Towards Enabling FAIR Dataspaces Using Large Language Models

Benedikt T. Arnold[1,2], Johannes Theissen-Lipp[1,2], Diego Collarana[1,3], Christoph Lange[1,2], Sandra Geisler[1,2], Edward Curry[4] and Stefan Decker[1,2]

[1] *Fraunhofer Institute for Applied Information Technology FIT, St. Augustin, Germany*
[2] *RWTH Aachen University, Aachen, Germany*
[3] *Universidad Privada Boliviana, Bolivia*
[4] *University of Galway, Galway, Ireland*

Abstract

Dataspaces have recently gained adoption across various sectors, including traditionally less digitized domains such as culture. Leveraging Semantic Web technologies helps to make dataspaces FAIR, but their complexity poses a significant challenge to the adoption of dataspaces and increases their cost. The advent of Large Language Models (LLMs) raises the question of how these models can support the adoption of FAIR dataspaces. In this work, we demonstrate the potential of LLMs in dataspaces with a concrete example. We also derive a research agenda for exploring this emerging field.

Keywords
Dataspaces, FAIR Data Principles, Large Language Models

1. Introduction

In recent years, many dataspace verticalizations have emerged from European dataspace initiatives such as the *International Data Spaces (IDS)* [1] and *Gaia-X*[1] in different domains. Dataspaces are now also being established in domains with less affinity to digitization, such as culture. The strong grounding in Semantic Web technology (in the following also: semantics) promotes the realization of the *FAIR* principles in dataspaces. However, this raises the need for adopters of dataspaces to familiarize themselves with these technologies. This is a significant entry barrier and incurs costs for learning semantics or outsourcing the related tasks. The recent advent and broad adoption of *Large Language Models (LLMs)* raises the question of how this technology can be sensibly used to ease the adoption of FAIR dataspaces at a lower cost. In this work, we demonstrate the potential of LLMs for dataspaces with an example in section 2. In section 3, we propose a research agenda for investigating LLMs in FAIR dataspaces. Before, we introduce some important concepts:

The Second International Workshop on Semantics in Dataspaces, co-located with the Extended Semantic Web Conference, May 26 – 27, 2024, Hersonissos, Greece
benedikt.arnold@fit.fraunhofer.de (B. T. Arnold)
0000-0001-8594-880X (B. T. Arnold); 0000-0002-2639-1949 (J. Theissen-Lipp); 0000-0002-2583-0778 (D. Collarana); 0000-0001-9879-3827 (C. Lange); 0000-0002-8970-6282 (S. Geisler); 0000-0001-8236-6433 (E. Curry); 0000-0001-6324-7164 (S. Decker)

[1] https://gaia-x.eu/

Dataspaces The term *dataspace* was coined in 2005 and has evolved since then, with scientific definitions gathered in [2]. In this work, we refer to dataspaces as *a multi-sided data platform* connecting participants in an ecosystem [3]. Importantly, dataspaces do not incorporate an integration layer to bridge heterogeneity; instead, each data source remains unaltered, eliminating the need to transfer data to a centralized storage location. This omission of the integration aspect in data exchange reduces the initial workload and concentrates on an *as needed* best-effort strategy for data integration [4]. Significantly, the decentralized storage concept benefits dataspace participants as they maintain sovereignty over their data [5]. Dataspaces commonly use semantics to ensure a "common language" and foster FAIR data sharing [2].

FAIR Principles The FAIR Data Principles[2] aim at supporting the *Findability*, *Accessibility*, *Interoperability*, and *Reusability* of data and have gained widespread adoption. FAIR comprises these technical prerequisites: persistent identifiers (PIDs), rich metadata and open protocols – of which PIDs and rich metadata are closely linked with semantics [2]. In this paper, we highlight some of the most complicated and tedious associated tasks, which are: Enhancing existing semantic metadata schemas, creating instances from these schemas and understanding semantic data. We see a high potential to tackle these tasks more efficiently using LLMs – fostering the incorporation of the FAIR principles if done correctly.

Large Language Models Generative AI, in the form of Large Language Models (LLMs), represents a fundamental shift in text processing and knowledge generation methodologies. LLMs have shown excellent capabilities in understanding and generating human-like text, leading to breakthroughs in various downstream applications, including text summarization, content generation, and conversational systems [6]. LLMs such as GPT-4 [7] or Mixtral 8x7B[3] predict the next tokens in an auto-regressive manner, given an input sequence (prompt). They are first trained on a massive and diverse dataset in an unsupervised manner to produce base models and subsequently fine-tuned on a smaller and more specific dataset, aiming at performance improvement in specific tasks, like instruction-following [6].

