important to precizely look at the expected answer, and to check whether the answer is correctly providing the answer. The answer may be formulated differently, but the information in the golden standard answer should be present.

The golden standard answer will contain disclosed information for the different years. For each of the years, indicate whether the information was correctly provided.

Below is the score scheme:

- **Fully correct** (The essence of the answer is what it should be, including the correct unit size and the correct number, but more information may be given by the model) = 5
- **Almost correct** (There essence of the answer is almost correct, though there are minor inconsistencies, such as a small numerical mistake, or unit mistake.) = 4
- **Partially correct** (The model gave a the (almost correct) answer, though also provides wrong information) = 3
- **Incorrect, but in the right format** (the answer is wrong, but the model provided the right format, indicating understanding of the assignment) = 2
- **Completely incorrect** (the model gave an unrelated answer, not showing understanding of the assignment, nor the context) = 1

Figure 4.3: Scoring instructions for GenAI model evaluation task.

4.5. PHASE 5: MODELING 2

Based on the results in modeling phase 1, we select the best performing model on our selected datasets to improve its performance. Since retrieval appears to be the main bottleneck after evaluating our results of the modeling phase 1, our selection of the baseline model for modeling phase 2 is based on the retrieval performance only. During modeling phase 2, our goal is, to improve this retrieval performance, as we hypothesize that this will have the largest effect on the resulting performance of the QA-pipeline. The best performing architecture of modeling phase 1 is M3DOCRAG, which will therefore serve as the basis for modeling phase 2.

M3DOCRAG uses ColPali as its retriever. The main idea between ColPali is Contextual Late Interaction, which was first implemented in the paper of ColBERT [92]. In their implementation, this means that the document images are divided into patches, which are all embedded, instead of the pages as a whole. Then, for all query tokens, the maximum similarity is selected with one of the patch tokens and summed to obtain a retrieval score for the page. Then, based on the indicated `k`, a number of pages is retrieved.

$$\text{Score}(q,d) = \sum_{i=1}^{|q|} \max_j \text{sim}(q_i, d_j) \tag{4.2}$$

Here, q_i is the embedding of the i-th query token, d_j is the embedding of the j-th document patch token, and $\text{sim}(\cdot,\cdot)$ refers to the dot product [68].

Since embedding all document patches would be a lengthy process, M3DocRAG was implemented so that the image patches of the document pages only need to be embedded once. Then, when the user asks a query, only the query tokens need to be embedded, to enable the calculation of the similarity score. This, combined with the storage of them in an FAISS database, significantly speeds up the process.

4.5.1. FINE-TUNING

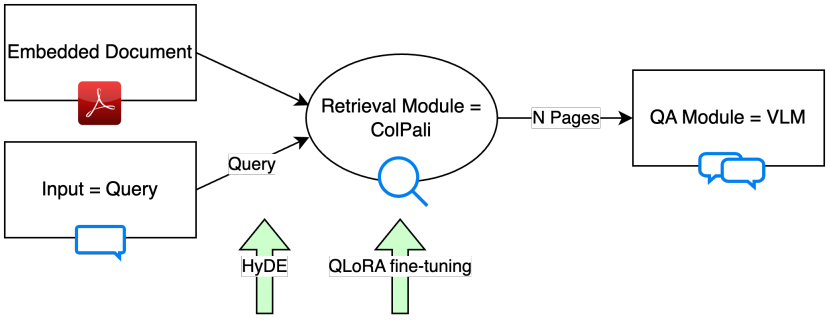

Figure 4.4: Diagram of tested solution, here the VLMs in the QA Module are either Qwen2-VL, or GPT-4o

Fine-tuning implies that a model is not trained from scratch, but a previous checkpoint with parameter settings is reused and adjusted, based on more data. Fine-tuning has shown to be effective in various ML Architectures, including that of Transformers and Visual Language Mod-

els, as Section 2.1 points out.

Fine-tuning VLMs comes with substantial resource requirements. In contrast to when a model is solely used for inference, training a transformer model requires its gradients to be stored, causing significant overhead and memory requirements. To mitigate this overhead problem, parameter-efficient fine-tuning (PEFT) methods such as LoRA [76] and QLoRA are created [93].

Colpali uses PaliGemma as the base model, for which they create an adapter for the fine-tuning tasks. We reuse the pre-trained "vidore/colpaligemma-3b-mix-448-base" as our base_model, and also use the pre-trained adapter created by the authors of Colpali, named "vidore/colpali", as our starting point for fine-tuning. To enhance the performance of M3DOCRAG on QA based on sustainability reports, we apply LoRA fine-tuning [76], and hypothetical document embeddings (HyDE) [94]. The architecture of our solution is presented in Figure 4.4.

During fine-tuning ColPali, we use QLoRA, which applies a combination of quantization and LoRA. Quantization implies that the precision of the weights is altered by removing a large part of the decimals. Typically, this results in Int4 or Int8 quantizations, significantly reducing the amount of data that needs to be stored during training. However, this also could come at the cost of inference speed, and accuracy of the model. LoRA implies that instead of fine-tuning, and adjusting all the weights of the pre-trained model, two low-rank matrices are multiplied to get an approximation of the original weight matrix. The rank here is configurable and determines the memory, speed, and effectiveness of the fine-tuning. We do not modify the rank, and reused the rank as used in 'vidore/colpali' itself.

The objective of fine-tuning is to equip the model with the required knowledge about sustainability reporting. Because of the minor performance on the EY Classifications and EY Quantifications, we aim to add knowledge about the ESRS for the former and CSRD for the latter, which are the main subjects in the questions in those two datasets. We test different settings for the quantization and the number of training documents outside of the hyperparameter tuning configuration. In this way, we expand our search space iteratively, depending on the results.

TRAINING DATA

As training data, we used only reports that implement the new CSRD regulations, obtained from a collection collected created by Key ESG [4]. Here, we select reports from the financial sector as this is more closely aligned with the data used by the stakeholders within EY and the final evaluation data. The list of reports used for fine-tuning is available in Appendix A. Since we want to equip the models with knowledge about sustainability reporting, we shorten the downloaded reports so that they only consist of pages from the sustainability sections. Since the EY datasets ask questions about the full annual report, and not just about the sustainability reports, this means that the training data differs from the evaluation data. We find that this does not negatively affect the performance when fine-tuning ColPali.

We let Qwen2-VL annotate the data, following the approach of ColPali, which was to use Claude

[4] https://www.keyesg.com/article/access-the-first-wave-of-csrd-reports

for the generation of the queries [68]. We evaluate a subset of the queries, to iteratively change the prompt, until satisfying annotations were obtained. Appendix B shows the final prompt. To ensure that the model also focuses on other modalities, the query explicitly asks Qwen2-VL to return, per page, a query about a table, a figure, and from text.

Once Qwen2-VL returns queries, each pair of questions and answers is loaded as a row in the database, with a third column representing the subject page of the original document. The data is then split into a training and a testing dataset using `train_test_split` from `model_selection` module from `sklearn`, using a test_size of 0.1, and applying a shuffle. Finally, the data are wrapped into `datasets.DatasetDict` and then stored as parquet.

Fine-tuning is performed on a SLURM cluster, on a single NVIDIA a40 GPU, having a VRAM of 48GB. This allowed training of the ColPali LoRA-adapter with 4-bit, 8-bit, or no quantization. To obtain the optimal hyperparameter tuning settings, the Weights & Biases package was used [5]. This package makes the setup of testing agents possible while requiring minimal implementation configurations. For the ranges of hyperparameters, we utilized an exemplary notebook created by the ColPali authors. Extensive analysis key aspects during fine-tuning can be analyzed using this package, such as the contribution of the configurations to model performances and the GPU usage. For hyperparameter tuning, we use Bayes optimization and at least run 10 variations of the models before selecting the best.

Fine-tuning of the model is done in multiple iterations. After each run, we test the best model of the run on our selected datasets for evaluation (see Figure 4.1. We choose not to train on the evaluation sets to ensure that there is no information leakage. The tunable hyperparameters are given in Figure 4.5.

Hyperparameter Settings

- **Num_train_epochs** = [1, 2, 3, 4]
  Total number of times the entire dataset is used during training.
- **Learning_rate** = $\in$ (0.00001, 0.0001)
- **Gradient_accumulation_steps** = [4, 6, 8]
  Number of steps to accumulate gradients before performing a backward/update pass.
- **Weight decay** $\in$ (0.01, 0.1)
  A regularization parameter that penalizes large weights.
- **Early_stop_patience** = [2, 3, 4]
  Number of validation checks to wait before early stopping.

Figure 4.5: Tunable hyperparameters in Weights & Biases

Furthermore, we test variations of the quantization and training data. Here, we distinguish between int4, int8, or no quantization, and between a dataset containing 5 or 10 sustainability parts of reports. If a gradual increase is shown in performance, this can be an indication to

[5] https://wandb.ai/site/

further fine-tune on more data.

4.5.2. HYPOTHETICAL SEARCH

Hypothetical Document Embedding (HyDE), is an information retrieval technique that uses LLMs to generate a passage of text based on the query, in such a way that it is likely to find this text in the document [94]. Then, this generated text can be used for similarity search, instead of only the query. HyDE has been shown to often significantly increase retrieval accuracies with minimal effort link.

To realize HyDE, we chose to use GPT-4o because it has a later cut-off (latest creation date of the data) date than Qwen-2-VL-7B, which we use Qwen2-VL-7B for most other tasks. The cut-off date for Qwen2-VL is June 2023. For GPT-4o, this date is June 2024, and thus it is likely to be more informed on CSRD, ESRS or the EU Taxonomy. For generating the hypothetical documents, we loop through our dataset and give a system instruction to GPT-4o, accompanied by the initial state of the query (see Figure 4.6).

HyDE Prompt

You are an expert in the sustainability domain.
Your goal is to provide a passage which could answer the question:
'{question}'
The passage should be formatted as if it were part of a sustainability report.
It should be between 100 and 200 words long, and must be written in a professional tone.
If you think all is said about the topic, but you have fewer than 100 words, add something related.

Figure 4.6: HyDE prompt

4.6. PHASE 6: EVALUATION 2

The objective of evaluation phase 2 is to obtain insight into whether our improved retrieval module leads to sufficient performance increase so that our solution can be used in the described business setting of sustainability benchmarking. Here, the focus is mainly on retrieval performance. During this evaluation phase, we compare the results against the baseline, which we instantiate based on results from the metrics of the best model tested in evaluation round 1. For the retrieval performance analysis, we use the same method as in evaluation phase 1.

To gain an understanding of the performance that is required from a retrieval module, we also analyze the QA performance of Qwen2-VL-7B and GPT-4o, which has much more parameters, and is therefore likely to outperform Qwen2-VL. This comparison is valuable, because this could give incentives for future directions, for example, to filter the document locally, though then use a proprietary MLLM to do the QA.

In addition, we evaluate the study results with a focus group of seven experts from an accounting team within EY that was focused on sustainability benchmarking. The participants have

different roles. Most of them are consultant, but also a manager and partner attended the focusgroup. We apply an ex-vivo testing strategy, by presenting KPIs and qualitative examples to the users, as described by Bertolino et al. [95]. The goal of this evaluation is to assess whether the model adheres to the initially set business requirements and to identify next steps for the technique of document AI in the business setting of Sustainability Benchmarking. During this focus group, we discuss key performance indicators (KPIs) along with qualitative extraction examples. Based on the results, an open discussion followed.

Lastly, we assess some crucial ML Criteria during this evaluation phase to determine the suitability of the resulting architecture for deployment. Those criteria are the following [3]:

- Ideally, the tool would be able to explain itself how it came to its conclusions. The best way to explain this case is to make a quote + a page number. In this way, it is also very easy to cross-check.
- Also, the tool should be reliable. If the tool is hallucinating and it is difficult to distinguish whether certain answers are true or not, the tool could have a large negative impact on the business. Instead, a reliable answer to the questions and an indication if it cannot is required.
- Finally, for deployment in a business setting, observing fairness and robustness is crucial. By observing fairness, it can be ensured that the geological location, race, language, and other sensitive traits of the report do not influence the extraction quality or trend of the report.

4.6.1. SUMMARY: EXPERIMENTAL SET-UP

During our study, we:

- Define business requirements together with experts in the field of sustainability reporting.
- Compare state-of-the-art Document AI methods for IE from sustainability reports.
- Observe retrieval as the main bottleneck for IE from sustainability reports using Document AI on lengthy documents and therefore select the model with the best retrieval performance as a baseline, with the objective of improving this retrieval performance.
- Apply quantization and LoRA combined as QLoRA for fine-tuning the selected Document AI architecture, and further utilize HyDE to increase retrieval performance.

5

RESULTS AND DISCUSSION

This section presents and discusses the results collected in evaluation phases 1 and 2. The models evaluated in the first phase are existing Document AI methods applied to sustainability reports. The models tested in the second phase are variations of M3DocRAG, which have been improved using QLoRA fine-tuning or HyDE. Finally, in the discussion, we discuss the managerial and theoretical implications of this work.

5.1. RESULTS

We start by explaining the requirements in the business setting of performing a sustainability benchmark analysis, based on the meetings we had with sustainability experts within EY. Then, we discuss the evaluation of modeling phase 1, where we tested the best state-of-the-art methods for QA on multipage documents. evaluation phase 1 revealed several shortcomings of the current state-of-the-art, which we address in modeling phase 2. Using fine-tuning and hypothetical document embeddings, we aim to reduce those shortcomings. The degree of success in obtaining this objective is evaluated in evaluation phase 2. Finally, we discuss the results of the focus group validation.

