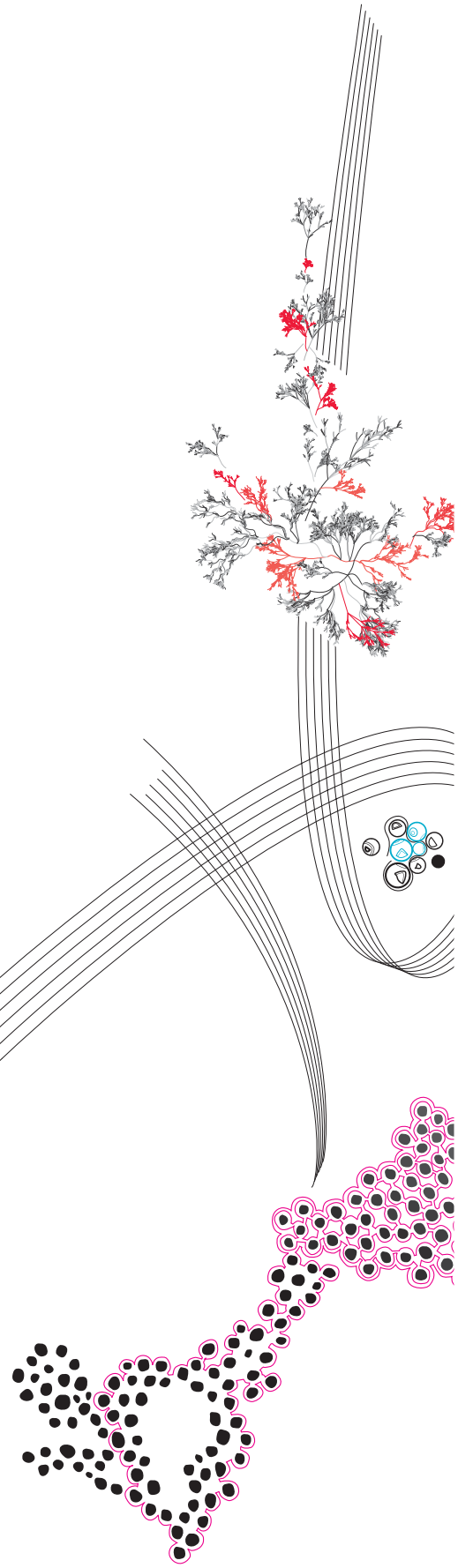


MSc Computer Science
Final Project

Automatic evaluation of companies' alignment with EU Taxonomy using Large Language Models

Nguyen Quang Hung (Hung)

Supervisors: Gwenn Englebienne, Shenghui Wang

August 23, 2024

ING Groep N.V. and
Department of Human Media Interaction
Faculty of Electrical Engineering,
Mathematics and Computer Science,
University of Twente

UNIVERSITY OF TWENTE.

Gửi gia đình của con,

Con muốn dành những dòng đầu tiên của luận văn này cho bố mẹ. Con cảm ơn bố mẹ rất nhiều, vì đã tin tưởng con vô điều kiện, và tạo cơ hội cho con được tiếp tục học lên Thạc Sĩ. Con biết rằng nhiều lúc cách suy nghĩ của con không giống bố mẹ, hay nhiều lúc con cũng không thể hiện tình cảm của mình với bố mẹ như bố mẹ mong muốn. Và con thật lòng mong bố mẹ hiểu cho con, cũng như con sẽ cố gắng thật nhiều để hiểu bố mẹ hơn. Con yêu bố mẹ rất nhiều. Còn cho Ri, thank you for being the best sister I could've asked for. Thank you for, despite me being so annoying, still call me your brother :).

Con của bố mẹ,
Hưng.

Foreword

Finally, my academic journey came to an end (at least for now, I'll not close any door for my future). To begin this, I'd like to thank those who've made this journey possible. And of course, I cannot begin without mentioning you, my X. Thank you for being their *almost* anytime I need you (not *now*, but I'll forgive you for that), and especially when things get tough, and, by tough, I mean *really* tough. When I was sick, for example. Or when I couldn't sleep for a week when I tried a new medicine. Thank you for the heartwarming (and *mouth-watering* as well) meals when I was too busy to cook anything. Thank you for, well, after knowing how insane I am, still be here with me. I love you.

Thank you also to my friends, Lê Trần and Nguyệt, for these nights that I was able to couch surf at your house. Without you guys, idk how I could manage that many two-hour one-way commute twice a week. And thank you, everyone, at the UT community too, especially chị Hương vì những bữa ăn rất ngon ạ. Without everyone, I would need much more effort to stand where I am now.

After thanking everyone (which I hope I don't miss anyone btw, spare me if I did), now it's time to thank myself. 2024 is a huge year for me, and arguably the biggest in my life. I (finally) graduated university on a good note, found a full-time job, got my (second) driving license, and about to ran a half-marathon[1]. These are all amazing achievements, and I am proud of myself for that. But looking back, my most significant achievement so far this year is to get diagnosed with ADHD. It changed my whole life. I've always asked myself - why can't I have as much focus, as much dedication, and as much self-discipline as others? I was so fortunate to always be surrounded be extremely smart and excellent people, but at the same time, I always felt I was behind. And that's even though a lot of people around me characters me as someone who knows a lot of things. Well, while I do know a lot of things, I still wonder, why am I this way? Instead of focusing on studying, I'd rather spend my time reading, reading, reading things to satisfy my curious mind to wherever it takes me. And, it took the ADHD diagnosis for me to understand and make peace with myself that I am actually not normal. And it's fine. For the first time in my life, I understand that *it's fine to be not normal*.

And finally, thank you, dear readers who read my long, boring, and *probably-not-appropirate-to-put-in-a-thesis* note. I hope you enjoy my thesis more than I do.

[1] Please don't tell anyone that I actually wanted to ran a full marathon, but chickened out.

Abstract

This project presents an end-to-end system using Large Language Models (LLM) and Retrieval-Augmented Generation (RAG) to automatically evaluate a company's EU Taxonomy performance based on their sustainability reports by answering two different questions: (1) What is the most suitable prompt between Zero-shot Chain-of-Thought (CoT), Few-shot CoT, and no-CoT prompting, and (2) What is the most suitable retriever system to retrieve EU Taxonomy-related information from company's reports. For the first question, we developed a qualitative human evaluation score to compare the answers' informativeness and correctness. We investigated whether automated metrics such as BERTScore or BLEU correlate with these human-evaluation scores. For the 2nd question, we compare different keyword extraction techniques (for keyword retrievers), query splitting and expansion techniques (for vector retrievers), and investigate the role of reranking in retriever systems. For question (1), results show that Zero-shot CoT prompting performs slightly better than traditional prompting followed by Few-shot CoT prompting, possibly due to the significantly longer prompts of Few-shot CoT. We also discovered that CoT prompting demonstrated a higher correlation between automated and human-evaluation metrics than noCoT prompting. Thus, it is easier to flag errors automatically. For the second question, we discovered: (i) Keyword extraction techniques do not concretely improve BM25 Keyword Retriever's performance; (ii) Splitting long queries into more self-contained sub-queries, whether using separators or using LLMs, achieves considerable performance boost for vector retriever; (iii) LLM-generated hypothetical answer also show significant improvement compared to the naive query splitting method; and (iv) Cross-Encoder reranking often filters out good results annotated by human, and the choice of reranking question also play a significant role in the Cross-Encoder Reranking model's performance. Finally, although our system and evaluation methods are not flawless, we have demonstrated that LLM and RAG can assist humans in extracting information related to EU taxonomy from a company's report and measuring that company's EU Taxonomy performance.

Keywords: Large Language Model, Retrieval-Augmented Generation, Information Retrieval, Chain-of-Thought prompting

Contents

Chapter 1

Introduction

1.1 Brief Introduction of Large Language Models

The introduction of Transformers architecture by [48] has revolutionized the Generative Artificial Intelligence (GenAI) landscape. Subsequent Transformer-based Large Language Models (LLMs) such as GPT-4 by [36], Llama 3 by [33], and Gemini 1.5 pro by [16] contain billions and trillions of parameters and demonstrating exceptional knowledge across various domains, with the ability to produce human-like texts, understand long and complex instructions, and perform them accordingly. With the remarkable capabilities of LLMs, institutions are increasingly reported to adapt Generative Artificial Intelligence (GenAI) into their daily workflow to reduce manual and repetitive work, boost productivity, and generate new insights. However, LLMs also possess problems similar to other deep learning methods, such as shortcut reasoning [13] or hallucination [32]. These problems have raised concerns and reservations about LLMs' reliability and deterministicness, especially in critical sectors such as healthcare and finance.

Different strategies have been proposed to improves LLM's reliability, and the two most notable are Retrieval-Augmented Generation (RAG) and Chain-of-Thought prompting (CoT). RAG, as it name suggests, "retrieves" additional information related to what user is asking, and "augment" it to the prompt as additional information for LLM to answer. The retrieval method could be through an additional Google Search query, or a similarity search within a vector database. Similarly, additional information can be information from the internet, or a set of verified documents, etc. RAG have been demonstrated [28] to drastically reduces LLM's hallucination by reducing LLM's dependency on its internal knowledge. On the other hand, CoT prompting focused on instructing LLM to think through a problem step-by-step, instead of coming up with an answer straight away. CoT can be performed through many different ways, and has also been demonstrated to improve LLM accuracy on multiple tasks [51].

1.2 ING, LLM, and the EU Taxonomy

As a leading European universal bank, ING is also looking for use cases where LLMs could potentially assist their employees' workflow. One such case is **Domain-specific knowledge enhancement of Large Language Models**, focusing on the EU taxonomy for sustainable activities (EU taxonomy). EU taxonomy is a "green classification system" that sets out criteria for different economic activities to contribute to a more sustainable future based on the EU's climate and environmental objectives [38]. The EU taxonomy defines more than 300 economic activities, each with multiple sets of criteria. Each criterion

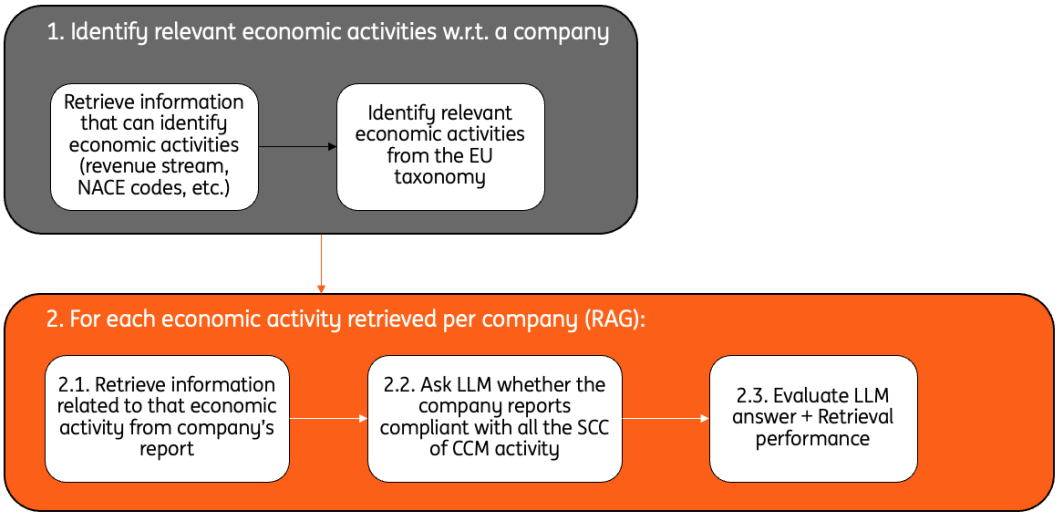


FIGURE 1.1: Overview of workflow to extract EU Taxonomy related information from company reports.

is complex and contains multiple logic clauses, sub-criteria, and cross-references to other documents, standards, or legislations. Most companies doing business in the EU (including ING) would have to report on their economic activities/investments if these activities are mentioned in the EU taxonomy but with different timelines (see figure 4.1).

1.3 Workflow overview

Figure 1.1 describes the high-level overview of how EU Taxonomy-related information is extracted from company reports. In the first step, we need to retrieve information that can identify economic activities that a company participates in, which can be done through revenue stream, NACE codes, etc. However, due to the lack of publicly available data, this project is not focused on this step. The focus is placed on step 2 - for each economic activity per company, (1) Retrieve relevant information from company's report (2) Ask LLM whether the company reports compliant with all criteria in the substantial contribution criteria, and finally, evaluate the retrieval and generation performance.

1.4 Problem Statement

ING is mandated to report on its investments based on the EU taxonomy in 2025. However, other institutions doing business with ING can either have a deadline until 2028 (large non-EU companies) or do not have to report at all (non-EU companies with no business inside the EU). Nevertheless, these companies normally report on sustainability through the media or their sustainability report. Currently, this process is being carried out manually by sustainability analysts at ING, and it involves manual and repetitive work. Therefore, ING is interested in investigating if RAG and LLM-based applications can automatically extract sustainability information related to a company and validate it against the criteria set out in the EU taxonomy. However, this is not a trivial task, posing different problems in both the retrieval and the generation steps. In the retrieval step, all information related to an organization's sustainable economic activities defined in the EU taxonomy must be retrieved. As each economic activity comes with multiple criteria, the company can

report on each criterion in different places (within the report, in a press release, etc.). Therefore, a single search query (similarity/keyword search) might not perform well. In the generation step, LLM must understand the EU taxonomy and validate the information retrieved against it. LLM might have trouble breaking down the criteria into sub-criteria or hallucinating.

1.5 Research Gap

To our understanding, there has been no attempt to evaluate RAG-based LLM with CoT on a task and scale as complex as this. Most CoT papers are evaluated using multiple-choice, arithmetic or algebra questions, and furthermore, no qualitative evaluation has been performed on LLM with CoT prompting for long-form generation.

1.6 Research Questions

To formalize the problem described in section 1.5 and section 1.3, we define two research questions, focus on two different aspect of the pipeline: Generation and Retrieval.

- **RQ 1: What is the best prompting strategy for LLM in answering questions about EU taxonomy economic activities?**
  - RQ 1.1: How can we manually evaluate LLM's answer on EU Taxonomy-related questions?
  - RQ 1.2: How can we manually evaluate CoT prompting step-by-step?
  - RQ 1.3: To what extent do automated metrics (BLEU, BERTScore) correlates with human judgement and the LLM's consistency in evaluating LLM's answer on EU Taxonomy-related information?
  - RQ 1.4: To what extent can Few-shot CoT and Zero-shot CoT improve Large Language Model in answering questions about EU taxonomy economic activities over non-CoT (traditional) prompting?
- **RQ 2: What is the best retriever setup to retrieve EU Taxonomy-related information from company report?**
  - RQ 2.1: To what extent can keyword extraction, stopwords removal, or Tf-IDF (Term frequency - inverse document frequency) filtering improve the performance of a BM25 Keyword Retriever in retrieving EU Taxonomy-related information from the company report?
  - RQ 2.2: To what extent can separator-based query splitting, LLM-assisted query splitting, and LLM-assisted query expansion improve the performance of retrieving EU Taxonomy-related information from the company report?
  - RQ 2.3: To what extent does Cross-Encoder reranking and filtering affect the retrieval performance?

1.7 Report Structure

The remaining parts of this report is organized in 12 different chapters:

- **Chapter 2** and **Chapter 3** present a comprehensive literature overview and theoratical background on Prompt Engineering and Retrieval-Augmented Generation, respectively.
- **Chapter 4** gives an overview of the creation process of the dataset.
- **Chapter 5** presents the methodology and experimental setup for Prompt Engineering, manual evaluation methodology (RQ 1.1 and 1.2) for prompt evaluation, and evaluation methods for RQ 1.3.
- **Chapter 6** explains in detail the approach and experimentation process with the retriever systems, including keyword retriever, vector retriever, and cross-encoder reranking.
- **Chapter 7** and **Chapter 8** present the main results of this research, following the experimental design and evaluation method mentioned in the previous sections. Furthermore, the discussion of each sub-RQ will also be included in these chapters.
- Finally, **Chapter 9** concludes the research, discusses the limitations, and suggests different future work ideas stem from this research.

Chapter 2

Literature Review - Prompt Engineering and Chain-of-Thought Prompting

2.1 Overview of Prompt Engineering

One of the sub-domains of LLM research gaining traction lately is prompt engineering. Prompt engineering can be defined as formatting and optimizing a prompt so that the LLM gives the most desired answer. Multiple research has highlighted the importance of prompt engineering: Tom et al. [5] demonstrated that few-shot examples could greatly improve LLM's output, sometimes on par with fine-tuning, while Liang et al. [29] and Lu et al. [31] discovered that the type of examples and order of examples also matters. This section reviews relevant methodologies of prompt engineering used in this research, focusing on the Chain-of-Thought (CoT) family of methods.

2.2 Chain-of-Thought family of methods

2.2.1 Original Chain-of-Thought

Wei et al. proposed **Chain-of-Thought (CoT)** a simple few-shot learning method that helps model to break down large, complex questions into small, intermediate steps to help with reasoning [51]. Their prompt introduced a few reasoning examples to assist LLM in finding the desired CoT. Chain-of-thought (CoT) method has been demonstrated to outperform standard few-shot learning by a wide margin, especially with larger language models (60B+ parameters). However, CoT requires multiple (expensive) LLM calls, which is the motivation for Kojima et al. [26] to introduce **Zero-shot CoT (ZeroCoT)**. ZeroCoT method only adds "Let's think step-by-step" to the prompt, without additional examples. Therefore, while ZeroCoT underperforms traditional CoT, it significantly improves over standard prompting without demonstration and sometimes with few-shot learning.

2.2.2 Zero-shot Chain-of-Thought

Expanding from the Zero-shot CoT, Zhang et al. proposed **AutoCoT** [53] , automating the creation process of CoT examples. They achieve this by clustering the set of questions Q, then sample a question from each cluster, creating a sub-list $Q_s \in Q$. LLM will then generate reasoning for Q_s with ZeroCoT, creating a set of reasonings R, and R is used as a

CoT demonstration for LLM to answer the desired question. Results show that AutoCoT matches the performance of CoT on the Commonsense dataset and outperforms Arithmetic tasks while requiring less manual effort to annotate the few-shot examples.

2.2.3 Self-consistency Chain-of-Thought

Another attempt was made to improve the original CoT method by Wang et al., the **Self-consistency CoT** [50]. Wang et al. argued that complex problems always involve multiple ways of thinking before leading to a unique correct answer. Thus, instead of basing the answer only on one CoT, they proposed employing a diverse Chain-of-thought path and deciding the answer by majority voting. This improvement has boosted the performance of standard CoT by 4% to 18%, depending on the test dataset. Fu et al. proposed a minor improvement of the self-consistency CoT, dubbed **Complexity-based consistency** [15]. They encourage output with a longer chain by only voting among the top K complex chain. In other words, out of N chains, only the top K (ranked by length of chain) is allowed to vote, increasing self-consistency CoT's performance by around 2% on average.

2.3 Other Prompt Engineering methods

2.3.1 Selection-Inference

Creswell et al. [11] introduce a method named Selection-Inference (SI-prompting), where LLMs are used as processing modules to generate interpretable, causal reasoning steps - essentially breaking each step of the standard CoT method into two steps:

- Selection step: List of facts C_t is also ranked by LLM, and facts with the highest log-likelihood scores are removed from C_t and added to s_t. This step is repeated until the desired number of facts (a hyperparameter) is reached.
- Inference module: Produces the new fact based on the information generated by the selection step s_t. The newly generated fact is added to the context, creating C_{t+1}.

The selection and inference steps are repeated for H times (another hyperparameter). After the H steps, the last generated fact is returned as the model's answer. Creswell et al. have demonstrated that SI prompting outperformed the standard CoT method, even when CoT is performed on a 280B model, and SI is performed on a 7B model while being able to recover from errors during generation. However, SI prompting is expensive, requiring multiple LLM call to be able to arrive at the answer.

