LONDON'S GLOBAL UNIVERSITY
Department of Computer Science

Benchmarking the efficiency of Large Language Models on ESG data extraction in RAG system

Supervisor:

Canhui Liu

Faculty of Engineering

Department of Computer Science

University College London

A Project Report Presented in Partial Fulfillment of the Degree

MSc Artificial Intelligence for Sustainable Development

September 2024

Abstract

This study benchmarks the efficiency of large language models (LLMs) for environmental, social, and governance (ESG) data extraction in retrieval-augmented generation (RAG) systems. An evaluation dataset of 2,039 Q&A pairs based on HKEX ESG Reporting Guide KPIs and 15 ESG reports was constructed. Gold answer was made using ChatGPT-4 followed by manual check. The answer accuracy and speed of 14 LLMs with different architechtures and sizes were assessed. Findings reveal performance variations among the models, offering insights into their capabilities and limitations in ESG data extraction. This study contributes valuable insights for technology providers, guiding enhancements in ESG data extraction methodologies and highlighting the potential for further research in LLM efficiency.

Keywords— Benchmarking - Large Language Model (LLM) - ESG Data Extraction - Retrieval-Augmented Generation (RAG)

Table of Contents

List of Abbreviations

AI: Artificial Intelligence

ESG: Environmental, Social, Governance

HKEX: Hong Kong Exchanges and Clearing Limited

KPI: Key Performance Indicator

LLM: Large Language Model

LM: Language Model

NLM: Neural Language Model

NLP: Natural Language Processing

OCR: Optical Character Recognition

PLM: Pre-trained Language Model

Q&A: Question and Answer

RAG: Retrieval-Augmented Generation

SLM: Statistical Language Model

1 | Introduction

In an era where data drives decision-making, accurately extracting environmental, social, and governance (ESG) related data is crucial for informed decision-making in sustainable investment [1]. However, a comprehensive evaluation method or framework for the efficiency of LLMs in this context is still lacking. This report delves into the heart of this challenge, benchmarking the efficiency of LLMs within RAG systems for ESG data extraction.

ESG is a criterion used to evaluate how well a firm addresses various ethical and sustainable business challenges [2]. As the concept of ESG gains recognition and is embraced by a wide range of stakeholders, the ability to accurately and effectively extract ESG-related data from an increasing volume of disclosures becomes essential. Timely and precise ESG information is critical for stakeholders, including investors, regulators, and companies, to make informed decisions regarding investments, corporate strategies, and risk management [3].

Although ESG reports are crucial for every stakeholder, extracting ESG data from these reports is challenging. Traditionally, gathering, preprocessing, and analysing ESG data has been done by human analysts [4]. While stock exchanges and business websites provide investors with access to ESG reports, the large volume and diversity of these reports create substantial integration issues for compiling disclosure data at the corporate or industry level [5]. This makes the data extraction process labor-intensive and time-consuming, with a significant likelihood of human error [4].

With the development of artificial intelligence (AI), LLMs, alongside the RAG framework, can enhance the capacity to extract ESG data. A LLM is a deep learning algorithm (particularly transformer architectures) that has been trained on a vast amount of text data and has hundreds of billions or more parameters [6]. It can handle a wide

range of natural language processing (NLP) tasks such as text generation, translation, summarization, and question answering [7]. Models like GPT-4 [8], LLaMA 2 [9], and Claude [10] have demonstrated exceptional natural language understanding capabilities and wide applicability in information technology [11].

However, some limitations of LLMs include hallucinations (i.e., untrustworthy outputs produced by LLMs) [12], data currency, and lengthy contexts [5]. To overcome these problems, the RAG framework was developed. It uses semantic similarity computation to retrieve pertinent document chunks from external knowledge sources [13]. RAG improves the credibility and accuracy of model outputs through dynamic information retrieval [5], especially for knowledge-intensive tasks [13], such as ESG data extraction.

Recent research shows increasing interest in using LLMs and RAG approaches for ESG data extraction, with promising results in terms of accuracy and efficiency. However, their performance in accurately and efficiently extracting ESG-specific data has not been comprehensively evaluated.

The lack of comprehensive benchmarking of LLMs on ESG data extraction can lead to adverse consequences for several reasons. First, stakeholders such as investors and regulators have high expectations for the accuracy and reliability of ESG data. Without benchmarking, LLMs may not meet these expectations, leading to a loss of trust and potential legal or financial consequences. Second, we cannot determine which LLM performs the best or which is better suited for the ESG data extraction task without comparing the performance of a large number of different LLMs. Third, benchmarking can spur innovation by identifying areas for improvement. The absence of benchmarking may hinder the development of LLMs for ESG data extraction, as there will be no clear direction for where to focus research and development efforts.

Therefore, the overall aim of this study is to evaluate and compare the efficiency of LLMs in extracting ESG data in RAG system, identifying strengths and limitation of different

models in this context, and thus giving the further direction to optimise performance for real-world applications.

The specific aims and objectives of this project include:

1. Conduct a comprehensive review of existing research on LLMs, RAG systems and their implement on ESG data extraction task to identify gaps in the current literature.

2. Construct an evaluation ESG dataset suitable for testing the performance of LLMs on ESG data extraction.

3. Select a range of LLMs with different architectures and sizes for comparison.

4. Design a robust framework for benchmarking the efficiency of LLMs in ESG data extraction.

5. Execute benchmarking experiments to evaluate the performance of each LLM, and analyze the results to determine the efficiency of each LLM in extracting ESG data.

The report unfolds as follows. Chapter 2 delves into the foundational concepts of ESG, LLMs, and RAG. This chapter also reviews key studies that have explored LLM and RAG techniques in ESG data extraction and identifies the research gap. Chapter 3 describes the process of constructing an evaluation dataset, the LLMs selected for comparison, and the methods used for benchmarking model outputs. Chapter 4 systematically presents all the results of the LLMs' performance and analyzes incorrect model outputs. Chapter 5 discusses the results, critically evaluates the findings, and highlights the strengths and weaknesses of the study. Chapter 6 summarizes the key findings and conclusions, providing a comprehensive evaluation of the study. Chapter 7 focuses on potential future work for the improvement of this study.

2 | Related Work and Background

In this chapter, the background information of LLM and RAG would be provided, then a systematic literature review was conducted, focus on the studies on implementing LLMs for ESG data extraction in RAG system. Section 2.1 introduces the concept, history and usage of ESG. 2.2 provides the background knowledge for LLMs, including its significance, its Transformer architecture, the development of language model, its emergent abilities and limitations. Section 2.3 indicates how RAG can address the challenges of LLM, and three key steps in a typical RAG process. Section 2.4 reviews the previous work on implementing LLM and RAG for ESG data extraction, and discussed the gaps and opportunities for current research. Section 2.5 gives a summary of this chapter.

2.1 ESG

ESG stands for Environmental, Social, and Governance. It is a criteria utilized to evaluate how well a firm performs on a range of ethical and sustainable business challenges [2]. There is a long history of thought surrounding corporate governance and the interactions of firms with stakeholders, local communities, the environment, and society at large [14]. The pivotal event that in this transition towards ESG was a 1999 speech by Kofi Annan at the World Economic Forum in Davos, whereby corporate leaders were directly urged to join the UN in advocating principles that would lay the groundwork for a sustainable global economy. In 2004, The UN Global Compact Initiative's "Who Cares Wins" report, which aims to improve the way environmental, social, and corporate governance (ESG) aspects are taken into account when making investment decisions. This report marked the official introduction of the term ESG [14].

ESG disclosure measures a company's transparency and is essential for evaluating its performance in relation to ESG factors. ESG disclosure guidelines for corporations have been

implemented by a number of stock exchanges, such as Hong kong Exchanges (HKEX). ESG data have important usages for different stakeholders. For investors, they help them analyze long-term benefits, company reputation, and risk management, and thus make better investment decisions. Firms with strong ESG performance are often viewed as more stable and responsible, attracting more investment [14]. For regulators, they are increasingly concentrating on how ESG variables affect financial stability and corporate governance. For the public, ESG data provide lasting value for the goals of the United Nations and the guiding principles of the Global Compact, which more generally seek to promote social benefits, security, and sustainable development [14].

2.2 Large Language Models (LLM)

Language Model (LM) is the computational model that is capable of both compreheding and generating human language [15]. Generally speaking, LM seeks to forecast the likelihood of future (or absent) tokens by modeling the generative likelihood of word sequences [6]. Mathematically, LM aims to predict the next token y, given a context sequence X. To train the model, the probability of the supplied token sequence conditioned on the context is maximised. The probability can be represented by $P(y \mid X) = P\left(y \mid x_1, x_2, \ldots, x_{t-1}\right)$, where t represents the current position, and $x_1, x_2, \ldots, x_{t-1}$ indicate the tokens in the context sequence. The conditional probability can be broken down into a product of the probabilities at each position by applying the chain rule:

$$P(y \mid X) = P\left(y \mid x_1, x_2, \ldots, x_{t-1}\right) \tag{2.1}$$

where T represents sequence length. This is how LM generates a whole text sequence by autoregressively predicting each token at each position [15].

Although LM is one of the key methods to achieve machine language intelligence [6],

it faces difficulties include overfitting, the problem of uncommon or unseen words, and the challenge of capturing complex linguistic occurrences [15]. To overcome these obstacles, researchers keep enhancing LM architectures and training techniques [15]. The development of LM can be separated into four main stages. The first stage is the statistical language models (SLMs), which predict word according to Markov assumption (e.g. utilizing the most recent context to predict the next word). The second stage is neural language models (NLMs), which built the word prediction function based on the distributed word representations. The third stage is pre-trained language models (PLMs), which are models trained on a large general set of data and then fine-tuned for a specific task. The last stage comes to LLMs, which is scaling PLMs to increase model performance on downstream tasks [6]. The scaling of LLMs include three aspects: model parameters, dataset size and total compute [11].

LLMs are the language models have been trained on vast amounts of data, enabling it to comprehend and produce natural language and other kinds of content for a variety of uses. Their ability to recognize complex linguistic patterns and carry out a vast range of language-related tasks is made achievable by billions of parameters [16]. LLMs function by utilizing deep learning techniques and enormous volumes of textual data. The foundation of LLMs is typically a transformer architecture [16]. Transformer was developed by researchers at Google in 2017. Its capacity to handle sequential data effectively, enable parallelization, and capture long-range dependencies in text has transformed the field of NLP. Attention mechanisms is the core of the Transformer architecture that powers most LLMs. It enables the modeling of dependencies in input or output sequences without taking into account how far apart they are [17].

