Enhancing Digital Libraries with Automated Definition Generation

Andrea Zielinski
andrea.zielinski@isi.fraunhofer.de
Fraunhofer Institute for Systems and Innovation Research ISI
Karlsruhe, Germany

Simon Hirzel
simon.hirzel@isi.fraunhofer.de
Fraunhofer Institute for Systems and Innovation Research ISI
Karlsruhe, Germany

Sonja Arnold-Keifer
sonja.arnold-keifer@isi.fraunhofer.de
Fraunhofer Institute for Systems and Innovation Research ISI
Karlsruhe, Germany

Abstract

Scientific domains encompass many concepts that require a concise term definition to enable a common understanding among researchers, in particular for interdisciplinary fields. In digital libraries, information access and sharing is often facilitated by terminology databases. However, building up such resources is expensive to produce manually and requires expert knowledge.

Automatically generating definitions for scientific terms has become a hot research topic recently that can reduce the manual burden. However, current methods heavily rely on large language models (LLMs) that store factual knowledge in their parameters, so that knowledge cannot be easily updated for emerging scientific terms. Furthermore, a major shortcoming of these models is that they are prone to hallucination and their output is difficult to control. To bridge these gaps, we propose to address the task of definition generation through guided abstractive summarization, incorporating key information from external resources. At test time, we augment the model with retrieved abstracts from *Scopus* and use automatically extracted topics and keywords as guidance, both essential for definition generation. To this aim, our approach takes into account two relevant sub-tasks in the process, a) predicting the topic class and b) generating hypernym candidates for the term.

Our proposed pipelined approach for automatic guided definition generation achieves significant performance improvement over the standard baselines as well as relevant prior works on this problem. We use BLEU, ROUGE and BERTScore to automatically evaluate the quality of the systems on our benchmark and carry out a human evaluation to assess fluency, relevancy, coherence and factuality of the output. Our experiments show that LLMs can provide fluent and coherent definitions, and are often on par with human created definitions. Yet, there is still room for improvement on identifying relevant content and improving factual correctness.

CCS Concepts

• **Knowledge Representation Formalisms and Methods**; • **Natural Language Processing**;

Keywords

natural language generation, terminology, definitions, large language models, indexing, information retrieval, information systems, digital libraries, natural language processing

ACM Reference Format:
Andrea Zielinski, Simon Hirzel, and Sonja Arnold-Keifer. 2024. Enhancing Digital Libraries with Automated Definition Generation. In *The 2024 ACM/IEEE Joint Conference on Digital Libraries (JCDL '24), December 16–20, 2024, Hong Kong, China.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3677389.3702536

JCDL '24, December 16–20, 2024, Hong Kong, China

ACM ISBN 979-8-4007-1093-3/24/12.
https://doi.org/10.1145/3677389.3702536

1 Introduction

Terminological resources are necessary to ensure the quality and consistency of scientific terms. They are generally defined by experts and contain information intended to represent their meaning, expressed in a such a way that they are interpretable and helpful to expert and non-expert users.

In the energy field, one of the most representative terminological resources is the EnArgus wiki, the central German information system for energy research funding [23, 48]. It is a widely-used resource which incorporates all the terminology with respect to energy and sustainability based on a systematic literature of funded research projects. It includes approximately 2,500 energy-related concepts whose relationships are expressed in a comprehensive ontology [57]. The terms are interlinked by cross-references and each term is assigned a subject field. The high quality terminology database is expensive to produce manually, taking teams of experts several years to complete, reflecting the terminology at the time of creation. Automatic methods that are able to generate high quality articles or assist editors to enhance article quality are needed, preferably for different languages. This may facilitate faster production, in particular for emerging research topics, without sacrificing rigor.

Therefore, the task of generating term definitions automatically has recently evolved as a new research area in NLP that considers the task as a language modeling task with the goal to estimate the joint probability of a sequence of words in a language [18].

While results are overall promising for common vocabulary terms, it has been observed by various authors [18, 25] that definition modeling faces several challenges when dealing with technical and non-English terms [46]. While large language models (LLMs) memorize a vast amount of factual knowledge, exhibiting strong performance across diverse tasks and domains, the performance diminishes when it comes to less-popular or low-frequency concepts

in domain-specific applications [18]. Likewise, major challenges exist for retrieving up-to-date knowledge, and less common long-tail information that is exempt from scholarly databases [38].
Inspired by the work of [4], we investigate two different approaches as baselines: a) abstractive question answering by leveraging knowledge encoded in LLMs by fine-tuning them for the task; and b) prompting LLMs by providing a concise task description without additional training. In particular, we compare BART [34] based on the encoder-decoder architecture with decoder-only models from the GPT family [9].
To overcome the problem of data scarcity, we also seek to integrate background information related to EnArgus terms by means of scholarly publications (e.g., *Scopus* abstracts), since adding retrieved evidence text passages has shown to improve performance [15, 35]. We investigate two different approaches: a) an abstractive summarization method, using topics and keywords as guidance, ensuring that the key information from the source text is incorporated (controlled summarization) [21, 67] and b) retrieval-augmented prompts on GPT-4, where we ingest top abstracts from *Scopus* and prepend them to the question (retrieval-augmented generation).
Our contributions are as follows:

- We explore various recent neural models for the task and evaluate how well they capture scientific terminology and are able to generate valid scientific definitions for terms from the energy field.
- We explore the performance for this task in a multilingual setting, comparing English to German models.
- We evaluate advanced neural models that are able to incorporate support information as well as guiding keywords and topic information into the definition generation process.
- We evaluate how well our models match human definitions of terminology. To this aim, we build up a human-annotated corpus with expert annotations of generated text.

After presenting the related work (Section 2), we describe the main task and related subtasks (Section 3) and analyze the EnArgus dataset (Section 4). We detail our methodology (Section 5) and give experimental details for all tasks. In our empirical study, we experiment with baseline models (Section 6.1) and explore various strategies to enhance/guide the generated output (Section 6.2), followed by an evaluation (Section 7), and conclusion (Section 8).

2 Related work

2.1 Automated Definition Generation

The early works of automatically generating definitions date back to Noraset et al. [2017], who introduce the task, focusing on dictionary definitions of frequent English words. The authors experiment with static word2vec embeddings and test various neural models, where a gated recurrent neural network (RNN) model, combined with a character-level convolutional neural network (CNN) performed best. Since static embeddings are not able to account for polysemous words or handle newly invented words, in subsequent work, examples of contextual usage were added to the word representations [44].
Balachandran et al. [2018] focus on definition generation for technical terms from the software domain, building on the work of [47]. The authors seek to accommodate for the *genus-differtia* structure of definitions by incorporating word co-occurrence information and the ontological category of the target word. To this aim, domain-specific word embeddings are trained from a huge background corpus (QA forum).
More recent approaches are typically based on pre-trained large language models and contextualized representations that are able to encode the meaning of individual words more accurately, as demonstrated by recent probing studies [2, 41]. Huang et al. [24] use the encoder-decoder architecture and fine-tune a T5 model [52] on word-context pairs for common English words and Wikipedia terms. Generated definitions can be further improved by regulating their degree of specificity via a re-ranking module. Likewise, the complexity level of the definition can be adjusted to the user at decoding time by re-ranking [4]. The authors frame the task as an abstractive question qnswering task and evaluate various transformer models for two scientific domains. Their best model is the BART-large model, fine-tuned on pairings of the term and definition, and supplemented with definitions contained in scholarly abstracts. For a review on the task, see [18].
However, these methods all rely on knowledge encoded within LLMs during training, which is opaque and hidden in the models' parameters, resulting in poor factual knowledge retrieval capabilities [26, 66].
This study aims to utilize information from research abstracts to guide LLMs towards more factually correct definitions. Specifically, we propose large language models like BART and GPT-4 that are able to summarize the content of multiple research abstracts with respect to certain aspects in an abstractive manner. In this work, we focus on comparing baseline models with more recent summarization methods that are tailored to align with specific objectives that are relevant for definition generation, in particular keywords and topic information, i.e. controllable text summarization and retrieval-augmented generation. A comprehensive survey can be found in [58] and [37].
To the best of our knowledge, controlled summarization on scholarly abstracts has not been exploited yet for the task of definition modeling. Moreover, we compare German to English models in solving the task.