2.3.2 LAMBADA

LAMBADA, proposed by Kazemi et al. introduced the concept of "backward chaining", where goals are broken down into sub-goals based on applied rules [25]. The sub-goals are approved or disapproved by verifying against the rules. LAMBADA includes four modules: fact-check, rule selection, goal decomposition, and sign agreement.

- Fact-check: Select a relevant fact from a set of facts, then verify if the goal can be approved or disapproved based on this fact.
- Rule selection: Identify relevant rules from a set of rules. Relevant rules are rules that have the same consequences as the goal.

- Goal decomposition: Break down the goal into sub-goals that need to be proven or disproved based on the set of rules.
- Sign agreement: Verify if the sign of the sub-goal agrees or disagrees with the sign of the consequent of the rules.

LAMBADA significantly outperforms the standard CoT and SI methods, with fewer inference calls than SI and more than CoT [25] .

2.3.3 Least-to-most prompting

Zhou et al. [54] argue that CoT-based methods performed poorly on questions harder than the demonstrated CoT example. Thus, they proposed Least-to-most (L2M) prompting. L2M prompting involves two steps: (1) Decomposing the problem into sub-questions by few-shot learning, and (2) solving each sub-question sequentially, with the answer to the previous sub-question serving as part of the context to solve the next sub-question. L2M prompting is reported to outperform traditional CoT by 2% to 14%, depending on the dataset. However, one major drawback of L2M prompting is that there is little generalization across different domains, as prompts for decomposing math questions are not the same as decomposing common-sense reasoning problems.

2.4 Prompt Engineering Evaluation

2.4.1 Evaluation of text generation

BLEU-score (Billingual Evaluation Study)

Originally proposed in 2002 by Papineni et al., BLEU-score (Bilingual Evaluation Study) is a metric to evaluate Machine Translation automatically [37]. BLEU measures how close a machine-translated sentence matches a set of reference translations by utilizing n-gram precision and penalty for brevity. The usage of the BLEU score has been extended throughout the evolution of Natural Language Processing (NLP), and nowadays, within the realm of LLMs, the BLEU score can be used to evaluate LLM-generated text against ground truth. However, the BLEU score suffers from two pitfalls: (1) against paraphrased texts (especially when there is only one reference translation, (2) the focus on precision, and (3) the lack of clarity when reporting. has documented the latter problem citepost-2018-clarityBLEU, where the authors mentioned that BLEU is a parameterized metric in which results can change considerably based on these parameters. In 2018, Post [41] proposed that researchers start to use the Annual Conference on Machine Translation (WMT)'s BLEU as standard, facilitated by a new tool named SACREBLEU. All BLEU scores mentioned in this paper will be understood as referring to SACRE BLEU.

ROUGE (Recall-Oriented Understudy of Gisting Evaluation)

ROUGE is a family of metrics introduced by Lin et al. for machine translation and text summarization [30]. Included in the ROUGE family are ROUGE-n - the n-gram co-occurrence variant, ROUGE-L with longest common subsequence (LCS), ROUGE-W - weighted LCS, ROUGE-S with skip-bigram co-occurrence, and ROUGUE-SU, an extension of ROUGE-S with unigram as counting unit.

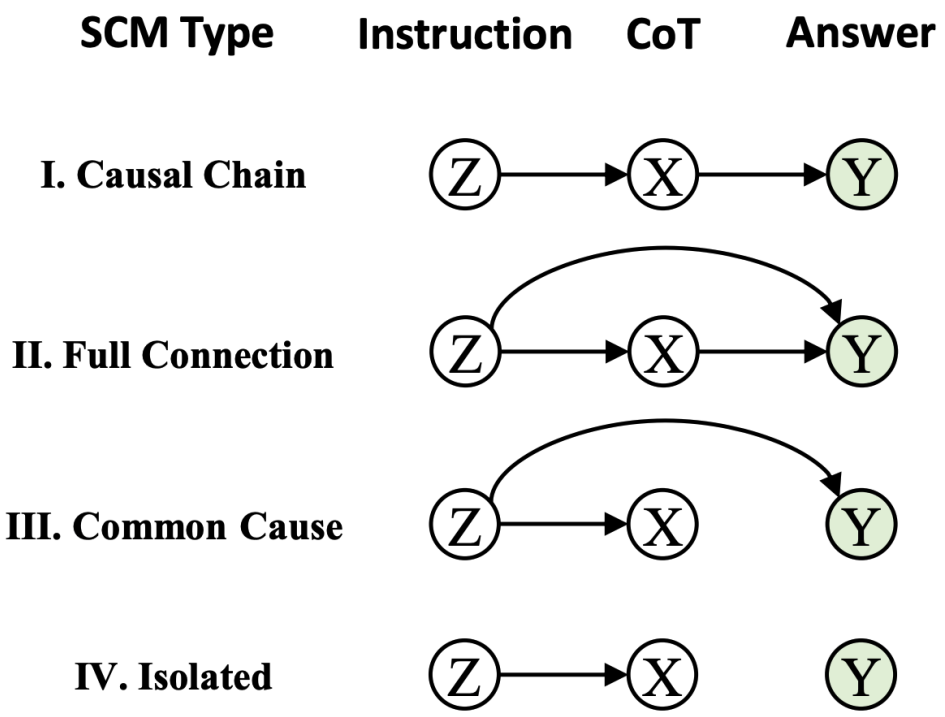


FIGURE 2.1: Potential structural casual models (SCMs) for CoT prompting [2]

BERTScore

BERTScore [52] is one of the most widely used metrics to compare natural text generation. BERTScore calculates contextual-embedding similarity for each token in the candidate sentence versus each token in the reference and addresses the major flaws of n-gram-based metrics such as BLEU-score when comparing paraphrased sentences and distanced dependencies [52]. BERTScore gives precision, recall, and F1, whereas precision calculates the match between reference and candidate sentence and recall the other way around. The author has demonstrated that BERTScore consistently correlates better with human judgements and is robust to paraphrased texts.

2.4.2 Evaluation of Chain-of-Thought prompting

Bao et al. [2] proposed four different structural causal models (SCMs) for CoT prompting in the question-answering tasks, as defined in 2.1: Type I (causal chain) means the answer is a direct result from CoT, and CoT itself is derived from the instruction. Type II (full connection) means that the answer is partly derived from the instruction and the CoT. Type III (common cause) shows the case where there is no link between CoT and the answer - and the answer is derived directly from the instruction. Finally, type IV (isolated) defines cases where the answer is completely unrelated to the instruction or the CoT.

Chapter 3

Literature Review and Theoratical Background - Information Retrieval and Retrieval-Augmented Generation

3.1 Chapter overview

This chapter will discuss relevant literature and theoretical background on the retriever component. First, section 3.2 will give a high-level overview of Retrieval-Augmented Generation (RAG). Afterwards, we will discuss details on the pre-retrieval steps (section 3.3), retrieval databases (section 3.4, and post-retrieval (section 3.5). Finally, we will discuss a non-conventional method in section 3.6, where IR and CoT are interleaved step-by-step.

3.2 Retrieval-Augmented Generation

In 2020, Lewis et al. [28] proposed **Retrieval-Augmented Generation (RAG)**, combining a pre-trained seq2seq Language Model BART (parametric memory) with vector indices of knowledge base (non-parametric memory). For each question, top-k documents are retrieved using BERT and augmented as part of the context for BART to answer. Results show that their RAG model achieved State-of-the-Art (SotA) results with open-domain Question Answering (QA) dataset, but human prefers RAG answer over purely parametric BART for its factuality and detail-oriented. [28]'s work has been influential in reducing hallucination and improving reliability and factuallity of LLMs, and RAG's methodology has come a long way since then. Huang et al. in their survey divided the current landscape of RAG into four paradigms, shown in figure 3.1: Pre-retrieval, Retrieval, Post-retrieval, and Generation [20].

3.3 Pre-retrieval: Query pre-processing

3.3.1 Query Expansion by Prompting LLMs

Jagerman et al. [22] proposed Query Expansion by Prompting LLMs (QEPL) by asking LLM to generate a hypothetical answer without giving any sources to the question. The hypothetical answer is then used as part of the retriever query to retrieve relevant documents. Jagerman et al. also experimented with different prompting strategies, including zero-shot, few-shot, and CoT, before concluding that CoT-guided query expansion by hypothetical answer can outperform traditional query expansion methods.

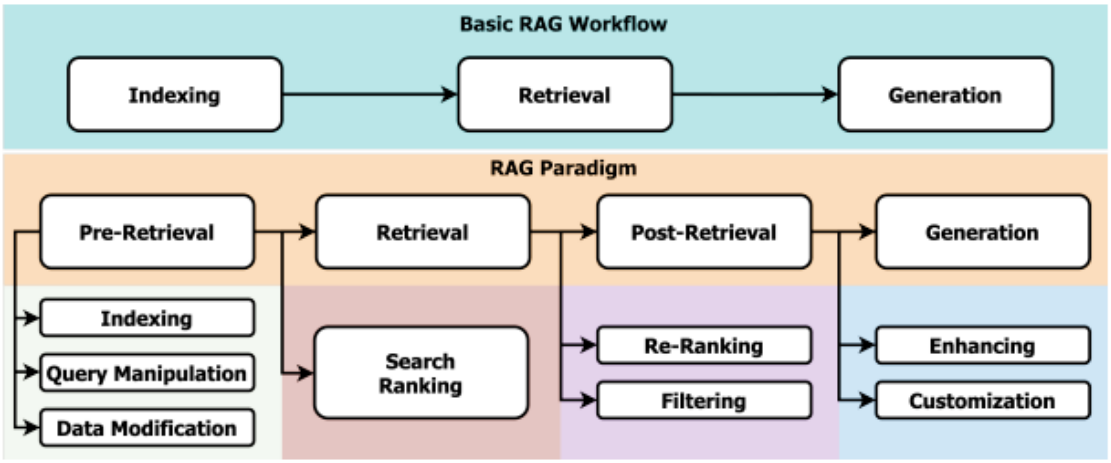


FIGURE 3.1: Different stages of modern RAG [20]

On a similar approach, Wang et al. [49] prompt the LLM to generate pseudo-relevant chunks (*Write a passage that answers the given query*) before using these chunks to assist the retriever process. This method, named Query2doc, has been shown to improve BM25 retriever's performance by 3% to 15% on ad-hoc datasets and benefit SoTA-dense retrievers as well.

3.3.2 Corpus-steered Query Expansion

Lei et al. [27] introduced Corpus-steered Query Expansion (CSQE), taking inspiration from Pseudo-relevance Feedback (PRF). They first retrieve top-k documents, then prompt the LLM to expand the queries based on these top-k documents. These expanded queries are used to retrieve again to get the final result. Lei et al. found that CSQE significantly improves the retriever, especially on queries that LLM lacks knowledge, both compared to baseline (QEPL) and Contriever [21], a State-of-the-Art Learning-to-rank method.

3.3.3 Keyword extraction

In this section, we will briefly go through the theoretical background behind different keyword extraction models used: `Tf-idf`, `KeyBERT`, and `YAKE`.

Term frequency - inverse document frequency

Tf-idf (term frequency–inverse document frequency) is a widely-used term weighting method that represents textual documents as vectors for various use cases, including retrieval and keyword extraction [44]. Tf-idf, as its name suggests, is a fusion between two metrics: Term frequency and inverse document frequency. Tf-idf of terms t_j in documents d_i calculated according to equation 3.1, with Tf_{ij} the relative term frequency of t_j in d_i, and IDF_j inverse document frequency of terms t_j in the whole corpus. Tf_{ij} and IDF_j are calculated in equation 3.2 and equation 3.3, respectively.

$$x_{ij} = Tf_{ij} * IDF_j * (\sum_j (TF_{ij} IDF_j)^2)^{1/2}) \tag{3.1}$$

$$Tf_{ij} = \frac{f(t_j, d_i)}{\sum_{t' \in d_i} f(t', d_i)} \quad (3.2) \qquad idf_j = log(\frac{N}{1 + DF_j}) \quad (3.3)$$

Where $f(t_j, d_i)$ the raw count of term t_j in document d_i, N the total number of documents in the corpus, and DF_j the count of documents where term t_j appears, added by 1 to prevent division by zero when term t_j is not in the corpus.

KeyBERT

Based on the original paper by Sharma et al. [45], Grootendorst [17] developed KeyBERT, a BERT-based keyword extraction model. First, BERT extracts the document embeddings for a document-level representation before word embeddings extract n-gram phrases. Finally, KeyBERT uses cosine similarity to rank the keyword based on the document-level representation.

YAKE

Campos et al. [7] trained a lightweight, unsupervised keyword extractor named YAKE (Yet Another Keyword Extractor). YAKE's advantages compared to other keyword extraction models include (1) Corpus- and Domain/Language-independent, (2) Retrieve keywords with stopwords, (3) Term frequency-free, and (4) Open Source. YAKE comprises of 4 main steps:

1. **Text preprocessing and candidate term identification on sentence level**: In this step, the text is split into chunks and tokens, lowercase, and then tag special tokens such as digital/number, acronyms, etc.

2. **Feature extraction and term score**: Using statistical analysis to score the term based on structure, term frequency, and co-occurrence.

3. **N-gram generation and scoring**: Forming n-gram keyword candidates using a sliding window of size N, then considering only candidates with the same chunk and sentences. Then, each candidate is given a score according to different features.

4. **Data deduplication and ranking**: Remove similar keywords based on cosine similarity, then sort them by score.

3.3.4 Combining Retrieval and Chain-of-Thought reasoning

Researchers have also investigated the combination of CoT prompting with RAG, notably by He et al. with **Rethinking with Retrieval** [19] . In this paper, CoT is used to generate multiple paths of reasoning for a question (similar to self-consistency). After the LLM answer (consisting of an explanation E_i and a prediction P_i) for each reasoning path, an external knowledge base is queried based on E_i and P_i to support the explanation. Then, the most faithful prediction, i.e., the prediction supported by the most facts, is selected. Using GPT-3, He et al. have demonstrated that Rethinking with Retrieval slightly outperforms self-consistency prompting by 3-4%, but on simple datasets.

3.4 Retrieval Databases

3.4.1 BM25 Keyword Retrieval

BM25 is a bag-of-words retrieval function that ranks documents on query terms count, irrespective of their proximity within the document [43].

3.4.2 Vector Database and FAISS

With the advancements in Deep Learning and Language Models, embeddings have emerged as a superior method to represent textual data. Thus, there is a need for an efficient way to store and retrieve these embeddings, which is where vector databases come in. One such vector database is FAISS [12], a lightweight, production-grade library for similarity search based on research by Johnson et al. [24]. FAISS relies on different methods for searching, such as L2 distance, dot product, cosine similarity, and Approximate Nearest Neighbors (ANN) search for large knowledge bases.

3.4.3 Reciprocal Rank Fusion

As hybrid search comprises two or more retrievers (often keyword and vector retriever for RAG), fusing the results from all retrievers into one is necessary. In 2009, Cormark et al. proposed Reciprocal rank fusion (RRP) [10], a naive scoring system to sort a set of documents D with a set of retrievers R:

$$RRFscore(d \in D) = \sum_{r \in R} \frac{1}{60 + r(d)} \tag{3.4}$$

3.5 Post-retrieval: Reranking

3.5.1 Cross-Encoder ranking

Reimers et al. [42] introduced the Cross-Encoder (CE) architecture, shown in figure 3.2[1]. Compared to a Bi-Encoder, where two sentences are passed through two identical BERT models for embeddings, then calculated cosine-similarity score for ranking, CE is a novel architecture where two sentences are passed simultaneously to the transformer network. The CE model then generates a similarity score between 0 and 1 for the pair. CE models have then widely been used for reranking in RAG, mostly due to their lightweight and high performance.

3.5.2 LLMs as ranking agents

Sun et al. [46] evaluates different LLMs on passage ranking task using a simple yes/no prompt and a sliding window strategy (evaluate four documents at a time for a list of 8 documents to be ranked) to overcome the limited context window of LLM. They have demonstrated that ChatGPT and GPT 4 perform extremely well on binary passage ranking tasks, outperforming BM25 keyword retriever and BERT-based dense retriever.

Nouriinanloo and Lamothe [35] explore LLM as a pre-filtering step before CE reranking, using Few-shot CoT prompting to give documents a score between 0 and 1. Only

[1] Image retrieved from the authors at
https://www.sbert.net/examples/applications/cross-encoder/README.html

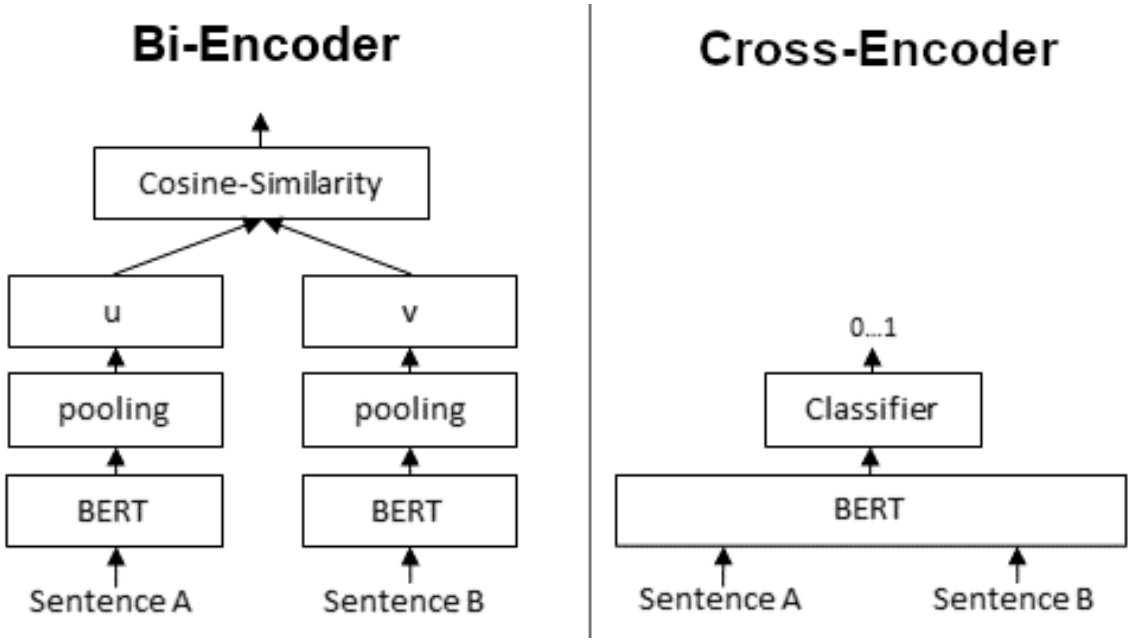


FIGURE 3.2: Bi-Encoder models vs. Cross-Encoder models [42]

documents with a score higher than a certain threshold move forward to the next reranking step, which could be a BERT-based cross-encoder reranking model. They conclude that using smaller LLMs (Mixtral 8x7B instruct with 4-bit quantization) in this approach can have comparable performances with much larger models, although it requires expert input (for few-shot examples).

Déjean et al. [14] performed a comprehensive comparison between cross-encoders and LLM rerankers on the MS MACRO dataset, with the reranker models being deBERTa-v3 large and ELECTRA-large, around 300M parameters, and the LLM used are GPT-3.5 and GPT-4. They discovered that while GPT-3.5-turbo and GPT-4 are very effective in reranking passages, CE rerankers remain competitive against these LLMs, with the benefit of being faster and more efficient. However, they also noted that CE rerankers could perform differently on in- and out-of-domain datasets, while LLMs do not exhibit that problem.