LLMs have shown two key characteristics. One is in-context learning, which means generate text based on a given context or prompt without requiring additional retraining or gradient update [6]. This makes LLMs possible to produce responses that are more logical and relevant for the given context, which makes them appropriate for interactive

and conversational applications. Another key feature is Reinforcement Learning from Human Feedback (RLHF). By using human-generated responses as rewards, this strategy entails fine-tuning the model over time so that it can learn from its mistakes and perform better [15].

The popular method of interacting with LLMs is prompt engineering, in which users create and give prompt texts to direct LLMs in producing desired responses or finishing particular tasks. In addition, users can participate in dialogue interactions, in which they converse with LLMs in natural language. Another example question-and-answer interactions, in which they ask the model questions and get responses [15].

However, some limitations of LLMs exist. One of the largest limitations is that they may have "hallucinations" (i.e. models generate content that is inaccurate or non-factual). Other limitations include non-transparent and untraceable reasoning processes and outdated knowledge etc. [13].

2.3 Retrieval-Augmented Generation (RAG) Systems

To address the challenges of LLMs, RAG systems were first introduced by Lewis et al. in 2020 [18]. It uses information from external databases to increase the model output's accuracy and credibility, especially for knowledge-intensive tasks. Compiling information that is relevant to the user's query. Together with the initial query, these articles create a thorough prompt that makes LLMs able to produce a knowledgeable response. It also allows integration of domain-specific knowledge and continuously updating of knowledge [13].

External data used by RAG are the new data that are not part of the LLM's initial training set. They may originate from a variety of data sources, including databases, document repositories, and APIs. The information can be found in a number of formats, including files, database entries, and long-form text. The data is then represented by

vectors that the model can understand and stored in a vector database. The external data and their embedding representation are synchronously updated in order to preserve up-to-date information for retrieval. Either recurring batch processing or automated real-time processes can be used for this.

The procedure of how RAG works include three steps:

1. Retrieve: The user query is transformed into a vector representation in order to match the vector databases. The relevancy was calculated using vector computations and mathematical representations. External data with high relevance to the query will be retrieved [19].

2. Augment: The retrieved information is integrated with the original user prompts. The RAG system prepares this combined input for the generative model [20]. By including pertinent retrieved data in context using prompt engineering techniques, the RAG model enhances user input (also known as prompts). LLMs are able to produce precise responses to user inquiries due to the augmented prompt [19].

3. Generation: The augmented input is processed by a generative model, which constructs responses based on the original query and retrieval data. The model synthesises the information and generates coherent and contextually relevant answers, effectively bridging knowledge gaps that may exist in the model training data [21].

Trantorinc (2024) [22] explained RAG in a figure (Figure 2.1), which provides a more intuitive illustration of the RAG process.

RAG have several main advantages. First, RAG saves the cost. Compared to the high computational and financial expense when retraining LLM, RAG provides a more cost-effective way to add new data to the LLM. Second, LLM can give users the most recent information. RAG enables developers to feed the generative models with the most recent data, statistics, or research. Using RAG, they can establish a direct connection between

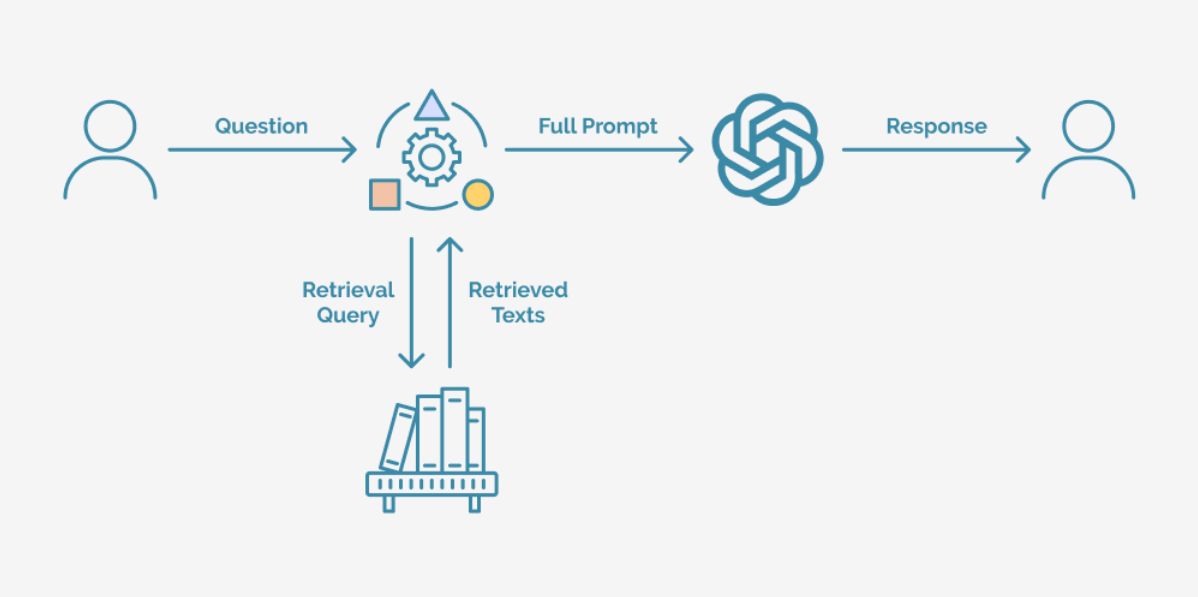

Figure 2.1: RAG Procedure Explanation

the LLM and real-time social media feeds, news websites, or other regularly updated information sources. Third, RAG increases user trust. Because The LLM with RAG technique is able to provide precise information with source attribution. References or citations to sources can be included in the output [19].

2.4 Previous Work

The development of LLMs has completely changed how we handle a variety of activities, such as ESG data extraction. LLMs can makes the process of ESG data extraction become more accurate and efficient [23], especially combined with RAG systems.

Recent academic literature has used LLMs and RAG system to extract and analyse ESG data. For example, Ontiveros (2024) [24] developed and evaluated an IT tool based on LLM and RAG for companies' ESG compliance consulting. Gupta et al. (2024) [25] leveraged LLM with Knowledge Graph-based Retrieval Augmented Generation (KG-RAG) to help question-answering about company ESG performance and sustainability

news articles. Ardic et al. (2024) [26] extracted ESG information from Sustainability Reports in Turkish using LLM and RAG. Zou et al. (2023) [5] designed ESGReveal, which is an LLM-based method to extract structured data from ESG reports. Bronzini et al. (2024) [27] used LLMs to derive structured ESG insights from sustainable reports.

The recent work indicates a growing interest in leveraging LLMs and RAG techniques for ESG data extraction, with promising results in terms of accuracy and efficiency. These developments could completely change the processing and analysis of ESG data, providing a more sophisticated knowledge of company sustainability strategies.

However, despite the promising applications, there remains a significant gap in the research landscape. To the best of our knowledge, studies that concentrate on a comprehensive assessment of LLMs particularly in relation to ESG data extraction in RAG systems are lacking. This gap is noteworthy because the unique characteristics of ESG data. For example, its complexity nature (including diverse sources, inconsistencies in how companies disclose their ESG performance, etc.) [28] and the need for high precision in analysis [29]. These may present distinct challenges and opportunities for LLMs and RAG techniques that are not fully understood. To bridge the gap, this study aims to develop a comprehensive benchmark framework to evaluate the efficiency of LLMs on ESG data extraction in RAG system.

2.5 Summary

To summarize, this chapter provides the relevant background knowledge for the subject of this study and reviews the recent work to find the gaps and opportunities in this field. First, we provided the background knowledge of ESG, LLM and RAG, showed LLM and RAG work, why they are important, and their applications in ESG data extraction. Previous work shows potential of LLMs with RAG techniques for extracting ESG data, but studies that focus on the evaluation of LLMs particularly in relation to

ESG data extraction in RAG systems are lacking. This gap is significant due to ESG data's complexity nature and the necessity for high precision in analysis, and LLMs and RAG techniques may not fully understood, and thus it limits the user trust to using them for ESG data extraction task without evaluation. This study aims to fix this gap by providing a comprehensive evaluation for this task.

3 | Data and Methods

This chapter explains the data and methods used to evaluate LLMs on ESG data extraction in RAG system. Section 3.1 illustrates the process of constructing the evaluation dataset. It explains how question part and answer part for the gold answer dataset were constructed and the reason and process for further splitting the dataset. Section 3.2 describes selected LLMs for comparison and the specific methods used for evaluation.

3.1 Data

This study constructed a question and answer (Q&A) dataset for evaluation. The following subsections introduce the question and the answer part, respectively. Due to the critical role Hong Kong Exchanges and Clearing Limited (HKEX)[1] has played in the development of ESG operations in China [5], this study used the HKEX ESG Reporting Guide to construct the question data, and the ESG reports of companies listed on the HKEX to construct the answer data.

3.1.1 Question Part for Gold Answer Dataset

To construct question data, I used the HKEX ESG Reporting Guide[2]. It acts as a market regulator that gives companies listed on our stock exchanges a transparent framework for the disclosure, application, and implementation of ESG through listing regulations, norms, and education [30]. The guide sets out the "comply or explain" provisions. It requires issuers to report the "comply or explain" provisions in ESG disclosures. If the issuers choose not to report on any one or more of these provisions, they must state the reasons in their ESG report [31].