2.2 Controllable text generation

We build on recent work for controllable text generation [21, 64] and controllable text summarization [16, 42, 67]. LLMs encode a significant amount of knowledge implicitly in their parameters [9, 52], including factual [50] and relational knowledge [55]. However, deep neural networks are essentially black-box models that lack interpretability, and therefore generation of text satisfying certain control conditions can help produce more focused output [64]. Actually, text generation in an unconstrained setting is an ill-defined task where multiple generated summaries are equally relevant [30].
In tasks like abstractive summarization, providing the model with discrete guiding signals, such as keywords, has shown to improve their performance in terms of relevancy and factuality [13, 21]. Moreover, query-based abstractive summarization techniques have been defined that are able to generate a concise summary related to a given query [31, 32, 59]. In this approach, the query generally

refers to a certain question that might be spread across multiple documents. Interestingly, query-based summarization can also be combined with lexical constraints such as keywords [21].
Control mechanisms have been established in generic summarization models for both fine-tuned transformer models based on models like BART and the instruction-tuned GPT-family models. It has been shown that prompt-based adaptation is often superior to conventional fine-tuning in settings with scarce resources, without requiring extensive parameter updates [33, 39].

2.3 RAG

Retrieval-augmented generation (RAG) is a popular framework that combines seq2seq models with external knowledge bases by using vector indices of Wikipedia [35]. It consists of a Dense Passage Retrieval [27] and BART [34], but can also be used with other LLMs and extract knowledge from any other external knowledge base. Using both parametric and non-parametric memory generally leads to reduced hallucinations and higher interpretability in tasks like question answering and summarization [20, 62]. Even though the system might retrieve relevant passages, pre-training of the language model is still required, which might be expensive [38]. RAG has also been explored in the context of scholarly digital libraries for generating automatic literature reviews [8]. Our work is related to in-context Retrieval-Augmented Language Modeling (RALM), grounding the model during generation by conditioning on relevant documents retrieved from an external knowledge source. In this setting, prepending the selected documents to the language model's input text does not involve altering the model's weights [53].

3 Task Description

Definition modeling, initially proposed by Noraset et al. [47], is a specific application of language modeling where the model estimates the likelihood of textual definitions for terms. This involves predicting the probability of a sequence of words that form a coherent and accurate definition that captures its essential meaning. In this study, we focus on the generation of terminology definitions within the scientific domain [4, 14, 24] that includes the following subtasks:

- *Topic Classification:* assign a topic class to a term
- *Hypernym Prediction:* assign the lowest top-level concept, i.e. the most specific concept in the sub-class hierarchy

3.1 Defining Terms

Even though a definition is often in free-text format, it exhibits a conceptual structure composed of the features selected as relevant for defining the term.
Its underlying logical form is generally that of the classical Aristotelian definition of a species via *genus* and *differentia* (as in: a transistor (species) is a semiconductor device (genus) used to amplify or switch electrical signals and power (differentia). Thus, a definition can be decomposed into two parts containing the corresponding textual information units, i.e. *genus* and *differentia*.
Generally, a basic building block for specifying the *genus* is by means of an ontology or taxonomy, where the classes are organized with subsumption or meronomy relations. The part of the definition that plays the *genus* role is thus often by way of the *is-a* relation or the *part-of* relation [17, 51].

Table 1: Statistics on our dataset, terms and definitions

Dataset	Language	Terms	Avg. Length [min, max]
EnArgus	de	2,484	31.50 [22, 36]
EnArgus	en	2.484	29.5 [20, 35]

4 Dataset

In this section, we describe the creation of the EnArgus dataset and briefly present some statistics to provide a quantitative overview.

4.1 Data collection

Our dataset consists of individual definitional articles covering a broad range of energy research topics. These articles are sourced from a widely-used open-source platform: the EnArgus Wiki[1], part of an information system for energy research funding, containing 2,484 entries. Additional corpus statistics are displayed in Table 1. The wiki contains energy-related 173 concepts whose relationships are expressed in a comprehensive ontology [57]. The terms are interlinked by cross-references and each term is assigned one of 7 topic classes (i.e. Energy supply, Energy transport, Energy storage, Energy utilization, Energy and material flows, Markets and players, Basic concepts and fundamentals).

The terms (single words, compounds, or phrases) vary from highly specialized (e.g., layered charge storage) to common words (e.g., clothes dryer). We evaluate our models for the task on the respective benchmark, i.e. German or English, split into train, eval and and test set (90/5/5%). For the human evaluation, we randomly sampled 100 term-definition pairs.

4.2 Data pre-processing

We clean the corpus, removing non-definitional text that does not belong to the core definition, such as etymology and example usage. We use the first 1-3 sentences from each definition which generally contains the core definition and follows the *genus-differentia* structure of definitions (see section 3.1).
EnArgus articles were originally written in German and have been translated into English using DeepL Translate[2], followed by a post-editing step to ensure high quality.

5 Methodology

We briefly discuss the sub-tasks in our pipeline in the subsequent sections. We next present a variety of models that we use to benchmark the feasibility and difficulty of the task of definition generation of scientific terms.

[1] https://www.enargus.de/wiki/
[2] https://www.deepl.com/de/translator

5.1 The various prediction sub-tasks

Here we describe the prediction sub-tasks of text generation, i.e. topic classification and hypernym prediction, and detail our experimental setup. Both sub-tasks have considerable potential to improve the quality of the generated text.

Topic Classification. Topic classification in NLP involves assigning a text into predefined categories. Recent works on text classification leverage large pre-trained language models for predicting the label [29]. Also BERT models [12] have been successfully applied to the task, due to powerful word representations through the transformer encoder model architecture [37].
Topic classification is crucial for disambiguating terms like *jacket* which can refer to *an item of clothing* or, in the energy domain, *a possible form of foundation for offshore wind turbines*. In our case, it is a single label classification task. We use BERT fine-tuned on our training data. Furthermore, we compare German BERT to English SciBERT [7], trained on a corpus of 1.14M scientific publications, comparing them to GPT-4 with a prompt utilizing few-shot learning, which includes the classification instruction, a list of topics and examples of each topic class featuring the subject domain in EnArgus[3].