3.6 Interleaving Retrieval with CoT Reasoning

Instead of post-retrieval generation, [47] took a more unconventional route with **Interleaving Retrieval with CoT Reasoning (IRCoT)**: First, K documents are retrieved based on the user question, and then two steps are repeated until termination: (1) Reasoning step: Generate next rationale based on the question, retrieved documents (so far), and generated rationales. (2) Retrieve step: Based on the generated rationale in step (1), K more documents are retrieved. The process terminates when a desired number of steps is reached, or the generated rationale contains "answer is". [47] made a few conclusions with IRCoT: Better than one-step retrieval, effective in an Out-of-distribution (OOD) setting, and generates CoT with fewer factual errors.

Chapter 4

Dataset

For this research, we curated our dataset using publicly available information on the EU Taxonomy and different companies' sustainability reports or equivalent. This chapter will further describe the creation dataset, beginning with a deeper dive into the EU Taxonomy below.

4.1 The EU Taxonomy

The EU taxonomy defines different economic activities and how they can contribute to a more sustainable future [38]. These economic activities are divided into 16 sectors: "Water supply, sewerage, waste management and remediation", or "Arts, entertainment and recreation". For an activity to be "aligned", it must satisfy three different conditions [38]: **(1) Substantially contributed** to one of the six environmental objectives (listed below), **(2) Do No Significant Harm (DNSH)** to other five objectives, and **(3) Comply with minimum safeguards** and **technical screening criteria**[1] in the EU Taxonomy regulation. The criteria for substantial contribution and DNSH differ on an economic activity basis. Subsequently, the six climate environmental objectives are:

1. Climate change mitigation (mitigation)
2. Climate change adaptation (adaption)
3. Sustainable protection of water and marine resources (water)
4. Transition to a circular economy (circular economy)
5. Pollution prevention and control (pollution)
6. Protection and restoration of biodiversity and ecosystems (biodiversity)

The EU's Corporate Social Reporting Directive (CSRD) [39] set out criteria and timeline for mandatory EU taxonomy reporting, for different type of companies. The timeline is visualized in figure 4.1. For instance, economic activity 5.1 "Construction, extension, and operation of water collection, treatment, and supply systems". This activity can substantially contributes to the "mitigation" goal or the "adaptation" goal, but to simplify, we will consider only the mitigation goal. The substantial contribution criteria are **one of the following:**

[1] In this research, we will only consider the first two conditions.

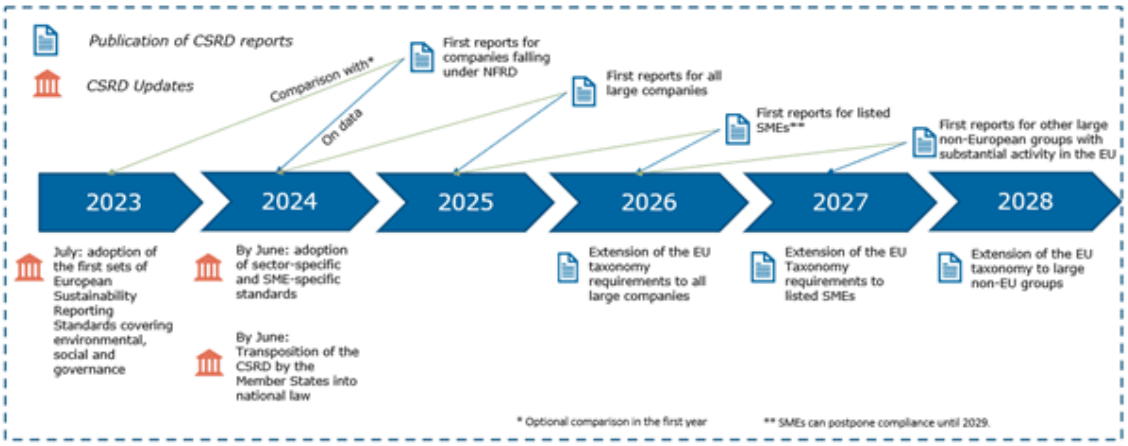


FIGURE 4.1: EU taxonomy mandatory reporting timeline. **2024**: All companies falling under the EU's Non-financial Reporting Directive (NFRD); **2025**: All large EU companies, and all large [2], listed non-EU companies ; **2026-2029**: All EU and non-EU listed small and medium enterprises (SMEs), except micro-enterprises; **2028**: Large non-EU companies with turnover $\geq$ 150 million €, with large EU-based subsidiary. [18]

1. Net average energy consumption for abstraction and treatment equals to or is lower than 0.5 kWh per cubic meter produced water supply.

2. The leakage level is calculated using the Infrastructure Leakage Index (ILI) (231). That calculation is to be applied across the extent of water supply (distribution) network where the works are carried out, i.e. at water supply zone level, district metered area(s) (DMAs) or pressure managed area(s) (PMAs).

Meanwhile, EU taxonomy does not define a specific DNSH criteria for any other climate goals. Consider a hypothetical large company A, with headquarter in the EU. A is doing business within the scope of economic activity 5.1 as described above, and therefore have to report on whether they are substantially contributing to one of the climate goals or not. In their annual sustainability report, A discloses that they have 50 water treatment plants across Europe, and currently 30 of them have a net average energy consumption lower than 0.5 kWh per m^3 water produced. In this project, we would like to automatically retrieve this information and have the LLM to validate if the company is reporting based on the EU taxonomy criteria or not.

4.2 Building the dataset

We identified 15 companies from different sectors with our colleagues at ING. Companies marked with an asterisk (*) have already reported on their EU taxonomy performance according to EU reporting standards. The miscellaneous category represents a company that operates in more than one sector. Furthermore, as these companies have already released their sustainability report for the financial year 2023 (FY2023), we will use data from FY2023. Detailed links to each company's report are presented in appendix A. Afterwards, for each company's report, we further identify possible economic activities for

[2]Large companies are companies that satisfy 2 out of 3 following criteria: (1) 250+ employees, (2) Total balance sheet $\geq$ 20 million € , or (3) Turnover $\geq$ 40 million € [9]

Sector	Company	Activities
Waste Management	Suez*	5.1, 5.3, 5.5-5.10, 4.8, 4.25
	Renewi	5.3, 5.5, 5.8-5.10, 4.3, 4.23
	Biffa	5.5, 5.9, 5.10, 4.8
Marine Transport	Euronav	6.10, 6.12
	Golden Ocean	6.10
	Maersk	6.2, 6.6, 6.10, 6.12, 6.16, 6.19, 7.6
Metal	AcrelorMittal*	1.3, 3.5, 3.9, 4.1, 4.3, 5.9, 7.1
	ThyssenKrupp*	3.1, 3.2, 3.6, 3.9, 5.9, 6.6
Energy	Trafigura	3.8, 3.10, 4.1, 4.3
	Iberdrola*	3.10, 4.1, 4.3, 4.5, 4.9, 4.10, 7.3-7.6
Automotive	BMW*	3.3, 6.5
Miscellaneous	Hitachi	3.9, 3.19, 3.20, 5.1, 5.3, 8.1
	General Electric	3.1, 7.6, 4.26, 4.28
	Vopak*	1.4, 4.1, 4.12, 4.16, 5.3, 5.4, 7.2-7.6, 8.2
	Norsk Hydro*	3.2, 3.8, 3.10, 4.5

TABLE 4.1: Eligible activities identified per company

which the company is eligible. To narrow the scope, we only look for activities contributing to a company's turnover.

- For companies already reported by EU standard, the task is simple: From the turnover table according to EU reporting standard, we can already identify eligible economic activities. For each activity, we then identify chunks of information scattered in the company reports related to the activity description and its substantial contribution criteria.

- For companies that do not report by EU standards, we identified eligible activities based on (1) their competitors' activity, (2) The turnover table in their financial report, and (3) skimming through the sustainability report and manually identifying. Afterwards, the process is similar: identify information related to the activity description and its substantial contribution criteria for each eligible activity.

Finally, based on the available chunks of information, we analyze whether a company has aligned with an economic activity or not. We refer to appendix A for an example.

Disclaimer

As the dataset is curated for experimental purposes only, it has not been verified by a seasoned Sustainability Analyst. We also consider only textual and tabular data from the reports; any graph or figure is lost. Table 4.1 shows the eligible activities identified per each company.

Chapter 5

RQ 1 - Prompt Engineering Methodology, Experimental Setup, and Evaluation

5.1 Chapter Structure

In this chapter, we will describe our complete experimental setup in prompt engineering, including approaches to different prompting techniques - Zero-shot no-CoT prompting, Zero-shot CoT, and 5-shot CoT (for details on the content of the prompts, we refer to appendix B). For details on the content of the prompts, we refer to appendix B. Furthermore, we will describe our approach to manual evaluation strategies and compare them against automated evaluation methodologies. The chapters are structured as follows:

- Section 5.2 describes the experimental setup, including how different prompts are structured, what kind of information and examples are included in the prompt, and how we instruct the LLM to generate the desired output.
- Section 5.3 describes our approach to the manual evaluation of LLM outputs.
- Section 5.4 describes how we calculate different automated metrics and their correlation to human evaluation metrics.

5.2 Experimental Setup

5.2.1 Prompt Design

In order to evaluate different prompt engineering methodologies, all relevant information must be provided to the LLM. This section describes three main aspects considered when crafting different prompts to ask LLM about evaluating a company's EU Taxonomy performance.

- What is the background information (related to the company, EU Taxonomy, reporting, etc.) that the LLM need to know to answer the question?
- What are the thought processes (Chain-of-thought) that humans use to answer this question?
- How can the output be structured in order for a streamlined evaluation process?

Background information provided in the prompt

We provide the LLM with its role as a Sustainability Analyst at a large European bank, followed by a brief introduction to the EU Taxonomy and the scope of consideration. For experimental purposes and time limitations, the scope includes only the turnover/revenue of a company and the Substantial Contribution Criteria of that activity towards the Climate Change Mitigation (CCM) goal. Furthermore, the prompt also includes the current Financial Year in which the report is carried out. This information is important for LLM to (1) identify the correct piece of information within the report, since most companies also report on their future target and ambitions, and (2) Determine which criteria apply since some sub-criteria only apply from a period in the future. Finally, the format of the data (HTML) and type of data (text and table) is also given.

Chain-of-Thought for evaluating EU Taxonomy alignment

We define the following as the Chain-of-Thought the LLM should follow:

1. **Break down the criteria into a set of sub-criteria**: In this stage, the goal for LLM is to thoroughly demonstrate its understanding of the criteria and its complex nature. For instance, if all the sub-criteria need to be satisfied or only one (or a few) of them, any metrics/regulations need to be considered. This step also verifies (and ensures) whether or not LLM uses internal memory to understand the criteria.

2. **Identify relevant content from company report**: This step aims to identify the relevant information from the source regarding each sub-criteria. As the source contains different information, some unrelated to the economic activity in question, this step is crucial in ensuring that LLM selects the right information to derive a conclusion.

3. **Evaluate if the company satisfy the substantial contribution criteria or not**: In this final stage, the final answer is given by the LLM.

Furthermore, we ask the LLM to answer in a JSON format, with four fields corresponding to each thinking step (step 3 consists of two fields, one for reasoning and one for conclusion). We also employed a JSON format validator, and if the answer is incorrect, the LLM is asked to generate again, emphasizing the fact that there was a JSON format error. The JSON answer template is:

```
{
    "Step 1: Break down the criteria": <STEP 1 REASONING as string>,
    "Step 2: Break down what the company report on each sub-criteria":
        <STEP 2 REASONING as string>,
    "Step 3.1: Conclude and explain if company satisfy
    the substantial contribution criteria": <STEP 3.2 REASONING as string>
    "Step 3.2: Conclusion based on step 3.1": True/False
}
```

5.2.2 Traditional prompting and Zero-shot CoT prompting

Traditional prompting is considered a baseline for comparing Few-shot CoT and Zero-shot CoT prompting. In traditional prompting, we only provide the LLM with the instruction, "Perform step-by-step analysis of COMPANY on this ECONOMIC ACTIVITY, based on

the following report snippet. Do not just answer yes or no; provide detailed information on how you come up with the conclusion". The JSON template for traditional prompting consists of two steps:

```
{
    "Analysis: Conclude and explain if company satisfy
    the substantial contribution criteria": <STEP 3.2 REASONING as string>
    "Conclusion: Conclusion based on step 3.1": True/False
}
```

To simplify the terminologies, from now on, we will also call the `Analysis` step of Traditional prompting `Step 3.1`, and the `Conclusion` step as `Step 3.2`. For Zero-shot CoT prompting, we provided the LLM with the CoT thinking step as described in section 5.2.1; however, no examples are given.

5.2.3 Few-shot CoT prompting

We identified ten rows from the original dataset as examples for Few-shot CoT prompting. The examples are crafted with two constraints:

- **Contain companies with different reporting standards**: EU companies that report partly compliant, EU companies that report no compliance since it is not their deadline yet, and non-EU companies that do not report based on EU taxonomy.
- **Contain economic activities from different sectors**, with both single- and multiple substantial contribution criteria.

Furthermore, the answers are written concisely, and bullet points are used whenever possible. The sources given as examples are also shortened to reduce the token length of the prompt. Finally, five examples are picked randomly from the ten examples used in the Few-shot CoT prompting to ensure a diverse reasoning path.

5.2.4 Overall LLM consistency

For each prompt, we ask the LLM to generate five times separately, which allows for diverse reasoning paths. We then prompt the LLM again to combine all the answers together into one final answer, as this is commonly done in practice to group different reasoning paths

5.3 Manual Evaluation of Prompt Engineering

Since the evaluation target is open-ended text generation, there are no fully automated analysis methods - even with ground truth datasets available, except for methods that utilize LLM to evaluate. However, in this paper, we will not consider the LLM-based evaluation method due to the non-deterministic nature of LLM and its sensitivity to prompts. Instead, we will employ human (qualitative) and automated evaluation based on static metrics such as BLEU and BERT-score. We will investigate the correlation between human evaluation and automated metrics, as well as between consistency and automated metrics. Furthermore, as Chain-of-Thought is the main prompting strategy, evaluation should not focus solely on the final conclusion—LLM should also provide correct reasoning steps, and the conclusion must be derived from these reasoning steps.

5.3.1 Terminology

To avoid further confusion, this section presents different terminologies mentioned later on:

1. *Row*: One of the 86 original rows in the ground truth dataset represents an economic activity of a company.
2. *Generation*: For each prompting technique (Zero-shot CoT, Few-shot CoT, and traditional), LLM is asked to generate five times. A *generation* is one of these five LLM-generated texts.
3. HE_n: The Human-Evaluated score for step n.

5.3.2 Criteria for answer correctness

For each step of the CoT prompting and the final answer of the traditional prompting experiment, the generated text is evaluated based on these criteria:

1. Is there any hallucination? Hallucination is any output that appears coherent and plausible but is completely incorrect and unfaithful [23]. All hallucination, although in small quantities, is treated as harmful.
2. Can the LLM understand the complex criteria relationship in the EU Taxonomy SCC?
3. Does the LLM utilize its internal memory as part of the answer? Contrary to hallucination, internal memory information is any incorrect information not present in the prompt.
4. Does the LLM utilize the right section of sources to derive the answer?
5. Is the LLM-generated answer factually correct, or does it contain any logic flaws?
6. How consistent is the LLM-generated text? i.e., if we ask LLM to generate five times with the same prompt, can we expect a similar answer?
7. Does LLM follow the prompt as instructed, or does it take any shortcut (ignore steps or does not perform it adequately)?
8. (Only for CoT prompting) Does the next reasoning step utilise information from the previous step? In other words, how is the Chain-of-Thought connected?

5.3.3 RQ 1.1 - Evaluate CoT prompting

Evaluating Stage 1: Breaking down the criteria

In stage 1, the LLM was asked to break down the criteria and demonstrate its understanding of the EU Taxonomy SCC's complex relationship. Correctness criteria (section 5.3.2) 1, 2, 3, and 6 are considered. We define a three-point scale for step 1 (HE_1), shown in table 5.1. In this scale, we define "entities" as any regulations (example: Article 29(2-5) of Directive (EU) 2018/2001), standards (ISO 14067:2018), values (50-100 MW), date (01/01/2026), or other economic activities reference (4.3) mentioned in the criteria. While the prompt does not provide the regulations, standards, and other economic activities, they can be implemented in future work.

Evaluating Stage 2: Identify relevant information from company report

For the second stage, we will assess whether useful and relevant information has been derived from the company report and if there is any hallucination/ internal memory usage. The relevant correctness criteria (section 5.3.2) are 1, 3, 4, and 6. The score range HE_2, among examples for each score, is given in table 5.2.

Evaluating stage 3: Conclusion

The answer will be directly compared to the ground truth for the last stage. The relevant correctness criteria (section 5.3.2) are 1, 3, 4, 5, and 6. We also define a score range from -1 to 2 HE_3, visualized with examples in table 5.3.

5.3.4 RQ 1.1 - Evaluate traditional prompting

For traditional prompting, since there are no stages, all the correctness criteria (section 5.3.2) are evaluated at the same time, using a similar scoring system mentioned in 5.3.

5.3.5 RQ 1.2 - Evaluating CoT Correctness

Inspired by Bao et al.'s structural causal models (SCM) [2], we designed a scoring system to evaluate CoT prompting. As the LLM is required to "think" in three steps: break down the criteria, identify relevant information from the company report snippet, and give a conclusion, we rate a score of 0 whenever information in the next step is independent and unrelated to the previous step. Formally:

$$\forall (y,x) \in [(2,1),(3,1),(3,2)], C_{yx} = \begin{cases} 0 & \text{if } y \perp\!\!\!\perp f(x) \\ 1 & \text{otherwise} \end{cases} \tag{5.1}$$

C_{yx} represents the chain score between step y and x. With this scoring system, there are seven possible combinations of SCMs, shown in figure 5.1. Moreover, table 5.4 will explain and give examples of cases where a score of 0 is given.

Score	Explanation	Examples	Reason
-1	There are hallucinated entities.	The substantial contribution criteria for this activity is: Manufacture iron and steel where GHG emissions do not exceed specific thresholds: Hot metal = 1,331 tCO2e/t product, Sintered ore = 0.163 tCO2e/t product, ... **including the amount of emissions assigned to the production of waste gases in accordance with point 10.1.5(a) of Annex VII to Regulation (EU) 2019/331.**	Imagined a criteria on GHG emissions (in bold).
0	The criteria could not be understood correctly, either due to missing entities or shortcut.	To qualify as substantially contributing to climate change mitigation, the activity of iron and steel manufacturing must meet specific GHG emission thresholds for different process steps.	Does not mention all entities in the criteria.
1	The criteria is correctly understood, but less than two entities are missing	The substantial contribution criteria for this activity is the manufacturing process of hot metal and sintered ore where GHG emissions do not exceed specific thresholds for different manufacturing processes. ...	Does not mention specific threshold.
2	All entities are correctly mentioned, and the criteria is correctly understood.	The substantial contribution criteria for this activity is the manufacturing process of iron and steel where GHG emissions do not exceed specific thresholds: Hot metal = 1,331 tCO2e/t product, Sintered ore = 0.163 tCO2e/t product, ...	Similar to the criteria

TABLE 5.1: Evaluation Score for Stage 1 HE_1 - Breaking down the criteria

Original Sources: For activities 4.3 Electricity Generation from solar photovoltaic (PV) technology, ArcelorMittal's renewable energy project in India, which includes solar PV farms will automatically fulfil the substantial contribution criteria to climate change mitigation. AcrelorMittal is also investing in covering our plant's rooftop in solar panels to feed the production.			
Score	Explanation	Examples	Notes
-1	Hallucination	ArcelorMittal reports that the renewable energy project in India is expected to generate 2400 GWh of electricity annually, which is equivalent to 1.5% of India's total electricity consumption in 2019. This is a substantial contribution to climate change mitigation as it will reduce greenhouse gas emissions by 2.1 million tonnes of CO2 equivalent annually. The project is expected to be operational by 2022.	Hallucination
0	No useful information is retrieved / LLM utilizes internal memory	AcrelorMittal reports that they have 27,000 solar panels at the roof of their plant in Belgium	Utilizes internal memory (correct facts, but not mentioned in the report)
1	Not all useful information is retrieved	AcrelorMittal reports that they are installing solar panels on the rooftop of their plants.	
2	All useful information is retrieved	ArcelorMittal reports that they are investing in a Solar PV project in India, which will automatically fulfills the SCC. Moreover, they are also installing solar panels on the rooftop of their plants.	