[1] https://www.hkex.com.hk/?sc_lang=en

[2] https://www.hkex.com.hk/Listing/Sustainability/ESG-Academy/Rules-and-Regulations?sc_lang=en

Here is the structure of the reporting guide. There are two ESG subject areas in this guide: Environmental (Subject Area A) and Social (Subject Area B). Corporate governance is not in this report and is separately in the Corporate Governance Code. Each subject area contains different aspects. Each aspect outlines the general disclosures and key performance indicators (KPIs) that issuers are required to report on to show how well they have done. The following table shows the structure of the reporting guide:

Subject Area	**Aspect**	**Number of Indicators**
A. Environmental	A1. Emissions	6
A. Environmental	A2. Use of Resources	5
A. Environmental	A3. The Environment and Natural Resources	1
A. Environmental	A4. Climate Change	1
B. Social	B1. Employment	2
B. Social	B2. Health and Safety	3
B. Social	B3. Development and Training	2
B. Social	B4. Labour Standards	2
B. Social	B5. Supply Chain Management	4
B. Social	B6. Product Responsibility	5
B. Social	B7. Anti-corruption	3
B. Social	B8. Community Investment	2

Table 3.1: Structure of HKEX ESG Reporting Guide

There are 36 KPIs in total. Each KPI can be converted to one or more questions. The questions can be categorised as either quantitative or qualitative. The questions that start with "what is" are categorised as quantitative questions, and the questions that start with "is there any description of" are categorised as quanlitative questions. In order to obtain quantifiable evaluation results and make the subsequent model performance easily comparable, the answers for all quantitative questions are designed as a number with unit (if any). If the answer is not given in the ESG report, the answer will be labelled as "No Answer". The answers for all quantitative questions are designed to be either "yes" or "no".

For example, KPI B6.2 is "number of products and service related complaints received and how they are dealt with". This KPI can be converted to two questions:

- Question 1 (quantitative): What is the number of products and service related complaints received?
- Question 2 (qualitative): Is there a description of how complaints related to products and services are handled?

If the answers for those two questions are given in the ESG report, the answer to Question 1 is supposed to be a specific number, and the answer to Question 2 is supposed to be either "yes" or "no".

The advantage of this kind of question design is that the evaluation of the answer will become a simple binary problem (i.e. either "true" or "false"). This makes the result quantifiable and enables easy comparison of the model result. Finally, the dataset consists of 66 questions for each ESG report, of which 34 are qualitative questions and 32 are quantitative questions.

3.1.2 Answer Part for Gold Answer Dataset

The answers to the questions can be found in the ESG reports of companies listed on HKEX. We found 7280 ESG reports for diffenert companies listed on HKEX which issued from fiscal year 2013 to 2023. Due to limited time and computational cost, a representative sample of 15 companies' ESG reports were selected. The reports are selected based on the following rules: First, reports are issued in fiscal year 2023 to ensure timeliness. Second, the reports content are (almost) all written in English. Many reports are half in Chinese and half in English, but some LLMs only support English and this may cause the inconsistencies in the evaluation process. Third, the KPIs and their corresponding sections or pages in the report are clearly listed, which makes us easier to find the relevant information. The companies of selected ESG reports and their stock code are listed in

Table 3.2.

Company	**Stock Code**
Oriental Enterprise Holdings Limited	18
Mexan Limited	22
Dickson Concepts (International) Limited	113
Moiselle International Holdings Limited	130
National Electronics Holdings Limited	213
Alco Holdings Limited	328
Edvantage Group Holdings Limited	382
Sincere Watch (Hong Kong) Limited	444
Lisi Group (Holdings) Limited	526
Public Financial Holdings Limited	626
Pico Far East Holdings Limited	752
Summi (Group) Holdings Limited	756
Asia Cassava Resources Holdings Limited	841
China Water Affairs Group Limited	855
Vision Values Holdings Limited	862

Table 3.2: Selected ESG Reports List

The report file is in PDF format, which is difficult for LLMs to read directly. To pre-process the reports, I searched and compared different PDF parsing tools, including Unstructured[3], Marker[4] and PyMuPDF4LLM[5] and LlamaParse[6]. Table 3.3 indicates the application, features and price of each tool.

[3] https://github.com/Unstructured-IO/unstructured
[4] https://github.com/VikParuchuri/marker
[5] https://pymupdf4llm.readthedocs.io/en/latest/
[6] https://cloud.llamaindex.ai/parse

Tool name	**Applications**	**Features**	**Price**
Unstructured	1. Pretraining Models 2. Fine-tuning Models 3. Retrieval Augmented Generation (RAG) 4. Traditional ETL	1. Supports for image extraction. 2. Robust Core Functionality for efficient data processing, includes partitioning, cleaning, extracting, staging, chucking, embedding.	1. The free API usage is limited to 1000 pages per month. 2. Fast Strategy: $1 per 1000 pages processed. Hi-Res Strategy: $10 per 1000 pages processed.
Marker	1. Converts PDF to markdown 2. Optical Character Recognition (OCR) 3. Line Detection 4. Layout Analysis 5. Reading Order 6. LaTeX OCR	1. Supports for image extraction. 2. Formats tables and code blocks. 3. Removes headers/footers/other artifacts. 4. Supports all languages.	1. Free for noncommercial research and personal purposes. 2. If the organization has less than $5M USD in recent 12-month revenue and lifetime VC/angel funding, the tools are free. For revenue or funding exceeding $5M USD, a commercial license is required.
PyMuPDF4LLM	Extract PDF content for LLM & RAG environments	1. Support for image and vector graphics extraction. 2. Support for multi-column pages. 3. Support for page chunking output.	1. Open-source AGPL version is free. 2. For commercial use, contact Artifex Software Inc. for information about obtaining a commercial license.
LlamaParse	Parse and clean data for downstream LLM use cases such as advanced RAG	1. State-of-the-art table extraction. 2. Support for image extraction. 3. Provide natural language instructions to parse the output in the exact format you want.	1. 1k free pages a day. 2. Paid plan: 7k free pages a week, then $0.003 for each page.

Table 3.3: PDF parsing Tool Comparison

After experimenting all the tools, I finally chose LlamaParse. The most important reason

is that LlamaParse performs the best in extracting tables among all these tools, whereas all others often face problems such as misplaced data, misformatting, and misrecognised characters for tabular data, which can significantly affect the accuracy and usability.

The relevant content from the ESG reports is served as "original information" in the gold answer dataset. Since the aim of this study is to benchmark the efficiency of LLMs on ESG data extraction for RAG ready system, we assume LLMs have already retrieved these "original information", then compare the model performance in extracting data from these "original information".

To construct gold answers based on the questions and original information from reports, I used ChatGPT-4, one of the powerful and well-known LLMs at the time of writing.

The following is the prompt that I used:

```
I will provide you with a passage of text and a question. Based on the type of question, please follow the steps below to provide your answer. Determine the type of question according to the beginning of the question, and answer the following questions according to the different types:
1. For quantitative questions (" what is "), you should provide a corresponding number based on the information in the text. If the text uses expressions such as "not less" or "more than" to represent quantities, the original representation is used. If the text does not mention any relevant data, respond with "No Answer".
2. For qualitative questions ("Is there any description"), when answering "Is there any description" questions, ensure you carefully verify the information. If the text clearly lacks the required information, answer "no". Otherwise, answer "yes".
```

After ChatGPT-4 generated the answer, I manually checked each answer to ensure the correctness. Now, the dataset contains 15 ESG reports, and as mentioned before, there are 66 questions for each ESG report, of which 34 are qualitative questions and 32 are

quantitative questions. Therefore, there are 510 (15×34) qualitative Q&A pairs and 480 (15×32) quantitative Q&A pairs in total.

3.1.3 Further Split Gold Answer Dataset

To better evaluate the ability of the models to extract single data, I split the Q&A dataset in a more refined way. The reason for doing this is that the original quantitative Q&A pairs are too complicated to analyze and evaluate. Take one Q&A pairs as an example, the original question is:

```
What is the number of suppliers by geographical region?
```

The original answer is:

```
2023:
Hong Kong: 140
Taiwan: 0
PRC: 5
Japan: 16
Korea: 19
North America: 96
Europe: 186
Others: 23

2022:
Hong Kong: 178
Taiwan: 2
PRC: 2
Japan: 18
Korea: 22
North America: 156
Europe: 239
Others: 26
```

This complicated answer not only made subsequent model answer evaluation very difficult, but also prevented us from evaluating the model to extract a single piece of data.

To solve this problem, the comprehensive questions are broken down into multiple, more specific questions. The libraries I used include openai [32] (which provides convenient access to the OpenAI API), pandas [33] (a data analysis and manipulation tool), and json [34] (to store and transmit structured data).

Firstly, the primary variables such as "type" and "year" in each problem were explicitly labelled. For example, the original question mentioned before:

```
What is the number of suppliers by geographical region?
```

The question is converted to:

```
What is the number of suppliers for {type} in {year}?
```

Then, gpt-3.5-turbo is used to efficiently process each converted question and the corresponding answer. The following is the prompt for splitting Q&A pairs:

```
You will receive a pair of question and answer, please follow the steps
    below to give me the answers broken down by years:
1. Split answers by year
2. Record answers in different years and under different types in a
   JSON format.
3. Ensure the JSON contains the keys "year", "type", and detailed
   parameters in answer if it have.
  If no detailed classification is provided, set the "details" value to
    "null".
4. The JSON format should look like this:
[
  {
    "year": "2023",
    "data": [
      {
```

```
        "type": "type_here",
        "value": "value_here"
        "details": {
          "details_infomation_here": "details_value_here",
          "details_infomation_here": "details_value_here"
        }
      },
      {
        "type": "type_here",
        "value": "value_here"
        "details": {
          "details_infomation_here": "details_value_here",
          "details_infomation_here": "details_value_here"
        }
      },
      ...
]
5. Only output the JSON with the specified keys and in the specified
   order.
```

Listing 3.1: Prompt for Splitting

Using this prompt, LLM helps break down the original question into more specific questions by years and types, and output the JSON format to store the data. The converted question

```
What is the number of suppliers for {type} in {year}?
```

is now breaking down into multiple questions:

```
What is the number of suppliers for Hong Kong in 2023?
What is the number of suppliers for Taiwan in 2023?
What is the number of suppliers for PRC in 2023?
What is the number of suppliers for Japan in 2023?
What is the number of suppliers for Korea in 2023?
```

```
What is the number of suppliers for North America in 2023?
What is the number of suppliers for Europe in 2023?
What is the number of suppliers for Others in 2023?
What is the number of suppliers for Hong Kong in 2022?
What is the number of suppliers for Taiwan in 2022?
What is the number of suppliers for PRC in 2022?
What is the number of suppliers for Japan in 2022?
What is the number of suppliers for Korea in 2022?
What is the number of suppliers for North America in 2022?
What is the number of suppliers for Europe in 2022?
What is the number of suppliers for Others in 2022?
```

The corresponding answers are also split by gpt-3.5-turbo. By applying this splitting process to all quantitative Q&A, the original 510 pairs of qualitative questions were split into 1529 pairs. Therefore, the final dataset contains 2039 Q&A pairs, of which 510 qualitative Q&A pairs and 1529 quantitative pairs. The dataset contains the columns "question", "type" (of the question, either "qualitative" or "quantitative"), "original information" (from ESG report) and "gold answer".