Hypernym Prediction. The task of hypernymy detection is to classify whether a given pair of terms is in a hypernymy relation [3]. Knowledge of the hypernymy relation is critical for the definition generation task, due to the conceptual structure of definitions composed of the *genus* and *differentia* (see section 3.1). For assigning a hypernym to a term, the term's embeddings have been exploited, using either static embeddings [5, 47], or contextual embeddings [1, 63]. Since semantically related words are expected to be close in vector space, the nearest neighbors can be computed by semantic similarity measures, such as cosine similarity, as is done in many word similarity tasks (e.g. SimLex-999), the notion of similarity is left under-specified, i.e. it can refer to different types of relations. We evaluate the capability of various embeddings from the Massive Text Embedding Benchmark (MTEB) [45] and select the leading Semantic Textual Similarity (STS) embeddings[4]. We build negative pairs from random concept pairs in the EnArgus ontology and evaluate on a benchmark that consists of approx. 1,000 term pairs balanced between positive and negative hypernym pairs. For comparison, we also generate similarity scores with GPT-4[5].

Retrieval. Retrieval is based on Hearst patterns [22] applied to EnArgus terms, which are indexed in Scopus. We use lexico-syntactic patterns (e.g., NP_y is a NP_x) that have shown strong performance for acquiring hypernyms in an unsupervised way with a high precision [41, 54]. We retrieve the whole abstract and leave further compression methods that aim to extract answer spans in the returned abstracts for future work.
Since our goal is to find a sufficiently small subset of documents that correspond to the given topic category, we also apply the SciBERT-based topic classifier and select the top-5 abstracts from *Scopus* according to the citation counts, favoring the most-cited articles.

6 Definition generation: main task

We use transformer-based models for text generation. While the baseline models draw upon their inherent parametric knowledge, the more advanced models have access to external (non-parametric) knowledge that is integrated into the prompt. Guidance is achieved by automatically extracted keywords, i.e. single words or phrases, used to represent a hypernym and/or the topic class.

6.1 Baseline Models

Definition Generation. We use the open-source BART autoregressive decoder model [34] for comparison with the closed-source instruction-tuned GPT-4 (OpenAI, 2023) model. The BART transformer model has been pre-trained with a masked language modeling (MLM) objective and is fine-tuned for text generation on <term-definition> pairs. We chose different foundational models:

- German BART[6] and English BART[7] models
- GPT-4[8], a chatbot comprising approximately 1.8 trillion parameters and a max context length of 128K tokens[9].

Prompts. We apply different prompting strategies to produce definitions. We represent the input as X for the term, Y the definition, topic category T, and hypernym H. We consider hypernym candidates that are listed in the EnArgus ontology.

- X→Y: ask the model to generate an expert-style definition. Answer the question 'What is term X?'. We refer to this approach as GPT4.
  Prompt: You will be given words or phrases, please provide expert-style concise and accurate definitions in German/English for each word or phrase. Assume that the words/phrases are related to "energy" and "sustainability". Term X.
- X,H,T→Y: ask the model to generate a definition, given a topic category and a hypernym. We refer to this approach as GPT4+H+T.
  Prompt: You will be given words or phrases, please provide expert-style concise and accurate definitions in German/English for each word or phrase. Assume that the words/phrases are related to "energy" and "sustainability" and correspond to the topic category T. Please use a hypernym from the following candidates [..]. Term X.

6.2 Controlled Summarization with Support

Alternatively, we frame definition generation as a controlled multi-document summarization tasks. We experiment with models that tackle the definition generation task with a two-step approach: 1) retrieve relevant *Scopus* abstracts; 2) summarize the retrieved content and focus on question answering, thereby incorporating information from the topic classification and hypernym prediction

[3]Prompt: Classify the following text into one of the following categories: A: Energy supply, B: Energy transport, C: Energy storage, D: Energy utilization, E: Energy and material flows, F: Concepts, markets and players, G: Basic concepts and fundamentals. As context, here are some examples: [..]
[4]https://huggingface.co/spaces/mteb/leaderboard
[5]We ask the model to generate a similarity score for word pairs consisting of the term and a hypernym candidate. Prompt: Please assign a semantic similarity score between 0 and 1 for a pair of terms, reflecting their degree of semantic similarity. Here are some examples. Examples: <Carbon dioxide,species of weak acidic gas>: 0.9, <Carbon dioxide,greenhouse gas>: 0.9,<Carbon dioxide,renewable energy> 0.2, <Carbon dioxide,waste product>:0.8
[6]https://huggingface.co/Shahm/bart-german, i.e. a fine-tuned version of facebook/bart-base on mlsum-de dataset
[7]https://huggingface.co/facebook/bart-large
[8]gpt-4-1106-preview; available via the API from https://platform.openai.com
[9]https://explodingtopics.com/blog/gpt-parameters

subtasks.
Our goal is to test if advanced systems are able to control the content of the generated summaries along aforementioned dimensions based on evidence from scientific corpora, resulting in more accurate and coherent term definitions.
We focus on summarization from multiple (i.e. generally 1-4) abstracts, which are concatenated. When automatically summarizing the text, the model should retain the most relevant parts related to the definition of the term from multiple text passages, thereby avoiding repetitive text and ensuring fluency and consistency.

6.2.1 CTRLsum. CTRLsum is a framework for controllable summarization that includes keyword-focused and query-focused summarization based on a fine-tuned BART model [21]. In keyword-based summarization, the generated text should summarize the input, focusing on a given list of keywords, while query-based summarization seeks to produce summaries that answer particular questions of interest [59]. The approach enables to control multiple aspects at inference time, i.e. it can be flexibly adapted for both keywords and a query by descriptive prompts.
The following controlling options are used to generate summaries:

1 Conditioned summarization:
TERM => keywords
e.g.: "PERL cell => solar cell"

2 Prompt summarization:
"Q:What is the definition of TERM? A: => "

We also used the combination of natural language prompts with and without control keywords - refered to as CTRLsum+Q and CTRLsum+Q+H+T, respectivly. Moreover, we specified the minimum/maximum length of the generated output to 20/40 tokens. We use the CTRLsum implementation[10] and the pretrained BART model[11] [21] to yield a question-guided summary, illustrated in Figure 1. The control function uses *chiller* as keyword and *What is the definition of TERM?* as prompt to yield a question-guided summary. All keywords used in our experiments have been extracted automatically.

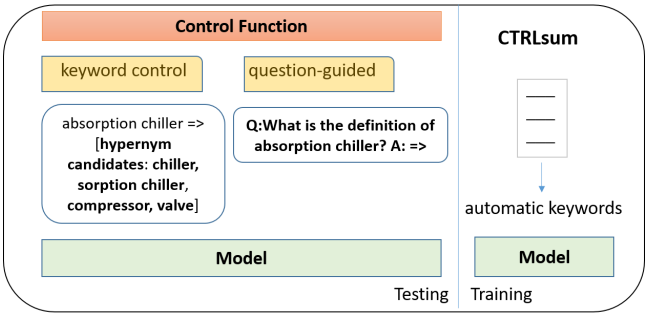

Figure 1: CTRLsum for Controlled Summarization

[10]https://github.com/salesforce/ctrl-sum
[11]https://huggingface.co/hyunwoongko/ctrlsum-cnndm

6.2.2 GPTsum. Prompting large language models, using only a task description, is an alternative to fine-tuned models trained on large summarization datasets for keyword-based summarization [9, 36]. Moreover, adapting the pretrained GPT model to a new setup is possible without requiring parameter updates [33, 39], dramatically cutting down model storage costs. A recent study by Goyal et al. [2022] shows that humans prefer summaries generated by prompting GPT over those from fine-tuned models. These summaries also exhibited a higher degree of factuality. The authors also compare CTRLsum to GPT in a keyword/aspect-focused setting, and find that guiding the model with additional discrete signals, such as entities or keywords, is more successful than with CTRLsum which suffers from lower controllability. Another shortcoming of the BART-based model is that it can perform abstractive summarization but model input size is limited to 1024 tokens.