TABLE 5.2: Evaluation Score for Stage 2 HE_2 - Identify relevant information from company report

Ground truth answer			
Renewi satisfied the criteria, since their recycling rate is 63.6% in FY23, which is higher than the criteria of "converting 50% of the processed separately collected non-hazardous waste into secondary raw materials that are suitable for the substitution of virgin materials in production processes".			
Sources			
In FY23, our recycling rate has increased to 63.6%. We've also aligned our recycling labelling for solid waste with international standards, whether these are EU or country-specific benchmarks. We are further developing our Mission75 programme, which aims to raise our recycling rate to 75%			
Score	Explanation	Examples	Notes
-1	Hallucination	Renewi reports their recycling rates in 2023 at 63.6%. Furthermore, their recycling plant is equipped with the latest technology to ensure that the waste is processed in an environmentally friendly manner. Therefore, they satisfied the criteria.	
0	The conclusion generated is factually incorrect, contains major logic flaws, and/or LLM utilizes incorrect information from the sources to derive answer.	Yes, Renewi satisfies the criteria since they aim to raise their recycling rate to 75%.	The prompt specifically asked to "Evaluating based on 2023's data".
1	The answer is factually correct and exactly as expected.	Yes, Renewi satisfies the criteria since their recycling rate in FY2023 is 63.6%, above the 50% threshold.	
2	The answer is factually correct, pointing out facts that were not mentioned in the ground truth.	While the company reports a 63.6% recycling rate and the production of secondary materials, it is not explicitly stated whether this rate corresponds to the conversion of separately collected non-hazardous waste into secondary raw materials suitable for substituting virgin materials.	We initially assume that recycling rate is similar to converting raw waste into secondary materials, but it was incorrect. In this case, LLM pointed out the error in the ground truth.

TABLE 5.3: Evaluation Score for Stage 3 (HE_3) - Conclusion

Sources Economic activity CCM 6.5, "Transport by motorbikes, passenger cars and light commercial vehicles: The Taxonomy-aligned shares for the three performance indicators are at a low single-digit level for the Financial Services segment. A further reason is the varied, stricter DNSH requirements for economic activity CCM 6.5, in particular those relating to Environmental Objective V "Pollution prevention and control", which lead to the exclusion of almost all PHEV and a significant restriction in the recognition of BEV (for details see section: Do no significant harm). Third-party brands are not included in the vehicle portfolio in the reporting on Taxonomy alignment for economic activity CCM 6.5. A lack of available data regarding the tyre categories or WLTP emissions values of third-party products makes it impossible to review compliance with the DNSH criteria in full.
GT of Step 1 - Criteria breakdown The SCC economic activity 6.5 for climate change mitigation are: (1) Category M1 and N1 (cars and vans): Until December 31, 2025: Specific CO2 emissions must be below 50g CO2/km (low-emission vehicles). From January 1, 2026: Specific CO2 emissions must be zero (zero-emission vehicles). (2) Category L (motorcycles and quadricycles): Tailpipe CO2 emissions must be 0g CO2e/km.
GT of step 2: Identify relevant section BMW reports that the taxonomy-aligned shares for the 3 performance indicators (CapEx, OpEx, turnover) are at a low single-digit level.
Examples of chain score (2,1) = 0 BMW reports that due to the strict DNSH requirements, particularly for pollution prevention, almost all PHEVs and of BEVs are excluded. The report also mentions challenges in applying DNSH criteria due to data limitations for third-party brands. **Reason for chain score (2,1) = 0:** Instead of identifying relevant information from the report according to the criteria, this answer reports a different section of the report. Although this section is related to EU taxonomy (DNSH criteria), it is not relevant information to answer the question
GT of step 3 Since BMW reports that the taxonomy-aligned shares for 3 KPIs are low, they are likely not satisfy the SCC.
Examples of chain score (3,1) = 0 and chain score (3,2) = 0 (compare to GT of step 2) While BMW reports activities aligned with the criteria for low- and zero-emission vehicles, the strict DNSH requirements and data limitations for third-party brands raise concerns. The report lacks clarity on the proportion of activities that fully comply with DNSH criteria. The exclusion of most PHEVs and restrictions on BEV recognition due to DNSH further complicate the assessment. Without a comprehensive overview of how these limitations affect the overall activity, it's challenging to definitively confirm if all substantial contribution criteria are met. **Reason for chain score (3,1) = 0**: This answer does not base the conclusion on the SCC mentioned in step 1, but rather on the DNSH criteria. **Reason for chain score (3,2) = 0:** This answer does not base the conclusion on the relevant report section mentioned in GT of step 2.

TABLE 5.4: Evaluation score examples for CoT prompting

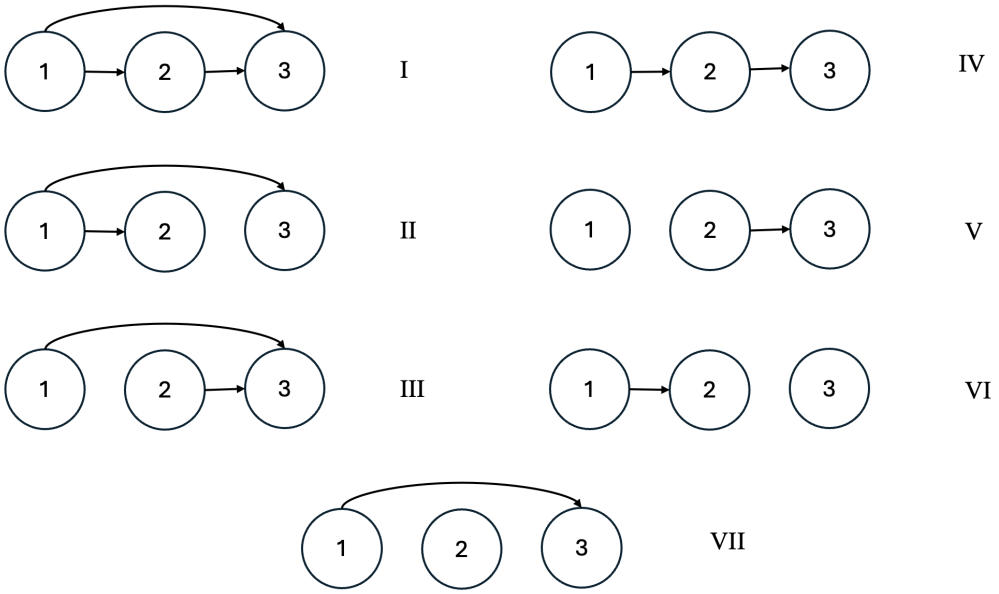


FIGURE 5.1: Seven possible Structural Causal Model types for EU Taxonomy reasoning

5.4 Correlation between automated metrics and human evaluation

5.4.1 Automated metrics

This section describes how automated metrics (BERTScore and BLEU score) are calculated. Please note that by BERTScore in this context, we refer to the F1 Score given by the BERTScorer module [52].

Scoring against the Substantial Contribution Criteria

Specifically for step 1 CoT, since the LLM is asked to break down the criteria into sub-criteria, we calculate BERTScore (BE) and BLEU score (BL) for this step against the original substantial contribution criteria. Formally, for the i^{th} *generation* of the k^{th} row:

$$BE_{SCC}^{i/k} = BERTScorer(LLM_1^{i/k}, SCC) \tag{5.2}$$

$$BL_{SCC}^{i/k} = BLEUscorer(LLM_1^{i/k}, SCC) \tag{5.3}$$

With $LLM_1^{i/k}$ the LLM-generated text for step 1 of *generation* i of row k, and SCC the corresponding Substantial Contribution Criteria.

Consistency scores

To answer RQ 1.3 on correlation between human judgement, consistency, and automated score, we calculate BERTScore (BE) and BLEU score (BL) of each *generation* against other four *generations*, for each step. Formally, for each step n in each *generation*:

$$BE_n^{i/k} = \frac{1}{4} \sum_{j=0, j\neq i}^{4} BERTScorer(LLM_n^{i/k}, LLM_n^{j/k}) \tag{5.4}$$

$$BL_n^{i/k} = BLEUscorer(LLM_n^{i/k}, \{LLM_n^{j/k}\ \forall j \in [0,4] \text{ and } j \neq i\}) \tag{5.5}$$

With $LLM_n^{i/k}$ the i^{th} generation for step n, and $\{LLM_n^{j/k}\ \forall j \in [0,4] \text{ and } j \neq i\}$ the list of other four LLM-generated text for step n of the same *row*. Since traditional prompting only has one step, we calculate the consistency score for this only step. Furthermore, we define a "disagreement" binary score: Within five generations of a row, if there are generations that output `True` for `Step 3.2` and other generations that output `False`, then disagreement is `True`. Formally:

$$DS_k = \begin{cases} 0 & \text{if } LLM_{3.2}^{1/k} \equiv LLM_{3.2}^{2/k} \equiv LLM_{3.2}^{3/k} \equiv LLM_{3.2}^{4/k} \equiv LLM_{3.2}^{5/k} \\ 1 & \text{otherwise} \end{cases} \tag{5.6}$$

With DS_k the disagreement score for the k^{th} row, and $LLM_{3.2}^{j/k}$ the i^{th} generation of row k for `Step 3.2`.

5.4.2 RQ 1.3 - Calculating correlation

Within five generations of a row, a diverse reasoning path may be a double-edged sword: It can lead to more informative answers if all reasoning paths are correct, but on the other hand, a diverse reasoning path could mean that some (or all) answers are based on incorrect information and containing logic holes. As rows with more diverse reasoning paths within its generations (i.e., different answers) tend to have lower automated scores, we will investigate whether lower automated scores correlate to lower human-evaluated scores. For `Step 1` of each *generation*, we calculate these correlations according to equation (5.7) and (5.8). For all steps of each *generation* (only step 3 for traditional prompting), the correlation is shown in equation (5.9) and (5.10). Furthermore, with the DS_k metrics, the correlation is calculated row-wise as per equation (5.11) and (5.12), with BE_3^k and BL_3^k the mean of 5 different generations' BERTScore and BLEU score of a row k. We utilize the **Pearson Correlation Coefficient** (denoted as *corr* below) to calculate the correlation.

$$corr\left(BE_{SCC}^{i/k}, HE_1^{i/k}\right) \tag{5.7}$$

$$corr\left(BL_{SCC}^{i/k}, HE_1^{i/k}\right) \tag{5.8}$$

$$corr\left(BE_n^{i/k}, HE_n^{i/k}\right) \tag{5.9}$$

$$corr\left(BL_n^{i/k}, HE_n^{i/k}\right) \tag{5.10}$$

$$corr\left(BE_3^k, DS_k\right) \tag{5.11}$$

$$corr\left(BL_3^k, DS_k\right) \tag{5.12}$$

Chapter 6

RQ 2 - Retriever Methodology and Evaluation

6.1 Technological setup

In this section, we will briefly describes the base technological setup for Retriever component, as shown in figure 6.1. For more details on theoretical background, please refer to Chapter 3. For the current project, the infrastructure supporting Retrieval-Augmented Generation is based on a Hybrid search of BM25 Keyword Retriever and FAISS Vector Database Retriever (50/50 share). Initial documents (PDF) are preprocessed using Azure's Document Intelligence module, which utilizes OCR technology to semantically parse PDFs. Then, the processed PDFs are chunked, and store in the Keyword Database and Vector Database - using Google's `textembedding-gecko@003` embedding model. For each search query, **5** initial documents are retrieved, and these documents will undergo Cross-Encoder (CE) reranking, score filtering, and finally deduplicating.

6.1.1 Documents Pre-processing

As mentioned above, the document pre-processing step is handled by Azure Document Intelligence (Azure DI). Azure DI's main function is to parse tables into HTML. After the documents are processed, they are first chunked by pages, then each pages are chunked again into chunks. The splitting algorithm is `Langchain`'s `RecursiveCharacterTextSplitter` module [8], which split text using separators such as `\n\n`, `\n`, or whitespace.

6.1.2 Databases

There are two main Retriever databases currently in use: BM25 Keyword Retriever [1] and FAISS Vector Database (Facebook AI Similarity Search) [12]. For BM25 Keyword Retriever, the queries are only preprocessed by whitespace split. For Vector Retriever, the embedding is calculated using Google's `textembedding-gecko@003` model, and the documents are retrieved using similarity search.

6.1.3 Retrieved Documents post-processing

Cross-encoder reranking

The CE-rerank step utilize `sentence_transformers`'s `Cross-Encoder` module [42], with sigmoid activation and `cross-encoder/ms-marco-MiniLM-L-6-v2` (CE model) as the trans-

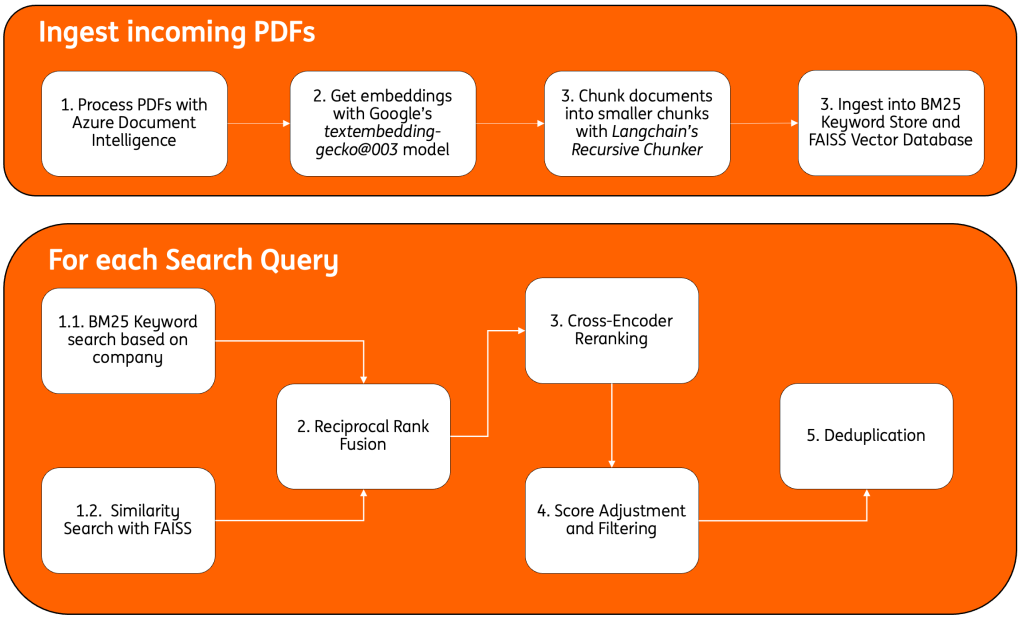


FIGURE 6.1: Current Retriever technology overview

former model to calculate the CE score. The CE model is also trained by Reimers et al. [42], based on Microsoft's Machine Reading Comprehension Dataset (MS-MACRO) [34].

6.1.4 Chunk Size Experiment

Before we begin experimenting with the retriever system, we conducted an experiment to find the optimal chunk size and chunk overlap, which is documented in Appendix C. Follow the results of the experiment decided that a chunk of maximum 2800 characters, with 300 characters overlap, is the most suitable value with the given dataset.

6.2 Experimental Setup

6.2.1 Naming convention for experiments

To simplify the experiment name for different retrievers and query splitting/expansion techniques, we define a set of naming convention:

- `kw`: Keyword retriever.
- `vector`: Vector retriever.
- `hb`: Hybrid retriever.

Furthermore, each query splitting/expansion techniques have their own code, mentioned in the section title of each technique below. These technique's codes can be combined with the retriever's code, for example: `hb_qs` refers to hybrid retriever, using base keyword retriever and separators-based query splitting.

6.2.2 RQ 2.1 - Keyword Retriever Experiments

The experiment is performed with Okapi BM25 Retriever from `langchain`, which in turn used the implementation of `rank_bm25` Python package [4]. We do not preprocess document based on these functions before ingesting into the BM25 retriever, but only preprocess the query which were sent to the BM25 retriever itself. Before each document's ingestion, the document are simply split by whitespace into tokens. Previously, we also experimented with pre-process the document before ingestion, which is documented in appendix **??**.

Stopwords removal (`sw`)

We remove all stopwords in the queries according to NLTK's list of stopwords [3].

Tf-IDF (`tfidf`)

To construct a Tf-IDF dictionary of all documents, we first preprocess the documents by removing all HTML tags, special character, and stopwords (also using NLTK's list of stopwords). Then, we use Scikit-learn's `TfidfVectorizer` [40] to construct the dictionary. When queries are pre-processed using the Tf-IDF dictionary, all words with score lower than 0.2 will be removed.

YAKE (`yake`)

We utilized YAKE (Yet Another Keyword Extractor) [7] to extract 3-gram keywords from the queries. YAKE has two main hyperparameters. The first one is `top` - denotes top-k keywords to return, which will be set to 5. Another hyperparameter is `dedupLim`, the pair-wise similarity filtering threshold based on Sequence Matcher. We keep `deDupLim=0.9` as the default recommendation. YAKE's implementation can be found at [6].

KeyBERT (`bert`)

Similar to YAKE, KeyBERT [17] is utilized to extract keywords. All the hyperparameters are kept as default.

6.2.3 RQ 2.2 - Vector Retriever Experiments

Separators-based query splitting (`qs`)

We split each query based on different end-of-sentence separators, followed by a whitespace or HTML tags (as the EU Taxonomy was formatted so that there were no whitespace if the end-of-sentence separator is followed by HTML tag).

Query splitting with LLM (`qe`)

Inspired by different keyword extraction algorithms (KeyBERT and YAKE), and the fact that queries with activity description and SCC are not self-contained, we ask the LLM to split the original queries into sub-queries. We prompt the LLM that "providing the activity description and SCC as queries to retrieve the document are insufficient, since these queries are complex and not self-contained. Your task is to split them into more self-contained sub-queries. Provide between 5 and no more than 20 sub-queries".

Pseudo-relevance Feedback with LLM (`prfqe`)

Taking inspiration from PRF and CSQE [27], we proposed an experiment to investigate whether LLM can perform pseudo-relevance feedback. For each sub-queries generated as per section 6.2.3, we retrieve a top-1 document, calculate CE score, then ask the LLM to extend the sub-queries further based on that document if the CE score is above 0.75.

Hypothetical answer query expansion with LLM (`qeha`)

This experiment is based on the method proposed by Jagerman et al. [22], where LLM is first asked to generate an answer to the prompt without any source (hypothetical answer). Then, the set of queries is expanded using the hypothetical answer, with the assumption that the hypothetical answer is a better match with the source in the vector database, even if the hypothetical answer is wrong.

6.2.4 RQ 2.3 - Filtering Experiments

Reranking questions

The Cross-Encoder model takes a query and one retrieved chunk, and give a higher score if it rates this chunk as highly relevant to the query. Since we have different query extension methods - especially for vector retriever, the score can be biased, especially with LLM-based query extension methods: LLM can output a query which retrieves a lot of relevant documents with high scores, but irrelevant to the task at hand. Therefore, we will experiment with another setup where all the retrieved chunks are ranked base on one common query: "Does COMPANY substantially contribute to ECONOMIC ACTIVITY?"

Filtering threshold

Apart from the queries, an important hyperparameter to consider is the filtering threshold, which determines the minimum score a document must have in order to be included in the final search results (after reranking). The range of filtering threshold to be experimented are 0.01, 0.025, 0.5, and then from 0.1 to 0.9 with an increment of 0.1.