Firstly, the success criteria that resulted from interviews conducted in the first phase of this study, with four experts in EY working in the sustainability field. Firstly, the solution should save at least 30% of the time currently spent extracting information from sustainability reports. If the reduction is lower, it is unlikely that the tool will be adopted, as this probably indicates a insignificant added value. A baseline, we take, is that it takes a team of five sustainability reporting experts 40 hours to make a benchmark comparison of 15 reports.

Furthermore, a localization of the evidence for the model response is required. Especially because the business extracts from sustainability reports must have an accuracy of more than 99%, which seems to be infeasible considering the performance on benchmarks such as MP-

DocVQA, which often revolve around 80%. Therefore, it is crucial that the resulting solution allows traceability of the evidence passages in the document, so that, even though the model extractions are not perfect, experts can be assisted in finding the right information.

5.1.1. EVALUATION PHASE 1: BENCHMARKING THE STATE-OF-THE-ART

We mainly evaluate the following models: Self-Attention, SVRAG, and M3DocRAG. We stopped early with evaluating mPlug-DocOwl2 after finding that it ran into out-of-memory errors as soon as the number of pages increased above 10. Although we tested various combinations of variations of the M3DocRAG architecture, we chose to report on the original implementations for this report for this evaluation phase, and maintain the parameters in the models the same as obtained in the available code. For M3DocRAG, this means that an Inverted File Index is activated, which could lead to slightly lower results compared to having it deactivated. Also, for QA, in this phase, M3DocRAG consisted of a quantized variant of Qwen2-VL.

Table 5.1: Average Recall@5, taken over all datasets

Architecture	**Retrieval**	**Avg. Recall@5**
M3DocRAG	vidore/colpali	0.37594
SVRAG	Custom	0.05424
Self-Attention	Custom	0.05188

Table 5.1 presents the average Recall@5 of the model in all data sets. The table shows that M3DocRAG performs best, with an average Recall@5 of 37.6%. SVRAG and Self-Attention underperformed, with a Recall@5 of 5.4% and 5.2%, respectively. The performance per dataset varies. Table 5.2 shows that the Recall@5 results in the Scope123 dataset were all above 80%, while the retrieval models do not obtain a better score than 20% in the ClimRetrieve, Quantifications and Classifications data sets.

Table 5.2: Recall@5 performance per architecture and dataset, with the mean number of pages per file

Dataset	**Architecture**	**Recall@5**	**Mean # Pages**
climRetrieve	Self-Attention	0.160638	67.38
	SVRAG	0.150051	
	M3DocRAG	0.118642	
Scope123	M3DocRAG	0.815789	141.00
	SVRAG	0.046053	
	Self-Attention	0.046053	
Quantifications	M3DocRAG	0.120690	425.80
	SVRAG	0.000000	
	Self-Attention	0.000000	
Classifications	M3DocRAG	0.171875	425.80
	SVRAG	0.031250	
	Self-Attention	0.000000	

Despite retrieval results seem to be insufficient due to low presence of relevant pages in the model response, state-of-the-art models have shown to be able to answer accurately on docu-

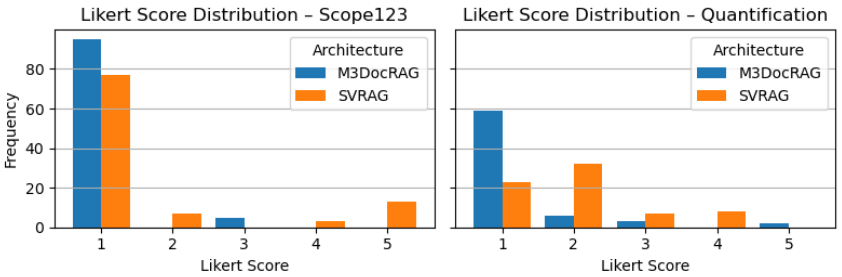

Figure 5.1: Likert-scores when analyzing the question answering performance on the Scope123 and Quantifications Datasets.

ments of 20+ pages, and thus could supplying more pages to the model, so that a relevant page is included, be a mitigation to this shortfall. Although we did not report Recall@k for larger k-values in this evaluation phase, we see in evaluation phase two that Recall@k increases when k increases.

In addition, we evaluate the performance of the QA. We analyze Quantifications and Scope123 using GPT-4o, to detect nuances, such as formatting or incorrect unit sizes. Based on the query, as presented in Figure 5.1, the model returns a score between 1 and 5. Where 1 is returned if the number is completely wrong, a number between 2 and 4 is returned if it is wrong to some extent, and 5 is returned if the answer is correctly given in the right format (see Figure 4.3. Table 5.3 shows the results. The G-Acc provides a certain degree of freedom to generative models for the format in which they provide their answer. When a grade of 5 is assigned, the G-Acc will be true. The criterion for a grade of 5, that was given to the evaluating GPT-4o, is: "The essence of the answer is what it should be, including the correct unit size and the correct number, but more information may be given by the model". This means that we apply CEM, as explained earlier in Chapter 3.

We evaluated the QA performance of SVRAG and M3DocRAG. Self-Attention is only a retrieval module and is therefore excluded from this analysis. The results show that the accuracy is low. In the Scope123 dataset, SVRAG had an accuracy of 13%, and M3DOCRAG did not provide one correct answer. Similarly, performance in the Quantifications dataset was low, with M3DocRAG obtaining a G-Acc of 2.9%, while SVRAG did not correctly answer any question.

Table 5.3: Likert and G-Acc scores per architecture and dataset

Dataset	**Architecture**	**Avg. Likert (1-5)**	**G-Acc**
Scope123	SVRAG	1.680000	0.130000
Scope123	M3DocRAG	1.100000	0.000000
Quantifications	M3DocRAG	1.285714	0.028571
Quantifications	SVRAG	0.114286	0.000000

When observing the frequency of given Likert scores, it is remarkable that on the Scope123 dataset, in the majority of cases, the model is unable to perform a simple task, namely to provide the output in the correct format. Here, the requested format is to output the scope emis-

sions in kilotonnes of CO_2eq, which is the most common metric for reporting Scope123 emissions. In the quantification dataset, the formatting was more complex. Here, the QA-module of M3DocRAG appeared to have more difficulties generating a response in the right format than SVRAG, which uses Qwen2-VL-7B-Instruct-GPTQ-Int4 and Intern2-VL, respectively. Later, we found that the implementation provided by Omar Alsaabi differs from the original paper, which uses Qwen2-VL-7B. We did not perform another evaluation with Qwen2-VL-7B because, based on the retrieval results, it became apparent that the focal point of this study needs to be the retrieval quality of the current state-of-the-art.

Since in the classification datasets, the main objective of the model is to correctly indicate whether information is present or not, we do not employ a GPT-based grader but define the rules ourselves, using a Python script. Based on this, we calculated the precision, recall, f1-score and accuracy of the model. The results are presented in 5.4. Both models obtain mediocre classification results, with SVRAG performing slightly better. When observing the true positive (TP), false positive (FP), true negative (TN), and false negative (FN) quantities in ClimRetrieve, we observe that both SVRAG and M3DocRAG almost always returned positive. This coincides with a slightly larger number of positives in the dataset (28 positive and 25 negative).

The average QA-performances on all datasets are 24.9% for M3DocRAG, and 35.4% for SVRAG. Thus, SVRAG scores higher on QA, however, SVRAG rarely found the relevant pages, meaning that it is coincidentally guessing the right answer. Considered that the QA model in M3DocRAG can still be improved by taking the original variant Qwen2-VL-7B, we choose to continue with M3DocRAG in modeling phase 2, based on its higher retrieval accuracy and the upward potential by employing the better Qwen2-VL.

Table 5.4: Classificationsmetrics including Recall@5, precision, recall, F1, and accuracy per architecture and dataset

Dataset	Architecture	TP	FP	FN	TN	Recall@5	Precision	Recall	F1	Accuracy
Classifications	SVRAG	20	9	8	13	0.031	0.690	0.714	0.702	0.660
Classifications	M3DocRAG	0	0	28	22	0.172	0.000	0.000	0.000	0.440
climRetrieve	SVRAG	28	21	0	4	0.150	0.571	1.000	0.727	0.604
climRetrieve	M3DocRAG	28	25	0	0	0.119	0.528	1.000	0.691	0.528

5.1.2. EVALUATION PHASE 2: ADJUSTING M3DOCRAG TO SUSTAINABILITY REPORTS

As described in Chapter 4, we employ M3DocRAG and propose applying fine-tuning and HyDE to improve the retrieval performance of the architecture. Because a newer version of the ColPali package is required to use the LoRA adapter, we slightly adjust the existing M3DocRAG pipeline, leading to a different performance than in evaluation phase 1. Therefore, we set a new baseline, where we use the modified pipeline. Here, the retriever is ColPali, which is loaded from the `vidore/colpali` adapter [1]. In addition, we deactivated the clustering module, resulting in slightly higher retrieval accuracies.

[1] https://huggingface.co/vidore/colpali

Table 5.5: Retrieval performance with different fine-tuning and HyDE settings, presented as a difference with the baseline. The metrics are averages over all datasets.

Quantization	FNT-Data/hypo	Avg. Recall@5	Avg. Recall@10
none	**Baseline**	0.402991	0.489085
Int4	HyDE	+0.028411	+0.058534
Int4	10 reports	+0.003452	-0.007623
Int4	5 reports	-0.011194	-0.016376
Int4	20 reports	-0.03725	-0.04075
none	10 reports	-0.079963	-0.064833

RECALL

During modeling phase 2, we tested fine-tuning and HyDE. We started by applying fine-tuning. The average results are presented in Table 5.5. When analyzing the average over all datasets, fine-tuning did not seem to be effective. However, Table 5.6 shows that the performances differ by dataset. In the Classifications dataset, the best fine-tuning performance was obtained by fine-tuning on 10 sustainability reports but did not lead to an improvement. The other fine-tuning settings performed worse on this dataset. However, applying HyDE leads to significant improvements in the classification dataset, where it increases the Recall@5 with 7.8%. Recall@10 increased by 10.9%. Considered that HyDE was tested on the `Int4 + 5 reports` configurations for fine-tuning, which initially led to a decrease of the retrieval performance, applying it to the baseline model will possibly increase this performance by another 1.5%. In the hypothetical situation where we can add up the performance increases of HyDE and fine-tuning on 10 reports, this would lead to an increase in Recall@5 performance on the classification dataset from 20.3% to 28.1%.

Table 5.6: Recall performance deltas compared with the baseline (italicized row per dataset).

Quantization	FNT-Data/hypo	Dataset	Recall@5	Recall@10
none	*Baseline*	Classifications	0.203	0.297
int4	HyDE prompt	Classifications	+0.078	+0.110
int4	10 reports	Classifications	0.000	-0.031
int4	5 reports	Classifications	-0.016	-0.047
int4	5 + 2 esrs/csrd	Classifications	-0.047	-0.078
int4	20 reports	Classifications	-0.078	-0.109
none	10 reports	Classifications	-0.078	-0.141
none	*Baseline*	ClimRetrieve	0.310	0.471
int4	20 reports	ClimRetrieve	+0.014	-0.035
int4	HyDE prompt	ClimRetrieve	+0.013	+0.042
int4	10 reports	ClimRetrieve	+0.010	-0.028
int4	5 + 2 esrs/csrd	ClimRetrieve	-0.004	-0.007
int4	5 reports	ClimRetrieve	-0.020	-0.043
none	10 reports	ClimRetrieve	-0.076	-0.047
none	*Baseline*	Quantifications	0.233	0.250
int4	HyDE prompt	Quantifications	+0.069	+0.129
int4	10 reports	Quantifications	+0.043	+0.043
int4	5 reports	Quantifications	+0.017	+0.034
int4	5 + 2 esrs/csrd	Quantifications	-0.017	+0.052
int4	20 reports	Quantifications	-0.026	-0.026
none	10 reports	Quantifications	-0.043	-0.052
none	*Baseline*	Scope123	0.866	0.939
int4	5 reports	Scope123	-0.026	+0.020
int4	10 reports	Scope123	-0.039	-0.026
int4	5 + 2 esrs/csrd	Scope123	-0.039	0.000
int4	HyDE prompt	Scope123	-0.046	-0.048
int4	20 reports	Scope123	-0.059	+0.007
none	10 reports	Scope123	-0.123	-0.020

In the ClimRetrieve dataset, again, HyDE appeared to be more effective, since it increased Recall@5 by 1.3% and Recall@10 by 4.2%. Again, most fine-tuning configurations did not lead to an improvement in this dataset. However, fine-tuning with 10 sustainability reports led to an increase of Recall@5 and Recall@10 of 1.1% and 2.9%, respectively. When adding this up, this could lead to an increase in Recall@5 from 31.0% to 33.3%, and Recall@10 from 51.22% to 54.12%.

In addition, on the Quantifications dataset, HyDE improved the performance most. Recall@5 increased by 6.9%, and Recall@10 increased by 12.9%, when HyDE was applied. When fine-tuning with a training set of 10 sustainability reports, with a quantization setting of Int4, the performance increased by 4.3% for both Recall@5 and Recall@10. Again, hypothetically, when combining the HyDE performance increase and fine-tuning on 10 reports, the resulting Recall@5 and Recall@10 could be 34.5% and 42.2%, respectively.

Lastly, on the Scope123 dataset, the suggested modifications do not seem to benefit the Recall@5. However, when evaluating Recall@10, an improvement of 2.0% can be measured when

fine-tuning on 5 reports. Thus, when only applying the fine-tuning on 10 sustainability reports, the final Recall@10 could become 95.8%.