Prompts. We apply the following prompting strategies (SUPPORT is for support abstracts, see 6.1 for X,Y,H,T):

- X,SUPPORT→Y: ask the model to generate a definition, given the support abstracts as context. We refer to this approach as GPTsum+Q.
  Prompt: Please provide an expert-style concise and accurate definition for the term X related to "energy" and "sustainability", using the following abstracts [..].
- X,SUPPORT,H,T→Y: ask the model to generate a definition, given the support abstracts as well as a list of topic and hypernym candidates as context. We refer to this approach as GPTsum+Q+H+T.
  Prompt: Please provide an expert-style concise and accurate definition for the term X related to "energy" and "sustainability", using the following abstracts. A definition should follow the hypernym and differentia structure (as in: a transistor (species) is a semiconductor device (hypernym) used to amplify or switch electrical signals and power (differentia).

7 Evaluation

In this section, we first describe the evaluation metrics that we chose for automatic and human evaluation of the model generated definitions to compare selected baselines with advanced approaches.

7.1 Evaluation metrics

We explore several methods to evaluate model-generated definitions automatically: ROUGE [40] and BLEU [49]. For ROUGE, we report ROUGE-1, ROUGE-2, and ROUGE-L. Considering the brevity of the definitions, we present scores derived from the unigram overlap (BLEU-1). Additionally, BERTScore [65] is employed as a metric for semantic matching, utilizing vector-based contextual similarities determined by the cosine similarity between sentence embeddings of each generated definition with respect to a reference definition. The sentence embeddings were generated using the bert-base-multilingual-cased model.
We conducted a human evaluation with regard to a) *Fluency*, i.e. grammaticality and readability, b) *Relevance*, i.e. selection of important content from the source, c) *Factuality*, i.e. no invented facts/components/terms (hallucinations) d) *Coherence*, i.e. the collective quality of all sentences that should build a coherent body of information about a topic [28, 60].

Table 2: Support documents and generated definitions from various approaches

ID	Abstracts from Scopus
1	Coke oven gas is a by-product of the coke plant, which contains mainly hydro-carbons. The coke oven gas is lanced into the kiln through a number of pipes attached to the kiln. [..]
2	Coke oven gas is an alternative hydrogen-rich fuel for vehicles, the water vapor in it will result in corrosion and seal damage of the engine combustion chamber. The paper describes the method and principle of COG dehydration, through the analysis of gas dehydration technology, determines the program of coke oven gas dehydration, and implements it in the coke oven gas stations. [..]
3	Coke oven gas is a by-product of coke production for steelmaking and by volume typically consists of 55,60% hydrogen, 23,27% methane and impurities. An estimated 650 million tonnes of coke oven gas are produced worldwide, with up to 50% re-utilised within steelmaking. [..]
4	Coke oven gas is an important byroduct of the steel industry as it is used as an energy substitute for natural gas. However, coke gas contains impurities and must be treated before use. One of the steps of the treatment is the absorption process, wherein ammonia and hydrogen sulfide are removed. Considering the increasing use of coke oven gas and, in parallel, the strengthening of compliance with emission restrictions, it is extremely important to make the absorption process more efficient. This work consists of the development of a coke oven gas purification process model to evaluate solutions that would allow greater removal of hydrogen sulfide. [..]

Approach	Gold Definition and Automatically Generated Definitions
Gold:	Coke oven gas (COG) is a byproduct of the coking process [3] used in steel production [3,4], where coal is heated in an oxygen-free environment to extract metallurgical coke, used as a fuel [2] and as a reducing agent in steelmaking [3,4].
BART:	The coke oven gas is a gas that is used in the production of electricity. It is also known as a coke-fired gas. It can also be used as a fuel [2].
GPT4:	Coke oven gas (COG) is a byproduct of the coking process used in steel production [3], where coal is heated in an oxygen-free environment to extract metallurgical coke, used as a fuel [2] and as a reducing agent in steelmaking. This gas is comprised of a mixture of various gases including hydrogen (H2), methane (CH4), carbon monoxide (CO), carbon dioxide (CO2), and nitrogen (N2), along with a range of volatile hydrocarbons [1]. Coke oven gas is often used as a fuel in steel plants, but it requires cleaning to remove impurities like tar, ammonia [4], and sulfur compounds before utilization to prevent pollution and ensure safety.
GPT4+H+T:	Coke oven gas is **a blast furnace gas (hypernym)** produced as a by-product of coke production in steelmaking, primarily composed of hydrogen and methane, and used as an alternative hydrogen-rich fuel and energy substitute for natural gas after purification.
CTRLsum+Q:	Coke oven gas is a by-product of the coke plant [1], containing mainly hydro-carbons [1].
CTRLsum+Q+H+T:	Coke oven gas is an important byroduct of the steel industry [1] as it is used as an energy substitute for natural gas [4].
GPTsum+Q:	Coke oven gas is a by-product of the coke production process [3] used in steelmaking [3,4], primarily composed of hydrogen, methane, and various impurities [3]. It serves as an alternative fuel [2] and requires treatment such as dehydration [2] and purification [4] to remove contaminants like ammonia and hydrogen sulfide before use [4].
GPTsum+Q+H+T:	Coke oven gas (species) is a **blast furnace gas (hypernym)** produced as a by-product of coke production in steelmaking [3,4], primarily composed of hydrogen and methane [3], and used as an alternative hydrogen-rich fuel [2] and energy substitute for natural gas [4] after purification.

Top Hypernyms: substance (0.24), thermal process (0.29), fuel cell-electrode-electrolyte unit (0.31), chemical compound (0.36), **blast furnace gas (0.47)**
Topic class: **Energy Supply**

We randomly selected 100 definitions from EnArgus for which definitions were generated with our proposed models.

Qualitative examination of the results is necessary due to the limitations of automatic measures [30]. Even though BERTScore has been shown to be more robust and correlate better with human evaluation than the metrics BLEU and ROUGE, which are good at evaluating fluency but punish length differences [11], automatic evaluation metrics are not capable of capturing features such as *factuality, coherence* and *relevance of content* on this task [10, 30]. Further, we employ six annotators for human evaluation (native speakers, domain experts) who judge the models' output on a Likert scale (evaluation score on a scale of 1 (worst) to 5 (best)).

7.2 Baselines and comparing systems

Evaluation - Topic Classification. To measure classification performance, we use standard performance metrics (e.g., accuracy, F1). Table 3 presents a summary of the performance achieved by our models, showing that the fine-tuned BERT models[12] outperform GPT-4 on the topic classification task (accuracy: 0.81 English, 0.79 German). As can be seen, GPT-4 only achieves an accuracy of 0.55 and 0.56 for the best prompting strategy in a few-shot setting for this task in German and English, respectively.