6.3 Evaluation

6.3.1 Mean Average Precision and Mean Average Recall

We evaluate the retriever systems based on two metrics: Mean Average Precision (mAP) and Mean Average Recall (mAR), both commonly used in IR evaluation. mAP and mAR are described in equation 6.1 and 6.2, respectively, with N is the number of rows, TP_i (true positives) is the number of correctly retrieved documents for row i, FP_i (false positives) is the number of incorrectly retrieved documents for row i, and FN_i (false negatives) is the number of correct but not retrieved documents for row i.

$$\text{mAP} = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i + FP_i} \tag{6.1}$$

$$\text{mAR} = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i + FN_i} \tag{6.2}$$

Since the retrieval task at hand is part of a RAG pipeline, LLM is expected to play a part in filtering unrelated documents. Thus, mAR is a more important metrics than mAP, as higher recall denotes that more correct documents are retrieved. However, mAP also have an important role in ensuring the LLM's efficiency and effectiveness. A high mAP indicates that these relevant documents are ranked highly, enabling the LLM to concentrate its constrained context window on the most crucial information.

6.3.2 One-by-one unique documents retrieved

Furthermore, to dive deeper into how much value each method (query extension/preprocessing) brings, we define a metric called "One-by-one unique documents retrieved" (UDR). This metric compares retriever A to retriever B, calculating how many (correctly) retrieved documents in retriever A's results that are not present in retriever B's results. The UDR score calculation between Retriever A and B (U^i_{AB}) at row i is shown in equation 6.3, R^i_A and R^i_B retriever results for retriever A and B, and GT_i the ground truth retriever results.

$$U_{AB} = \sum_{i=0}^{86} |\forall d \in \{R^i_A \cap GT_i \cap {R'_B}^i\}| \tag{6.3}$$

Chapter 7

RQ 1 - Prompt Engineering Evaluation Results and Discussion

7.1 RQ 1.1 - Manual Evaluation of LLM's answer

7.1.1 Evaluating step 1

Score	Zero-shot	Few-shot
0	16	8
1	20	35
2	394	387

TABLE 7.1: Step 1 score for each CoT prompt type

For step 1 - breaking down the criteria, the result is shown on table 7.1. Overall, Zero-shot and few-shot CoT perform comparably well in this step, with few-shot CoT have a slight edge with lower generations with score 0.

7.1.2 Evaluating step 2

Score	Zero-shot	Few-shot
-1	1	0
0	3	7
1	19	19
2	407	404

TABLE 7.2: Step 2 score for each CoT prompt type

The result for step 2 (table 7.2) confirms the observation in step 1 (table 7.1: That there is not much different between Zero-shot and Few-shot CoT prompting. While Zero-shot prompting has one hallucination incident, it is too small to conclude that Zero-shot prompting perform worse than Few-shot. Another interesting observation is that Few-shot CoT utilizes its parametric memory in three generations of the same row, and this behavior was not observed anywhere else other than CoT prompting.

7.1.3 Evaluating final conclusion

Score	Traditional	Zero-shot	Few-shot
-1	3	0	2
0	45	26	58
1	6	12	8
2	374	382	360
3	2	10	2

TABLE 7.3: Final conclusion for each CoT prompt types and traditional prompting

Logic hole	Traditional	Zero-shot	Few-shot
Misunderstand the criteria	29	22	36
Misunderstand the report	11	9	12
Use wrong information from the report to derive answer	1	2	4
Misunderstand both the criteria and the report	1	0	0
Misunderstand the task (analyze SCC on CCM)	0	5	3
Hallucination	3	0	2
Give correct analysis but wrong conclusion	0	0	2

TABLE 7.4: Counting different logic holes for each prompting strategies in step 3

For the result of step 3 (conclusion), we evaluate the final steps of the zero-shot and few-shot, together with traditional prompting. With the result in table 7.3, the most surprising result is that Few-shot prompting is the worst performer out of all three. In first place is zero-shot prompting, with the highest generation scoring 2 or higher by a considerable margin, followed by traditional prompting. Furthermore, while giving scores for each row, we also noted the reason behind it - for the row that scores one or less, denoted "logic hole", with the result displayed in table 7.4.

The most common logic hole is "misunderstanding the criteria, " where LLM often gets confused when the SCC mentions that only one or two of all criteria must be satisfied. One such example is the activity *Installation, maintenance and repair of charging stations for electric vehicles in buildings (and parking spaces attached to buildings)*, with the SCC being *Installation, maintenance or repair of charging stations for electric vehicles*. In some generations, LLM would mention that the company only reported that they have installed EV charging stations but did not report about maintenance or repair. In another case, the SCC for activity *Manufacture of aluminium* states that before 2025, only two out of three criteria need to be satisfied. LLM correctly mentions this information in step 1 but gets confused in the final step: *The company does not report on criteria 3. Therefore, they have not satisfied the SCC.*

"Misunderstanding the report" is another common logic hole across all three prompting paradigms and mostly happens when the report words things confusingly, even for humans. For instance, activity *Manufacture of energy efficiency equipment for buildings*

has an SCC about *Manufacturing insulating products with a lambda value lower or equal to 0,06 W/mK*. A company reports that they have "*assessed these panels against the insulating products criteria requiring a lambda value lower or equal to 0,06W/mK to identify substantial contribution*" but does not confirm that their product met the criteria.

7.1.4 Discussion

For CoT prompting, table 7.1 and 7.2 show that there is little difference between Few-shot CoT and Zero-shot CoT in the human-annotated score for the first two steps - both achieve very high performance. The most common mistake LLM makes in step 1 is not mentioning the criteria in full. With step 2, LLM occasionally summarizes the company report snippet, while the prompt specifically asks to "Identify sections relevant to EU Taxonomy from the snippet". However, `step 3` is where the performance differs: Table 7.3 shows that although most *generations* of Few-shot prompting still achieve adequate results (84.2%), it has the highest amount of error *generations* - with 68.

Since LLM's performance is known to have a negative correlation with the number of tokens it has to handle, and as Few-shot CoT prompts are considerably longer than both Zero-shot CoT and traditional prompts, we believe the length of Few-shot CoT prompts is part of the culprit. This is also supported by the fact that Few-shot CoT performs on par with Zero-shot CoT in the first two steps and only drops in step 3. Meanwhile, the difference in performance between Zero-shot CoT and traditional prompting is expected, as CoT prompting is known to improve LLM's reasoning capabilities [51], and the success of Zero-shot CoT over traditional prompting shows that our step-by-step method works.

CoT prompting's underperformance can also be attributed to its examples. As mentioned earlier, other CoT papers have not tested their methods on a problem with this scale - mostly arithmetic or basic reasoning problems. Our problems require much deeper analysis, so the examples might have limited the reasoning path that LLMs can take. In other words, LLM with a Few-shot CoT will always try to solve the task in ways that are most similar to the examples provided, but the examples might not provide the right tool.

Finally, we notice that in most ground truth rows, LLMs cannot derive a conclusive comment on whether a company satisfies the substantial contribution criteria, mainly because insufficient information is provided as part of the prompt. For instance, if a criterion requires compliance with a specific regulation, the majority of the time, LLMs will clearly state, "Since information about this regulation is not available, we cannot conclude that company A satisfies the criteria". We will discuss this further in the future work section below.

7.2 RQ 1.2 - Manual Evaluation of CoT Correctness

Chain type	Zero-shot	Few-shot
I	421	415
II	3	4
III	3	8
IV	3	0
V	0	3
VI	0	0
VII	0	0

TABLE 7.5: Results for CoT correctness analysis

The results for CoT correctness analysis are displayed in table 7.5, which shows Zero-shot CoT slightly outperform Few-shot CoT, but only by a small margin. The manual evaluation scores also show that 97.91% of zero-shot generations and 96.51% of few-shot generations fall within chain type I - the ideal scenario, and while Few-shot CoT does not have any Type IV chain, it has some type V chain (where information from step 1 - breaking down the criteria is not used for the analysis in step 2 and 3). Meanwhile, Zero-shot prompting has no type V chain but only type II, III, and IV. For the CoT correctness evaluation (table 7.5), Zero-shot and Few-shot prompting also perform on par with each other, both often missing a link in their chain (type II, III, and IV). While Few-shot CoT makes three type V errors (missing two links), the sample size is also too small (3/430) to conclude.

7.3 RQ 1.3 - Correlation between consistency, human judgements, and automated metrics

Table 7.6 shows the Pearson coefficients as described in equation 5.7 to 5.12. For the correlation between BERT/BLEU against DS (eq. 5.11 and 5.12), the correlation is negative because $DS_k = 1$ if and only if there are disagreements. Overall, the strongest correlations are between DS/BE and DS/BL (eq. 5.11 and 5.12), followed by correlation in equation 5.7, 5.8, 5.9 and 5.10 for $n = 1$. Few-shot CoT prompting also demonstrate a higher correlation between human-annotated scores/consistency metrics against automated metrics.

From table 7.6, it is clear that Few-shot prompting's answers for each step and the conclusion exhibit much higher correlation than traditional prompting, which can be attributed to the fact that CoT prompting is more self-contained in each step: Since in each step, the LLM with CoT are trying to answer the question using a pre-defined method, BERTScore and BLEU for *generations* that are different are more likely to be lower. On the other hand, traditional prompting does not define a common method to answer the question. Thus, the LLM is free to interpret the criteria and company report in any order and in any way; therefore, a lower BERTScore and BLEU might not correlate with the wrong answer.

Within CoT prompting, it is also observable that steps 2 and 3's human-evaluation scores do not correlate well with automated metrics. At the same time, the correlation is the strongest in equation 5.11, 5.12, followed by equation 5.7 and 5.8. This observation implies that BERTScore and BLEU are better at identifying if (1) the LLM understands the report correctly or not and (2) if there is any disagreement between each *generations*.

	Few-shot CoT prompting		Zero-shot CoT		Traditional	
Equation	**Pearson**	**p-value**	**Pearson**	**p-value**	**Pearson**	**p-value**
(5.7)	**0.3904**	0	0.2950	0	-	-
(5.8)	**0.3789**	0	0.3163	0	-	-
(5.9) with $n = 1$	**0.3761**	0	0.2923	0	-	-
(5.10) with $n = 1$	**0.3346**	0	0.1996	0	-	-
(5.9) with $n = 2$	0.1430	0.0030	0.0733	0.1291	-	-
(5.10) with $n = 2$	0.1096	0.0230	0.0986	0.0409	-	-
(5.9) with $n = 3$	0.2673	0	0.1238	0.0102	0.0613	0.2042
(5.10) with $n = 3$	0.1263	0.0087	0.0165	0.7337	0.0424	0.3804
(5.11)	**-0.4527**	0	**-0.4250**	0	-0.2138	0.0481
(5.12)	-0.2909	0.0066	**-0.3655**	0.0005	-0.1486	0.1720

TABLE 7.6: Pearson correlations and p-values results

The Pearson correlation scores hovering around 0.4 (moderate correlations) also suggest that a system where answers with low BERTScore and BLEU are automatically flagged is feasible.

7.3.1 RQ 1.4 - CoT vs. traditional prompting

Evidence from RQ 1.1 and 1.3 suggests that CoT prompting, especially Zero-shot prompting, outperforms traditional prompting in several different ways:

1. **Higher human-evaluation score:** Zero-shot CoT prompting scores higher than traditional prompting, identify more relevant information and make fewer mistakes than traditional prompting.

2. **Generate more diverse reasoning paths:** This is evident also from the fact that Zero-shot CoT scores higher on human evaluation score, and have more **generations** with a score of 3, implying correct results that human annotator was not expecting.

3. **More consistent and interpretable:** Between each *generations*, traditional prompting answers the question differently. However, with CoT prompting, there has been a set of steps to answer the question. Thus, it is easier to evaluate where the problem is (step 1 or 2). Although traditional prompting is more consistent (generates fewer reasoning paths), its way of concluding can be vastly different.

Therefore, Zero-shot prompting significantly increases LLM's performance in answering EU taxonomy-related questions.

Chapter 8

RQ 2 - Retriever Improvement Results and Discussion

8.1 RQ 2.1 - Keyword Retriever results

We attempted different approaches to how the query is formulated and whether the document is pre-processed or not.

8.1.1 Pre-processing both queries and documents

In this approach, the queries and the documents are pre-processed by a keyword extractor (marked with * in the experiment codes). The results (figure 8.1) were significantly worse than the original keyword extractor, with decreased precision and recall.

8.1.2 Only pre-process query, but each keyword as one query

Figure 8.2 shows the results of sending each keyword found by keyword extraction models as a query into the BM25 Keyword Retriever (marked with ** in the experiment codes). As each query retrieves top-5 documents, using this approach results in nearly tripled, while the recall shows a slight increase. KeyBERT and YAKE have improved their performance by 8% and 11%, respectively (on mAR). Interestingly, `kw_tfidf`, although retrieving as many documents as keyBERT or YAKE, saw a drastic drop in precision and recall.

8.1.3 Only pre-process query, and concatenate all found keywords into one query

In this approach, the keywords found by KeyBERT, YAKE, or Tf-IDF for each original query are concatenated, which results in two keyword queries: One for activity description and one for substantial contribution criteria. The results for this approach, displayed in figure 8.3, shows `kw_tfidif` performing significantly better compared to the previous approaches, but `kw_bert` saw a significant drop.

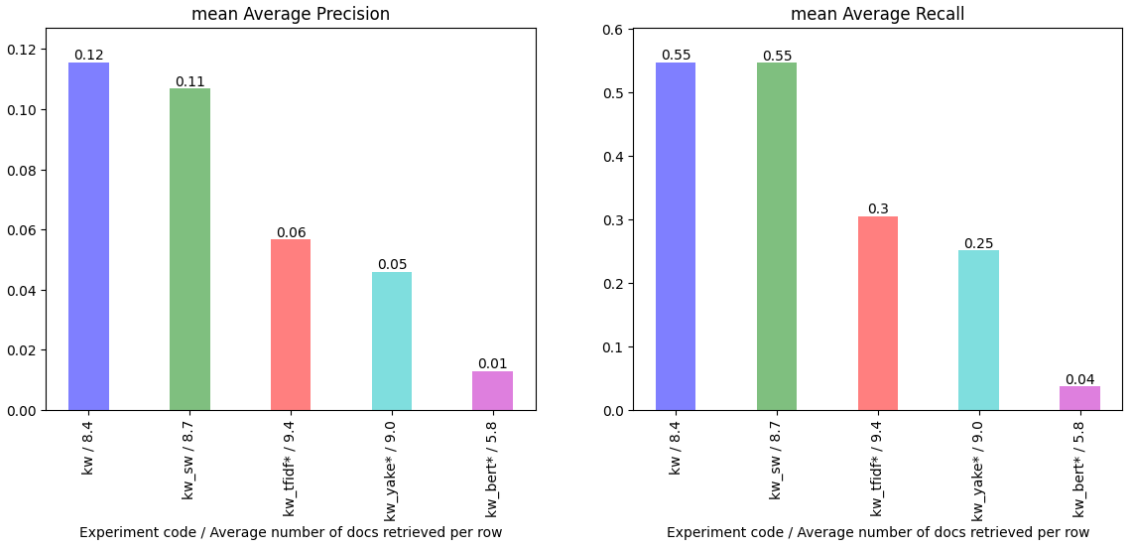


FIGURE 8.1: Keyword Extractor experiments where pre-processing functions are applied to both documents and queries

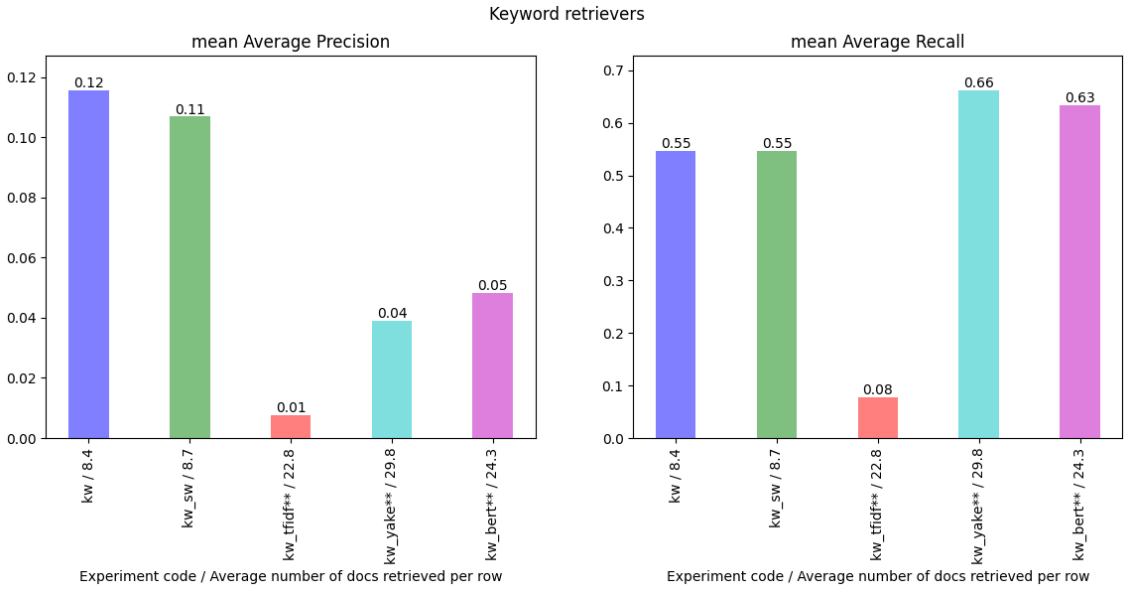


FIGURE 8.2: Keyword Extractor experiments where pre-processing functions only apply to query, and each keyword is a query

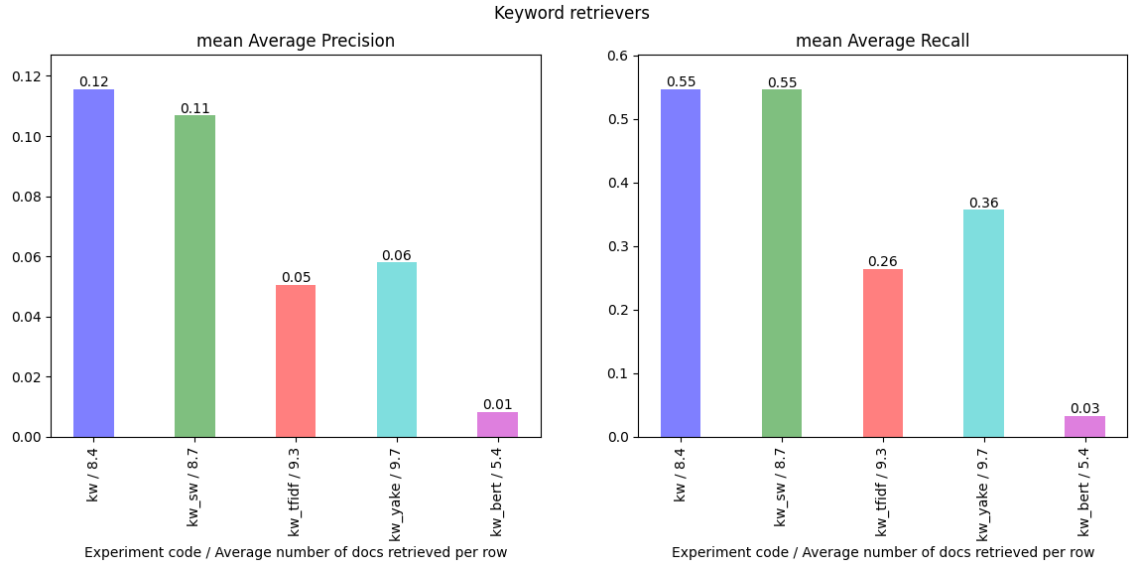


FIGURE 8.3: Keyword Extractor experiments where pre-processing functions only apply to query, and all keywords are appended to a single query

8.1.4 Discussion

In section 8.1, we show the result of keyword retrievers with three different settings: pre-process both queries and documents (figure 8.1), pre-process only query - but each keyword as a query (figure 8.2), and pre-process only query but concatenate all keywords from one query into a new query (8.3).