Ablation study number of reports We observed a slight increase in the performance of the fine-tuned models using QLoRA (Int4) fine-tuning, when the number of documents the model was trained on increased from 5 to 10. Therefore, we performed an ablation study, solely focusing on the influence of the number of documents used in fine-tuning, hypothesizing that increasing the number further might lead to a further performance increase. The results are presented in Figure 5.2.

The results show that only on the ClimRetrieve dataset, our hypothesis is not challenged. On the EY Classifications and EY Quantifications datasets, a clear decline is seen. On the Scope123 dataset, the influence of increasing the number of documents to 20 is positive, but negligible. In general, it does not seem that increasing the number of documents leads to a better performing model.

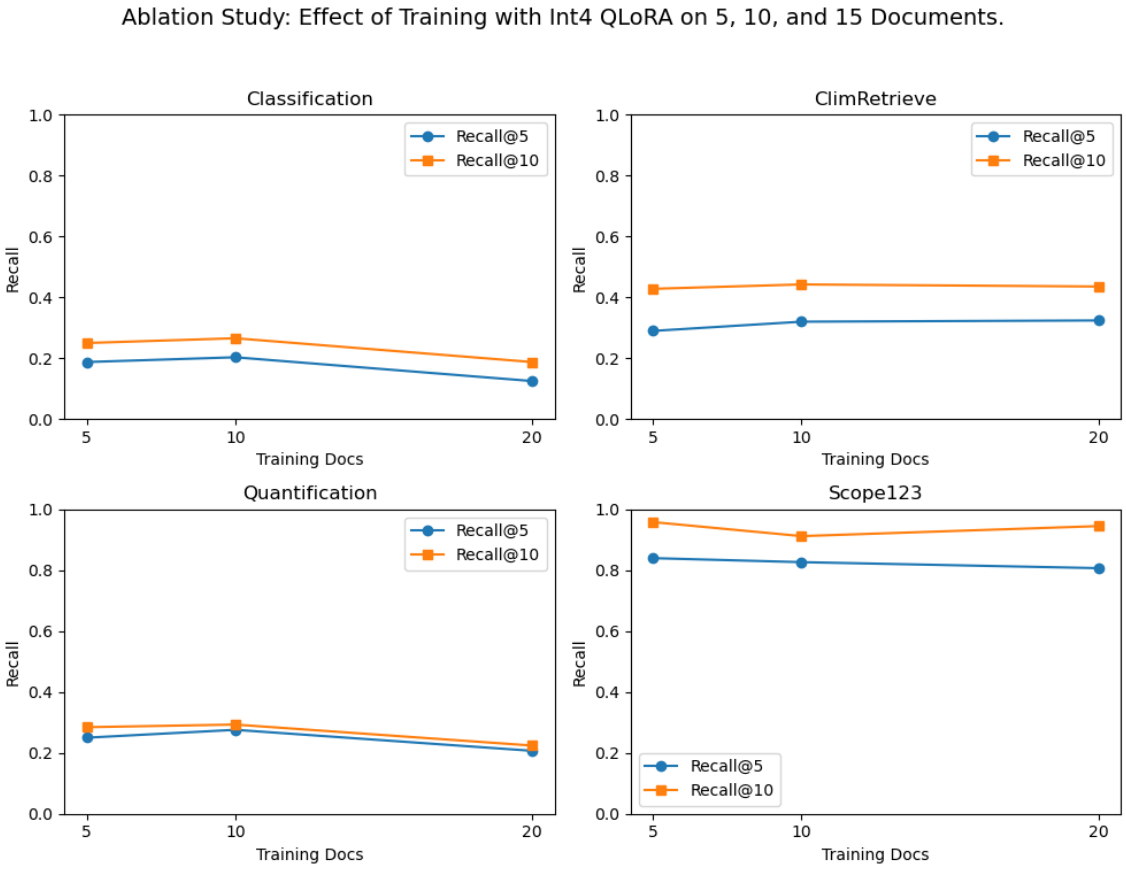

Figure 5.2: Ablation Study for fine-tuning with QLoRA Int4, where the number of training documents varies.

FIRST PAGE INDEX

Besides observing the Recall@k metric, we further analyze the first relevant page indexes to investigate the feasibility of a Document AI pipeline for IE from sustainability reports. For this,

we select the best performing model on each dataset and observe the average index of the first relevant page. This tells us the number of pages to which we can compress the files without excluding the relevant pages.

We analyze the first relevant page index by letting the retrieval model return the highest-scored 100 pages, and observe what the index was of the first relevant page, when ordered based on the given similarity score, from high to low. When the relevant page was not included, we assign 100 as the first relevant page index, which leads to slightly lower averages.

Quantization	FNT-Data/hypo	Dataset	First Rlvnt Page (mean)	Std Dev	Not Found (mean)	Mean # Pages
Int4	10 reports	Classifications	37.125000	33.489871	0.125000	425.80
Int4	HyDE	ClimRetrieve	7.686275	15.505470	0.019608	67.38
Int4	HyDE	Quantifications	48.155172	42.859543	0.310345	425.80
Int4	5 reports	Scope123	3.513158	15.901776	0.026316	141.00

Table 5.7: First relevant page statistics and average document length by dataset and configuration. Lower is better.

By analyzing the percentiles of the first relevant page index, we can get an indication of the minimum number of pages that the model needs to receive, to at least have one relevant page with a certain probability. We find that the average first relevant page index in the Classifications and Quantifications datasets is high: 37.1 and 48.2 on average. Furthermore, we find high standard deviation meaning that a significant number of retrievals have a first relevant page considerably higher than the averages.

Since applying Document AI for QA for sustainability report benchmarking requires high QA-performance, a high retrieval performance is required as well. We analyze the 80th and 95th percentile, to get insight in the number of pages that are required to retrieve, to ensure that at least one relevant page is included in the output of the retrieval model, 80% or 95% of the time. The results are depicted in Figure 5.3. We find that 24 pages need to be given to the QA-model to include at least one relevant page on the ClimRetrieve dataset, and 6 pages on the Scope123 dataset. For classification and quantification, this number is more than 100.

Figure 5.3: Distribution of the first relevant pages on the best performing models

QA-PERFORMANCE

Since the retrieval results appear to be insufficient to meet the business requirements for performing a benchmarking analysis as described in Section 4.1, we analyze the performance of the QA module for the hypothetical situation where the retrieval is always 100%. This gives an indication what the required performance of the retrieval module must be. We analyze the situation that all right pages are included in either 5, 10 or 20 pages. Also, we compare against GPT-4o, which is a significantly larger model, and available via the OpenAI-API. We analyze the results for each of the datasets. In this section, we specify the number of pages by adding a (x) behind the model name, where $x \in \{5, 10, 20\}$. The results are presented in Figure 5.4.

On the EY Classifications dataset, where the model has the task to detect whether certain information about, or related to the ESRS, is disclosed in the document (see Section 4.2), the best performing GPT-4o (5) obtains an accuracy of 84%, while the best model of Qwen2-VL (5) obtains an accuracy which is 6% lower. The performance of both models decreased when the number of pages given to the model was increased to 10. The f1-scores of all models were 0, indicating that model performance is likely favored by a class imbalance, which becomes clear when we analyze the confusion matrix (see Figure 5.5). The confusion matrix shows that GPT-4o (5) only returned a negative response and thus was unable to detect whether the information was present in the document. Thus, because the Classifications dataset is imbalanced, with 8 positive examples and 42 negative examples, the model coincidentally predicts negative correctly when it should have, more often than when it should not have, leading to a high accuracy. All other variations of Qwen2-VL and GPT-4o had this problem (see Appendix D for all results).

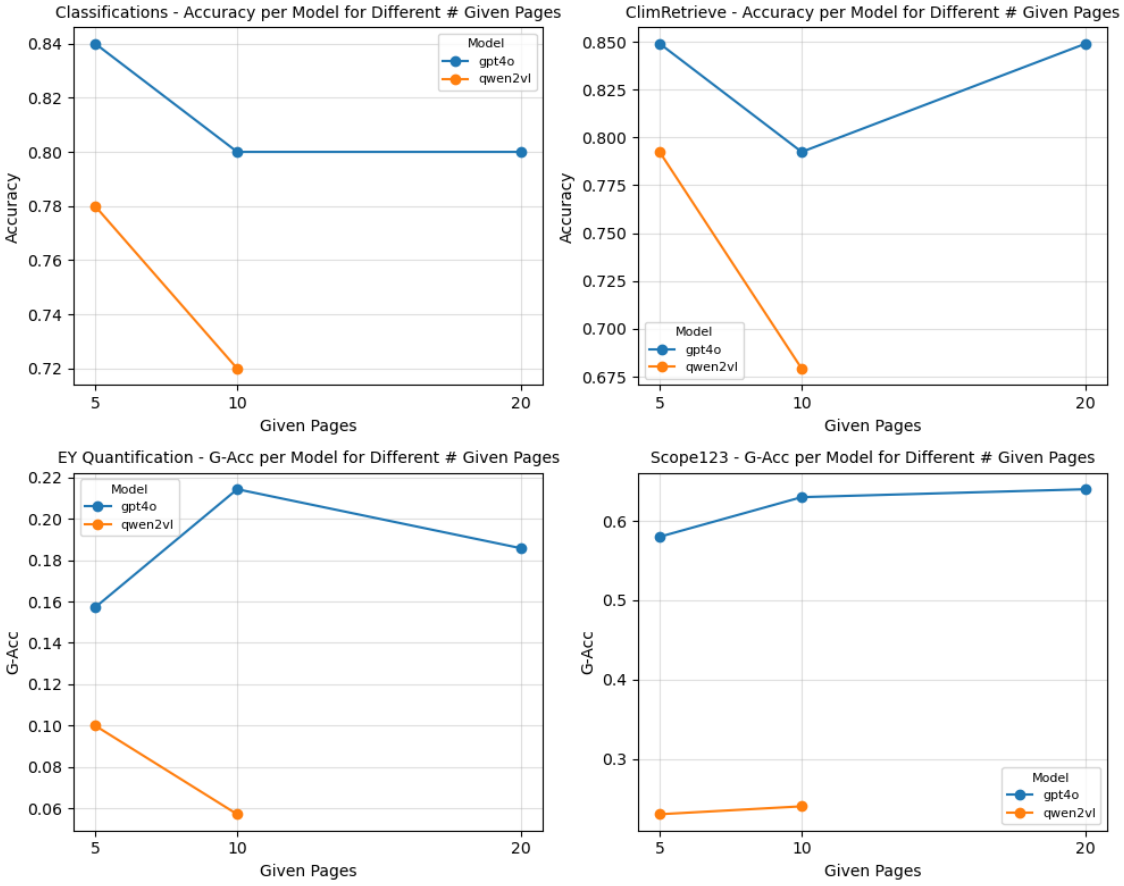

Figure 5.4: The accuracy and G-Acc for 5, 10 or 20 pages, on the Classifications, ClimRetrieve, Quantifications and Scope123 datasets. Here, Qwen2-VL (20) consistently led to an out-of-memory error, and is thefore not depicted.

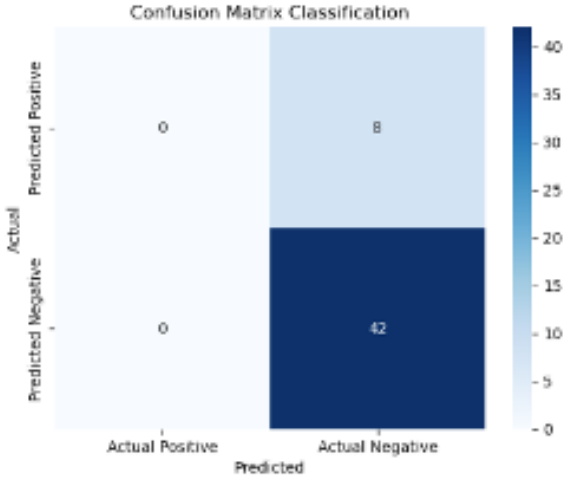

Figure 5.5: Confusion matrix for EY Classifications, for GPT-4o (5)

Similarly, GPT-4o outperformed Qwen2-VL on the dataset of ClimRetrieve, as shown in Figure 5.4. In the ClimRetrieve dataset, the task is to determine whether certain information is present in the report. Again, Qwen2-VL (5) obtained the best results, with 85%, and when the given pages were increased to 10, the results drastically decreased, from 79% to 68%. Increas-

ing the number of pages to 10 also leads to a performance decrease for GPT-4o; however, when the model received 20 pages, it performed equally as good as when it received 5 pages. When observing the f1 scores, we see that they are mostly similar to the accuracy, indicating that the model is able to detect positives and negatives, which is also shown by the confusion matrix in Figure 5.6. Here, you can see that the model has a high recall, detecting almost all positive cases. The precision is slightly lower, since six times out of 25 times, the model predicted a positive, while it should have been negative. See Appendix D for the accuracy, precision, recall, and f1 score for all configurations that we tested.