Evaluation - Hypernym Prediction. In this work, we exploit how well semantic relatedness is captured for EnArgus terms in state-of-the-art embeddings, i.e. MiniLM-L12, MiniLM-L12-multilingual MPNet-multilingual, and GPT-4 with a proper prompting strategy

[12]https://huggingface.co/dbmdz/bert-base-german-cased and https://github.com/allenai/scibert

Table 3: Classification Performance on EnArgus testset

Model	Language	f1	p	r	accuracy
BERT	de	0.78	0.80	0.78	**0.79**
GPT-4	de	0.56	0.59	0.56	0.56
SciBERT	en	0.81	0.82	0.81	**0.81**
GPT-4	en	0.54	0.55	0.58	0.55

(see 6.1). The results have been measured against the manually built EnArgus ontology as ground truth dataset that captures subclass information for all EnArgus terms. Both Pearson rank coefficients and Spearman rank coefficients are calculated, where semantic similarity for term-hypernym pairs is expected to be high. Moreover, we compute top-k (k=10, 25) semantic neighbors for EnArgus terms and calculate mean average precision.

We report the best performing models in Table 4. We found that rare domain-specific terms are not sufficiently represented in open-source contextual embeddings, as has been observed before [41]. Even though *all-MiniLM-L12-v2*[13] and *multilingual-e5-large-instruct*[14] show a strong positive correlation for term-(non)hypernym pairs for both English and German, they perform poor in terms of MAP. However, EnArgus terms and hypernyms are surprisingly well captured in GPT-4 (MAP@10: 96.5 English, 94.5 German).

We noticed that our models often failed to predict the correct hypernym for multi-word terms (e.g. *bio-waste*), while the lexical head already reveals the correct subclass (e.g. *waste*). It is well known that subword tokenization, such as Byte Pair Encoding [56], used by large language models such as GPT as a preprocessing step, do not always lead to a valid morphological decomposition [6].

Table 4: Classification Performance on EnArgus testset.

Model	Lang	Spear	Pears	MAP@10	MAP@25
m-e5-large	de	0.93	0.99	14.7	16.8
all-MiniLM	de	0.93	0.72	11.6	22.1
GPT-4	de	0.89	0.39	**94.5**	**96.5**
m-e5-large	en	0.96	0.98	19.2	51.5
all-MiniLM	en	0.94	0.97	24.2	56.6
GPT-4	en	0.92	0.38	**96.5**	**98.5**

Table 5: Natural Language Text Generation.

Model	Lang	BLEU	R-1	R-2	R-L	BERT
BART	de	0.05	0.13	0.15	0.04	0.66
GPT-4	de	0.14	0.26	0.08	0.21	0.74
BART	en	0.21	0.35	0.16	0.30	0.77
GPT-4	en	0.08	0.17	0.05	0.13	0.71
GPT4+H+T	en	0.17	0.34	0.11	0.27	0.74

[13]https://huggingface.co/sentence-transformers/all-MiniLM-L12-v2
[14]https://huggingface.co/intfloat/multilingual-e5-large-instruct

Table 6: Abstractive Summarization from Scopus Abstracts.

Model	BLEU	R-1	R-2	R-L	BERT
CTRLsum+Q	0.12	0.25	0.07	0.21	0.72
CTRLsum+Q+H+T	0.30	0.30	0.10	0.24	0.73
GPTsum+Q	0.25	0.35	0.13	0.28	0.76
GPTsum+Q+H+T	0.08	0.47	0.22	0.38	0.81

Evaluation - Retrieval. In our experiments, we used Hearst patterns rather than semantic search, since the occurrence of the query term in the abstract is a necessary condition for relevance. The number of retrieved *Scopus* abstracts per keyword query is high (mean: 168,299), but shrinks drastically when applying the Hearst filter (mean: 56.2). The number of non-relevant retrieved items (false positives) (e.g. *Ammonia is a well-known example* is relatively low, yet not all retrieved abstracts possess a concise definition (P@5: 0.87). For 18% of EnArgus terms we could not retrieve any valid definition.

8 Results and analysis

We present the results of automatic evaluation of the model generated definitions, comparing the selected baselines with advanced systems.

8.1 Quantitative analysis - Evaluation of Definition Generation

Table 2 (and Table A.2 in the Appendix) show examples of generated definitions from different models. We manually annotated factual evidence from *Scopus* in the generated text using inline citations. Evaluation scores for the generated output definitions for baseline models are presented in Table 5.

We report BLEU, ROUGE-1, ROUGE-2, ROUGE-L, BertScore F1 on our dataset, evaluated on the EnArgus dataset with GPT-4 and fine-tuned BART.

As can be seen, the English BART model achieved the highest BERTScores, followed by the German GPT-4 model. Regarding multilinguality, we observe that the German BART model performs worse than the English BART model. These results suggest that a 2-stage process, i.e. natural language generation (NLG) based on a monolingual English model followed by translation of NLG results into German (i.e., generate-translate approach), might be superior to the monolingual German approach, despite the noise produced by automatic machine translation.

Table 6 presents the results for the CTRLsum and the GPT4sum models, evaluated on the EnArgus test set, guided with prompts and/or keywords. Compared to GPT-4 in the vanilla summarization setting, retrieval augmented GPT-4 models demonstrate improved performances across all automatic evaluation metrics, ROUGE, BLEU and BERTScores (e.g., 0.27 versus 0.38 on R-L). This indicates the superiority in controlled settings with support abstracts (GPTsum+Q) and keywords (GPTsum+Q+H+T).

Moreover, GPT4sum+Q+H+T outperforms CTRLsum+Q+H+T by a large margin. In contrast to their counterparts that use only a prompt for guidance, i.e. GPT4sum+Q and CTRLsum+Q, they show better quantitative results. Comparing the output of CTRLsum+Q

to CTRLsum+Q+H+T manually, we noticed that keyword guidance could not further enhance the quality of the generated definitions. In fact, our results indicate that CTRLsum+Q+H+T has only limited controllability for keywords. We observe that less than 10% of generated definitions included the selected keywords, as opposed to 94% of all generated definitions with GPTsum+Q+H+T.

8.2 Human Evaluation

To assess the quality of the models, six humans (domain-experts) were instructed to evaluate the automatically generated definitions from a random and blind sample, also comparing them to original EnArgus definitions. Our test set consists of 100 term-definition pairs, i.e. published in the respective dataset. All output texts were evaluated in German and English. The annotators were shown only the generated definition and not the original term definition. We calculated Krippendorff's alpha (Krippendorff, 2004) to compute data reliability which is estimated for 6 coders. The results imply small disagreements among raters (fluency = 0.16, reliablity = 0.15, factuality= 0.22, coherence = 0.23). The results of the human evaluation are reported in Table 7, given as mean average over the six evaluation scores. Interestingly, the blind test revealed that evaluators could not always distinguish expert-written definitions from automatically generated definitions, indicating that overall quality is quite high. Regarding multilinguality, evaluators confirmed the superiority of the English models (in contrast to German), in particular with respect to the pre-trained German BART model that generated only truncated non-sense output and obtained the lowest possible ratings.