For the first setting, we discovered that pre-processing both queries and documents significantly affects the performance of the keyword retrievers, with the recall rate dropping as much as ten times. The second setting,(`kw_bert` and `kw_yake`) slightly improves the performance of `kw` by around ten percentage points; however, comes at a cost since, on average, approximately ten more documents were retrieved. Finally, with the third setting, the performance is comparable to the first. Interestingly, only `kw_yake` maintains the performance over all three settings, while `kw_bert` does not perform in the first and third, and `kw_tfidf` performs much worse in the second.

At this moment, we do not have any explanation for how the performance of each keyword extraction technique differs on each setting or how splitting each keyword into a query significantly improves the performance. One assumption is that the increase is purely by chance—more queries mean more retrieved documents. Since this experiment raises more questions than answers, we decided not to utilize keyword extraction techniques in hybrid retrievers going forward.

8.2 RQ 2.2 - Vector Retriever

Figure 8.4 visualizes the difference between mAP and mAR of all vector retriever experiments. It is clear thaor t the LLM-based hypothetical answer query extension method (`vector_qeha`) significantly improves the recall performance of the vector retriever by a wide margin compared to the original retriever, and almost double the performance of the

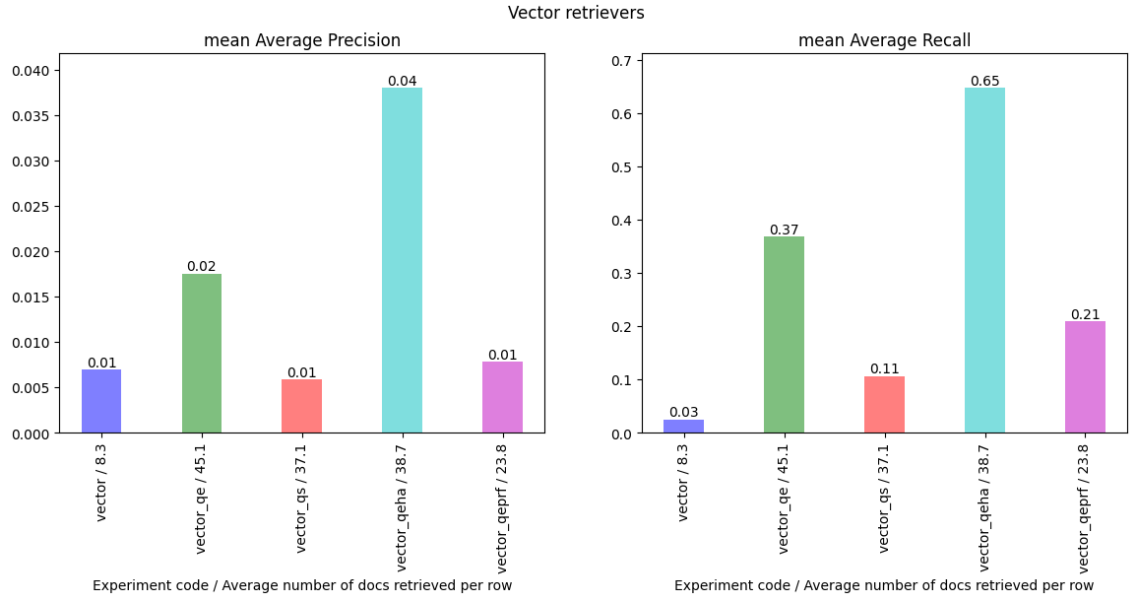


FIGURE 8.4: mAP and mAR comparison for all Vector Retrievers experiment

second-best performing vector retriever - the LLM-based query splitting method. Meanwhile, separator-based query comes in 3rd place closely (0.7 recall) with minimal effort and resource utilisation. Vector retriever with LLM-based pseudo-relevance feedback perform significantly worse, coming fourth at only 23% recall.

However, since the number of retrieved documents also increased 7-8 times (due to many more queries), mAP did not improve. Another interesting observation is that while comparing vector retrievers' UDR score among themselves and versus keyword retrievers (figure 8.6), it is clear that the original vector retriever does not bring any additional value compared to the original keyword retriever—meanwhile, LLM- and separator-based query splitting brought 15 and 3 new documents, respectively.

Before experimenting with the retriever, we hypothesise that since the query, containing an activity's description and its substantial contribution criteria, is not self-contained, vector retrievers would have difficulty finding relevant documents. The experiment with the base vector and keyword retriever confirmed this hypothesis - figure 8.5 shows that base keyword retriever (`kw`) and hybrid retriever (`hb`) have exactly similar performance (on both precision and recall), while figure 8.6 shows that base vector retriever (`vector`) does not retrieve any unique document compared to `kw`.

Thus, query splitting and extension techniques, especially `vector_qe` (LLM-based), improve upon `vector` by a considerable margin - retrieving 50 more unique documents according to figure 8.6. There is also a considerable difference between `vector_qs` (naive separators-based) and `vector_qe`, with the latter also retrieving 41 more unique documents than the former while not retrieving many more documents (6 more per row). The best performing query extension techniques - extending with hypothetical answers (`vector_qeha`), retrieve 95 more unique documents compared to the original `vector`, and 52 compared to `vector_qe`. Furthermore, since the current prompt for query splitting (see appendix B) is still naive (without any examples or iterative process), it is possible

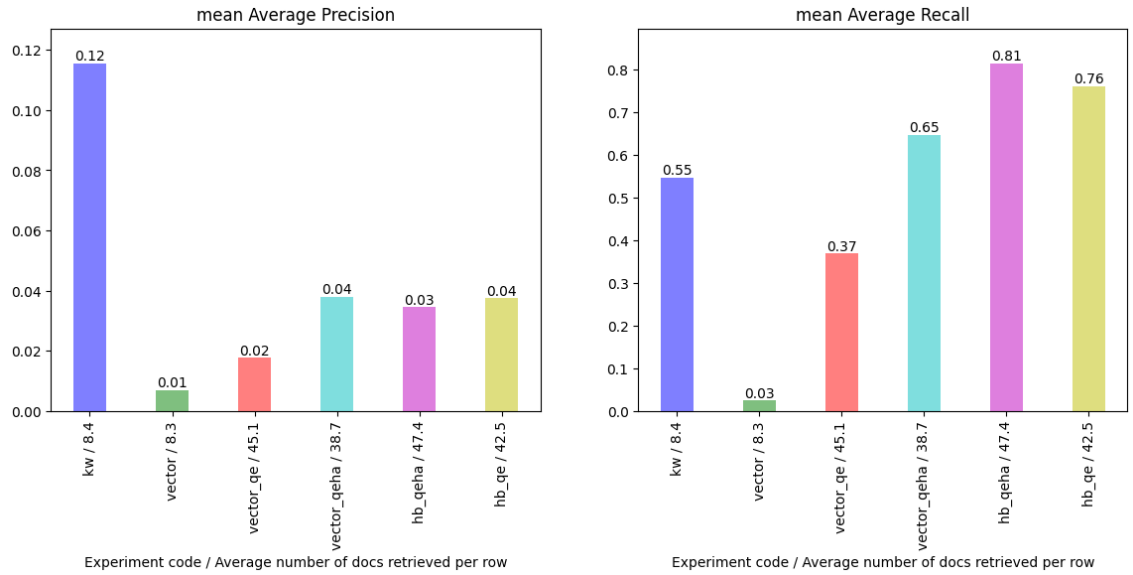


FIGURE 8.5: mAP and mAR comparison for Hybrid and selected Keyword/Vector retrievers

that LLM-based query expansion can still improve the vector retriever's performance much further.

8.3 Hybrid Retriever - combining Keyword and Vector Retriever without reranking

Before continuing with the Cross-Encoder Reranking result, we are interested in how the hybrid retriever improves using the different combinations of vector retrievers (LLM- and separators-based query splitting) and the original keyword retriever. Due to the uncertainty of how keyword retrievers behave, we have decided to only use the original keyword retriever for hybrid retrievers. These hybrid retrievers' mAP-mAR and UDR scores are presented in figure 8.5 and 8.6, respectively. From these figures, we can make three observations:

1. Hybrid retrievers with original keyword and vector retrievers (hb) perform the same as original keyword retrievers (kw), which proves that original vector retrievers do not add any value.

2. Hybrid with LLM- and separators-based query splitting methods improve recall performance by 26% and 15% , respectively, while hybrid with hypothetical answer increases the performance by a further 5% compared to LLM-based query splitting.

3. Hybrid with separator-based query splitting does not bring any new unique documents compared to LLM-based query splitting.

8.4 RQ 2.3 - Cross-Encoder Reranking

To perform experiments regarding the filtering threshold, we chose the two best-performing retrievers: Hybrid with an LLM-based query extension (`hb_qe`) and Hybrid with LLM hypothetical answer query extension (`hb_qeha`). Figure 8.7 and 8.8 displays the mAP and

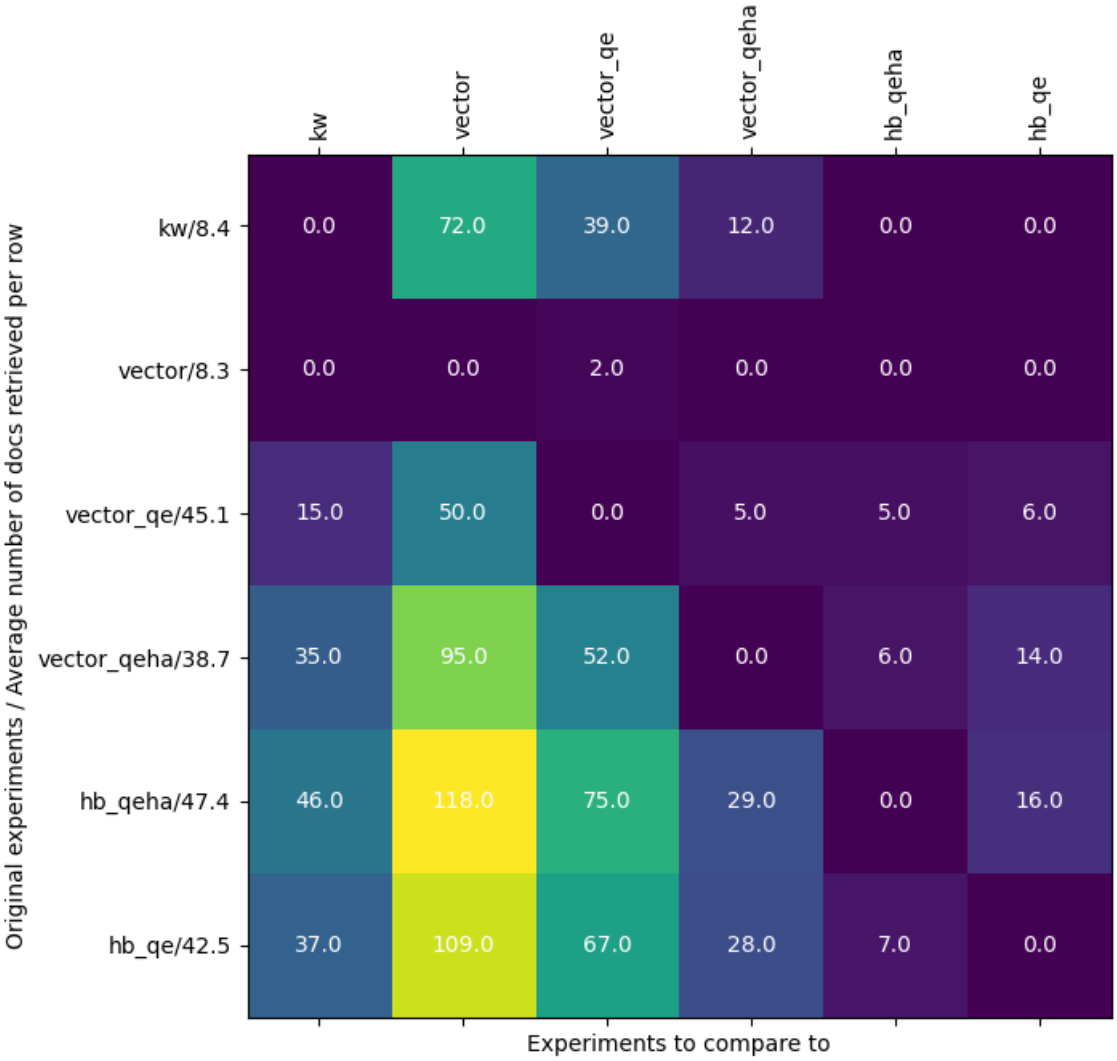


FIGURE 8.6: One-by-One UDR Score matrix - Hybrid and selected Keyword/Vector retrievers

mAR scores for each of the filtering thresholds for `hb_qe` and `hb_qeha`, respectively - among with the average number of documents retrieved per query[1]. In these two figures, the retrieved documents are ranked against the query used to retrieve them, i.e., the query extended by LLM. An additional metric on each column's label: Average number of UDR lost per row versus when no filtering occurs. We calculate this by dividing UDR (of that column against the no filtering experiment) by the number of rows (86). Furthermore, Figure 8.9 and 8.10 displays the mAP and mAR scores for each of the filtering thresholds for `hb_qe` and `hb_qeha`, respectively, where all documents are ranked against one common question - as denoted in section 6.2.4.

The mAP and mAR scores for reranking threshold in figure 8.7 for `hb_qe` are largely within expectation: Higher threshold (t) cause recall and amount of documents retrieved to decrease while increasing precision due to increased relevance. However, we can observe a significant drop from $t = 0$ (no filter) to 0.05 - two-thirds of the retrieved documents are below this score and are thus filtered out, causing a 9% drop in the recall. In other words, we lost 0.2 unique documents per row but filtered out 28 unrelated documents (on average), which is a good tradeoff. When the filtering threshold increases, the change in value becomes less significant. At $t = 0.3$, the value seems to be the most ideal: only a 6% drop in recall from $t = 0.05$ but with only half the documents. After $t = 0.3$, the decrease in retrieved documents becomes insignificant, while the recall drop is more observable.

On the other hand, figure 8.8 shows a different story: Since chunks retrieved by hypothetical answers are ranked very high (against the answer) - with around half of the chunks scores higher than 0.3 and more than one-third of the chunks score higher than 0.9. This high score implies that ranking retrieved chunks against hypothetical answers is biased, and the reranked does not significantly filter out unrelated documents. Thus, figure 8.9 and 8.10 display the results when all retrieved chunks are ranked based on one common question "Does COMPANY substantially contribute to ECONOMIC ACTIVITY?".

With `hb_qe`, the new reranking question does not significantly alter performance at low to medium filtering threshold (0.01 to 0.5). However, at a higher filtering threshold, mAP for new reranking questions still increases steadily (from 0.5 to 0.8) - while for the original ranking method, it fluctuates at around 0.17, and mAR does not decrease as much. Overall, the new reranking question does not significantly affect the `hb_qe` experiment. However, with `hb_qeha`, the results in figure 8.10 are in stark contrast with figure 8.8, and closely follows the trend of figure 8.7: At $t = 0.01$, more than two third of the retrieved chunks are removed; mAR drops at a higher pace, reaching 0.46 at $t = 0.3$ (as opposed to 0.7 in fig. 8.8); at $t = 0.9$, only 1 document remain per row, versus 13.5 in fig. 8.8. We can conclude that, for `hb_qeha`, reranking with one common question is better for mAP, while the original reranking method is better for mAR.

[1] At filtering threshold = 0.0, it is the same as the original experiment.

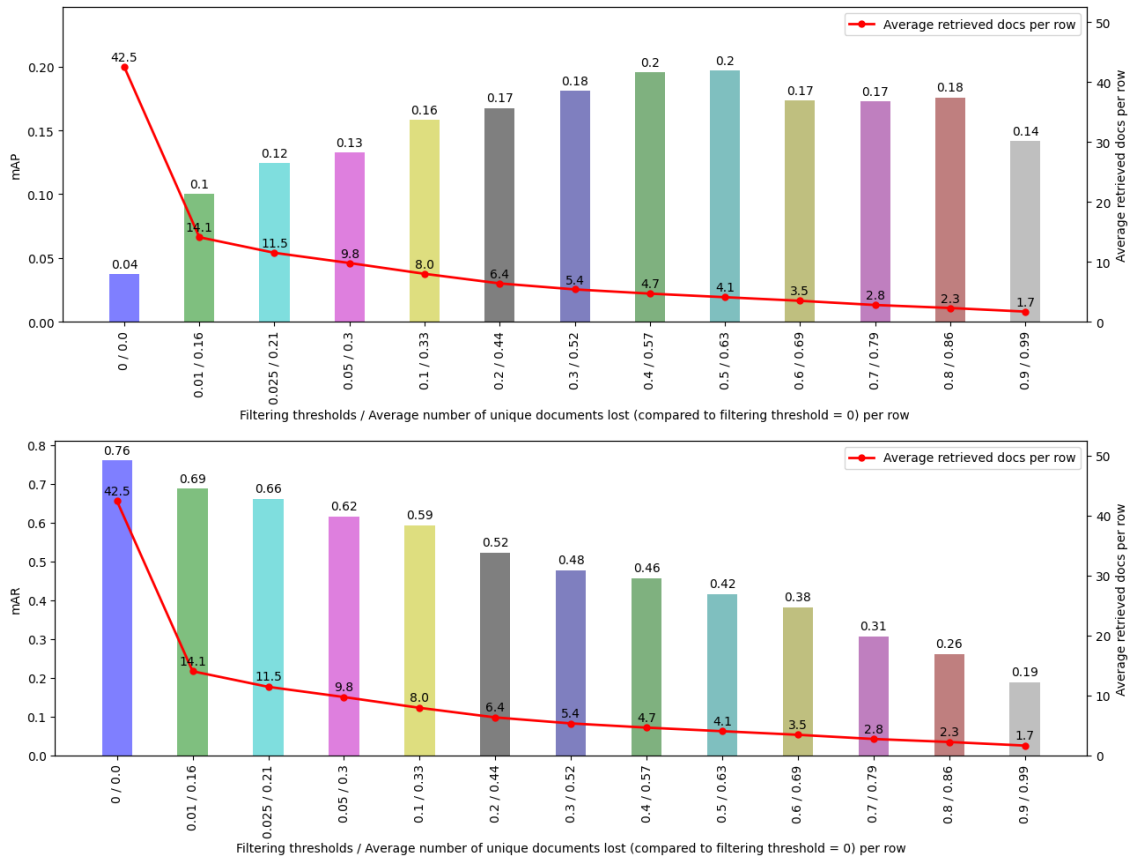


FIGURE 8.7: mAP (top) and mAR (bottom) comparison for different filtering thresholds for `hb_qe`. mAP and mAR values are denoted by the bar plots.