3.2 Evaluation Methods

After the gold answer dataset was constructed, other different models also generated answers based on the question and original information. Then, the model-generated answers are compared with the gold answer to evaluate the accuracy of the answers. In addtion, The running time for the model to generate all the answers is also compared.

3.2.1 LLMs Used

This study compares a total of 14 LLMs, all of which are now mainstream LLMs at the time of writing. Tables tables 3.4 to 3.7 list all LLMs used. The LLMs were divided into

three scales based on the number of parameters. Model sizes between 40 and 10B were classified as large scale models, model sizes below 5B were classified as medium scale models, and model sizes ranging from 5 to 0B were classified as small scale models. For each scale, four different models were selected for comparison. In addition, comparisons were also made using the GPT series of models, including gpt-4 and gpt-3.5-turbo. The tables contain the information about the models' size (i.e. number of parameters), core capabilities, potential usages, and cost (including deployment requirement, input and output token price).

Model	**Size**	**Core Capabilities**	**Potential Usage**	**Cost**
gpt-4 [35]	1760B	Broader general knowledge and advanced reasoning capabilities	High-intelligence models that reach human performance on a range of academic and professional benchmarks	Clusters of 128 A100 GPUs for efficient deployment Input token price: $30.00, Output token price: $60.00 per 1M Tokens
gpt-3.5-turbo [36]	175B	Can comprehend and generate natural language or code; well-suited for chat, but they are also effective for non-chat tasks	A fast, inexpensive model for simple tasks	Run in full precision would necessitate around 700 GB of VRAM Input Token Price of $1.50 per 1M Tokens and an Output Token Price of $3.00 per 1M Tokens

Table 3.4: GPT Series Model Comparison

Model	**Size**	**Core Capabilities**	**Potential Usage**	**Cost**
yi-34b-bnb-4bit [37]	34B	1. language understanding 2. commonsense reasoning 3. reading comprehension	Suitable for personal, academic, and commercial (particularly for small and medium-sized enterprises) purposes.	Minimum VRAM: 72 GB, Recommended GPU Example: 4 × RTX 4090 (24 GB), 1 × A800 (80 GB). Input token price: $3.00, Output token price: $3.00 per 1M Tokens
gemma-2-27b [38]	27B	1. Dialogue 2. Reasoning 3. Mathematics 4. Code generation	Ideal for a range of text generation jobs, such as summarizing, reasoning, and question answering. Small size makes it possible to deploy them in environments with limited resources.	Can be deployed on a single NVIDIA H100 Tensor Core GPU or TPU host Input token price: $0.80, Output token price: $0.80 per 1M Tokens
llama-2-13b [39]	13B	1. Reasoning 2. code generation 3. instruction following	Intended for usage in English for research and commercial purposes. Can be customized for a range of natural language generating tasks.	At least 26GB VRAM, with options like the RTX 6000 ADA 48GB being recommended Input Tokens: $0.30 per 1M tokens Output Tokens: $0.30 per 1M tokens
mistral-nemo-base-2407 [40]	12B	1. reasoning 2. World knowledge 3. Coding accuracy	Automate large scale text generation and processing, an internal assistant with RAG and function calling, coding assistant, tailor your applications to your customer base.	Perform inference with the NVIDIA A100 or NVIDIA L40 Input token price: $0.15, Output token price: $0.15 per 1M Tokens

Table 3.5: Large Language Model Comparison

Model	Size	Core Capabilities	Potential Usage	Cost
gemma-2-9b [41]	9B	1. Dialogue 2. Reasoning 3. Mathematics 4. Code generation	Ideal for a range of text generation jobs, such as summarizing, reasoning, and question answering. Small size makes it possible to deploy them in environments with limited resources.	can be deployed on a single NVIDIA H100 Tensor Core GPU or TPU host Input token price: $0.20, Output token price:$0.20 per 1M Tokens
llama-3-8b [42]	8B	1. Reasoning 2. Code generation 3. Instruction following	Intended for usage in English for research and commercial purposes. Can be customized for a range of natural language generating tasks.	Around 16GB of disk space and 20GB of VRAM (GPU memory) in FP16 Input token price: $0.14, Output token price:$0.20 per 1M Tokens
Qwen2-7b [43]	7.6B	1. Language understanding 2. Language generation 3. Multilingual capability 4. Coding 5. Mathematics 6. Reasoning	A wide range of language and logic-based tasks.	About14.92 GB of VRAM is needed for inference. Input Tokens:$0.10 per million tokens Output Tokens: $0.50 per million tokens
mistral-7b [44]	7.2B	1. Reasoning 2. Mathematics 3. Code generation	Automate large scale text generation & processing, an internal assistant with RAG and function calling, coding assistant, tailor your applications to your customer base	About 14.92 GB of VRAM for inference when using float16/bfloat16 precision. Input token price: $0.15, Output token price:$0.20 per 1M Tokens

Table 3.6: Medium Language Model Comparison

Model	**Size**	**Core Capabilities**	**Potential Usage**	**Cost**
gemma-2-2b [45]	2.6B	1. Dialogue 2. Reasoning 3. Mathematics 4. Code generation	Ideal for a range of text generation jobs, such as summarizing, reasoning, and question answering. Small size makes it possible to deploy them in environments with limited resources.	Efficient deployment on CPU and on-device applications
gemma-2b [46]	2.5B	1. Dialogue 2. Reasoning 3. Mathematics 4. Code generation	Ideal for a range of text generation jobs, such as summarizing, reasoning, and question answering. Small size makes it possible to deploy them in environments with limited resources.	Efficient deployment on CPU and on-device applications
Qwen-2-1.5b [47]	1.5B	1. Language understanding 2. Coding 3. Mathematics 4. Chinese language understanding	A wide range of language and logic-based tasks.	Efficient deployment on CPU and on-device applications
tinyllama [48]	1.1B	1. Problem-solving 2. Commonsense reasoning	Supports a wide range of applications that have limited computation and memory requirements.	Efficient deployment on CPU and on-device applications

Table 3.7: Small Language Model Comparison

3.2.2 Method of Benchmarking Model Answers

This study used above models generated the answers based on question and original information from ESG report, following the same steps and prompt as constructing the

gold answer dataset. To accurately and efficiently compare model-generated answers with the gold answer, this study implemented an automated assessment. This subsection describes the method of automated assessment in detail. The library used include pandas [33] (a data analysis and manipulation tool) and re [49] (i.e. regular expression matching operations).

As mentioned before, in the gold answer dataset, answers for qualitative questions are either "yes" or "no", and no additional explanations have been added. For quantitative questions, if the original information has mentioned the relevant data, answers would be specific numbers with units (if any). If the original information does not mention any relevant data, the answers would be "no answer". Also, no additional explanations have been provided. Based on this answer format, the study used different assessment methods for qualitative and quantitative questions.

For qualitative questions, the specific keywords in the model-generated answers are checked. If the keywords "yes" or "no" (case insensitive) are found in the model-generated answer, check whether it is the same as the gold answer. If they are the same, returns "true", otherwise it returns "false".

For quantitative questions, there are several steps before extracting the numbers in the answers. First, the answer is converted to a string and all commas in the string are removed. The reason for doing this is to make sure that all numbers are formatted uniformly, without commas as thousands separators (e.g., 2,100 to 2100). This helps with subsequent value extraction and comparison. Second, find all numbers and percentages using the regular expression. If it contains a percentage sign (%), convert it to a decimal (e.g., 50% to 0.5).

If the gold answer is "no answer", then it checks if the model answer is also "no answer" (or equivalent). If so, the function returns "true", otherwise it returns "false". If the gold answer contains relevant data, extract the numbers in the model-generated answers and

check if the value in the model answer is equal to the value in the gold answer. Given the precision of floating point numbers, a very small threshold of 1e-6 is used to determine if the two numbers are "close enough" to be matched.

The model answer may contains more than one numbers. For example, for question:

```
What is the intensity (e.g. per unit of production volume, per facility ) of hazardous waste produced (in tonnes) in 2023?
```

the gold answer is:

```
0.0016 tonnes per square meter
```

the answer generated by gpt-3.5-turbo is:

```
The intensity of hazardous waste produced in 2023 is 0.0016 tonne/m2.
```

The numbers extracted from model-generated answer is a list:

```
[2023, 0.0016, 2]
```

which additionally contains the year number (2023) and the number from the unit (2). Therefore, if at least one of the values in the model answer matches with the gold answer, the function would return "ture", otherwise it would return "false".

3.3 Summary

This chapter describes how the evaluation Q&A dataset was constructed and the method used to evaluate LLMs on extracting ESG data in RAG system. To construct a gold answer dataset, the question data were constructed based on the HKEX ESG Reporting Guide. Each KPI in the reporting guide was converted to one or more questions categorised as either quantitative or qualitative. Quantitative question answers are specified as numbers with units for comparability (with "no answer" used if the information is missing from the ESG report), while qualitative answers are binary ("yes" or "no").

This question design simplifies evaluation to a binary "true" or "false" assessment, making results quantifiable and easily comparable. The answer data were constructed based on the 15 ESG reports of companies listed on HKEX. Gold answer is constructed using ChatGPT-4 with manual check to ensure the answer correctness. To facilitate the assessment of model performance to extract single data, quantitative qustions were further split using gpt-3.5-turbo, the resulting gold answer dataset consisted of 2039 question-answer pairs, of which 510 were qualitative and 1529 were quantitative.

A total of 14 LLMs were compared in this study, they were categorised into small, medium and large models, with four models at each scale. 2 GPT models were also compared as a reference. Model answers were generated based on the question and original information, then compare with the gold answer. An automated assessment was carried out to compare model answers with the gold answer based on the string matching techniques.