We observe a small improvement on factuality for all variants of GPT4sum over the baseline models due to the model's ability to identify relevant pieces of information from the provided abstracts. While GPTsum models were able to summarize multiple abstracts into one coherent piece of text in an abstractive way, we noticed that CTRLsum was much more extractive and often chose information from the first provided abstract. However, we find that also GPT4sum models sometimes were not to the point and selected irrelevant details reported in abstracts, e.g. *A pedelec is a type of bicycle that is equipped with an electric motor to assist the rider's pedal force, providing power assistance based on the pedaling effort to enhance mobility and reduce physical strain.* Instead the output produced by GPT4+H+T is much more concise, e.g. *A pedelec is an electric vehicle (hypernym) equipped with an electric motor that assists the rider's pedaling, providing additional power to make cycling easier, especially on inclines (differentia).* Thus, support documents do not always result in higher factuality scores, partly, because the definition provided in the evidence snippets is not precise enough, and partly, because the model could not identify the definition part of the abstract despite guidance.

Table 7: Human Evaluation with respect to Fluency (Flu), Relevance/Completeness (Rel), Factuality (Fac), Coherence (Coh).

Model	Language	Flu	Com	Fac	Coh
BART	de	1.0	1.0	1.0	1.0
GPT4	de	3.6	4.3	4.4	4.4
Human	de	4,3	3.9	4.7	4.6
BART	en	4.3	3.7	4.4	4.1
GPT4	en	3.3	3.5	4.4	3.9
CTRLsum+Q	en	3.5	2.6	2.8	3.2
CTRLsum+Q+H+T	en	3.2	2.7	3.6	3.9
GPTsum+Q	en	4.5	3.6	4.5	4.2
GPTsum+Q+H+T	en	4.5	3.6	4.6	4.2
Human	en	4,3	3.9	4.7	4.5

9 Conclusion

Our work systematically studies the performance of various state-of-the art LLMs for generating scientific definitions in English and German. We propose a novel approach for the task, following the retrieval-augmented generation paradigm for summarizing top-ranked abstracts retrieved from *Scopus*. Furthermore, we enhance summarization by means of guiding signals, leveraging a domain-specific ontology for hypernym prediction and using ML classifiers for topic classification. Overall, our approach outperforms supervised models like a fine-tuned BART model or the GPT-4 baseline model on key facets such as fluency, relevancy, comprehensiveness and factuality. However, augmenting GPT-4 with irrelevant information in some cases led to overriding correct knowledge already possessed knowledge.

Our work reveals new insights for dealing with the task in German. In particular, the German BART model still cannot compete with its English counterpart, and lacks native resources for augmenting and guiding the model. In future work, we like to extend the definition generation task to produce more comprehensive definitions, including e.g. application fields, and develop and evaluate methods for definition generation with citations. Thus, for individual facts there should be a reference to the corresponding knowledge sources, so that scientific claims can be more easily verified.

Limitations

With regard to the human evaluation, it should be noted that the human evaluators, though domain experts, are not necessarily equally familiar with all terms in the broad range of definitions which may result in heterogeneous evaluation quality across the different criteria, especially with regard to factuality. It should also be noted that only the introductory definition parts were used, i.e. the first 1-3 sentences of the more comprehensive EnArgus article). Further investigations are needed to verify how the models perform for long definitions instead of the core definitions only.

In particular for novel energy terms, we observe low frequency in *Scopus* abstracts. In some cases, we were not able to extract abstracts with definitional sentences. In theses cases, we eliminated the term from the test set. While *Scopus* is regarded as a high-quality scholarly database which contains millions of articles covering various disciplines, the reliability and accuracy of individual abstracts vary significantly. Existing studies have shown that retrieved incorrect information might harm the model's generation process [43, 61]. While our study is limited to terms from the energy domain, it can be applied to any other domain.

Acknowledgments

We gratefully acknowledge the support of the German Federal Ministry for Economic Affairs and Climate Action (BMWK) for the EnArgus 3.0 project, grant agreement number 03EI1062A. The responsibility of this publication lies with the authors.

References

[1] Dzhal Antonov and Davide Buscaldi. 2023. Assessing the Impact of Word Embeddings for Relation Prediction: An Empirical Study. In *Joint Proceedings of the Second International Workshop on Knowledge Graph Generation From Text and the First International BiKE Challenge co-located with 20th Extended Semantic Conference (ESWC 2023), Hersonissos, Greece, May 29th,2023 (CEUR Workshop Proceedings, Vol. 3447)*. CEUR-WS.org, 192–204.

[2] Marianna Apidianaki. 2023. From word types to tokens and back: A survey of approaches to word meaning representation and interpretation. *Computational Linguistics* 49, 2 (2023), 465–523.

[3] Maurizio Atzori and Simone Balloccu. 2020. Fully-unsupervised embeddings-based hypernym discovery. *Information* 11, 5 (2020), 268.

[4] Tal August, Katharina Reinecke, and Noah A Smith. 2022. Generating scientific definitions with controllable complexity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 8298–8317.

[5] Vidhisha Balachandran, Dheeraj Rajagopal, Rose Catherine Kanjirathinkal, and William Cohen. 2018. Learning to define terms in the software domain. In *Proceedings of the 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text*. 164–172.

[6] Khuyagbaatar Batsuren, Ekaterina Vylomova, Verna Dankers, Tsetsuukhei Delgerbaatar, Omri Uzan, Yuval Pinter, and Gábor Bella. 2024. Evaluating Subword Tokenization: Alien Subword Composition and OOV Generalization Challenge. *arXiv preprint arXiv:2404.13292* (2024).

[7] Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676* (2019).

[8] Francisco Bolaños, Angelo Salatino, Francesco Osborne, and Enrico Motta. 2024. Artificial Intelligence for Literature Reviews: Opportunities and Challenges. https://doi.org/10.48550/arXiv.2402.08565

[9] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[10] Meng Cao, Yue Dong, Jiapeng Wu, and Jackie Chi Kit Cheung. 2020. Factual error correction for abstractive summarization models. *arXiv preprint arXiv:2010.08712* (2020).

[11] Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2020. Evaluation of text generation: A survey. *arXiv preprint arXiv:2006.14799* (2020).

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[13] Zi-Yi Dou, Pengfei Liu, Hiroaki Hayashi, Zhengbao Jiang, and Graham Neubig. 2020. GSum: A general framework for guided neural abstractive summarization. *arXiv preprint arXiv:2010.08014* (2020).

[14] Liana Ermakova, Eric SanJuan, Stéphane Huet, Olivier Augereau, Hosein Azarbonyad, and Jaap Kamps. 2023. CLEF 2023 SimpleText Track: What Happens if General Users Search Scientific Texts?. In *European Conference on Information Retrieval*. Springer, 536–545.

[15] Angela Fan and Claire Gardent. 2022. Generating full length wikipedia biographies: The impact of gender bias on the retrieval-based generation of women biographies. *arXiv preprint arXiv:2204.05879* (2022).

[16] Angela Fan, David Grangier, and Michael Auli. 2017. Controllable abstractive summarization. *arXiv preprint arXiv:1711.05217* (2017).

[17] John Flowerdew. 1992. Definitions in science lectures. *Applied linguistics* 13, 2 (1992), 202–221.