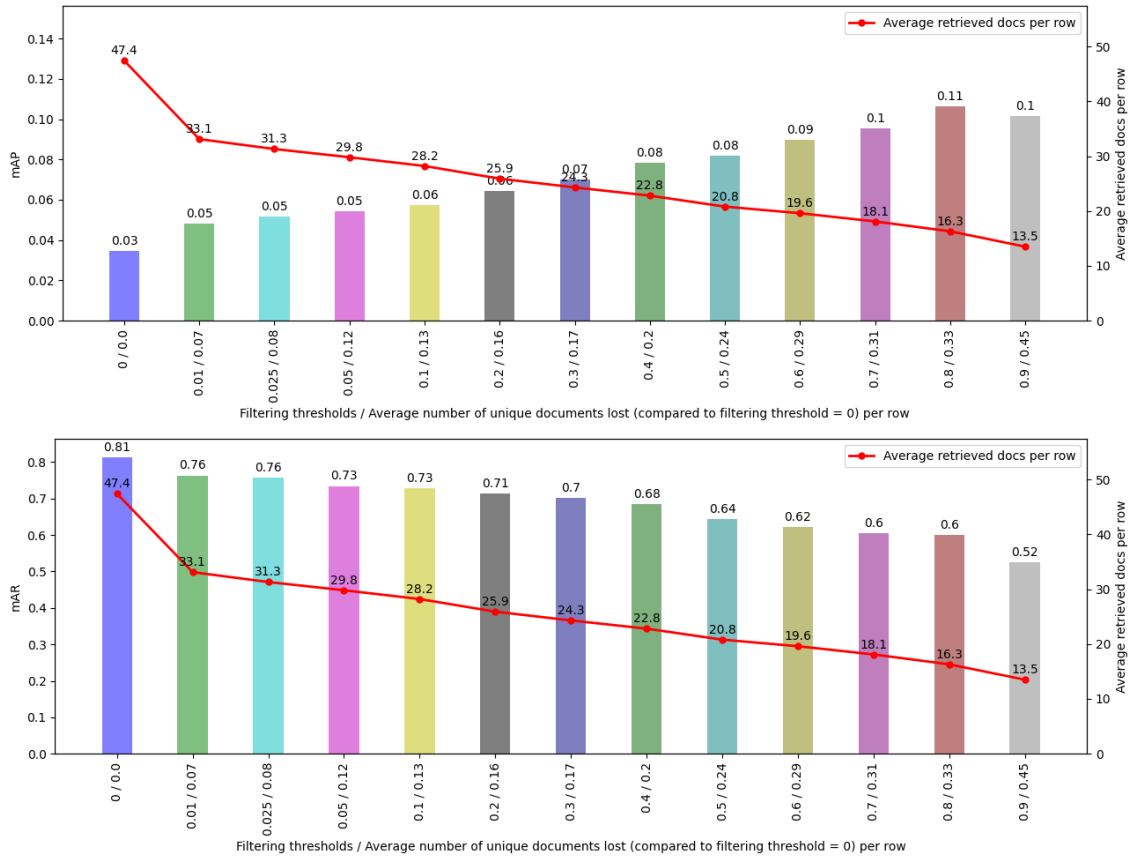


FIGURE 8.8: mAP (top) and mAR (bottom) comparison for different filtering thresholds for `hb_qeha`. mAP and mAR values are denoted by the bar plots.

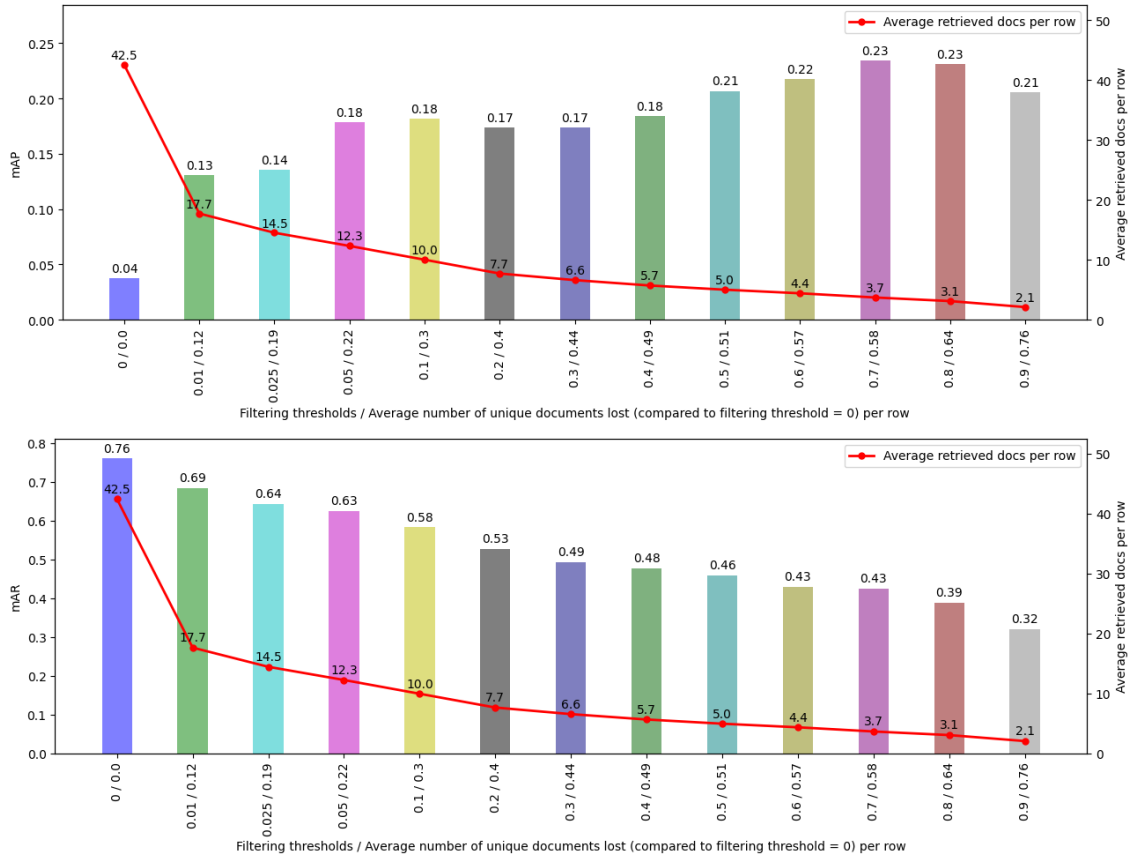


FIGURE 8.9: mAP (top) and mAR (bottom) comparison for different filtering thresholds for `hb_qe`, with one common reranking question. mAP and mAR values are denoted by the bar plots.

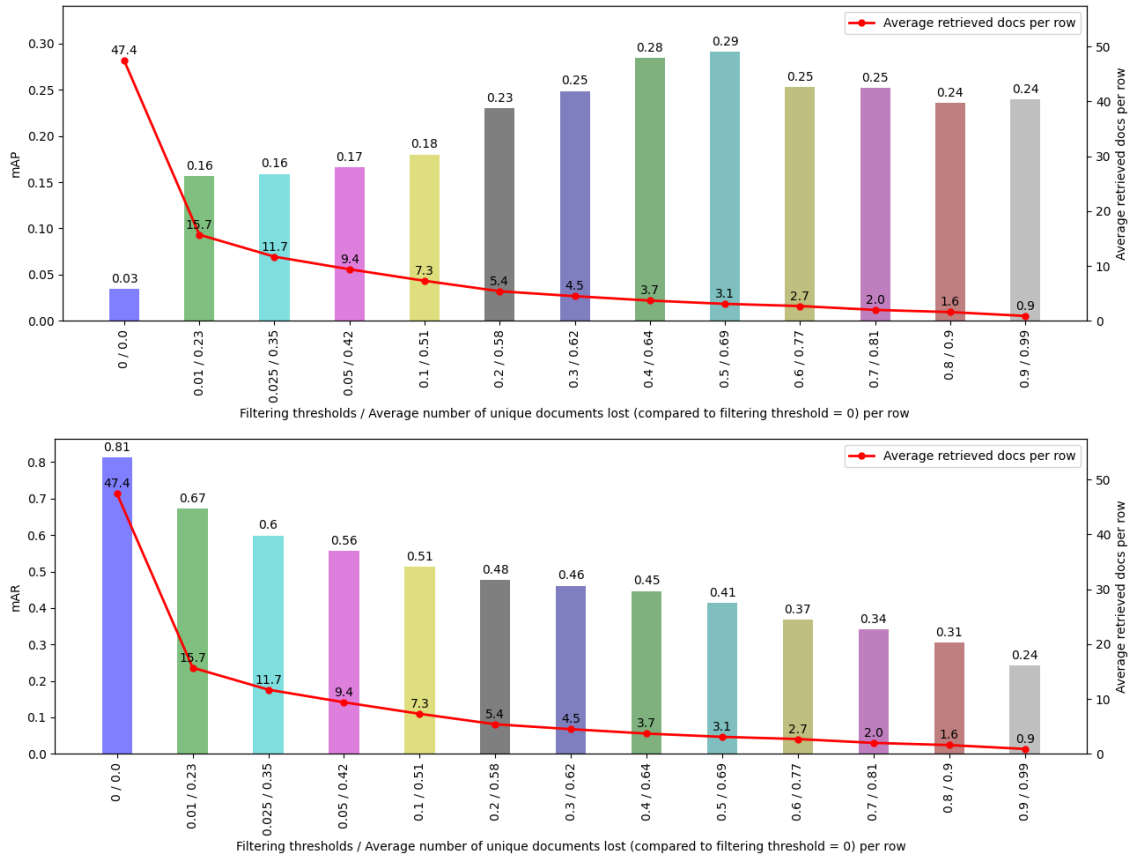


FIGURE 8.10: mAP (top) and mAR (bottom) comparison for different filtering thresholds for `hb_qeha`, with one common reranking question. mAP and mAR values are denoted by the bar plots.

Chapter 9

Conclusion

9.1 Conclusion

We have successfully demonstrated a framework for automatic question-answering on EU Taxonomy performance by Large Language Models using a two-step approach. In the first step, a hybrid retriever (composed of BM25 Keyword Retriever and Vector Retriever with LLM-based query extension) is used to retrieve relevant information on the company report, using the activity description and substantial contribution criteria of that activity as a query. In the second step, we prompted LLM using zero-shot CoT prompting and our three-stage approach: Breaking down the criteria, identifying relevant information from company reports, and concluding whether the company satisfied the criteria.

For the first step, we also compared different Keyword extraction techniques for the BM25 Keyword Retriever and query splitting techniques for the vector retriever. We have also evaluated the role of the filtering threshold in cross-encoder reranking in filtering out non-relevant documents to our query. Our findings can be summarized as:

1. Keyword Retriever, equipped with keyword extraction techniques such as YAKE and KeyBERT, improves the performance of BM25 Keyword Retriever. However, the improvements are not concrete, as demonstrated by how the performance differs when three different settings are applied.

2. Non-self-contained queries can significantly harm vector retriever's performance. We also investigated different query splitting techniques and discovered that Zero-shot, no-CoT LLM-based query splitting can significantly improve vector retriever's performance, especially in combination with LLM-generated hypothetical answers.

3. For LLM-based query extension method, Cross-Encoder reranking using a small language model can reduce a lot of irrelevant documents but also significantly lower the recall rate - and often rate relevant documents very low score. Meanwhile, for query extension method based on LLM-generated hypothetical answers, Cross-Encoder Reranking rate most documents above average score, implying potential bias. This bias is partly mitigated when reranking all documents using a common question (instead of against the search query). However, this new reranking method, while improving precision, also suffer from lower recall rate.

For the second step, we have also developed a rating scale to evaluate each stage of CoT prompting manually, a rating scale to evaluate CoT correctness, and another rating scale to compare all LLM output concerning answering questions on EU Taxonomy. We have

also investigated the correlation between these manual metrics and automated metrics such as BERTScore and BLEU, and results show a moderate correlation between CoT prompting's answer and its automated metric score, thus showing that automated metrics could be used to flag suspicious answers. Finally, we can conclude, based on this evidence, that:

1. Zero-shot CoT prompting significantly improves LLM's output regarding traditional prompting in answering EU Taxonomy questions.

2. While Few-shot CoT can outperform Zero-shot in more specific and narrow problems, we observe that Few-shot CoT lags in terms of human-evaluated score compared to Zero-shot CoT. Therefore, further evaluation is needed to justify Few-shot CoT prompting for an open-ended QA task as complex as this.

9.2 Limitations

The core limitation of our research is the dataset, which is not curated by a seasoned professionals and expert in the EU Taxonomy. Therefore, the ground truth within the dataset might be incomplete or inaccurate. Furthermore, due to limited public data available, we could not test our approach in full - as presented in figure 1.1. Specifically, we could not evaluate if LLM can independently identify EU Taxonomy activities relevant to a company or not, and thus, it is still a manual process. Furthermore, due to the limited time frame of the research, it is not possible to investigate methods to provide more related information to the LLM, such as regulations within a criteria - which could be crucial to answer the question. Furthermore, we only evaluate the retriever's performance by comparing if the chunk id it retrieves match the chunk id in the ground truth dataset or not. However, information in the company report can be repetitive and thus, chunk id might not be the best method of evaluation, since it does not take into account semantic similarity.

Another major limitation is that due to technical reasons, we could not consider images data within the company report - while companies often use graphical means to convey their idea. We considered using LLM with vision to generate a summary for each graph, but since (1) vision LLMs are still unreliable on this task, and (2) we have no means to detect which part of the company's report is a figure, and passing all the report for LLM to pre-process would be too costly.

9.3 Future works

Based on our limitations, we recommend the following for future works:

1. **Apply a modular approach**: Current LLM research focuses on deploying different modules that LLM can choose to invoke if needed. For example, when met with a criteria that contains regulation, LLM can choose to invoke the "regulation explained" module to get an extensive overview of that regulation.

2. **Investigate methods for automatic evaluation of LLM responses**: The current evaluation process of LLM-generated text for this task is very time-consuming and error-prone.

3. **Curate a better dataset**: As mentioned, our dataset is not curated by professionals and experts in the field. Therefore, a better dataset could also assist in developing a better evaluation method.

4. **Evaluation of Retriever system based on semantic similarity**.

5. **Investigate the role and performance of Cross-Encoder reranking**: We also noted that since the context window of the current CE model is much smaller than the chunk size, it might cause the CE model to give incorrect scores. Therefore, further experiments are needed to confirm the correlation, and perhaps the Cross-Encoder should be considered when setting chunk size.

9.4 Acknowledgement

I would like to thank my ING colleagues for being part of this project: Luna, Jef, and Yuefeng, for introducing me to ING, and in particular, Luna for introducing me to Cees and the Emerging Technology team. Furthermore, I thank Cees and Gamze for the guidance at the start of the project, Ceren for the continuous support and supervision, and other team members Matthijs, Mo, Bauke, Mariana, Kate, and Nick for supporting my thesis.

From the UT side, I would like to express my gratitude to dr. Shenghui Wang, dr. Gwenn Englebienne, thank you for your expertise and guidance on this project from the beginning. And finally, I thank dr. Maurice van Keulen for being part of my Graduation committee.

Bibliography

[1] Giambattista Amati. *BM25*, pages 257–260. Springer US, Boston, MA, 2009. doi:10.1007/978-0-387-39940-9_921.

[2] Guangsheng Bao, Hongbo Zhang, Linyi Yang, Cunxiang Wang, and Yue Zhang. Llms with chain-of-thought are non-causal reasoners, 2024. arXiv:2402.16048.

[3] Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. 01 2009.

[4] Dorian Brown. Rank-BM25: A Collection of BM25 Algorithms in Python, 2020. doi:10.5281/zenodo.4520057.

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

[6] Ricardo Campos, Vítor Mangaravite, Arian Pasquali, Alípio Jorge, Célia Nunes, and Adam Jatowt. Yet Another Keyword Extractor (Yake), 2019. URL: https://github.com/LIAAD/yake.

[7] Ricardo Campos, Vítor Mangaravite, Arian Pasquali, Alípio Jorge, Célia Nunes, and Adam Jatowt. Yake! keyword extraction from single documents using multiple local features. *Information Sciences*, 509:257–289, 2020.

[8] Harrison Chase. LangChain, October 2022. URL: https://github.com/langchain-ai/langchain.

[9] The European Comission. Commission delegated directive (eu) 2023/2775 of 17 october 2023 amending directive 2013/34/eu of the european parliament and of the council as regards the adjustments of the size criteria for micro, small, medium-sized and large undertakings or groups, 2023. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32022L2464.

[10] Gordon V. Cormack, Charles L A Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of*

the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '09, page 758–759, New York, NY, USA, 2009. Association for Computing Machinery. doi:10.1145/1571941.1572114.

[11] Antonia Creswell, Murray Shanahan, and Irina Higgins. Selection-inference: Exploiting large language models for interpretable logical reasoning. In *The Eleventh International Conference on Learning Representations*, 2023. URL: https://openreview.net/forum?id=3Pf3Wg6o-A4.

[12] Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. The faiss library. 2024. arXiv:2401.08281.

[13] Mengnan Du, Fengxiang He, Na Zou, Dacheng Tao, and Xia Hu. Shortcut learning of large language models in natural language understanding. *Commun. ACM*, 67(1):110–120, dec 2023. doi:10.1145/3596490.

[14] Hervé Déjean, Stéphane Clinchant, and Thibault Formal. A thorough comparison of cross-encoders and llms for reranking splade, 2024. URL: https://arxiv.org/abs/2403.10407, arXiv:2403.10407.

[15] Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. Complexity-based prompting for multi-step reasoning, 2023. arXiv:2210.00720.

[16] Gemini Team and Google. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context, 2024. arXiv:2403.05530.

[17] Maarten Grootendorst. Keybert: Minimal keyword extraction with bert., 2020. doi:10.5281/zenodo.4461265.

[18] Jordan Hairabedian. Who is the eu taxonomy for and what are the benefits of alignment?, 2023.
https://eco-act.com/blog/eu-taxonomy-benefits-of-aligning/.

[19] Hangfeng He, Hongming Zhang, and Dan Roth. Rethinking with retrieval: Faithful large language model inference, 2022. arXiv:2301.00303.

[20] Yizheng Huang and Jimmy Huang. A survey on retrieval-augmented text generation for large language models, 2024. arXiv:2404.10981.

[21] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*, 2022. URL: https://arxiv.org/abs/2112.09118, arXiv:2112.09118.

[22] Rolf Jagerman, Honglei Zhuang, Zhen Qin, Xuanhui Wang, and Michael Bendersky. Query expansion by prompting large language models, 2023. URL: https://arxiv.org/abs/2305.03653, arXiv:2305.03653.

[23] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12), mar 2023. doi:10.1145/3571730.

[24] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus, 2017. URL: https://arxiv.org/abs/1702.08734, arXiv:1702.08734.

[25] Mehran Kazemi, Najoung Kim, Deepti Bhatia, Xin Xu, and Deepak Ramachandran. LAMBADA: Backward chaining for automated reasoning in natural language. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6547–6568, Toronto, Canada, July 2023. Association for Computational Linguistics. URL: https://aclanthology.org/2023.acl-long.361, doi:10.18653/v1/2023.acl-long.361.

[26] Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213. Curran Associates, Inc., 2022. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/8bb0d291acd4acf06ef112099c16f326-Paper-Conference.pdf.

[27] Yibin Lei, Yu Cao, Tianyi Zhou, Tao Shen, and Andrew Yates. Corpus-steered query expansion with large language models, 2024. URL: https://arxiv.org/abs/2402.18031, arXiv:2402.18031.

[28] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc., 2020. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf.

[29] Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Alexander Cosgrove, Christopher D Manning, Christopher Re, Diana Acosta-Navas, Drew Arad Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue WANG, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri S. Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Andrew Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. Holistic evaluation of language models. *Transactions on Machine Learning Research*, 2023. Featured Certification, Expert Certification. URL: https://openreview.net/forum?id=iO4LZibEqW.

[30] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL: https://aclanthology.org/W04-1013.

[31] Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland, May 2022. Association for Computational Linguistics. URL: https://aclanthology.org/2022.acl-long.556, doi:10.18653/v1/2022.acl-long.556.

[32] Gary Marcus. The next decade in ai: Four steps towards robust artificial intelligence, 2020. arXiv:2002.06177.

[33] Meta AI. Introducing meta llama 3: The most capable openly available llm to date, 2024.
https://ai.meta.com/blog/meta-llama-3/.

[34] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. MS MARCO: A human generated machine reading comprehension dataset. *CoRR*, abs/1611.09268, 2016. URL: http://arxiv.org/abs/1611.09268, arXiv:1611.09268.

[35] Baharan Nouriinanloo and Maxime Lamothe. Re-ranking step by step: Investigating pre-filtering for re-ranking with large language models, 2024. URL: https://arxiv.org/abs/2406.18740, arXiv:2406.18740.

[36] OpenAI. Gpt-4 technical report, 2024. arXiv:2303.08774.

[37] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA, 2002. Association for Computational Linguistics. doi:10.3115/1073083.1073135.