4 | Results

This chapter presents the comprehensive results of the study, which include both qualitative and quantitative findings. The model performance is evaluated by two metrics: accuracy and running time (speed). In Section 4.1, the overall results are outlined, providing a clear summary of key insights. Following this, the qualitative results and quantitative results are discussed respectively in Section 4.2 and Section 4.3. In Section 4.4, two of the models were chosen for error analysis to observe why the model answered incorrectly, which contribute to a deeper understanding of the research outcomes.

4.1 Overall Results

Table 4.1 indicates the overall results for all models, including the name of the model, the size of the model in billion, the size scale of the model, the number of "true" answers (i.e. correct answer), the number of "false" answers (i.e. incorrect answer), accuracy (the number of "true" answers / the number of all answers), total running time in seconds.

From the result, it can be seen that there is a great difference in model performance across different models. GPT-3.5-turbo achieves a maximum accuracy of 89.5% among all tested models. Followed by Qwen-2-7b and gpt-4, with accuracy of 86.9% and 81.2%, respectively. Gemma 2 models with different sizes also have relatively high accuracy (75.4% for gemma-2-27b, 67.7% for gemma-2-9b and 63.9% for gemma-2-2b), and their performance in accuracy is similar. Tinyllama only achieves the accuracy of 29.9%, which is the lowest accuracy among all tested models. GPT-4 has the shortest running time (1316 seconds), and gpt-3.5-turbo has the second shortest run time (2095 seconds). Except for the GPT series models, gemma-2b model and Qwen 2 models with 7B and 1.5B parameters have relatively short running time (2513, 2574 and 2844 seconds, respectively). gemma-2-27b and yi-34b-bnb-4bit have the longest running time (11213 seconds and 11532 seconds).

Model	Size (B)	Scale	True	False	Accuracy	Running Time (s)
gpt-3.5-turbo	175.0	GPT Series	1824	215	89.5%	2095
Qwen-2-7b	7.0	medium	1772	267	86.9%	2574
gpt-4	1760.0	GPT Series	1655	384	81.2%	1316
gemma-2-27b	27.0	large	1537	502	75.4%	11213
gemma-2-9b	9.0	medium	1381	658	67.7%	5512
gemma-2-2b	2.6	small	1303	736	63.9%	4851
mistral-nemo-base-2407	12.0	large	1270	769	62.3%	7616
gemma-2b	2.5	small	1199	840	58.8%	2513
llama-3-8b	8.0	medium	982	1057	48.2%	3030
Qwen-2-1.5b	1.5	small	964	1075	47.3%	2844
mistral-7b	7.0	medium	905	1134	44.4%	5784
yi-34b-bnb-4bit	34.0	large	882	1157	43.3%	11532
llama-2-13b	13.0	large	832	1207	40.8%	7211
tinyllama	1.1	small	610	1429	29.9%	4139

Table 4.1: Overall Result

To investigate how model size may affect the answer accuracy and running time, Figure 4.1 shows the relationship between model size and accuracy, Figure 4.2 indicates the relationship between model size and running time. There is no significant difference in accuracy performance between models of different sizes, and no particular scale is better than others. The possible reason is that the difference in evaluated model size is not very large (the smallest model is 1.1B, and the largest model is 34B), resulting in no significant difference in model performance.

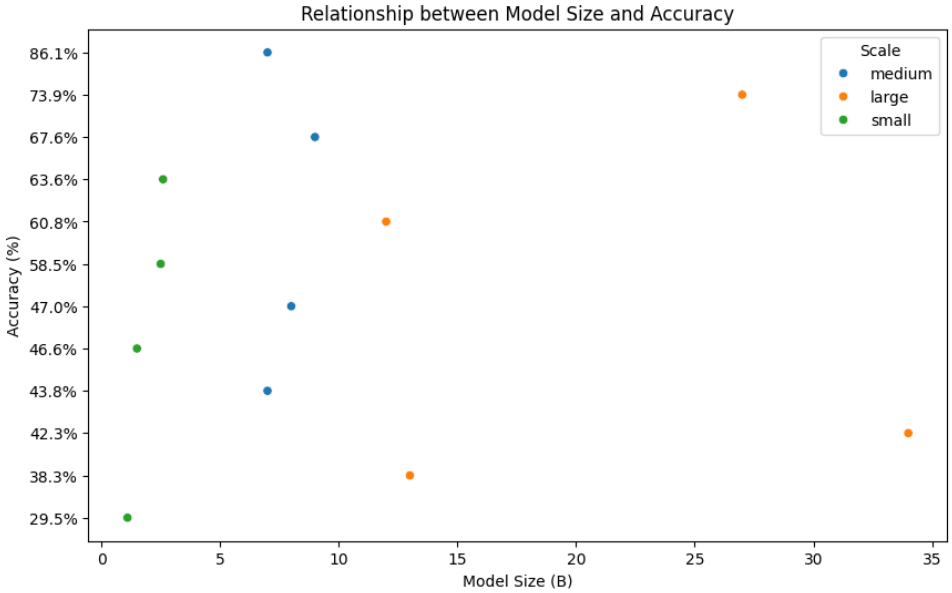

Figure 4.1: Relationship between model size and accuracy. Small scale models are represented by green points, medium scale models are represented by blue points, and large scale models are represented by orange points.

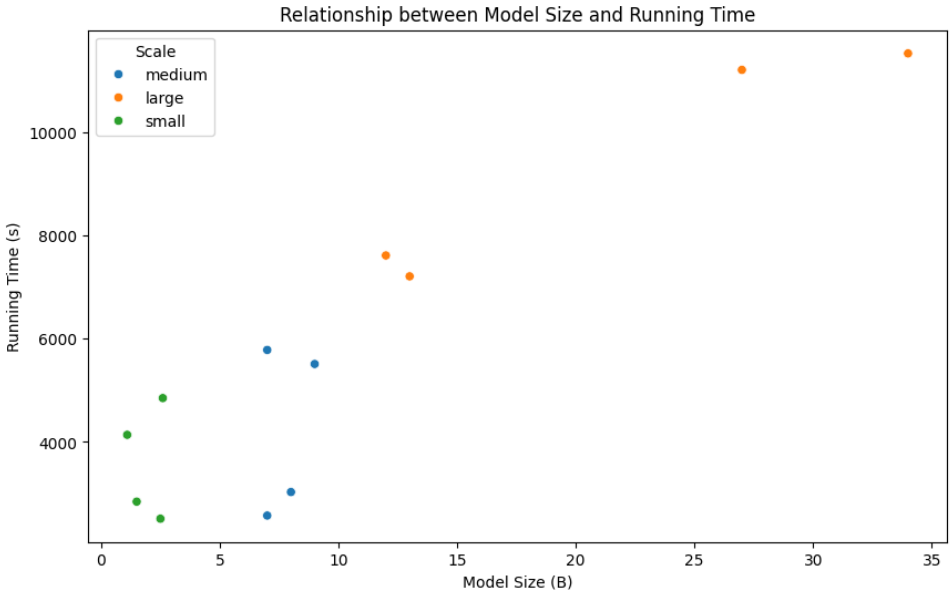

Figure 4.2: Relationship between model size and running time. Small scale models are represented by green points, medium scale models are represented by blue points, and large scale models are represented by orange points.

For running time, as the model size increases, the running time also tends to increase. This suggests that larger models may require more computational resources and take longer to run compared to smaller models. However, there is some variability in the data, with some points deviating from the general trend.

Considering the accuracy and running time trade-off, GPT series models have relatively high accuracy and short running time among all tested models. Qwen-2-7b has moderately longer running time than GPT models, but it still achieves relatively high accuracy while maintaining relatively short running times comparing to other models. It is the best performing model except for GPT models. Although gemma-2-27b has relatively higher accuracy, its running time is too long. Also, yi-34b-bnb-4bit has the longest running time, but its accuracy is relatively low.

4.2 Qualitative Results

Next, we discuss the qualitative and quantitative results separately. Table 4.2 shows the model performance for answering qualitative questions. GPT-3.5-turbo achieves the highest accuracy of 88.8%. Followed by Qwen 2 models 1.5B and 7B, with the accuracy of 88.6% and 88.0%. mistral-7b also performs relatively well, with accuracy of 80.2%, higher than the accuracy of gpt-4 (75.3%). Five models have an accuracy of less than 50%, the lowest of which is llama-2-13b (32.2%).

Model	False	True	Accuracy
gpt-3.5-turbo	57	453	88.8%
Qwen-2-1.5b	58	452	88.6%
Qwen-2-7b	61	449	88.0%
mistral-7b	101	409	80.2%
gpt-4	126	384	75.3%
gemma-2-9b	138	372	72.9%
gemma-2-27b	148	362	71.0%
yi-34b-bnb-4bit	159	351	68.8%
gemma-2b	241	269	52.7%
mistral-nemo-base-2407	298	212	41.6%
llama-3-8b	328	182	35.7%
tinyllama	335	175	34.3%
gemma-2-2b	342	168	32.9%
llama-2-13b	346	164	32.2%

Table 4.2: Qualitative Results

4.3 Quantitative Results

Table 4.3 shows the model performance for answering quantitative questions. GPT-3.5-turbo still achieves the highest accuracy (89.5%). Gemma 2 models with different sizes and mistral-nemo-base-2407 performs relatively well (ranging from 75.4% to 66.1%). tinyllama has the lowest accuracy, which is 28.1%.

Model	False	True	Accuracy
gpt-3.5-turbo	158	1371	89.7%
Qwen-2-7b	206	1323	86.5%
gpt-4	258	1271	83.1%
gemma-2-27b	354	1175	76.8%
gemma-2-2b	394	1135	74.2%
mistral-nemo-base-2407	471	1058	69.2%
gemma-2-9b	520	1009	66.0%
gemma-2b	599	930	60.8%
llama-3-8b	729	800	52.3%
llama-2-13b	861	668	43.7%
yi-34b-bnb-4bit	998	531	34.7%
Qwen-2-1.5b	1017	512	33.5%
mistral-7b	1033	496	32.4%
tinylma	1094	435	28.4%

Table 4.3: Quantitative Results

4.4 Error Analysis

4.4.1 Error Analysis for Qwen-2-7b

To understand where the models are making mistakes and find the potential ways to improve model performance, I looked into the incorrect answers for Qwen-2-7b (achieves the highest overall accuracy except for GPT models) and tinyllama (achieves the lowest overall accuracy).