[18] Noah Gardner, Hafiz Khan, and Chih-Cheng Hung. 2022. Definition modeling: literature review and dataset analysis. *Applied Computing and Intelligence* 2, 1 (2022), 83–98.

[19] Tanya Goyal, Junyi Jessy Li, and Greg Durrett. 2022. News Summarization and Evaluation in the Era of GPT-3. *ArXiv* abs/2209.12356 (2022). https://api.semanticscholar.org/CorpusID:252532176

[20] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*. PMLR, 3929–3938.

[21] Junxian He, Wojciech Kryscinski, Bryan McCann, Nazneen Rajani, and Caiming Xiong. 2022. CTRLsum: Towards Generic Controllable Text Summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (Eds.). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 5879–5915. https://doi.org/10.18653/v1/2022.emnlp-main.396

[22] Marti A. Hearst. 1992. Automatic Acquisition of Hyponyms from Large Text Corpora. In *COLING 1992 Volume 2: The 14th International Conference on Computational Linguistics*. https://aclanthology.org/C92-2082

[23] S Hirzel, P Plötz, C Rohde, B Teufel, L Oppermann, K Heiwolt, R Ruland, J Krassowski, C Beier, L Sikorski, et al. 2017. What's going on in energy efficiency research? A platform to enhance the transparency of energy research funding in Germany. In *European Council for an Energy-Efficient Economy (ECEEE Summer Study), 2017*. https://doi.org/handle/publica/396914

[24] Han Huang, Tomoyuki Kajiwara, and Yuki Arase. 2021. Definition modelling for appropriate specificity. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 2499–2509.

[25] Jie Huang, Hanyin Shao, and Kevin Chen-Chuan Chang. 2021. CDM: Combining Extraction and Generation for Definition Modeling. *ArXiv* abs/2111.07267 (2021). https://api.semanticscholar.org/CorpusID:244117000

[26] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *Comput. Surveys* 55, 12 (2023), 1–38.

[27] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, 6769–6781. https://doi.org/10.18653/v1/2020.emnlp-main.550

[28] Huan Yee Koh, Jiaxin Ju, Ming Liu, and Shirui Pan. 2022. An empirical survey on long document summarization: Datasets, models, and metrics. *ACM computing surveys* 55, 8 (2022), 1–35.

[29] Rania Kora and Ammar Mohammed. 2023. A Comprehensive Review on Transformers Models For Text Classification. In *2023 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*. IEEE, 1–7.

[30] Wojciech Kryściński, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Evaluating the factual consistency of abstractive text summarization. *arXiv preprint arXiv:1910.12840* (2019).

[31] Sayali Kulkarni, Sheide Chammas, Wan Zhu, Fei Sha, and Eugene Ie. 2020. Aquamuse: Automatically generating datasets for query-based multi-document summarization. *arXiv preprint arXiv:2010.12694* (2020).

[32] Md Tahmid Rahman Laskar, Enamul Hoque, and Jimmy Xiangji Huang. 2022. Domain adaptation with pre-trained transformers for query-focused abstractive text summarization. *Computational Linguistics* 48, 2 (2022), 279–320.

[33] Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691* (2021).

[34] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (Eds.). Association for Computational Linguistics, Online, 7871–7880. https://doi.org/10.18653/v1/2020.acl-main.703

[35] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems* 33 (2020), 9459–9474.

[36] Jiahuan Li, Yu Bao, Shujian Huang, Xinyu Dai, and Jiajun Chen. 2020. Explicit semantic decomposition for definition generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 708–717.

[37] Qian Li, Hao Peng, Jianxin Li, Congying Xia, Renyu Yang, Lichao Sun, Philip S Yu, and Lifang He. 2022. A survey on text classification: From traditional to deep learning. *ACM Transactions on Intelligent Systems and Technology (TIST)* 13, 2 (2022), 1–41.

[38] Xiaoxi Li, Jiajie Jin, Yujia Zhou, Yuyao Zhang, Peitian Zhang, Yutao Zhu, and Zhicheng Dou. 2024. From matching to generation: A survey on generative information retrieval. *arXiv preprint arXiv:2404.14851* (2024).

[39] Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190* (2021).

[40] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*. 74–81.

[41] Chunhua Liu, Trevor Cohn, and Lea Frermann. 2023. Seeking Clozure: robust Hypernym extraction from BERT with anchored prompts. In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (* SEM 2023)*. 193–206.

[42] Yixin Liu, Alexander R Fabbri, Jiawen Chen, Yilun Zhao, Simeng Han, Shafiq Joty, Pengfei Liu, Dragomir Radev, Chien-Sheng Wu, and Arman Cohan. 2023. Benchmarking generation and evaluation capabilities of large language models for instruction controllable summarization. *arXiv preprint arXiv:2311.09184* (2023).

[43] Seiji Maekawa, Hayate Iso, Sairam Gurajada, and Nikita Bhutani. 2024. Retrieval Helps or Hurts? A Deeper Dive into the Efficacy of Retrieval Augmentation to Language Models. *arXiv preprint arXiv:2402.13492* (2024).