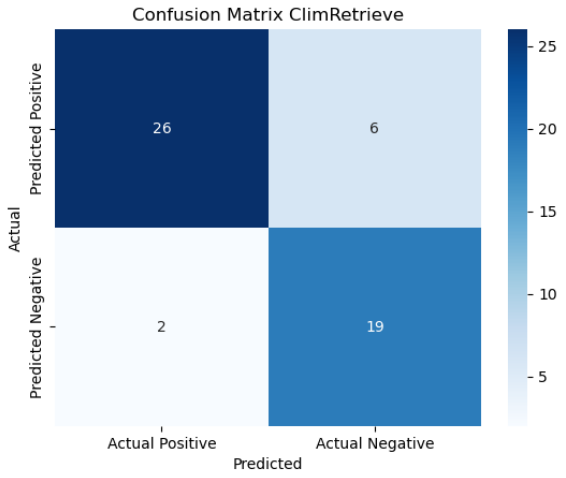

Figure 5.6: Confusion matrix for ClimRetrieve, for GPT-4o (5)

In contrast to the datasets of ClimRetrieve and EY Classifications, which focus on binary classification, the EY Classifications and Scope123 datasets aim to evaluate the IE performance of the models. The performance was evaluated using the G-Acc (see Section 4.4.2). Also, on the task of information extraction GPT-4o appears to consistently outperform Qwen2-VL (see Figure 5.4. In the EY Quantifications dataset the highest accuracy obtained was 21%, by GPT-4o (10). In contrast, the highest accuracy obtained by Qwen2-VL (5) was 10%. Again, the performance of Qwen2-VL decreased when more pages were given to the model, while this was not the case for GPT-4o.

Also in the Scope123 dataset, GPT-4o (20) obtained the highest G-Acc, which was 64%, while the highest Qwen2-VL score (10) was 24%. However, in this case, Qwen2-VL was not negatively affected by an increase in the number of pages given to the model, as now both Qwen2-VL and GPT-4o increased with 1% and 6%, respectively.

In general, therefore, it can be seen that GPT-4o outperforms Qwen2-VL by a large margin, when measuring accuracy and G-Acc. Furthermore, GPT-4o appears to be largely unaffected by an increase in the number of given pages, while overall the performance of Qwen2-VL drops significantly, when going from 5 to 10 given pages. Another finding is that although all relevant pages are included in the $given\_pages = 5$ configuration, for information extraction, GPT-4o seemed to benefit significantly from including more pages.

FOCUS GROUP WITH SUSTAINABILITY EXPERTS

The results of this study were presented to a group of seven sustainability experts from EY. After which an open discussion was held to discuss what needs to be improved to apply this in a business setting. The experts placed much emphasis on the accuracy required for a benchmark analysis. No mistakes can be made, and thus, they indicate that the current state of AI can never be fully trusted upon. Therefore, the results of the sustainability reports experiments were insufficient to apply them directly in a business setting. However, they indicate that if a user-friendly human-in-the-loop system can be built, the tool can be of great value if it is able to obtain performance such as those in the Scope123 and ClimRetrieve datasets.

5.2. DISCUSSION

We identify retrieval as the bottleneck when applying state-of-the-art document AI for QA on sustainability reports. When comparing the models in Evaluation phase 1, we find that M3DocRAG obtains the best retrieval results, with a Recall@5 only 37.6%. The Recall@5 of the other two models revolves around 5%. Remarkably, SVRAG performs much worse on retrieval than M3DOCRAG, while SVRAG applies the same technique for the retrieval module as ColPali, using exactly the same data as ColPali [64]. The difference between the two architectures is that ColPali employs PaliGemma as base model, and SVRAG in our implementation employs InternVL-4B. Here, InternVL has 4B parameters, compared to PaliGemma having 3B parameters in the implementation we use. Potentially, a larger base model in this case led to a lower bias and overfitting, meaning that the model is better able to fit on the training data due to flexibility, though therefore performs worse on data not used in training. Although we found in our literature review that the performance of an LLM benefits from a larger model size (see Section 2.1), the opposite may be true when a model is fine-tuned using LoRA on an out-of-domain dataset.

Moreover, the paper of ColPali highlights that patching of the document images leads to a performance decrease for documents containing more text [68]. Here, they found that, especially on the sustainability reports, which are text heavy, the performance is lower. The function of patching is to disregard white spaces in the document that contain no information, to reduce the context length. However, when the information is dense, this could lead to a loss of information. SV-RAG uses patching in a different way, which results in usually fewer patches per page. This could contribute to this lower performance on text-dense sustainability reports.

Furthermore, we observe a significant difference between the retrieval performances across datasets, which may be attributable to differences in document sizes. The documents in the EY datasets are substantially larger (counting ≈ 400 pages) than those in the other datasets (counting ≈ 100 pages). However, the difference in performance can not be completely attributed to the document length, since the retrieval performance on the Scope123 dataset is better than that on the ClimRetrieve dataset, which counts half of the pages. We assume that this comes from the fact that the Scope123 emissions are one of the most commonly used terms in sustainability reporting, due to which it is highly unlikely that it was not included in the data used

to train a pre-trained model. Moreover, Scope123 is aimed at directly extracting something that is written in the report, without the requiring any reasoning or summarization. In contrast, the questions in ClimRetrieve sometimes require the model to reason, or connect different parts in a text, which makes it less of a direct extraction task and more an abstract extraction, making it more challenging to fetch the relevant pages. Especially when taking into account how ColPali fetches the relevant pages, namely by adding summing up all maximum similarity scores between the patches and query tokens (which often come down to single words) [68] (see Equation 4.2), the model might be less prone to give a high score to a question which asks about something which is written differently in the report, because the words in the question differ from the words in the relevant part of the text.

When comparing the state-of-the-art models in evaluation phase 1, we also analyze the QA performance. For this, we use G-Acc, as previously done by Ma et al. [32]. As expected, M3DocRAG and SVRAG perform poorly on QA. However, especially in M3DocRAG, not only the retrieval, but also the formatting is insufficient. This is probably because the original architecture by Omar Alsaabi uses a quantized version. Indicating that although quantization might be interest to enable local computing, it is not recommended when accurate question answering based on the relevant pages is required. Due to time restrictions and the already clear incentive that the retrieval performance needs improvement first, we do not test with a not-quantized version of Qwen2-VL in evaluation round 1 anymore.

After evaluation phase 1, we conclude that retrieval is the bottleneck when using document AI for QA on sustainability reports. Therefore, in modeling phase 2, as being the best retrieval model in phase 1, M3DocRAG serves as the baseline model. As a retrieval component, we again use the ColPali model obtained by 'vidore / colpali' as the baseline. However, we observe a slightly higher performance of M3DocRAG during this phase, which could result from two modifications to the pipeline. Firstly, by deactivating IVF clustering, the retrieval performance increases. Furthermore, we slightly adjust the pipeline, so that a newer version of the colpali_engine that allows LoRA, could be implemented, which also could have lead to a slightly higher performance of the baseline in the evaluation phase 2.

In evaluation phase 1, we especially see shortcomings in the retrieval performance when extracting information on the EY datasets. These documents are lengthy, and contain questions specifically aimed at ESRS and CSRD topics, being relatively new topics, making it likely no information about them was included in the training data for PaliGemma, which is the base-model of ColPali. Therefore, we hypothesize that by fine-tuning the ColPali adapter on those datasets, we could equip it with the knowledge required to fetch the right pages.

Using QLoRA fine-tuning we aim to improve the current retrieval accuracy of M3DocRAG. We find that QLoRA fine-tuning can effectively reduce the model size, while increasing the performance. The performance differs per dataset. We test different configurations of the quantization strategy (Int4, Int8 or none) and also the number of training documents. Int4 quantization performs better than using Int8, or no quantization during fine-tuning. This might result from

the fact that the training dataset consists of different documents than the datasets on which we finally evaluate the resulting fine-tuned models, which are the EY Classifications, EY Quantifications and Climretrieve and Scope123 datasets. Potentially, Int4 coincidentally applies to an extent a required regularization, which allows in the end for a better generalization to other type of documents.

We perform an ablation study, where we tested different variations of the number of documents used to realize the training dataset. The variations tested are 5, 10 and 20 documents. We find, that for most datasets, an increase of the number of documents in the training data, does not necessarily lead to a performance increase on our selected evaluation dataset, which might be because the documents in the evaluation dataset are different from the training data. Thus, increasing the number of documents while allowing for hyperparameter tuning with the same parameter ranges, likely allows for overfitting in a setting of 20 documents, and potentially underfitting in a setting for 5 documents.

In addition to fine-tuning, we test HyDE. There appears to be a negative correlation between the benefit of applying HyDE and the baseline performance of the model on the dataset. On the Classifications and Quantifications datasets, we realize a significant performance increase of more than 6.8% on the Recall@5. Also, ClimRetrieve benefited from HyDE, while the performance of Scope123 remains roughly equal when averaging the Recall@5 and Recall@10. We think that HyDE especially benefits the performance on the Classifications, ClimRetrieve and Quantifications datasets because questions are asked about topics it might not have sufficient knowledge about. The Classification dataset consists of questions about the ESRS, the Quantification, consists of questions using terminologies such as financed emissions or Sustainable Finance Disclosure Regulation (SFRD) and in ClimRetrieve, consists of questions about the TCFD. In contrast, Scope123 directly asks about the number for either the scope 1, scope 2 or scope 3 emissions, requiring minimal domain knowledge furthermore, since searching with exact match will lead to the specific information in the document. Thus, it seems that HyDE mitigates to a certain extent the absence of understanding of those terminologies, when a more capable MLLM is used as prompt generator. What could also contributes to the performance increase, is that GPT-4o has more knowledge due to a later cutoff date (the date at which the last data ingestion took place).

To gain a better understanding of the extent to which the documents can be compressed, we also look at the average first relevant page index, and find that on the ClimRetrieve and Scope123 respectively, 24 and 6 pages need to be retrieved to include at least one relevant page 95% of the time. Although 95% is not sufficent to suffice to the business requirement, it could be a basis for a human-in-the-loop approach. To investigate the potential performance, we give the correct pages to the QA model, in a setting of providing a total of 5, 10 or 20 pages. We find that GPT-4o significantly outperforms Qwen2-VL on QA. Also, the performance of Qwen2-VL decreases significantly when the number of input documents increases. Since performance of both Qwen2-VL and GPT-4o does not exceed 60% for any number of given pages on the

Scope123 dataset, a fully automatic IE pipeline appears to be infeasible on this dataset. However, combining the 24 pages retrieved on ClimRetrieve with GPT-4o, might appear to potentially yield valuable performance for a human-in-the-loop approach. However, when 24 pages are retrieved, only one of the relevant documents is present, while in the ClimRetrieve dataset, the average number of relevant pages per question is 2.8, showing that a larger performance increase might be needed before implementing this in practice, even when combining with a human-in-the-loop approach.

Furthermore, the results show that for the classification tasks, where the aim is to determine whether the company discloses certain information or not, both GPT-4o and Qwen2-VL obtain better performance than on IE tasks. In addition, the performance of GPT-4o decreases only on the Classifications dataset, when the number of pages is increased from 5 to 20, while on other datasets, the performance increases. This performance increase could be due to valuable information for answering the question being present in the additional pages, enabling the QA module to better answer the question. If this is true, this could indicate that the relevant pages in the datasets are inadequately labeled, or that splitting the document in pages might be a suboptimal approach.

Taking into regard the resulting best retrieval performances we obtained on our evaluation dataset, we conclude that the current state of publicly available Document AI methods can be valuable for making the analysis of sustainability reports more efficient. As long as documents are not exceeding 150 pages and there is an overlap in wording between the queries and the document, high retrieval accuracy can be acquired. Though, when documents are very large, specific (new) terminology knowledge is required, and questions are abstractive (hardens col-retrieval), the current state of document AI might not be sufficient.

5.2.1. MANAGERIAL IMPLICATIONS

Several findings of this study have implications for managers or practitioners in the field of (non-financial) audit, sustainability and organizations affected by the CSRD.

Firstly, with our literature review on the usability of LLMs for IE from sustainability reports, we found that LLMs alone are unlikely to obtain an accuracy higher than 99%, because of their sole focus on the text in a document. We also found that the highest accuracy obtained by an LLM when performing extraction of scope 1, 2 and 3 emissions on sustainability reports was 85%, pointing out that it will not suffice for adhering of the requirements in a sustainability benchmarking setting, as described in Section 4.1. However, for certain tasks, Document AI can greatly reduce the time spent when information needs to be found in the document.

Also, the current state of Document AI is not able to directly extract the relevant information, when multi-page documents are given to the model. However, especially on sustainability reports, a human-in-the-loop process can be fruitful. In such a process, a retrieval module and QA module can be combined, to finally present the QA response together with the fetched page(s) to a user, who checks whether the model extracts the correct pages and information.

Then, if the information was not correct, the process can be repeated, optionally by excluding the recently fetched pages, or if the correct pages were fetched, by reducing the pages to the manually correct page. In this way, considering the performance on sustainability reports in this study, on certain tasks, the number of pages needed to be analyzed can be reduced significantly. For example, when the goal is to extract scope emissions from sustainability reports, the relevant page can be included in the first two pages 80% of the time, showcasing the potential efficiency gain for practitioners performing a benchmarking analysis.

Furthermore, this work helps organizations in reducing processing times and costs of already existing Document AI methods. By effectively reducing the number of pages based on their relevancy, less pages need to be processed by a VLM. Since the documents are long, directly using all document pages as input to a VLM will lead to significant processing times and costs if a cloud solution is used via an API. By creating a pipeline where the documents are filtered based on the required pages, before sending a request to the model or API, the time and costs are reduced.

Moreover, our experiments show that with a combination of QLoRA fine-tuning and HyDE, similar or better performance can be acquired on all datasets while reducing the size of the document by applying Int4 quantization, making this technology more accessible to organizations with limited access to computing resources, and reducing the required money spent to access AI solutions. In addition, QLoRA speeds up inference [93].