Table 4.4 is the contingency table for Qwen-2-7b, showing the distribution of qualitative and quantitative questions in terms of their answer correctness ("true" or "false"). There are 267 incorrect answers in total, of which 61 are qualitative and 206 are quantitative.

	Qualitative	Quantitative	Total
False	61	206	267
True	449	1323	1772
Total	510	1529	2039

Table 4.4: Contingency table for Qwen-2-7b

4.4.1.1 Qualitative Error Analysis for Qwen-2-7b

For qualitative questions, the types of errors have been grouped into four categories in Table 4.5.

Gold Answer	Model Answer	Cases
Yes	No	1
Yes	No Answer	12
No	Yes	28
No	No Answer	20

Table 4.5: Incorrect qualitative results analysis for Qwen-2-7b

Of these 61 qualitative incorrect questions, 13 questions had a gold answer of "yes", but the model generated 1 "no" and 12 "no answer". 48 questions had a gold answer of "no", but the model generated 28 "yes" and 20 "no answer".

It is interesting to note that the cases of questions where the model misclassifies the answer as "yes" (28) are much larger than the cases of questions where the model misclassifies the answer as "no" (1). The possible reason is that the questions with gold answer of "no" is often harder to judge. For example, for question:

```
Is there any description of how hazardous and non-hazardous wastes are
   handled?
```

given the original information:

```
Currently hazardous and non hazardous wastes are not considered
   material to the Group and we have yet to collect and consolidate
```

```
relevant information. We will continue to review and disclose
further information as appropriate in the future.
```

The gold answer is "no", because although the original information states that the Group has not yet considered these wastes as material and has not collected or consolidated relevant information, there is no description of "how" hazardous and non-hazardous wastes are handled.

However, the answer given by Qwen-2-7b is:

```
Yes, the text states that hazardous and non-hazardous wastes are not
considered material to the Group and we have yet to collect and
consolidate relevant information. However, we will continue to
review and disclose further information as appropriate in the future
.
```

These questions require the model more carefully verify the information and better reasoning skills to answer correctly. Fine-tuning may improve the model ability to reduce this type of error.

Another worth-noting point is that Qwen-2-7b generates 32 "no answer", but the answer is supposed to be either "yes" or "no" for all qualitative questions, as written in the prompt for generating model answer (see Listing 3.1.2). It indicates that the model does not strictly follow the prompt. Also, fine-tuning may improve the model ability and make model effectively follow the prompt.

4.4.1.2 Quantitative Error Analysis for Qwen-2-7b

For quantitative questions, the types of errors have been grouped into five categories in Table 4.6. For a clearer explanation, an example is provided for each error type. Situations where these errors may often occur will also be discussed.

Error Type	Explanation	Cases
Incorrect data	Misallocated data or incorrect calculation.	149
No answer	The original information contains the relevant data, but the model answer is "no answer".	32
Unit error	Fail to convert units as required by the question.	14
Incomplete answer	Model only reports part of its reasoning process, but did not complete answering the question.	7
Hallucination	Fabricated data generated from model.	4

Table 4.6: Incorrect quantitative results analysis for Qwen-2-7b

Type 1: Incorrect Data Error

Example question:

```
What is the total workforce for Male in 2023?
```

Table 4.7 is the data table from the original information.

KPIs	Unit	FY2022	FY2023
Social			
Workforce			
Total Workforce	Person(s)	58	51
Workforce by Employment Type			
Full-time	Person(s)	58	50
Part-time	Person(s)	0	1
Workforce by Gender			
Male	Person(s)	27	25
Female	Person(s)	31	26
Workforce by Employment Level			
Senior	Person(s)	15	15
Intermediate	Person(s)	7	5
General	Person(s)	36	31
Workforce by Age Group			
<30	Person(s)	2	1
30-50	Person(s)	40	39
>50	Person(s)	16	11
Workforce by Geographical Region			
Hong Kong	Person(s)	58	51
Macau	Person(s)	0	0

Table 4.7: Example for Incorrect Data Error: The Original Data Table

The gold answer is "25", but the answer generated by Qwen-2-7b is:

```
The total workforce for Male in 2023 is 26.
```

The number "26" is indeed in the original data table, but this is the total workforce for female, rather than male. Therefore, the model misallocated the data.

This type of error often occurs when the original data tables contain complex data structures (such as nested tables, multiple levels of linked data, or complex formulas) or large amount of data, it may be difficult for the model to understand and process the information. Further fine-tuning the model may reduce this type of error.

Type 2: No Answer Error

Example question:

```
What is the number of suppliers for Taiwan in 2022?
```

Table 4.8 is the data table from the original information.

Suppliers	**2023**	**2022**
Total Number of Suppliers	485	643
By Geographical Region		
Hong Kong	140	178
Taiwan	0	2
PRC	5	2
Japan	16	18
Korea	19	22
North America	96	156
Europe	186	239
Others	23	26

Table 4.8: Example for No Answer Error: The Original Data Table

The gold answer is "2", but the answer generated by the model is:

```
No Answer
```

Same as the incorrect data error, no answer error often occurs when the data is complex or large in size. Further fine-tuning the model may reduce this type of error.

Type 3: Unit Error

Example question:

```
What is the intensity (energy consumption intensity) of energy
   consumption (kWh in '000s) in 2023?
```

This is the original data table (Table 4.9):

Type of Consumption	Unit	2023	2022
Direct energy consumption			
Diesel consumption	MWh	4.90	3.55
Petrol consumption	MWh	52.74	34.65
Indirect energy consumption			
Electricity consumption	MWh	3884.84	3916.01
Total energy consumption		3942.48	3953.42
Total energy consumption intensity		3.52	3.47

Table 4.9: Example for Unit Error Error: The Original Data Table

The gold answer is:

```
3,520 kWh/FTE
```

Qwen-2-7b generated answer is:

```
The intensity of energy consumption in 2023 is 3.52 MWh/FTE.
```

The model successfully extracted the original data, but it did not convert units as required by the question.

A potential way to reduce this error is to improve the prompt. For example, write explicitly in the prompt:

```
Pay attention to the units of data. If units of data in the question
   does not match the units of data in the original information,
   convert the units as required in the question.
```

Type 4: Incomplete Answer Error

Example question:

```
What is the Scope 1 Emissions greenhouse gas emissions (in tonnes) in
   2022?
```

Table 4.10 shows the original data:

GHG Emissions	Unit	FY2023	FY2022
Scope 1	tCO2e	796.41	809.18
Scope 2	tCO2e	31526	43820
Scope 3	tCO2e	71.06	50.47
Total	tCO2e	32393.23	44679.30
Intensity	tCO2e per m2 floor area	0.42	0.30

Table 4.10: Example for Incomplete Answer Error: The Original Data Table

The gold answer is:

```
809.18 tonnes of CO2e
```

The answer generated from Qwen-2-7b is:

```
Step 1: Identify the type of question
The question starts with "What is", so it is a quantitative question.

Step 2: Determine the relevant information from the text
The text provides the Scope 1 emissions for both 2022 and 2023. The
   relevant information is:
```

The model did not finish generating a direct answer. Instead, it only shows part of its incomplete reasoning process. There may be an issue with the model's response generation, and the response is cut off due to limitations in the interface or an error during the generation process.

Type 5: Hallucination Error

Example question:

```
What is the employee turnover rate for Female in 2022?
```

Gender	2023		2022	
	Number of People	Percent	Number of People	Percent
Female	127	55%	191	69%
Male	34	57%	22	37%

Table 4.11: Example for Hallucination Error: The Original Data Table

The gold answer is "69%", but the answer generated from Qwen-2-7b is:

```
The employee turnover rate for Female in 2022 is 6.
```

However, the number "6" does not exist anywhere in the original information. In other words, this is fabricated data generated from the model. Hallucination is one of major limitations for LLMs.

4.4.2 Error Analysis for tinyllama

Table 4.12 is the contingency table for tinyllama. There are 1429 incorrect answers in total, of which 335 are qualitative and 1094 are quantitative.

	Qualitative	Quantitative	Total
False	335	1094	1429
True	175	435	610
Total	510	1529	2039

Table 4.12: Contingency table for tinyllama

Since the number of incorrect answers is very large, the specific error types and corresponding number of cases are not listed here. However, for most of incorrect questions, the model did not answer the question, but repeated part of the original information and prompt. The following are two representative examples, a qualitative question and a quantitative question, respectively. Most of tinyllama's incorrect answers have the same problems as these two examples.

Example question 1:

```
Is there any description of resources contributed (e.g. money or time)
   to the focus area?
```

This is the original text:

```
## Engaging the Community

Giving back to the community has been a key to the Group's corporate
   social responsibility. We endeavour to create a sustainable business
    as well as a harmonious and sustainable community. This year, the
   Group donated RMB520,000 to Ningbo Yinzhou District Charity
   Federation and Daozhen Township Miao Autonomous County Civil Affairs
    Bureau. The donation would be used to support education and
   development in the communities where we operate.
```

The gold answer is "yes", while the tinyllama generated the following:

```
The Group has donated RMB520,000 to Ningbo Yinzhou District Charity
   Federation and Daozhen Township Miao Autonomous County Civil Affairs
    Bureau. The donation would be used to support education and
   development in the communities where we operate.
```

It repeated the content from the original text, but did not answer the question (i.e. answer "yes" or "no").

Example question 2:

```
What is the intensity (e.g. per unit of production volume, per facility
   ) of hazardous waste produced (in tonnes) in 2023?
```

Original data table is:

Indicator	Unit	Year ended 31 October 2023	Year ended 31 October 2022
Total hazardous waste produced by production plants	tonnes	11	16
Total hazardous waste produced by production plants per unit of floor area	kg/m^2	0.14	0.21

Table 4.13: Example 1: The Original Data Table

The gold answer is "0.14 kg/m2", but the answer generated by tinyllama is:

```
| 2023 | 2022 |
|--------------|--------------|--------------|--------------|
| 11 | 16 |

### Determine the type of question according to the beginning of the
   question, and answer the following
```

The model extracted part of the original data table and part of the prompt, but did not answer the question.