[44] Timothee Mickus, Kees Van Deemter, Mathieu Constant, and Denis Paperno. 2022. Semeval-2022 Task 1: CODWOE–Comparing Dictionaries and Word Embeddings. *arXiv preprint arXiv:2205.13858* (2022).
[45] Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2022. MTEB: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316* (2022).
[46] Smaranda Muresan and Judith L Klavans. 2013. Inducing terminologies from text: A case study for the consumer health domain. *Journal of the American Society for Information Science and Technology* 64, 4 (2013), 727–744.
[47] Thanapon Noraset, Chen Liang, Larry Birnbaum, and Doug Downey. 2017. Definition modeling: Learning to define word embeddings in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 31.
[48] Leif Oppermann, Simon Hirzel, Alexander Güldner, Karoline Heiwolt, Joachim Krassowski, Ulrich Schade, Christoph Lange, and Wolfgang Prinz. 2021. Finding and analysing energy research funding data: The EnArgus system. *Energy and AI* 5 (2021), 100070. https://doi.org/10.1016/j.egyai.2021.100070
[49] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*. 311–318.
[50] Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066* (2019).
[51] James Pustejovsky. 2012. *Semantics and the Lexicon*. Vol. 49. Springer Science & Business Media.
[52] Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *ArXiv* abs/1910.10683 (2019). https://api.semanticscholar.org/CorpusID:204838007
[53] Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics* 11 (2023), 1316–1331.
[54] Stephen Roller, Douwe Kiela, and Maximilian Nickel. 2018. Hearst patterns revisited: Automatic hypernym detection from large text corpora. *arXiv preprint arXiv:1806.03191* (2018).
[55] Tara Safavi and Danai Koutra. 2021. Relational World Knowledge Representation in Contextual Language Models: A Review. arXiv:2104.05837 [cs.CL] https://arxiv.org/abs/2104.05837
[56] Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural Machine Translation of Rare Words with Subword Units. *ArXiv* abs/1508.07909 (2015). https://api.semanticscholar.org/CorpusID:1114678
[57] Lukas Sikorski, Michael Dembach, and Rudolf Ruland. 2015. EnArgus - Ontology Based Search. In *Joint Proceedings of the Posters and Demos Track of 11th International Conference on Semantic Systems - SEMANTiCS 2015 and 1st Workshop on Data Science: Methods, Technology and Applications (DSci15) 11th International Conference on Semantic Systems - SEMANTiCS 2015, Vienna, Austria, September 15-17, 2015 (CEUR Workshop Proceedings, Vol. 1481)*, Agata Filipowska, Ruben Verborgh, and Axel Polleres (Eds.). CEUR-WS.org, 5–7. https://ceur-ws.org/Vol-1481/paper2.pdf
[58] Ashok Urlana, Pruthwik Mishra, Tathagato Roy, and Rahul Mishra. 2023. Controllable Text Summarization: Unraveling Challenges, Approaches, and Prospects–A Survey. *arXiv preprint arXiv:2311.09212* (2023).
[59] Jesse Vig, Alexander R Fabbri, Wojciech Kryściński, Chien-Sheng Wu, and Wenhao Liu. 2021. Exploring neural models for query-focused summarization. *arXiv preprint arXiv:2112.07637* (2021).
[60] Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020. Asking and answering questions to evaluate the factual consistency of summaries. *arXiv preprint arXiv:2004.04228* (2020).
[61] Yile Wang, Peng Li, Maosong Sun, and Yang Liu. 2023. Self-knowledge guided retrieval augmentation for large language models. *arXiv preprint arXiv:2310.05002* (2023).
[62] Jing Xu, Arthur Szlam, and Jason Weston. 2021. Beyond goldfish memory: Long-term open-domain conversation. *arXiv preprint arXiv:2107.07567* (2021).
[63] Geonil Yun, Yongjae Lee, A-Seong Moon, and Jaesung Lee. 2023. Hypert: hypernymy-aware BERT with Hearst pattern exploitation for hypernym discovery. *Journal of Big Data* 10, 1 (2023), 141.
[64] Hanqing Zhang, Haolin Song, Shaoyu Li, Ming Zhou, and Dawei Song. 2023. A survey of controllable text generation using transformer-based pre-trained language models. *Comput. Surveys* 56, 3 (2023), 1–37.
[65] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675* (2019).
[66] Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. 2023. Siren's song in the AI ocean: a survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219* (2023).
[67] Yusen Zhang, Yang Liu, Ziyi Yang, Yuwei Fang, Yulong Chen, Dragomir Radev, Chenguang Zhu, Michael Zeng, and Rui Zhang. 2023. MACSum: Controllable Summarization with Mixed Attributes. *Transactions of the Association for Computational Linguistics* 11 (2023), 787–803. https://doi.org/10.1162/tacl_a_00575

A Research Methods

A.1 Hyperparameters

In case of GPT4-turbo, we use a zero-shot setting for generating the definition of the term, if not stated otherwise.
We fine-tuned the BART-large (provided by Facebook AI on HuggingFace) on our evaluation dataset using Adam as an optimiser with a constant learning rate of 0.0003 and a batch size of 16. Fine-tuning was terminated when the value of cross-entropy loss measured on a validation set stopped decreasing for 3 continuous epochs. For testing, min length was set to 20 and max length to 40 tokens.

A.2 Evaluation Example 'coke oven gas'

Scopus Abstracts for "coke oven gas". EID - ABSTRACT
2-s2.0-84883420227 - "Kiln of a lime plant, a part of an integrated steel plant, converts limestone (CaCO3) into lime (CaO) by heating at a temperature of around 900C using coke oven gas as a fuel. Coke oven gas is a by-product of the coke plant, which contains mainly hydro-carbons. The coke oven gas is lanced into the kiln through a number of pipes attached to the kiln. The lance pipes are cracking prematurely within 6months of their service against an expected service life of 3years. The analysis of cracking of the lance pipe has been presented. The investigation includes visual observation, chemical analysis, characterization of microstructures using optical and scanning electron microscopes, EDS analysis, and measurement of hardness. The lances are found to be made of AISI 302 grade of austenitic stainless steel. Visual observation shows longitudinal cracks on the pipe wall associated with irregular yellowish surface having multiple pits. Analysis of the deposit on the pipe wall shows the presence of sulphur. Microstructural examination shows intergranular corrosion with thick porous layer of scale on the surface. Microstructural and EDS analysis indicate sulphidation and chromium carbide networks along the grain boundary as well as sigma phases within the grains. Sulphidation, sensitization due to grain boundary carbides and sigma phases led to intergranular corrosion causing cracking of the pipe wall."
2-s2.0-84886279061 - "Coke oven gas is an alternative hydrogen-rich fuel for vehicles, the water vapor in it will result in corrosion and seal damage of the engine combustion chamber. The paper describes the method and principle of COG dehydration, through the analysis of gas dehydration technology, determines the program of coke oven gas dehydration, and implements it in the coke oven gas stations. The result shows the coke oven gas after dehydration meets automotive requirements."
2-s2.0-85091220151 - "Coke oven gas is a by-product of coke production for steelmaking and by volume typically consists of 55,60% hydrogen, 23,27% methane and impurities. An estimated 650 million tonnes of coke oven gas are produced worldwide, with up to 50% re-utilised within steelmaking. However, the rest is flared, contributing to carbon emissions and wasting valuable and useful gases. This study has investigated the co-electrolysis of simulated coke oven gas with steam using commercially available solid

oxide electrolysis technology for the purposes of recovering hydrogen. The electrochemical performance of an anode supported button cell was characterised using open circuit potential measurements, current-voltage curves and electrochemical impedance spectroscopy. The product gas composition was analysed using quadrupole mass spectrometry. Co-electrolysis of simulated coke oven gas (30/70% methane/hydrogen) with 50% steam achieved a hydrogen amplification of 119% and a purity of 91.7% by volume, balanced mainly in carbon dioxide and carbon monoxide. Theoretically, this corresponds to a worldwide hydrogen production from coke oven gas of 87.6 million tonnes, which is in excess of the current global demand for hydrogen (70 million tonnes). Catalytic steam reforming of methane and the water-gas shift reaction increased the hydrogen content by 89% and a further 16% gain was due to electrochemical steam reduction. Co-electrolysing at high steam-to-carbon ratios was shown to increase hydrogen yield, improve cell performance, maximise methane and carbon monoxide conversion and inhibit carbon deposition."

2-s2.0-85092938633 - "Coke oven gas is an important byroduct of the steel industry as it is used as an energy substitute for natural gas. However, coke gas contains impurities and must be treated before use. One of the steps of the treatment is the absorption process, wherein ammonia and hydrogen sulfide are removed. Considering the increasing use of coke oven gas and, in parallel, the strengthening of compliance with emission restrictions, it is extremely important to make the absorption process more efficient. This work consists of the development of a coke oven gas purification process model to evaluate solutions that would allow greater removal of hydrogen sulfide. The developed model was validated with data from an industrial plant, with errors of less than 6% for the most relevant variables. Among the three configurations tested, the best configuration represented a 5% increase in removal efficiency. This result is in line with scientific efforts in the search for environmentally responsible solutions. In addition, the improvement in the performance of this process allows for the use of coal with a higher sulfur content (responsible for generating hydrogen sulfide) at processes upstream of coke oven gas purification. Since coal with a higher sulfur content is cheaper and abundant, unlike low sulfur coal, an economic analysis was developed that allowed the financial impact of this modification to be quantified. The proposed modification resulted in a reduction of 15% of raw materials costs."