Lastly, implementing a Gen (Document) AI solution requires some careful considerations. Firstly, when such a solution is required, its fairness, robustness, and reliability needs to be assessed. (M)LLMs are prone to hallucinate, though, those hallucinations are difficult to detect, due to the tendency of the AI to make the answers seem probable. Thus, until the models have been proven to extract information reliable, a human-in-the-loop process remains to be required. Also, utilizing an API should be done with care. Although sustainability reports are publicly available, a practitioners method to assess to information in the document may be private. Using an API could give away valuable information. Additionally, could an API lead to significant costs, especially if large documents are used for analysis. Instead, a locally runnable model requires a larger initial investment, but very low maintenance costs afterwards.

5.2.2. THEORETICAL IMPLICATIONS

Firstly, no study in the selected literature in the Section 2.2 explored the retrieval performance on documents exceeding 125 pages. We show, that all publicly available retrieval models, show a significant performance decrease when a large number of pages is added. Which shows the need for methods which are unaffected by an increase in document size.

Secondly, we applied HyDE, and showed that it can significantly increase the performance on page-retrieval in a setting with specific domain language [94]. No previous study in the selected literature analyzed the potential of HyDE.

Thirdly, we showed that using a VLM (Qwen2-VL) to annotate a dataset used for QLoRA fine-tuning can increase the retrieval performance, also when complex reports such as sustainability reports are used. This makes fine-tuning and training more accessible, and can support in developing more specialized models.

6

CONCLUSION

During our study, we evaluate the applicability of current state-of-the-art document AI architectures for IE on sustainability reports. We find that there is no one-size-fits-all solution, when applying QLoRA fine-tuning and HyDE to increase the low retrieval ability in sustainability reports of ColPali, which is the best retriever on our selected data. Instead, each task, query, and type of document has a better fitting configuration. We find that QLoRA fine-tuning can at most increase Recall@10 performance by 4.3%, while a HyDE prompt leads to an increase of 12.9% on one of the datasets. Overall, combining QLoRA fine-tuning and HyDE allows for significantly reducing the size of the retrieval module, while maintaining similar performance, making it a valuable approach for making page-retrievers accessible for practitioners with limited access to computing resources. We show that when wording in a query and relevant passage align, high page-retrieval accuracy can be obtained, which can significantly reduce the time spent on sustainability benchmarking, but also on similar practices where information extraction is required. However, we also observe an underperformance in tasks where documents are lengthy, include domain-specific terminology, and require reasoning. QLoRA and HyDE can address this to some extent, though more research in model architectures for extracting from multipage document will most likely be more effective in improving the IE performance. After evaluating state-of-the-art Document AI model performance against requirements in a sustainability benchmarking setting within EY, we recommend a human-in-the-loop approach, where the output of the model is checked by a human, before being used in downstream processes.

6.1. FUTURE RESEARCH DIRECTIONS

This study explores QLoRA fine-tuning and HyDE to improve a Document AI pipeline for IE from sustainability reports. Although several improvements in retrieval performance are realized, a larger performance increase is required. QLoRA fine-tuning and HyDE are both options which require little effort, and are therefore ideal for a slight improvement increase, but insuf-

ficient to provide a solution to the lacking retrieval performance. Therefore, other options to increase the retrieval performance should be investigated. Considered that the domain knowledge seems to be a bottleneck, especially because page-retrieval on the Scope123 dataset is high, unfreezing the model parameters and retraining the model is a promising research direction.

As long as the context in a document cannot be reduced to a size from which a QA module can effectively extract information, no Document AI methodology is able to acquire 99% accuracy. Thus, further research in page-retrieval from large documents is required. A promising architecture is created by Gu et al. [75]. In this architecture, OCR is applied to extract the images and text is further extracted as text. This reduces the context size heavily, which likely contributes to their stable retrieval performance, despite an increasing number of document pages, making it a promising future research direction.

Also, during our study, we use out-of-domain data, which is data on which the model is not trained, to evaluate our fine-tuned models. However, we also use an automatic pipeline for fine-tuning, where the annotations are created by a VLM. During training, the loss on the training data decreases to near zero, showcasing that the model adapts to the training dataset. However, this increased performance might not transfer to out-of-domain data. Considered that a benchmarking study requires much manual labor as it is currently, another interesting future research direction is to build a pipeline where a training dataset is created automatically for the report at hand, which can be used for fine-tuning, after which the fine-tuned model is used on the same report.

Lastly, an agentic approach could be an interesting approach to recover errors made by the retrieval module [96]. A potential solution, for example, might be to let Qwen2-VL analyze the retrieved document pages, on whether information to answer the question is present in the page, or a part of the page.

6.2. LIMITATIONS

This study had several limitations. Firstly, generative AI was exploited for the analysis of the model QA responses on the Scope123 and EY Quantifications datasets. Although we have refined the prompt until we observed desirable results by the grading model, we did not analyze all answers, which might give room for mistakes by the grading model.

Moreover, we did not address the explainability of the resulting solution, because all efforts were spent on getting to an acceptable performance for implementation in practice. ColPali offers interesting features for this, by creating similarity maps.

Furthermore, this study is limited to QA architectures of which the code is publicly available. This excludes a few potentially better architectures from this analysis, such as DocVLM, Arctic-TILT, and PDFWukong. Although M3DocRAG obtains very similar performance, on MP-DocVQA, it may be that certain models are to a lesser extent negatively affected by an increase in docu-

ment size. For example, PDF-Wukong was shown to maintain similar performance on documents, despite an increase of the number of pages [75].

Furthermore, we used Qwen2-VL to annotate the documents we used to fine-tune the retrieval model. However, the cutoff date for Qwen2-VL is June 2023, which means that it might not consist of all the information required to accurately extract information about topics such as the TCFD, CSRD and ESRS. Instead, GPT-4o has a cutoff date of June 2024. Although we analyzed the annotations and included questions about the ESRS and CSRD, annotations may be more accurate when created with GPT-4o.

Lastly, we used Scope123, ClimRetrieve, EY Classification, and EY Quantification data sets to analyze different tasks. However, the prompt setup between these datasets differ significantly (see Appendix E). Ideally, to allow for fair comparability between the different tasks, the prompt should be as equal as possible. Instead, we chose to optimize the prompt, which might have compromised comparability.

REFERENCES

[1] I. C. Wiest, D. Ferber, J. Zhu, M. van Treeck, S. K. Meyer, R. Juglan, Z. I. Carrero, D. Paech, J. Kleesiek, M. P. Ebert, D. Truhn, J. N. Kather. Privacy-preserving large language models for structured medical information retrieval. npj Digital Medicine 7 (2024). URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85204478329&doi=10.1038%2fs41746-024-01233-2&partnerID=40&md5=e2bdede0ab60994894ac794b89c6f3f8. doi:10.1038/s41746-024-01233-2.

[2] P. Wang, S. Bai, S. Tan, S. Wang, Z. Fan, J. Bai, K. Chen, X. Liu, J. Wang, W. Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191 (2024).

[3] S. Studer, T. B. Bui, C. Drescher, A. Hanuschkin, L. Winkler, S. Peters, K.-R. Müller. Towards crisp-ml (q): a machine learning process model with quality assurance methodology. Machine learning and knowledge extraction 3 (2021) 392–413.

[4] J. Vom Brocke, A. Hevner, A. Maedche. Introduction to design science research. Design science research. Cases (2020) 1–13.

[5] H. Alhaddi. Triple bottom line and sustainability: A literature review. Business and Management studies 1 (2015) 6–10.

[6] A. Hobbs. How the eu's new sustainability directive is becoming a game changer. EY Insights (2022). URL: https://www.ey.com/en_hu/insights/assurance/how-the-eu-s-new-sustainability-directive-is-becoming-a-game-changer.

[7] European Commission, The european green deal, 2025. URL: https://ec.europa.eu/stories/european-green-deal/.

[8] Committee of European Auditing Oversight Bodies, CEAOB guidelines on limited assurance on sustainability reporting, 2024. URL: https://finance.ec.europa.eu/document/download/8ac2df18-2ae1-4bc7-9d87-a4a740e48f5e_en?filename=240930-ceaob-guidelines-limited-assurance-sustainability-reporting_en.pdf.

[9] European Commission, Corporate sustainability reporting, 2025. URL: https://finance.ec.europa.eu/capital-markets-union-and-financial-markets/company-reporting-and-auditing/company-reporting/corporate-sustainability-reporting_en.

[10] C. Gaganis, F. Pasiouras, M. Tasiou, in: C. Gaganis, F. Pasiouras, M. Tasiou, C. Zopounidis (Eds.), Sustainable Finance and ESG, Palgrave Macmillan Studies in Banking and Financial Institutions, Palgrave Macmillan, Cham, 2023, pp. 109–138. URL: https://doi.org/10.1007/978-3-031-24283-0_6. doi:10.1007/978-3-031-24283-0_6.

[11] EY, EY 2021 global alternative fund survey, 2021. URL: https://www.ey.com/content/dam/ey-unified-site/ey-com/en-gl/insights/wealth-asset-management/documents/ey-2021-global-alternative-fund-survey.pdf.

[12] I. Hasan, C. K. Hoi, Q. Wu, H. Zhang. Social capital and debt contracting: Evidence from bank loans and public bonds. Journal of Financial and Quantitative Analysis 52 (2017) 1017–1047.

[13] J. Harrison, T. P. Lyon, J. W. Maxwell. Beyond the 'E' in ESG: The effects of governance factors on firm performance. Nature Humanities and Social Sciences Communications 6 (2019) 1–10. URL: https://www.nature.com/articles/s41599-019-0315-9.pdf. doi:10.1057/s41599-019-0315-9.

[14] B. Korca, E. Costa, L. Bouten. Disentangling the concept of comparability in sustainability reporting. Sustainability Accounting, Management and Policy Journal 14 (2023) 815–851.

[15] F. Berg, J. F. Kölbel, R. Rigobon. Aggregate confusion: The divergence of ESG ratings*. Review of Finance 26 (2022) 1315–1344. URL: https://doi.org/10.1093/rof/rfac033. doi:10.1093/rof/rfac033. arXiv:https://academic.oup.com/rof/article-pdf/26/6/1315/47018560/rfac033.pdf.

[16] SquareWell Partners, The Playing Field: A Look at the World's Largest 50 Asset Managers, 2022. URL: https://squarewell-partners.com/wp-content/uploads/2021/05/2022-SquareWell-Playing-Field-Top-50-FINAL-October.pdf, accessed: 17-Mar-2025.

[17] S. Tukiainen, ESG Ratings and Their Divergence from an Investor Perspective, Master's thesis, Aalto University, 2021. Retrieved from https://aaltodoc.aalto.fi/handle/123456789/111140.

[18] M. Bronzini, C. Nicolini, B. Lepri, A. Passerini, J. Staiano. Glitter or gold? deriving structured insights from sustainability reports via large language models. EPJ Data Science 13 (2024). URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85195597218&doi=10.1140%2fepjds%2fs13688-024-00481-2&partnerID=40&md5=8f15a5c870c193ef56585deb0716e421. doi:10.1140/epjds/s13688-024-00481-2.

[19] T. Reason, E. Benbow, J. Langham, A. Gimblett, S. L. Klijn, B. Malcolm. Artificial intelligence to automate network meta-analyses: Four case studies to evaluate the potential application of large language models. PharmacoEconomics - Open 8 (2024) 205 – 220. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85184504830&d

oi=10.1007%2fs41669-024-00476-9&partnerID=40&md5=8b7cafa342e897474aea83848f9db401. doi:10.1007/s41669-024-00476-9.

[20] M. A. Fink, A. Bischoff, C. A. Fink, M. Moll, J. Kroschke, L. Dulz, C. P. Heußel, H.-U. Kauczor, T. F. Weber. Potential of chatgpt and gpt-4 for data mining of free-text ct reports on lung cancer. Radiology 308 (2023). URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85171900198&doi=10.1148%2fradiol.231362&partnerID=40&md5=073950e2025b4eb22548888786f23a65. doi:10.1148/radiol.231362.

[21] A. Castro, J. Pinto, L. Reino, P. Pipek, C. Capinha. Large language models overcome the challenges of unstructured text data in ecology. Ecological Informatics 82 (2024). URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85200389928&doi=10.1016%2fj.ecoinf.2024.102742&partnerID=40&md5=e9618bf5bc3a6b0db8143dee3cb25e7d. doi:10.1016/j.ecoinf.2024.102742.

[22] G. Gomes Ziegler, Automating Information Extraction from Financial Reports Using LLMs, Master's thesis, TU Eindhoven, 2024.

[23] V. Sciannameo, D. J. Pagliari, S. Urru, P. Grimaldi, H. Ocagli, S. Ahsani-Nasab, R. I. Comoretto, D. Gregori, P. Berchialla. Information extraction from medical case reports using openai instructgpt. Computer Methods and Programs in Biomedicine 255 (2024). URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85198731618&doi=10.1016%2fj.cmpb.2024.108326&partnerID=40&md5=7cb1956d56e2c39e1cab6d3c0942e71a. doi:10.1016/j.cmpb.2024.108326.

[24] J. Van Der Elst, Extracting ESG Data from Business Documents, Master's thesis, École Polytechnique de Louvain, Université Catholique de Louvain, 2021.

[25] D. Balsiger, H.-R. Dimmler, S. Egger-Horstmann, T. Hanne. Assessing large language models used for extracting table information from annual financial reports. Computers 13 (2024). URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85207721064&doi=10.3390%2fcomputers13100257&partnerID=40&md5=25f1160f4df1b31db07b76ce75451380. doi:10.3390/computers13100257.

[26] Y. Zou, M. Shi, Z. Chen, Z. Deng, Z. Lei, Z. Zeng, S. Yang, H. Tong, L. Xiao, W. Zhou. ESGReveal: An LLM-based approach for extracting structured data from ESG reports. Journal of Cleaner Production (2024) 144572.