[38] The European Parliament and the Council of the European Union. Regulation (eu) 2020/852 of the european parliament and of the council, 2020.
https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A32020R0852.

[39] The European Parliament and the Council of the European Union. Directive (eu) 2022/2464 of the european parliament and of the council, 2022.
https://eur-lex.europa.eu/eli/dir_del/2023/2775.

[40] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[41] Matt Post. A call for clarity in reporting BLEU scores. In Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Christof Monz, Matteo Negri, Aurélie Névéol, Mariana Neves, Matt Post, Lucia Specia, Marco Turchi, and Karin Verspoor, editors, *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium, October 2018. Association for Computational Linguistics. URL: https://aclanthology.org/W18-6319, doi:10.18653/v1/W18-6319.

[42] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL: https://arxiv.org/abs/1908.10084.

[43] Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389, apr 2009. doi:10.1561/1500000019.

[44] Claude Sammut and Geoffrey I. Webb, editors. *TF–IDF*, pages 986–987. Springer US, Boston, MA, 2010.

[45] Prafull Sharma and Yingbo Li. Self-supervised contextual keyword and keyphrase retrieval with self-labelling, 2019. Preprints 2019, 2019080073. URL: https://doi.org/10.20944/preprints201908.0073.v1.

[46] Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. Is chatgpt good at search? investigating large language models as re-ranking agents, 2023. URL: https://arxiv.org/abs/2304.09542, arXiv:2304.09542.

[47] Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10014–10037, Toronto, Canada, July 2023. Association for Computational Linguistics. URL: https://aclanthology.org/2023.acl-long.557, doi:10.18653/v1/2023.acl-long.557.

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[49] Liang Wang, Nan Yang, and Furu Wei. Query2doc: Query expansion with large language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9414–9423, Singapore, December 2023. Association for Computational Linguistics. URL: https://aclanthology.org/2023.emnlp-main.585, doi:10.18653/v1/2023.emnlp-main.585.

[50] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL: https://openreview.net/forum?id=1PL1NIMMrw.

[51] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc., 2022. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf.

[52] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*, 2019.

[53] Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. Automatic chain of thought prompting in large language models, 2022. arXiv:2210.03493.

[54] Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V Le, and Ed H. Chi. Least-to-most prompting enables complex reasoning in large language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL: https://openreview.net/forum?id=WZH7099tgfM.

Appendix A

Dataset

A.1 Links to companies' reports

Sector	Company	Report Link
Waste Management	Suez*	Suez 2023 Sustainability Report
	Renewi	Renewi plc Sustainability Review 2023
	Biffa	Biffa Sustainability Report 2023
Marine Transport	Euronav	Euronav 2023 Annual report
	Golden Ocean	2022 Golden Ocean Annual ESG Report [1]
	Maersk	2023 Maersk Sustainability Report
Metal	AcrelorMittal*	2023 ArcelorMittal Integrated Annual Review
	ThyssenKrupp*	Annual Report 2022/2023 Thyssenkrupp
Energy	Trafigura	2023 Trafigura Sustainability Report
	Iberdrola*	2023 Iberdrola Sustainability Report
Automotive	BMW*	BMW Group Report 2023
Miscellaneous	Hitachi	2023 Hitachi Sustainability Report
	General Electric	GE Vernova 2022 Sustainability Report[1]
	Vopak*	Vopak Annual Report 2023
	Norsk Hydro*	Norsk Hydro Integrated Annual Report 2023

TABLE A.1: Sustainability report or equivalent of each company used in the dataset.

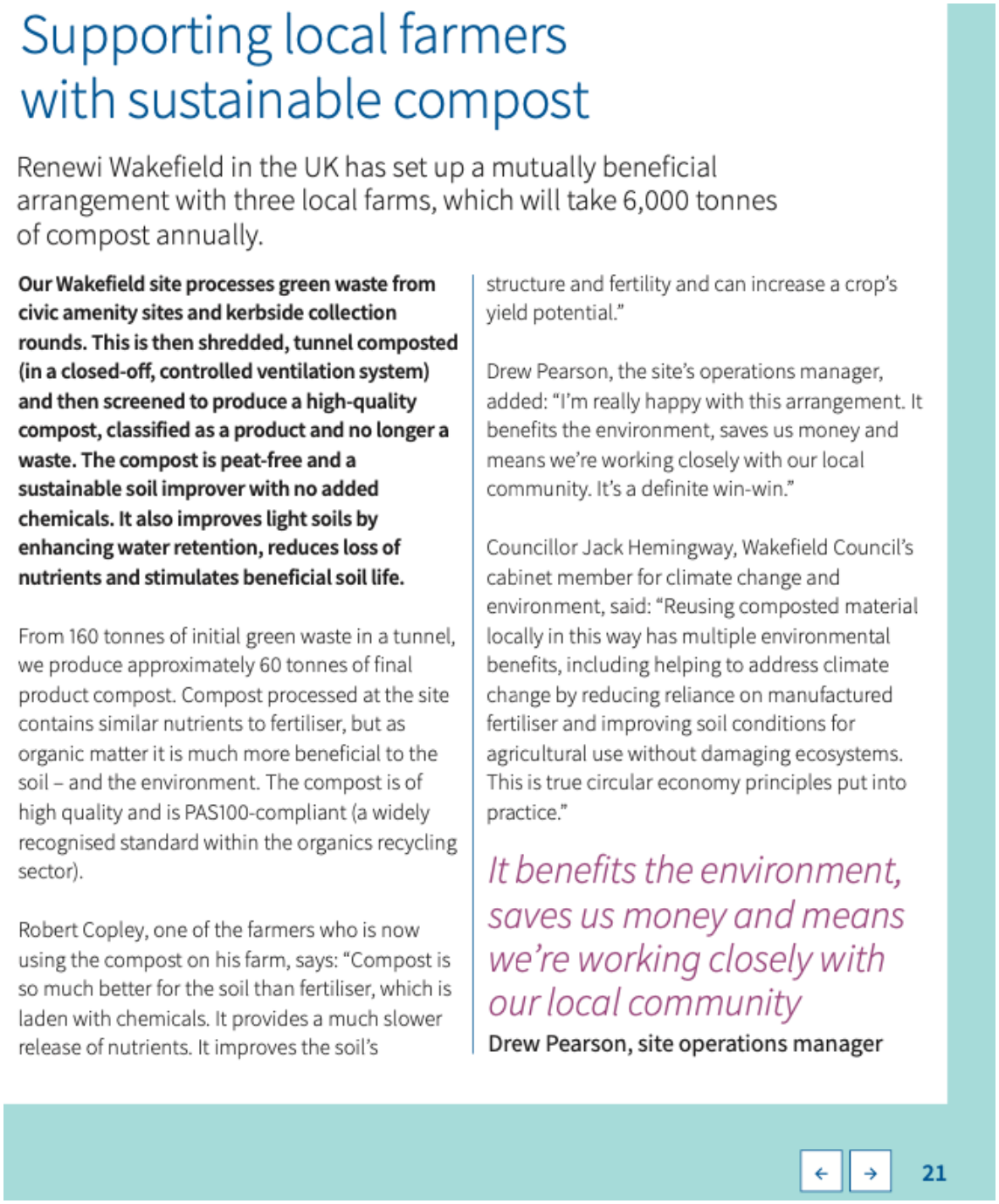
[1] As of 1st April 2024, Golden Ocean and GE Vernova has not released their 2023 ESG report.

A.2 Example

We take Renewi's economy activity 5.8 - Composting of bio-waste as an example of the dataset creation process. Suez, one of Renewi's competitors, also reports on this activity - therefore, 5.8 could be an activity in which Renewi performs business. The description for activity 5.8 is "Construction and operation of dedicated facilities for the treatment of separately collected bio-waste through composting (aerobic digestion) with the resulting production and utilisation of compost", and the substantial contribution criteria are: (1) The bio-waste that is composted is source segregated and collected separately, and (2) The compost produced is used as fertiliser or soil improver and meets the requirements for fertilising materials set out in Component Material Category 3 in Annex II to Regulation (EU) 2019/1009 or national rules on fertilisers or soil improvers for agricultural use.

With this information, on page 21 of Renewi's 2023 Sustainability Review, we identified a paragraph that reports on Renewi's compost business, as shown in figure A.1. In this paragraph, Renewi mentioned that they collect "green waste", implying that the source is segregated and collected separately (as per criteria 1). Renewi also mentioned that the compost is PAS100-compliant, a UK Government's standard for compost quality [2], satisfying criteria 2 on "meets the requirements for fertilising materials set out on national rules". Therefore, we can conclude that Renewi satisfied the substantial contribution criteria to Climate Change Mitigation for activity 5.8.

[2] https://assets.publishing.service.gov.uk/media/5a8039cfe5274a2e8ab4eeb1/Material_comparators_for_materials_applied_to_land_-_PAS100_compost.pdf



Supporting local farmers with sustainable compost

Renewi Wakefield in the UK has set up a mutually beneficial arrangement with three local farms, which will take 6,000 tonnes of compost annually.

Our Wakefield site processes green waste from civic amenity sites and kerbside collection rounds. This is then shredded, tunnel composted (in a closed-off, controlled ventilation system) and then screened to produce a high-quality compost, classified as a product and no longer a waste. The compost is peat-free and a sustainable soil improver with no added chemicals. It also improves light soils by enhancing water retention, reduces loss of nutrients and stimulates beneficial soil life.

From 160 tonnes of initial green waste in a tunnel, we produce approximately 60 tonnes of final product compost. Compost processed at the site contains similar nutrients to fertiliser, but as organic matter it is much more beneficial to the soil – and the environment. The compost is of high quality and is PAS100-compliant (a widely recognised standard within the organics recycling sector).

Robert Copley, one of the farmers who is now using the compost on his farm, says: "Compost is so much better for the soil than fertiliser, which is laden with chemicals. It provides a much slower release of nutrients. It improves the soil's structure and fertility and can increase a crop's yield potential."

Drew Pearson, the site's operations manager, added: "I'm really happy with this arrangement. It benefits the environment, saves us money and means we're working closely with our local community. It's a definite win-win."

Councillor Jack Hemingway, Wakefield Council's cabinet member for climate change and environment, said: "Reusing composted material locally in this way has multiple environmental benefits, including helping to address climate change by reducing reliance on manufactured fertiliser and improving soil conditions for agricultural use without damaging ecosystems. This is true circular economy principles put into practice."

It benefits the environment, saves us money and means we're working closely with our local community

Drew Pearson, site operations manager

21

FIGURE A.1: Renewi's 2023 Sustainability Review, page 21, showing information on Renewi's compost business

Appendix B

Prompts

B.1 Traditional prompt for answering questions about EU taxonomy

You are a Sustainability Analyst at a large bank in Europe. You have an extensive knowledge on different Sustainability topics, especially the EU taxonomy of economic activities.
In brief, the EU taxonomy is a list of economic activities and how they can substantially contribute to a climate goal. In this task, you will only looking at economic activities that contribute to the Climate Change mitigation goal. You are given some snippet of a company report that are related to an economic activity, along with how that economic activity substantially contribute to climate change mitigation. Your task is to verify, based on that snippet, if the SCC are satisfied.
Few remarks: (1) Base the answer solely on the snippet provided. (2) Pay attention to details in the criteria, and in some cases, not all the sub-criteria needs to be satisfied - it will be mentioned in the text, such as "Satisfying one of the following criteria".
(3) You are analyzing 2023's data. Therefore, if {{company}} reports data for multiple years and/or predictions for the future, ignore it. Only focus on data for Financial Year 2023 and criteria that needs to be applied before 2023. However, any investments for the future but performed in the year 2023 is taken into account.

Question: Perform step-by-step analysis of {{company}} on this economic activity {{activity}}, based solely on the following snippet of {{company}}'s Sustainability report The description of the activity is {{activity-description}}. The SCC for CCM of this activity are: {{criteria}}.
Company report snippet:
{{sources}}
Do not only answer yes or no, provide detailed information on how did you arrive at that conclusion.
In case the company already mentioned that they met the criteria, just report it and don't need to analyze further.
ONLY Answer in JSON in the following format:
{ "Analysis": <ANALYSIS as string>,
"Conclusion based on answer": True/False, }
Make sure your JSON file are syntactically correct. No need to encapsulate JSON inside a code block.

Table B.1: Traditional prompt. In brakets {{}} are variables to be changed.

B.2 CoT prompt for answering questions about EU taxonomy

You are a Sustainability Analyst at a large bank in Europe. You have an extensive knowledge on different Sustainability topics, especially the EU taxonomy of economic activities. In brief, the EU taxonomy is a list of economic activities and how they can substantially contribute to a climate goal. In this task, you will only looking at economic activities that contribute to the Climate Change mitigation goal in the company's turnover. You are given some snippet of a company report that are related to an economic activity, along with how that economic activity substantially contribute to climate change mitigation. Your task is to verify, based on that snippet, if all the substantial contribution criteria are satisfied.
Few remarks:
- Base the answer solely on the snippet provided;
- Data provided is in HTML format, and could include tables;
- Pay attention to details in the criteria. In some cases, not all the sub-criteria needs to be satisfied, either because only one (or more) needs to be satisfied, or the sub-criteria has a eligibility requirement;
- You are analyzing 2023's data. Therefore, if {{company}} reports data for multiple years and/or predictions for the future, ignore it. Only focus on data for Fiancial Year 2023. Focus on criteria that needs to be applied before 2023. However, any investments for the future but performed in the year 2023 is taken into account.
- VERY IMPORTANT: Your step-by-step reasoning should contain three steps: (1) Break down the criteiria into sub-criteria, (2) Break down what the company report on each sub-criteria, (3.1) Conclude and explain if company satisfy the substantial contribution criteria, and (3.2) Conclusion based on step 3.1.

Before you start, here are a few examples to demonstrate how to analyze step-by-step: {{examples}}
Question: Perform step-by-step analysis of {{company}} on this economic activity {{activity}}, based solely on the following snippet of {{company}}'s Sustainability report The description of the activity is {{activity-description}}. The Substantial contribution criteria for CCM of this activity are: {{criteria}}.
Company report snippet:
{{sources}}
ONLY answer in JSON in the following format. Use linebreaks and markdown lists frequently to enhance readability.
{ "Step 1: Break down the criteria": <STEP 1 REASONING as string>,
"Step 2: Break down what the company report on each sub-criteria" : <STEP 2 REASONING as string>,
"Step 3.1: Conclude and explain if company satisfy the substantial contribution criteria": <STEP 3.2 REASONING as string>
"Step 3.2: Conclusion based on step 3.1" : True/False}
Make sure your JSON file are syntactically correct. No need to encapsulate JSON inside a code block.

TABLE B.2: CoT prompt. The only difference between zero-shot and Few-shot CoT are the {{examples}}, shown in bold.

B.3 Query Splitting prompt

You are given an economic activity from the EU taxonomy, including its description and substantial contribution criteria . The goal of this task is to retrieve chunks of information related to this activity from a company report. Since it is done using a Vector database search with cosine similarity, we need to provide queries to the database. However, providing only the description and the substantial contribution criteria as queries are not enough to retrieve relevant information, since the query can be too long and contains too many information. Your task is to split the description and the substantial contribution criteria into sub-queries. Provide the answer as a list of sub-queries, like: ['sub-query-1', 'sub-query-2', ..., 'sub-query-n']. Give at least 5 sub queries at at max 20 sub queries. The activity's description: {{activity_description}} The substantial contribution criteria: {{scc}} Give your answer as a list. Your answer will be automatically evaluated if it's a python list or not, so make sure it's syntactically correct.

Table B.3: Prompt to split the queries into sub-queries.

B.4 Hypothetical Answer prompt

You are a Sustainability Analyst at a large bank in Europe. You have an extensive knowledge on different Sustainability topics, especially the EU taxonomy of economic activities. In brief, the EU taxonomy is a list of economic activities and how they can substantially contribute to a climate goal. In this task, you will only looking at economic activities that contribute to the Climate Change mitigation goal. You are given an economic activity, along with how that economic activity substantially contribute to climate change mitigation. Your task is to answer, based on your knowledge about {{company}}, if they satisfy the substantial contribution criteria. Question: Perform hypothetical analysis of {{company}} on this economic activity {{activity}}, based on your knowledge. The description of the activity is activity-description. The Substantial contribution criteria for CCM of this activity are: {{scc}}. Give both for- and against arguments on why {{company}} satisfies/not satisfy the criteria.

Table B.4: Prompt for Hypothetical answer generation.

B.5 Query Extension based on Hypothetical Answer prompt

You are given an economic activity from the EU taxonomy, including its description and substantial contribution criteria . The goal of this task is to retrieve chunks of information related to this activity from a company report. Since it is done using a Vector database search with cosine similarity, we need to provide queries to the database. However, providing only the description and the substantial contribution criteria as queries are not enough to retrieve relevant information, since the query can be too long and contains too many information. Therefore, you a given a hypothetical answer containing arguments on how company satisfies/ does not met the criteria. Your task is to split the hypothecial answers into sub-queries. Provide the answer as a list of sub-queries, like: ['sub-query-1', 'sub-query-2', ..., 'sub-query-n']. Give at least 5 sub queries at at max 20 sub queries. The activity's description: {{activity_description}} The substantial contribution criteria: {{scc}} Hypothetical Answer: {{hypothetical-answer}} Give your answer as a list. Your answer will be automatically evaluated if it's a python list or not, so make sure it's syntactically correct.

TABLE B.5: Prompt for Query Extension based on Hypothetical Answer.

B.6 Query Extension based on Pseudo-relevance Feedback prompt

You are given an economic activity from the EU taxonomy, including its description and substantial contribution criteria . The goal of this task is to retrieve chunks of information related to this activity from a company report. Since it is done using a Vector database search with cosine similarity, we need to provide queries to the database. However, providing only the description and the substantial contribution criteria as queries are not enough to retrieve relevant information, since the query can be too long and contains too many information. Therefore, we will perform pseudo-relevance feedback: retrieving the top 1 document, and your task is to perform query extension using this top-1 document, assuming it is relevant. Provide the answer as a list of sub-queries, like: ['sub-query-1', 'sub-query-2', ..., 'sub-query-n']. Give at least 5 sub queries at at max 20 sub queries. The activity's description: {{activity_description}} The substantial contribution criteria: {{scc}} The pseudo-relevance document: {{prf}} Give your answer as a list. Your answer will be automatically evaluated if it's a python list or not, so make sure it's syntactically correct.

TABLE B.6: Prompt for Query Extension based on Pseudo-relevance Feedback.

Appendix C

Chunk Size Experiment

C.1 Experimental Setup

As the prepared dataset contains the sources for each row and copy directly from the company's report (pre-determined GT chunk), the goal of this experiment is to find the optimal chunk size and chunk overlap that encapsulates all pre-determined ground truth chunks. We employ grid search to perform the experiment, with the chunk size ranging from 1000 to 300 and a step of 200, while the chunk overlap ranges from 0 to 500, with step of 100.

For each pre-determined GT chunk (d) of each row in the GT dataset , we compute the chunk overlap score $O_{d,c}$ (C.1) against each chunk $c \in C$, with C the list of all possible chunks after chunking. The chunk $c_{best} \in C$ with the highest overlap score is chosen. We then count all GT chunks d with $O_{d,c}$ lower than 0.8, as a metric to compare different $(chunk\_size, chunk\_overlap)$ combinations. In equation C.1, w represents a word - split by common separators such as whitespace, nextline, etc.

$$O_{d,c} = \frac{|w \in \{d \cap c\}|}{|d|} \tag{C.1}$$

C.2 Results

For all $(chunk\_size, chunk\_overlap)$ combinations within the defined range, only four combinations have zero low overlap count: (2800, 300), (2800, 400), (2800, 500), and (3000, 500). Thus, the combination (2800, 300) is chosen, as this is the lowest combination. Lower value in chunk size will take up less space in an LLM's context window, and smaller overlap will results in less chunks, both while keeping the same semantic meaning.