From the answer performance of tinyllama, it can be concluded that this model is not suitable for doing this data extraction task directly. As this the smallest model among all tested models, its poor performance may due to its small size which make it not able to hold all necessary information to generate a complete response. Also, further training or fine-tuning is needed to make it follow instructions more effectively.

4.4.3 Summary

The model results were evaluated based on two metrics: accuracy and speed. The performance of models varies greatly from one another. Except for GPT models, Qwen-2-7b performs the best, which achieves the highest accuracy of 86.9% with the relatively short

running time of 2574 seconds. While tinyllama only achieves the accuracy of 29.9% with running time of 4139 seconds. Qwen 2 models (with 1.B and 7B parameters) have high accuracy in qualitative questions, while Qwen-2-7b and gemma 2 models show high accuracy in quantitative questions.

An error analysis was carried out to investigate the common reasons for incorrect answers for Qwen-2-7b and tinyllama. For Qwen-2-7b, there are two common errors for qualitative questions. The first is misclassifying a question where the answer is "no" as "yes", and the second is generating "no answer". Quantitative error can be roughly categorised as five types: incorrect data error, no answer error, unit error, incomplete answer error and hallucination error. For tinyllama, most of incorrect answers are caused by the model simply repeated part of the original information and prompt, but not answer the question. Improving the prompt or further fine-tuning may reduce the number of incorrect answers.

5 | Discussion

In this chapter, the findings of the study will be critically examined to assess their alignment with the defined research aims and objectives. Additionally, both the strengths and weaknesses of the study will be analyzed to provide a comprehensive understanding of its overall impact.

5.1 Support for Research Aims

As mentioned in Chapter 1, the overall aim of this study is to benchmarking the efficiency of LLMs on ESG data extraction in RAG system, identifying the strengths and limitations of different LLMs in this context and providing insights into optimising their use for ESG data extraction. To achieve this overall aim, specific sub-goals were achieved respectively:

Firstly, the project involves a thorough review of existing literature on LLMs, RAG systems, and their implement on ESG data extraction. Through literature review, it can be found that there is still a research gap when it comes to evaluating LLMs for their specific role in extracting ESG data within RAG systems.

Secondly, an evaluation dataset was constructed. The dataset contains 2039 Q&A pairs. The question part is constructed based on KPIs listed on HKEX ESG Reporting Guide, the original information part is constructed based on 15 ESG reports of companies listed on HKEX. The gold answer was initially generated by ChatGPT-4 and then was checked manually to ensure the correctness.

Thirdly, a research was conducted on mainstream LLMs, including the model size, core capabilities, potential usages and cost. A total of 14 models with different architectures and sizes were selected for comparison. The models are categorised into large, medium and small scales according to their size, with four models in each scale. Two GPT models were also compared as a reference.

Fourthly, a robust benchmarking method was designed. By comparing the model generated answers with gold answer, model performance in ESG data extraction can be evaluated. The evaluation metrics used for comparison is accuracy and running time.

Finally, benchmarking experiments was conducted and the model results were comprehensively analysed, by comparing overall result, and the results of qualitative questions and quantitative questions separately. Also, specific error analyses were carried to the models with the highest accuracy (except GPT model) and lowest accuracy.

Based on the model characteristics (e.g. cost) listed in Chapter 3 and the results obtained in Chapter 4, we can understand the strengths and limitations of different LLMs on ESG data extraction in RAG system. For example, gpt-3.5-turbo achieves the highest accuracy (89.5%) and has the second shortest running time (2095 seconds) among all tested models, but its cost is much higher. Running it in full precision needs 700 GB of VRAM, input token price of $1.50 per 1M Tokens and an output token price of $3.00 per 1M Token. While Qwen-2-7b has the second largest accuracy (86.9%), relatively short running time (2574 seconds), and reasonable cost (about 14.92 GB of VRAM, input token price of $0.10 and output token price of $0.50 per 1M Token). This sheds light on the possibility of using Qwen-2-7b for ESG data extraction tasks.

Through analysing the incorrect answers generated by Qwen-2-7b (see Section 4.4), future direction for improving model performance are implied. For example, optimising prompt may reduce the unit error in quantitative questions. Fine-tuning may improve the model ability in extract ESG data, and thus reduce the misclassify problem or problem with no follow prompt for qualitative questions, incorrect data error and no answer error for quantitative questions.

5.2 Strengths and Weaknesses of the Study

There are several strengths of this study. First, the volumes of the constructed dataset are sufficiently large (contains 2039 Q&A pairs). It covers ESG data with different aspects and KPIs listed on HKEX Reporting Guide (see Table 3.1), and contains both qualitative and quantitative questions for evaluation. Adequate and comprehensive data allow for high reliability of subsequent assessments. Second, it employed an innovative methodology to automatically evaluate the model method. The question design in the dataset simplifies evaluation to a binary assessment (either "true" or "false"). The accuracy of models can be accurately and efficiently evaluated through string match. Third, the study has high replicability. It has clear documentation of methods and procedures, and availability of data and code (see Appendix A). If there are other models that we want to evaluate, it is easy to repeatedly use the same codes, or replicate the previous studies to validate findings.

However, there are some weaknesses of this study. The first weakness is the limited number of evaluation metrics. Due to time limits, this study only tested two evaluation metrics: accuracy and speed. Other dimensions of model performance evaluation are not taken into account, and thus the quality of model answer may not be fully evaluated. Second, as mentioned in Section 3.1.1, HKEX ESG Reporting Guide only covers environmental (E) and social (S) subject areas, but governance (G) subject area is not in this report. The content related to governance is separately reported in the Corporate Governance Code file. However, as the goal of the study is to evaluate the efficiency of LLMs on ESG data extraction in RAG system, Q&A pairs related to governance are also supposed to be in the dataset.

5.3 Summary

In this chapter, the findings of the research are critically examined to evaluate their alignment with the research aims. To achieve the research aim, key steps of this study including reviewing existing literature to find the research gap; constructing a dataset of 2039 Q&A pairs based on HKEX ESG guidelines and ESG reports; selecting 14 LLMs with different architectures and sizes for comparison; designing an automatic evaluation framework and benchmarking LLMs on their efficiency in ESG data extraction. The study identifies strengths, including a large and comprehensive dataset, innovative evaluation methods and good replicability. Also, the weaknesses, including limited evaluation metrics and incomplete governance coverage in the dataset.

6 | Conclusions

This study aims to benchmark the efficiency of LLMs in the context of ESG data extraction within RAG systems, analysing the advantages and drawbacks of various LLMs in this specific application and offering recommendations for enhancing their effectiveness in ESG data extraction.

The project conducts a comprehensive literature review on LLMs, RAG systems and ESG data extraction. Current work shows an increasing interest in using LLMs and RAG approaches for ESG data extraction, but reveals a research gap in comprehensive evaluation framework for LLMs' efficiency in this context.

An evaluation dataset of 2,039 Q&A pairs was created based on HKEX ESG Reporting Guide KPIs and 15 companies' ESG reports, with the gold answers generated by ChatGPT-4 and manually verified. The study assessed 14 mainstream LLMs, categorized by size, and designed an automatic benchmarking method to evaluate model performance using accuracy and running time metrics. Finally, benchmarking experiments were performed, analyzing results for both qualitative and quantitative questions, alongside specific error analyses for incorrect answers.

From the evaluation results, it can be found that there is a large difference in performance between the different models. gpt-3.5-turbo has the highest accuracy of 89.5% and tinyllama has the lowest accuracy of 29.9%. gpt-4 has the shortest running time of 1316 seconds, while yi-34b-bnb-4bit has the longest running time of 11532 seconds. Also, from the research, the cost of model also varies significantly. Usually, the larger size of the model, the higher cost.

Although gpt-3.5-turbo has high accuracy and short running time compared to other models, its cost is much higher (require 700 GB of VRAM, input token price of $1.50, output price of $3.00 per 1M token). Qwen-2-7b has the second largest accuracy (86.9%),

relatively short running time (2574 seconds), and reasonable cost (about 14.92 GB of VRAM, input token price of $0.10 and output token price of $0.50 per 1M Token). Therefore, Qwen-2-7b achieves the best trade-off between model performance and cost among all tested models. The findings indicate a significant advancement in the accuracy and speed of data extraction. By further improving the prompt and fine-tuning the model, the study underscores the potential suitability of Qwen-2-7b to enhance ESG data extraction in various applications.

This study has several strengths, including a large dataset of 2,039 Q&A pairs covering diverse ESG data as per the HKEX Reporting Guide, incorporating both qualitative and quantitative questions for reliable assessments. It uses an innovative methodology for binary evaluations ("true" or "false"), allowing accurate and efficient model assessments through string matching. Additionally, the study offers high replicability due to clear documentation of methods and procedures, with accessible data and code, facilitating easy evaluation of other models and validation of findings.

Despite the positive results, it is essential to acknowledge certain limitations within the study, including limited evaluation metrics, as only accuracy and speed were tested due to time constraints, which may hinder a comprehensive assessment of model performance. Additionally, while the HKEX ESG Reporting Guide includes only environmental (E) and social (S) aspects, it omits governance (G) content, which is found in the Corporate Governance Code. Since the study aims to evaluate LLM efficiency in ESG data extraction, it would benefit from including governance-related Q&A pairs in the dataset for a more complete analysis. Nonetheless, the insights gained provide a solid foundation for future research, paving the way for enhancements in LLM methodologies and broader applications in data extraction.

In conclusion, this study contributes valuable knowledge to the understanding of LLMs in ESG contexts and opens avenues for further exploration. The ongoing development and

refinement of these technologies hold promise for significantly advancing the efficiency and effectiveness of ESG data management in the future.

This research has strong practical relevance. The findings of this research can inform and benefit several key stakeholders such as technology providers and developers. They can use the findings to develop and improve their ESG data extraction tools and platforms. The study highlights the strengths and weaknesses of current LLMs and RAG systems, providing valuable insights for product development. By identifying key factors that affect efficiency through detailed error analysis, developers can implement targeted improvements to enhance the accuracy and speed of ESG data. The error analysis can guide developers in optimising the performance of LLMs and RAG systems, such as optimising prompt or fine-tuning the model.