[27] B. I. Hub, Project gaia enabling climate risk analysis using generative ai. bis, 2024.

[28] Z. Wang, Y. Zhou, W. Wei, C.-Y. Lee, S. Tata, in: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 5184–5193.

[29] C. Barboule, B. Piwowarski, Y. Chabot. Survey on question answering over visually rich documents: Methods, challenges, and trends. arXiv preprint arXiv:2501.02235 (2025).

[30] M. Mathew, D. Karatzas, C. Jawahar, in: Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp. 2200–2209.

[31] R. Tito, D. Karatzas, E. Valveny. Hierarchical multimodal transformers for multipage docvqa. Pattern Recognition 144 (2023) 109834.

[32] Y. Ma, Y. Zang, L. Chen, M. Chen, Y. Jiao, X. Li, X. Lu, Z. Liu, Y. Ma, X. Dong, et al. Mmlongbench-doc: Benchmarking long-context document understanding with visualizations. arXiv preprint arXiv:2407.01523 (2024).

[33] Z. Wang, Z. Chu, T. V. Doan, S. Ni, M. Yang, W. Zhang. History, development, and principles of large language models: an introductory survey. AI and Ethics (2024) 1–17.

[34] H. Snyder. Literature review as a research methodology: An overview and guidelines. Journal of Business Research 104 (2019) 333–339. URL: https://www.sciencedirect.com/science/article/pii/S0148296319304564. doi:https://doi.org/10.1016/j.jbusres.2019.07.039.

[35] R. Hahn, Understanding complex systems, https://www.youtube.com/watch?v=86pGVP5JnOw, 2023. Accessed: 2025-01-09.

[36] P. Varsha, A. Chakraborty, A. K. Kar. How to undertake an impactful literature review: Understanding review approaches and guidelines for high-impact systematic literature reviews. South Asian Journal of Business and Management Cases 13 (2024) 18–35. doi:10.1177/22779779241227654. arXiv:https://doi.org/10.1177/22779779241227654.

[37] D. W. Aksnes, L. Langfeldt, P. Wouters. Citations, citation indicators, and research quality: An overview of basic concepts and theories. Sage Open 9 (2019) 2158244019829575.

[38] N. Kannan, Y. Seki. Textual evidence extraction for ESG scores. FinNLP-Muffin 2023 - Joint Workshop of the 5th Financial Technology and Natural Language Processing and 2nd Multimodal AI For Financial Forecasting, in conjunction with IJCAI 2023 - Proceedings (2023) 45 – 54. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85184822418&partnerID=40&md5=295df3b290c542c75c2f7f7306699276.

[39] B. L. Guellec, A. Lefèvre, C. Geay, L. Shorten, C. Bruge, L. Hacein-Bey, P. Amouyel, J.-P. Pruvo, G. Kuchcinski, A. Hamroun. Performance of an open-source large language model in extracting information from free-text radiology reports. Radiology: Artificial Intelligence 6 (2024). URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85201401840&doi=10.1148%2fryai.230364&partnerID=40&md5=bf2b6bafcd08728fcdb45e8652e169aa. doi:10.1148/ryai.230364.

[40] García-Barragán, A. González Calatayud, O. Solarte-Pabón, M. Provencio, E. Menasalvas, V. Robles. Gpt for medical entity recognition in spanish. Multimedia Tools and Applications (2024). URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-851

91098262&doi=10.1007%2fs11042-024-19209-5&partnerID=40&md5=ff4502258e8daa8f5a2cea6983c77e61. doi:10.1007/s11042-024-19209-5.

[41] T. Labbe, P. Castel, J. M. Sanner, M. Saleh, in: Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), IEEE, 2023, pp. 1–4. URL: https://doi.org/10.1109/EMBC40787.2023.10340611. doi:10.1109/EMBC40787.2023.10340611.

[42] S. Kim, S. Kim, Y. Kim, J. Park, S. Kim, M. Kim, C. H. Sung, J. Hong, Y. Lee. Llms analyzing the analysts: Do bert and gpt extract more value from financial analyst reports? ICAIF 2023 - 4th ACM International Conference on AI in Finance (2023) 383 – 391. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85179851731&doi=10.1145%2f3604237.3627721&partnerID=40&md5=31621087ebb72014cc9d09536f9388e8. doi:10.1145/3604237.3627721.

[43] F. Maibaum, J. Kriebel, J. N. Foege. Selecting textual analysis tools to classify sustainability information in corporate reporting. Decision Support Systems 183 (2024). URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85196207417&doi=10.1016%2fj.dss.2024.114269&partnerID=40&md5=e22529fac172690896efd8d7ddb3ed0b. doi:10.1016/j.dss.2024.114269.

[44] A. Usmanova, R. Usbeck. Structuring sustainability reports for environmental standards with llms guided by ontology. ClimateNLP 2024 - 1st Workshop on Natural Language Processing Meets Climate Change, Proceedings of the Workshop (2024) 168 – 177. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85204423193&partnerID=40&md5=37d7cf082d5590cc536fab064c9a9cd2.

[45] A. Dimmelmeier, H. C. Doll, M. Schierholz, E. Kormanyos, M. Fehr, B. Ma, J. Beck, A. Fraser, F. Kreuter. Informing climate risk analysis using textual information – a research agenda. ClimateNLP 2024 - 1st Workshop on Natural Language Processing Meets Climate Change, Proceedings of the Workshop (2024) 12 – 26. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85204450168&partnerID=40&md5=12934b95fec86615d3c562bd5e219e3d.

[46] S. Jose, K. T. Nguyen, K. Medjaher, R. Zemouri, M. Lévesque, A. Tahan. Bridging expert knowledge and sensor measurements for machine fault quantification with large language models. IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM (2024) 530 – 535. doi:10.1109/AIM55361.2024.10637229.

[47] Y. Li, X. Peng, J. Li, X. Zuo, S. Peng, D. Pei, C. Tao, H. Xu, N. Hong. Relation extraction using large language models: a case study on acupuncture point locations. Journal of the American Medical Informatics Association 31 (2024) 2622 – 2631. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85206878773&doi=10.1093%2fjamia%2focae233&partnerID=40&md5=45a63af5bcb7ee3d118680a8b69692a9. doi:10.1093/jamia/ocae233.

[48] S. Yang, J. Zhu, J. Wang, X. Xu, Z. Shao, L. Yao, B. Zheng, H. Huang. Retrieval-augmented generation with quantized large language models: A comparative analysis. ACM International Conference Proceeding Series (2023) 120 – 124. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85192813473&doi=10.1145%2f3653081.3653102&partnerID=40&md5=a276fea953ad514ab1b330f2af12e602. doi:10.1145/3653081.3653102.

[49] Y. Cao, L. Yang, C. Wei, H. Wang. Financial text sentiment classification based on baichuan2 instruction finetuning model. 2023 5th International Conference on Frontiers Technology of Information and Computer, ICFTIC 2023 (2023) 403 – 406. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85188071930&doi=10.1109%2fICFTIC59930.2023.10454145&partnerID=40&md5=e77231bbbe131c4aa788803ef6b4dbe5. doi:10.1109/ICFTIC59930.2023.10454145.

[50] B. Rajan, S. Carradini, C. Lauer. The arizona water chatbot: Helping residents navigate a water uncertain future one response at a time. Conference on Human Factors in Computing Systems - Proceedings (2024). URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85194187355&doi=10.1145%2f3613905.3650919&partnerID=40&md5=22e803ad85aa09b0f4ff10644bbac793. doi:10.1145/3613905.3650919.

[51] A. Sonnenburg, B. van der Lugt, J. Rehn, P. Wittkowski, K. Bech, F. Padberg, D. Eleftheriadou, T. Dobrikov, H. Bouwmeester, C. Mereu, F. Graf, C. Kneuer, N. I. Kramer, T. Blümmel. Artificial intelligence-based data extraction for next generation risk assessment: Is fine-tuning of a large language model worth the effort? Toxicology 508 (2024). URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85202728648&doi=10.1016%2fj.tox.2024.153933&partnerID=40&md5=dad04e87bb23e5a13ac40c35555e53d5. doi:10.1016/j.tox.2024.153933.

[52] M. M. Dagli, Y. Ghenbot, H. S. Ahmad, D. Chauhan, R. Turlip, P. Wang, W. C. Welch, A. K. Ozturk, J. W. Yoon. Development and validation of a novel ai framework using nlp with llm integration for relevant clinical data extraction through automated chart review. Scientific Reports 14 (2024). URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85208603126&doi=10.1038%2fs41598-024-77535-y&partnerID=40&md5=9ebe7483d59f1720f8bb598da0d75c44. doi:10.1038/s41598-024-77535-y.

[53] V. Vizgirda, R. Zhao, N. Goel. Socialgenpod: Privacy-friendly generative ai social web applications with decentralised personal data stores. WWW 2024 Companion - Companion Proceedings of the ACM Web Conference (2024) 1067 – 1070. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85194498225&doi=10.1145%2f3589335.3651251&partnerID=40&md5=713514567dc37a74da0b7ca7cac62a73. doi:10.1145/3589335.3651251.

[54] S. M. Jones-Jang, Y. J. Park. How do people react to AI failure? automation bias, algorithmic

aversion, and perceived controllability. Journal of Computer-Mediated Communication 28 (2023) zmac029.

[55] Y. Huang, T. Lv, L. Cui, Y. Lu, F. Wei, in: Proceedings of the 30th ACM international conference on multimedia, pp. 4083–4091.

[56] Y. Ding, J. Lee, S. C. Han. Deep learning based visually rich document content understanding: A survey. arXiv preprint arXiv:2408.01287 (2024).

[57] T. Douzon, S. Duffner, C. Garcia, J. Espinas, in: International Conference on Document Analysis and Recognition, Springer, pp. 47–64.

[58] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, in: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers), pp. 4171–4186.

[59] J.-N. Li, J. Guan, W. Wu, Z. Yu, R. Yan. 2d-tpe: Two-dimensional positional encoding enhances table understanding for large language models. arXiv preprint arXiv:2409.19700 (2024).

[60] X. Dong, P. Zhang, Y. Zang, Y. Cao, B. Wang, L. Ouyang, S. Zhang, H. Duan, W. Zhang, Y. Li, et al. Internlm-xcomposer2-4khd: A pioneering large vision-language model handling resolutions from 336 pixels to 4k hd. Advances in Neural Information Processing Systems 37 (2024) 42566–42592.

[61] L. Fernandez-Luque, M. Imran. Humanitarian health computing using artificial intelligence and social media: A narrative literature review. International journal of medical informatics 114 (2018) 136–142.

[62] L. Gately. A narrative review of the potential use of generative artificial intelligence in educational research practices in higher education. Studies in Technology Enhanced Learning 4 (2024).

[63] L. Kang, R. Tito, E. Valveny, D. Karatzas, in: International Conference on Document Analysis and Recognition, Springer, pp. 219–232.

[64] J. Chen, R. Zhang, Y. Zhou, T. Yu, F. Dernoncourt, J. Gu, R. A. Rossi, C. Chen, T. Sun. Sv-rag: Lora-contextualizing adaptation of large multimodal models for long document understanding. arXiv preprint arXiv:2411.01106 (2024).

[65] C. Wohlin, in: Proceedings of the 18th international conference on evaluation and assessment in software engineering, pp. 1–10.

[66] D. Wicks. The coding manual for qualitative researchers. Qualitative research in organizations and management: an international journal 12 (2017) 169–170.

[67] Q. Dong, L. Kang, D. Karatzas, in: International Workshop on Document Analysis Systems, Springer, pp. 57–70.

[68] M. Faysse, H. Sibille, T. Wu, B. Omrani, G. Viaud, C. Hudelot, P. Colombo, in: The Thirteenth International Conference on Learning Representations.

[69] J. Cho, D. Mahata, O. Irsoy, Y. He, M. Bansal. M3docrag: Multi-modal retrieval is what you need for multi-page multi-document understanding. arXiv preprint arXiv:2411.04952 (2024).

[70] T. Blau, S. Fogel, R. Ronen, A. Golts, R. Ganz, E. Ben Avraham, A. Aberdam, S. Tsiper, R. Litman, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 15598–15607.

[71] M. S. Nacson, A. Aberdam, R. Ganz, E. B. Avraham, A. Golts, Y. Kittenplon, S. Mazor, R. Litman. Docvlm: Make your vlm an efficient reader. arXiv preprint arXiv:2412.08746 (2024).

[72] S. Appalaraju, P. Tang, Q. Dong, N. Sankaran, Y. Zhou, R. Manmatha, in: Proceedings of the AAAI conference on artificial intelligence, volume 38, pp. 709–718.

[73] Ł. Borchmann, M. Pietruszka, W. Jaśkowski, D. Jurkiewicz, P. Halama, P. Józiak, Ł. Garncarek, P. Liskowski, K. Szyndler, A. Gretkowski, et al. Arctic-tilt. business document understanding at sub-billion scale. arXiv preprint arXiv:2408.04632 (2024).

[74] Z. Tang, Z. Yang, G. Wang, Y. Fang, Y. Liu, C. Zhu, M. Zeng, C. Zhang, M. Bansal, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 19254–19264.

[75] J. Gu, X. Meng, G. Lu, L. Hou, N. Minzhe, X. Liang, L. Yao, R. Huang, W. Zhang, X. Jiang, et al. Wukong: A 100 million large-scale chinese cross-modal pre-training benchmark. Advances in Neural Information Processing Systems 35 (2022) 26418–26431.

[76] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, et al. Lora: Low-rank adaptation of large language models. ICLR 1 (2022) 3.

[77] A. R. Hevner, S. T. March, J. Park, S. Ram. Design science in information systems research. MIS quarterly (2004) 75–105.

[78] K. Peffers, T. Tuunanen, M. A. Rothenberger, S. Chatterjee. A design science research methodology for information systems research. Journal of management information systems 24 (2007) 45–77.

[79] T. Lee, H. Tu, C. H. Wong, W. Zheng, Y. Zhou, Y. Mai, J. Roberts, M. Yasunaga, H. Yao, C. Xie, et al. Vhelm: A holistic evaluation of vision language models. Advances in Neural Information Processing Systems 37 (2024) 140632–140666.

[80] A. F. Biten, R. Tito, A. Mafla, L. Gomez, M. Rusinol, M. Mathew, C. Jawahar, E. Valveny, D. Karatzas, in: 2019 International Conference on Document Analysis and Recognition (ICDAR), IEEE, pp. 1563–1570.

[81] D. Peer, P. Schöpf, V. Nebendahl, A. Rietzler, S. Stabinger. Anls*–a universal document processing metric for generative large language models. arXiv preprint arXiv:2402.03848 (2024).

[82] H. Ji, Q. Si, Z. Lin, W. Wang, in: Proceedings of the 32nd ACM International Conference on Multimedia, pp. 38–47.

[83] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi. Bertscore: Evaluating text generation with bert. arXiv preprint arXiv:1904.09675 (2019).

[84] C. Deng, J. Yuan, P. Bu, P. Wang, Z.-Z. Li, J. Xu, X.-H. Li, Y. Gao, J. Song, B. Zheng, et al. Longdocurl: a comprehensive multimodal long document benchmark integrating understanding, reasoning, and locating. arXiv preprint arXiv:2412.18424 (2024).

[85] Y. Liu, D. Iter, Y. Xu, S. Wang, R. Xu, C. Zhu. G-eval: Nlg evaluation using gpt-4 with better human alignment. arXiv preprint arXiv:2303.16634 (2023).

[86] S. Kim, J. Suk, S. Longpre, B. Y. Lin, J. Shin, S. Welleck, G. Neubig, M. Lee, K. Lee, M. Seo. Prometheus 2: An open source language model specialized in evaluating other language models. arXiv preprint arXiv:2405.01535 (2024).

[87] A. Hu, H. Xu, L. Zhang, J. Ye, M. Yan, J. Zhang, Q. Jin, F. Huang, J. Zhou. mplug-docowl2: High-resolution compressing for ocr-free multi-page document understanding. arXiv preprint arXiv:2409.03420 (2024).

[88] C. D. Manning, P. Raghavan, H. Schütze. Boolean retrieval. Introduction to information retrieval (2008) 1–18.

[89] N. Craswell, in: Encyclopedia of database systems, Springer, 2016, pp. 1–1.

[90] J. Gu, X. Jiang, Z. Shi, H. Tan, X. Zhai, C. Xu, W. Li, Y. Shen, S. Ma, H. Liu, et al. A survey on llm-as-a-judge. arXiv preprint arXiv:2411.15594 (2024).

[91] Y. Liu, D. Iter, Y. Xu, S. Wang, R. Xu, C. Zhu. G-eval: Nlg evaluation using gpt-4 with better human alignment. arXiv preprint arXiv:2303.16634 (2023).

[92] O. Khattab, M. Zaharia, in: Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, pp. 39–48.

[93] T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer. Qlora: Efficient finetuning of quantized llms. Advances in neural information processing systems 36 (2023) 10088–10115.

[94] M. Jostmann, H. Winkelmann, in: Wirtschaftsinformatik 2024 Proceedings, 115. URL: https://aisel.aisnet.org/wi2024/115.

[95] A. Bertolino, P. Braione, G. D. Angelis, L. Gazzola, F. Kifetew, L. Mariani, M. Orrù, M. Pezzè, R. Pietrantuono, S. Russo, et al. A survey of field-based testing techniques. ACM Computing Surveys (CSUR) 54 (2021) 1–39.

[96] X. Y. Lee, S. Akatsuka, L. Vidyaratne, A. Kumar, A. Farahat, C. Gupta. Reliable decision-making for multi-agent llm systems. arXiv preprint arXiv:2406.04092 (2025).

A

TRAINING DOCUMENTS

The documents used for training are listed below in a fixed order. This ordering implies that the top-k documents correspond to the dataset configurations with $k = 5$, 10, or 20 documents, respectively.

- Banco Sabadell SA_2024
- Swedbank AB_2024
- HSBC Continental Europe_2024
- Commerzbank AG_2024
- Nordea Bank Abp_2024
- Triodos Bank NV_2024
- Talanx AG_2024
- Credit Agricole SA_2024
- Nexi SpA_2024
- De Volksbank NV_2024
- Tatra banka as_2024
- Arion Bank Annual and Sustainability Report 2024
- SocieteGenerale2024
- Aker Horizons ASA_2024

- Helvetia2024
- MB_Report_ING_Group-2024-12-31-0-en
- EXOR 2024 Annual Report
- NykreditBank
- Allied Irish Banks PLC_2024
- DZ BANK AG_2024

B

PROMPT FOR TRAINING DATA ANNOTATION

prompt = """ You are an assistant specialized in Multimodal RAG tasks.

The task is the following: given an image from a pdf page, you will have to generate QUESTIONS that can be asked by a user to retrieve information from a large documentary corpus.

The questions should be about one of the following topics: - The Corporate Sustainability Reporting Directive (CSRD) - The European Sustainability Reporting Standards (ESRS) - Environmental, social and governance (ESG) factors - TCFD (Task Force on Climate-related Financial Disclosures)

Questions can be aimed at: - The disclosure of certain topics (yes/no) - The exact values (quantitatives) of certain indicators (e.g. certain emissions, or proportions of employees) - Getting to know the narratives behind a certain topic (e.g. the governance of the company)

The question should be about the subject of the page, and the answer needs to be found in the page.

Remember that the question is asked by a user to get some information from a large documentary corpus that contains multimodal data. Generate a question that could be asked by a user without knowing the existence and the content of the corpus.

Generate as well the answer to the question, which should be found in the page. And the format of the answer should be a list of words answering the question.

Preferably, include a QUESTION about a table, a QUESTION about a figure and a QUESTION about the text, but, only if those modals are present in the page. Don't ask two questions about

the same topic.

Generate at most THREE pairs of questions and answers per page in a dictionary with the following format, answer ONLY this dictionary NOTHING ELSE:

```
{
    "questions": [
        {
            "question_table": "XXXXXX",
            "answer": ["YYYYYY"]
        },
        {
            "question_text": "XXXXXX",
            "answer": ["YYYYYY"]
        },
        {
            "question_figure": "XXXXXX",
            "answer": ["YYYYYY"]
        }
    ]
```

where XXXXXX is the question and ['YYYYYY'] is the corresponding list of answers that could be as long as needed.

Note: If there are no questions to ask about the page, return an empty list. Focus on making relevant questions concerning the page.

Here is the page:"""

C

Selection of Papers and Extracted Information

Table A1 presents the synthesized information from the selected papers, structured around the defined research questions. The table consists of six columns: author, study description, benefits, requirements, validation methods, and disadvantages.

The author column references the specific study. The study description provides a brief summary to facilitate a quick understanding of the study's scope and findings. This is particularly useful because the purpose of each paper varies. Some studies were included for their insights on information extraction, while others were selected for their focus on requirements.

The benefits, requirements, validation methods, and disadvantages columns contain extracted information relevant to the research questions. Additionally, SLR identified research themes that emerged from the literature. These themes were primarily discovered through inductive coding, as described in Section 2.1.2, and are therefore not included in this table.

Table A1: Table summarizing research findings and characteristics

Author	Study description	Benefits	Requirements	Validation methods	Disadvantages
[50]	• A chatbot was created which can be used by citizens in Arizona to get information on how to reduce problems of drought in the region. • Used OpenAI GPT in combination with RAG. • RAG consists of 4 stages: ingestion, retrieval, synthesis, output generation. • By using the custom GPT, and making use of the API, it is possible to apply more filtering, add more documents for RAG with factual information, and add more requests. Also, it is possible to design a custom interface.	• Creating one ourselves; • Concise formulation of information the user asks for. • Provide factual information.	• Security checks: filter content which might violate OpenAI, prompt injection check which enables the system to correct malicious prompts, user intent check which verifies whether the user might have intent to harm self or others. • Checks and corrections for incorrect information/data in output evaluation. • RAG helped with factuality. • Balance between checks and processing time needed to be found. • Users did not want too long information (chatbot specific, so excluded). • Make use of API, and not a custom GPT.	• Comparison with OpenAI Custom GPT 'Aqua Advisor', to whom they asked similar questions. • User testing, qualitative. • Iteratively improving the product, by adding messages, adding files, etc.	• Due to the probabilistic nature of LLM, results are not always accurate. • RAG and factual checks are mitigations to this. • Also, with extensive user testing, some hard scripts were added to ensure factual information.
[20]	• Extracting data from a free-text report about lung cancer patients, returning it in a structured way. • Comparison between GPT-4 and ChatGPT. • 98.6% accuracy in extracting lesion diameters (GPT-4), while 84.0% for normal. • F1: 9.96 vs 0.91. • Also higher scores for factual correctness, lower confabulation, and accuracy for GPT-4.	• No need for training. • Potentially more efficient. • No need for retraining.	• NLP post-processing to deliver the information in a standardized format, otherwise it might differ a little from prompt to prompt. • Sensitive data handling. This might best be done by running an open source program on a local infrastructure.	Accuracy, F1 score, rate of confabulation.	• Confabulation. • Different output format, so need for post-processing to have a strict format. • Third-party application handling sensitive data.
[48]	• Using quantization + RAG to make a smaller model which can be run locally, which was necessary because of confidential info.	• Quantization is a way to compress LLMs. • LLMs are capable of retrieving data from free text, though are not so factual. • Solution to this is retrieval augmented generation. • Study used ChatGLM2, a Chinese variant of LLM, which is relatively small (6 billion params) and therefore it can be run locally.	• Local deployment due to confidentiality of the data.	ROUGE scores, BLEU, embedding similarity, running time, model size. Choice was a consideration of time and accuracy.	• Quantization was significantly slower than ChatGLM standard version, but this was also because the latter was run on GPU and the other 2 not.

Table A1 (continued)

Author	Study description	Benefits	Requirements	Validation methods	Disadvantages
[40]	• Named entity extraction from electronic health records (Spanish), comparing BERT vs. GPT for NER. • Prompting techniques. • Local BERT for also fine-tuning and such. GPT ran online.	• Similar performance in named entity recognition in electronic health records (Spanish). • While not needing extensive data annotation or model pretraining.	X	• Accuracy, precision, recall, F-score; BERT model was evaluated using k-fold. • Labeled corpus of named entities and the text.	• LLM is somewhat slower than a smaller alternative (BERT) (milliseconds).
[49]	• Extraction of financial information: sentiment analysis, event detection/information extraction using LLM (Baichuan2-7b).	• High accuracy. • Few-shot learning.	X	• Accuracy.	X
[52]	• Extracting spinal surgery data from electronic health records, with the help of a combination of NLP and LLM. • Identification of: surgery type, levels operated on, n of disks removed, detection of intraoperative incidental durotomies. • Shows potential of LLM approach in providing reliable data.	• Replacing fatigue and error-prone human extraction of electronic health report data. • Outperformed professionals in training. • By exploiting Turbo, also complex text could be examined, which is usually hard due to specialized terminologies and contextual nuances.	X (not done)	• 95% confidence interval around certain values, and then a precision, recall, and F1-score.	• Sometimes a hallucination.
[1]	Local deployment of LLM to preserve privacy, LLM is extracting quantitative data. Used Llama 2 for this. Use of single-shot and definition of thought prompting was used.	• Zero-shot learning / few-shot, • Possibility to extract from unstructured free text data.	• On-premise running, because data may not be shared to the cloud due to Dutch regulation. • Local running (so model cannot be too big). • Model still needs to be able to make deductions, as in extracting implicit information.	• Positive predictive value, sensitivity, specificity, negative predictive value, accuracy. It was tested on whether the model was able to detect whether the patients had one of the 5 illnesses. • Quantitative analysis, comparing output of the model to the ground truth, which was obtained by three blinded observers.	• Often runs in the cloud, which is not allowed in the EU for many sorts of data. • When running a little parameter version of Llama, it was not able to produce correct JSON output.
[21]	• Structuring ecological data from unstructured text. • Few-shot learning. • Comparing GPT-3.5, GPT-4, and LLaMA-2-70B.	• Handling complex data and capturing patterns in it. • Model size impacts the model's ability to generate a structured representation of the information. • GPT-4 was able to obtain similar performance to best software in industry.	• Resources for on-premise running.	Validation based on 4 things: • Ability to distinguish relevant/irrelevant sources. • Accuracy of the extracted information. • Ability to geocode the spatial entities in test. • Putting the info in structured format. • Metrics: specificity, precision, negative predictive value, accuracy, area under curve, bootstrap, confusion matrix, model size.	• For ChatGPT: it is costly to scan through thousands of articles. • The less accurate Llama can be run locally, though this requires than substantial resources.
[18]	Generation of knowledge graphs by employing LLMs, In-Context Learning, and the RAG-paradigm to extract structured insights related to ESG aspects from companies' sustainability reports.	• Semantic understanding of LLMs. • Ability of LLM to store factual knowledge. • Using natural language to give instructions to the LLM, to let it do certain tasks. • Few-shot learning.	• RAG: improves performance.	• Triple generation was validated by examining sentence coverage.	X