7 | Future Work

While this study has demonstrated the potential of LLMs in extracting ESG data within a RAG system and evaluated their efficiency in this context, several areas for improvement and exploration remain. This chapter outlines potential enhancements, limitations encountered during the research, and innovative ideas for future investigations.

As mentioned in Chapter 5, one limitation of this study is that LLMs were evaluated only by accuracy and speed. A multidimensional evaluation system could ensure a more comprehensive assessment of model performance. Other evaluation metrics, such as answer relevance and answer faithfulness, were not tested. Therefore, future research should consider additional evaluation metrics for comparing model performance. Another limitation is that the dataset does not include subjects related to governance. Future work should aim to utilize Corporate Governance Code reports to construct more Q&A pairs related to governance, thereby enhancing the generalisability of the findings.

For practical use, as the model's accuracy is not high enough to meet real-world needs, some enhancements should be made. For example, implementing fine-tuning techniques tailored to specific datasets can yield significant improvements. By refining models with domain-specific data, researchers can optimize performance, enabling LLMs to better understand the context of ESG information. Another way to improve performance is by optimizing prompts through analysis of incorrect model answers. These targeted approaches not only increase the reliability of the extracted data but also facilitate the development of practical tools that organizations can utilize for efficient information retrieval. Ultimately, these enhancements would make LLMs more valuable in real-world applications, allowing for more informed decision-making in ESG reporting and analysis.

References

[1] A. de Souza Barbosa, M. C. B. C. da Silva, L. B. da Silva, S. N. Morioka, and V. F. de Souza, "Integration of environmental, social, and governance (esg) criteria: their impacts on corporate sustainability performance," *Humanities and Social Sciences Communications*, vol. 10, no. 1, pp. 1–18, 2023.

[2] S. Mathis and C. Stedman, "What is esg (environmental, social and governance)?," *TechTarget*, 2024. [Online; accessed 2024-09-03].

[3] J. Lee and M. Kim, "Esg information extraction with cross-sectoral and multi-source adaptation based on domain-tuned language models," *Expert Systems with Applications*, vol. 221, p. 119726, 2023.

[4] F. Visalli, A. Patrizio, A. Lanza, P. Papaleo, A. Nautiyal, M. Pupo, U. Scilinguo, E. Oro, and M. Ruffolo, "Esg data collection with adaptive ai.," in *ICEIS (1)*, pp. 468–475, 2023.

[5] Y. Zou, M. Shi, Z. Chen, Z. Deng, Z. Lei, Z. Zeng, S. Yang, H. Tong, L. Xiao, and W. Zhou, "Esgreveal: An llm-based approach for extracting structured data from esg reports," *arXiv preprint arXiv:2312.17264*, 2023.

[6] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, *et al.*, "A survey of large language models," *arXiv preprint arXiv:2303.18223*, 2023.

[7] Elastic, "What is a large language model? a comprehensive llm guide." https://www.elastic.co/what-is/large-language-models/, 2023. Accessed: [Insert Date of Access Here].

[8] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, *et al.*, "Gpt-4 technical report," *arXiv*

preprint arXiv:2303.08774, 2023.

[9] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[10] L. Caruccio, S. Cirillo, G. Polese, G. Solimando, S. Sundaramurthy, and G. Tortora, "Claude 2.0 large language model: Tackling a real-world classification problem with a new iterative prompt engineering approach," *Intelligent Systems with Applications*, vol. 21, p. 200336, 2024.

[11] H. Naveed, A. U. Khan, S. Qiu, M. Saqib, S. Anwar, M. Usman, N. Akhtar, N. Barnes, and A. Mian, "A comprehensive overview of large language models," *arXiv preprint arXiv:2307.06435*, 2023.

[12] P. Béchard and O. M. Ayala, "Reducing hallucination in structured outputs via retrieval-augmented generation," *arXiv preprint arXiv:2404.08189*, 2024.

[13] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, and H. Wang, "Retrieval-augmented generation for large language models: A survey," *arXiv preprint arXiv:2312.10997*, 2023.

[14] E. Pollman, "The making and meaning of esg," *U of Penn, Inst for Law & Econ Research Paper*, no. 22-23, 2022.

[15] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, *et al.*, "A survey on evaluation of large language models," *ACM Transactions on Intelligent Systems and Technology*, vol. 15, no. 3, pp. 1–45, 2024.

[16] IBM, "What are large language models (llms)? | ibm." https://www.ibm.com/topics/large-language-models, Sept. 2024. [Online; accessed 2024-09-04].

[17] A. Vaswani, "Attention is all you need," *Advances in Neural Information Processing Systems*, 2017.

[18] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020.

[19] "What is RAG? - retrieval-augmented generation ai explained - AWS." https://aws.amazon.com/what-is/retrieval-augmented-generation/?nc1=h_ls, Sept. 2024. Accessed: [Online; accessed 2024-09-04].

[20] Acorn, "Understanding RAG: 6 Steps of Retrieval Augmented Generation (RAG)." https://www.acorn.io/resources/learning-center/retrieval-augmented-generation, Sept. 2024. [Online; accessed 2024-09-23].

[21] Pareto AI, "The Ultimate Guide to Retrieval-Augmented Generation (RAG)." https://pareto.ai/blog/retrieval-augmented-generation, Sept. 2024. [Online; accessed 2024-09-23].

[22] Trantor Inc., "What is RAG (Retrieval Augmented Generation)?." https://www.trantorinc.com/blog/what-is-rag-retrieval-augmented-generation, May 2024. [Online; accessed 2024-09-23].

[23] Seldon, "Harnessing the power of llms for automated data extraction." https://www.seldon.io/harnessing-the-power-of-llms-for-automated-data-extraction, June 2024. Accessed: 2024-08-28.

[24] A. Ontiveros, "Development and evaluation of an it tool for esg compliance consulting for companies using llms,"

[25] T. K. Gupta, T. Goel, I. Verma, L. Dey, and S. Bhardwaj, "Knowledge graph aided llm based esg question-answering from news," 2024.

[26] O. Ardic, M. U. Ozturk, I. Demirtas, and S. Arslan, "Information extraction from sustainability reports in turkish through rag approach," in *2024 32nd Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, IEEE, 2024.

[27] M. Bronzini, C. Nicolini, B. Lepri, A. Passerini, and J. Staiano, "Glitter or gold? deriving structured insights from sustainability reports via large language models," *EPJ Data Science*, vol. 13, no. 1, p. 41, 2024.

[28] alva, "Understanding the attributes of esg data," *alva*, 2022. Accessed: 2024-09-02.

[29] Brightest, "Esg data - what it is, types how to use it," *Brightest*, 2024. Accessed: 2024-09-02.

[30] "Rules and Regulations." https://www.hkex.com.hk/Listing/Sustainability/ESG-Academy/Rules-and-Regulations?sc_lang=en, Sept. 2024. Accessed: 2024-09-07.

[31] The Stock Exchange of Hong Kong Limited, Hong Kong, *Appendix C2 Environmental, Social and Governance Reporting Guide*, 2024. Accessed: 2024-09-07.

[32] OpenAI, "GitHub - openai/openai-python: The official Python library for the OpenAI API." https://github.com/openai/openai-python, 2024. [Online; accessed on 2024-09-17].

[33] pandas Development Team, "pandas - Python Data Analysis Library." https://pandas.pydata.org/, 2024. [Online; accessed on 2024-09-17].

[34] Python Software Foundation, "json — JSON encoder and decoder." https://docs.python.org/3/library/json.html, 2024. [Online; accessed on 2024-09-17].

[35] OpenAI, "GPT-4 Turbo and GPT-4 Model Documentation." https://platform.openai.com/docs/models/gpt-4-turbo-and-gpt-4, 2024. Accessed: 2024-09-23.

[36] OpenAI, "GPT-3.5 Turbo Model Documentation." https://platform.openai.com/docs/models/gpt-3-5-turbo, 2024. Accessed: 2024-09-23.

[37] The Hugging Face Community, "Yi 3.4B BNB 4-bit." https://huggingface.co/unsloth/yi-34b-bnb-4bit, 2023. [Online; accessed 2024-09-23].

[38] The Hugging Face Community, "GEMMA 2 27B." https://huggingface.co/unsloth/gemma-2-27b, 2023. [Online; accessed 2024-09-23].

[39] The Hugging Face Community, "LLaMA 2 13B." https://huggingface.co/unsloth/llama-2-13b, 2023. [Online; accessed 2024-09-23].

[40] The Hugging Face Community, "Mistral-Nemo Base 2407." https://huggingface.co/unsloth/Mistral-Nemo-Base-2407, 2023. [Online; accessed 2024-09-23].

[41] The Hugging Face Community, "GEMMA 2 9B." https://huggingface.co/unsloth/gemma-2-9b, 2023. [Online; accessed 2024-09-23].

[42] The Hugging Face Community, "LLaMA 3 8B." https://huggingface.co/unsloth/llama-3-8b, 2023. [Online; accessed 2024-09-23].

[43] The Hugging Face Community, "Qwen2-7B." https://huggingface.co/unsloth/Qwen2-7B, 2023. [Online; accessed 2024-09-23].

[44] The Hugging Face Community, "Mistral 7B." https://huggingface.co/unsloth/mistral-7b, 2023. [Online; accessed 2024-09-23].

[45] The Hugging Face Community, "GEMMA 2 2B." https://huggingface.co/unsloth/gemma-2-2b, 2023. [Online; accessed 2024-09-23].

[46] The Hugging Face Community, "GEMMA 2B." https://huggingface.co/unsloth/gemma-2b, 2023. [Online; accessed 2024-09-23].

[47] The Hugging Face Community, "Qwen2-1.5B." https://huggingface.co/unsloth/Qwen2-1.5B, 2023. [Online; accessed 2024-09-23].

[48] The Hugging Face Community, "Tiny Llama." https://huggingface.co/unsloth/tinyllama, 2023. [Online; accessed 2024-09-23].

[49] Python Software Foundation, "Python: Regular expression operations." https://docs.python.org/3/library/re.html, Sept. 2024. [Online; accessed 19-September-2024].

A | Source Code

Source code for all of the methods implemented in the project can be found in the GitHub repository:
https://github.com/BeiyiPan/Benchmarking-LLMs-on-ESG-data-extraction-in-RAG-system.