Table A1 (continued)

Author	Study description	Benefits	Requirements	Validation methods	Disadvantages
[44]	• Using ontology and knowledge graph to structurally extract information with an LLM from sustainability reports. The focus is on non-quantifiable report aspects.	• Benefit of LLM is that state of the art performance is obtained without much training.	X	• Comparison with annotators (3). • Topical match. • Vagueness.	• Slow, documents need to be divided in chunks: better preprocessing is needed. • Ontology requires further evaluation.
[45]	• To estimate loan risks for bank or regulators, estimating the climate risk is useful. Now companies need to create sustainability reports; this information can be used to especially estimate transition risks. This paper does this using, amongst others, LLM.	• High precision.	• Multimodal approach is needed to get to excellent performance.	• Annotated reports (39). • Focus on GHG emissions. • Annotated Scope 1,2,3, available in spreadsheet.	• Mediocre recall. • Preprocessing + text therein (conversion of PDF to machine-readable), cost-efficiency, validation against benchmarks. • Lack of data for many of the samples. • Different ways of measurement of GHG measurement.
[53]	• Architecture has been created in which Solid is leveraged (a platform to allow decentralized personal data storage) so that users within an ecosystem can determine what they want to share with an LLM so that questions to the LLM can be shared. Solving problems with privacy and LLMs.	• Users have more control over their data. • Data is not stored for long times at providers. • Less vendor lock-in; this system should allow users to easily switch to other service providers (e.g., another LLM).	• Limiting sensitive data sharing. • Privacy. • Limited vendor lock-in (for users this is of course desired).	X	X
[51]	• Fine-tuned ChatGPT Curie model (ChatGPT-3) was compared to two benchmarks: text-davinci-002 and text-davinci-003 models. • Detection of dangerous substance using LLM in test systems measuring biological activities of the substance.	• Fine-tuning increases domain understanding. • Fine-tuning leads to better complex task handling.	• Fine-tuning deemed needed when more than just simple information extraction needs to be done.	• Precision, recall, F1. • Manual assessment of quality of extraction by expert.	• For single pieces of information, fine-tuning did not increase performance.
[42]	• Extracting sentiment from financial analysts' reports, this is then used to define investment strategies.	• Pre-trained was able to outperform fine-tuned. • Fine-tuning shows to be effective in increasing domain knowledge. • Complex data can be handled.	• Smaller (non-generative models) need fine-tuning.	• Indirect.	X
[19]	• A process had been automated (end-to-end): a process of doing health economics research, which examines the feasibility of adding a mitigation in a certain location.	• LLM is able to make highly accurate extractions. • Can also reason based on information extraction.	X	• Accuracy comparison with manual evaluation as baseline.	• Difficult reproducing results because the same question might lead to different answers. • Also, having the API online version of GPT makes it hard to set a version, and there might be differences between the versions. • Token limit leads to partly passing documents to GPT.

Table A1 (continued)

Author	Study description	Benefits	Requirements	Validation methods	Disadvantages
[43]	• Several NLP methods were compared in extracting sustainability data from text.	• GPT (AI method, generative) can be used without fine-tuning (zero-shot/one-shot). • GPT's outperform previous LLM methods which required fine-tuning when extracting sustainability info from text.	• For pre-generative methods: fine-tuning is essential. • For (larger) GPT models, prompt engineering is more effective.	• For the manually labeled dataset of 75,000 sentences: Cohen's kappa was used to determine agreement amongst the labelers. • Validation of the model: model correlation, recall, precision, F1, precision-recall curves (AUC), lift curves.	• Many studies used ways to extract information, though did not validate properly.
[23]	• InstructGPT was tested on information extraction from medical reports. This was done by using 208 publications. 4 things had to be extracted, namely: age (82%), object (94%), body part (89%). Those percentages were a bit higher when medical reports were excluded where no metrics could be extracted.	• No need for data preprocessing, programming skills, extensive training data. • No high performant PC is needed because it can be accessed via API in the cloud. • Understands various languages.	X	• Accuracy, confusion matrix.	• For optimal performance, fine-tuning performed better, which is a higher effort.
[47]	• Obtaining optimal acupuncture locations by extraction from free text. Fine-tuned and pre-trained LLM models were tested.	• Able to extract from textual information. • Fine-tuning GPT enables it to learn nuances/complexities.	X	• Comparison with: • F1-score, precision, recall. • Micro-average F1 score. • Using an annotated dataset (manually by themselves) as the gold standard. • High averages for all of them (around 0.9) for the solution of fine-tuned GPT-3.5 model.	• Has more difficulty with points which are closer to each other, possibly due to higher complexity. • Model struggles to correctly grasp relations across multiple sentences. • Lack of domain knowledge in not-fine-tuned.
[25]	• Data extraction from tables; here, BARD vs. GPT-4 is being compared.	X	• External calculator to reduce hallucinations. • Combining LLM for table extraction techniques.	Accuracy.	• LLMs seem not ready to handle numerical data reliably. • Absent data leads to hallucinations.
[38]	• Fine-tuned LLM was used to extract textual evidence for ESG scores.	• SOTA performance.	X	• Precision. • F1 score, recall, accuracy.	• If during fine-tuning a word was often tied to a certain label, this might cause this label to be given also when this word is placed in a sentence for another context. Example words here are oxygen and environment.
[46]	Machine state estimation by combining free text with other metrics.	• Free text understanding.	X	• Indirect.	X
[39]	Extraction/classification of text.	• Ran locally. • No need to share sensitive data. • Made explainable through prompt engineering. • Accurate.	• Handling sensitive data → local development. • Cost. • Explainability.	• Sensitivity, specificity, recall, F1, accuracy, calculation time.	X

Table A1 (continued)

Author	Study description	Benefits	Requirements	Validation methods	Disadvantages
[41]	• Phenotype extraction from text using LLM.	X	X	• Precision. • Recall. • F1-score.	• Performance is highly dependent on prompt engineering. • LLM shows difficulty when more complex tasks are required.
[22]	• LLM has been used to filter information from annual reports. Here a part is often also the GHG/sustainability. It was shown that using image extraction might lead to some more hallucination. A dataset had been created of 1000 annotated papers.	X	• Multimodal approach is effective for processing of info from financial reports.	• For information extraction: recall, precision, F1, accuracy, perfect match rate, residual analysis.	• Performs worse with tabular and numerical data. Mitigations for this are needed. Several proposals in this paper.
[26]	• Combination of RAG and LLM to extract ESG data from ESG reports: so that a standardized structured dataset can be created (of HKEx companies). An extension methodology had been used to enhance LLM performance in the ESG analytical tasks; they call it ESGReveal, which adds ESG metadata that helps in building useful queries. • Gains of 9.9% in GPT-3, 2.5% in GPT-3.5. Outperformed previous literature with 20%, by GPT-4, 76.9% on data extraction tasks, 83.7 on data disclosure.	• LLM makes unstructured text analysis feasible.	X	• Accuracy for detecting the presence in the document, accuracy for having the correct value as well.	–

Table A1 (continued)

Author	Study description	Benefits	Requirements	Validation methods	Disadvantages
[27]	• Project Gaia extracts CO2 related information but in the future also other KPIs, from sustainability/corporate reports, and puts them in a structured database. With the goal of finally creating a database which can be used by analysts to do easy comparisons. Also, they want to add this to a web page so that this is easily accessible for anyone (for research purposes). • Multiple design choices were made to make extraction more easy.	• Understanding several languages.	• Shouldn't rely on hard definitions, but instead the meaning should align. This solves that there is ambiguity in the naming between different companies/sectors. • Some post-processing of errors, and also labeling of potential errors. Therefore, finding errors also. • Proper prompt engineering is needed (which was done for GAIA) to reduce hallucinations if information is not present. • Mitigations against hallucination: • Choosing the best model. • Prompt engineering. • Temperature to 0 in the prompts. • Those 3 things were able to significantly reduce the hallucinations.	• Accuracy, divergence rate. • Benchmarking in different ways. The above two ACC and DIV are for human annotation also: • Compared the percentage of reports which report Scope 1,2,3 according to GAIA and according to the benchmark TCFD. • Also comparison to commercial data sources reporting Scope 1 emissions.	• "LLMs' long response times, randomness (non-repeatability) in their responses and hallucinations pose a real challenge in designing an LLM-based application." • Currently, infographics (charts, and stuff) still pose a problem for LLM, or at least for GAIA. • Overconfidence. If the LLM is seeing certain data and it is not stated that something is true, the model tends to be very confident that something is not true; however, it could also be simply not mentioned. (bottom page 27)

D

QA-RESULTS

CLASSIFICATION

Table D.1: Classification performance per model where 5, 10 or 20 pages are given

Model	Given Pages	Accuracy	Precision	F1
Qwen2-VL	5	0.78	0.0	0.0
Qwen2-VL	10	0.72	0.0	0.0
Qwen2-VL	20	1.00	0.0	0.0
GPT-4o	5	0.84	0.0	0.0
GPT-4o	10	0.80	0.0	0.0

CLIMRETRIEVE

Table D.2: ClimRetrieve performance per model where 5, 10, or 20 pages are given

Model	Given Pages	Accuracy	Precision	Recall	F1
Qwen2-VL	5	0.792	0.815	0.786	0.800
Qwen2-VL	10	0.679	0.720	0.643	0.679
GPT-4o	5	0.849	0.813	0.929	0.867
GPT-4o	10	0.792	0.743	0.929	0.825
GPT-4o	20	0.849	0.813	0.929	0.867

QUANTIFICATION

Table D.3: Quantification performance per model where 5, 10, or 20 pages are given

Model	Given Pages	G-Acc	G-Acc-alt
Qwen2-VL	10	0.057	1.883
Qwen2-VL	5	0.100	2.000
GPT-4o	5	0.157	2.117
GPT-4o	20	0.186	2.457
GPT-4o	10	0.214	2.514

SCOPE123

Table D.4: Scope123 performance per model where 5, 10, or 20 pages are given

Model	Given Pages	G-Acc	G-Acc (Likert)
GPT-4o	10	0.63	3.68
GPT-4o	20	0.64	3.73
GPT-4o	5	0.58	3.51
Qwen2-VL	10	0.24	2.16
Qwen2-VL	5	0.23	2.23

E

EXAMPLE PROMPT PER DATASET

CLIMRETRIEVE

Query

Your task is to, based on the given context answer the question with "Yes" or "No". Do NOT elaborate further.

The question is:

Your task is to, based on the given context answer the question with "Yes" or "No".

The question is:

Do the environmental/sustainability targets set by the company align with external climate change adaptation goals/targets?

Golden Answer

Yes

CLASSIFICATION

As can be seen, the query the model also got the assessment to give a clarification. However, this was not used for the analysis, since a positive or negative result was extracted using a Python script.

Query

Does the institution disclose information on the theme of 'Climate Change' related to ESRS E1? If not, please indicate so by returning 'no'. If they do, how they refer to this disclosure, e.g.

by saying "Yes, our climate strategy", or "Yes, pollution", for the themes climate strategy and pollution respectively.

Golden Answer

No.

QUANTIFICATION

Here, we aim to have precise extractions per year. By letting the model disclose for each year separately whether something was disclosed, and what the exact value was.

Query

What were the total CO2 emissions in 2021, 2022 and 2023, if disclosed? Answer the question as in the following example:

2021: not disclosed 2022: 34 ktons CO2 2023: 10 Mt CO2

Use the unit which is also used in the document.

Golden Answer

2021: 26 ktons CO2 2022: 29 ktons CO2 2023: 29 ktons CO2

SCOPE123

The idea behind this query, is that first a description is given (description prompt), and then the task is given. And lastly few-shot learning is applied.

Query

scope_1 (double): total scope 1 emissions in metric tonnes of CO2eq.

Your task is to extract from the report the scope 1 emissions for the year 2021.

Respond by only giving the value in metric tonnes CO2eq.

So for example if the report says: "In 2021, our scope 1 emissions were 1000 metric tonnes CO2eq." You should respond with: 1000 If the report says: "In 2021, our scope 1 emissions were 5 kilotonnes of CO2eq." You should respond with: 5000

The scope 1 emissions for the year 2021 are:

Golden Answer

6000

F

ALL BENCHMARK RESULTS

Model names / Benchmark	DocVQA	MP-DocVQA	MP-DocVQA (800 docs)	MM-LongBenchDoc
Qwen-VL (very high on MP-DocVQA)	94.5	-	-	-
ILMXC24KHD	90	-	-	-
UDOP	87.8	-	-	-
TiLT (large)	87.05	-	-	-
D-Lava	85.91	-	-	-
TiLT (base)	83.92	-	-	-
LayoutLMv2 / LayoutLMv3	83.37	-	-	-
LayoutLM3Base	78.76	-	-	-
mPlug-docowl	62.2	-	-	-
ERNIE-Layoutlarge	0.88	-	-	-
SelfAttn scoring MPDocVQA	-	-	0.6199	-
DocVLM	92.8	84.5	-	-
M3DocRAG	-	84.4	-	21
Arctic-TILT	90.2	81.2	-	25.8
GRAM	-	80.3	-	-
Wukong	85.1	76.9	-	-
DocFormerV2 (large)	87.84	76.4	-	-
SV-RAG	-	71	-	34
mPlug-docowl2	-	80.7	69.1	-
RM-t5	-	-	64.01	-
HiVT5 + multipage document validation dataset	-	-	X	-

Table F.1: Benchmark performance of various models across DocVQA, MP-DocVQA (standard and extended), and MM-LongBenchDoc.