2. Tasks in FAIR Dataspaces and the Potential of LLMs

With a practical example, we demonstrate how tasks related to providing and consuming FAIR data via dataspaces can be aided by LLMs, in here GPT-4. Following this proof-of-concept, we outline paths to solve these tasks more sophisticatedly for more realistic cases in section 3.

Rich metadata benefits especially *Findability* and *Interoperability*. This can be achieved using semantics (cf. section 1). For simplicity, consider the specification of an offered dataset in listing 1. In the listing, two *SHACL (Shapes Constraint Language)*[4] shapes ensure that each dataset has a title and a usage policy. The policy is needed to make the dataset a valid dataspace offering for other participants to conclude a usage contract. Our example considers the cultural domain and how digitized paintings can be offered. A relevant ontology in the cultural domain is

[2]https://www.go-fair.org/fair-principles/
[3]https://mistral.ai/news/mixtral-of-experts/
[4]https://www.w3.org/TR/shacl/

the *Integrated Authority File (GND)*, associated with a *Knowledge Graph (KG)* that represents "the largest collection of cultural and research authority data in the German-speaking countries"[5]. We start our example with a museum curator who wants to offer her data in the dataspace and is the first to do so from the art domain. Therefore, they must extend the existing metadata schema with corresponding properties. The curator is familiar with domain-specific metadata standards but wants to focus on the concepts instead of syntactical subtleties of formats such as *Turtle*[6]. She knows that a painter can be represented as a painting's `gndo:firstArtist` that is a `gndo:DifferentiatedPerson`, which has the property `gndo:gndIdentifier` that provides the person's GND id. gndo denotes the GND namespace. Using this id is important as it can serve as a unique *Persistent Identifier* (PID) for this person and thus supports the FAIR principles. The property `gndo:dateOfProduction` gives the painting's creation date. With this *domain knowledge*, we employ the following prompt:

> **User:** As a Semantic Web expert, please extend the SHACL shapes given below such that I can also describe digital versions of paintings with them, using properties from the Integrated Authority File, the GND. For the painter, please use a 'gndo:firstArtist' that is a 'gndo:DifferentiatedPerson' which has its GND id as a property. For the year of creation, please use 'gndo:dateOfProduction'.
> `` ```turtle ... ``

Note that we have not included the exact identifier for the GND id in the prompt. The response of GPT-4, with surrounding explanatory comments omitted, are the shapes presented in listing 2 – importantly, the existing shapes were not harmed. The painter with their identifier and the date of production have been correctly added. However, the painter is also referenced with their `gndo:preferredNameForThePerson` (lines 17–22), which was not intended. Besides, the `gndo:dateOfProduction`'s datatype (line 33) is either an `rdfs:Literal` or `xsd:dateTime`. Hence, we ask GPT-4 to correct these mistakes using a subsequent prompt:

> **User:** Please do not include the preferred name of the painter, and consider that the dateOfProduction should be of type 'xsd:dateTime'.

The resulting shapes (listing 3) again remain consistent with their previous versions, and the issues are correctly resolved. To support people new to the metadata schema and semantics, GPT-4 can also be used to visualize it, e.g., in *UML*, using a text-based notation such as *PlantUML*[7].

Besides creating the initial schema, an important task is creating the instances based on the schema. We use the real example of Caspar David Friedrich's painting "Der Wanderer über dem Nebelmeer", which was painted around 1818, according to the GND[8]. For this, we can make use of the OpenAI API's *Function Calling* feature[9]. This allows GPT-4 to invoke externally defined functions in a two-turn interaction. We define a function to extract the id for a painter from the GND given the name with a *SPARQL* query. Based on the following prompt, we let GPT-4 create the instance, using the function if needed:

[5]https://gnd.network/Webs/gnd/EN/Home/home_node.html
[6]https://www.w3.org/TR/turtle/
[7]https://plantuml.com/
[8]https://d-nb.info/gnd/1069695750/about/lds.ttl
[9]https://platform.openai.com/docs/guides/function-calling?lang=python

User: You are given the following schema:
```turtle
... omitted for conciseness ...
```
Please create an instance of it for the painting "Der Wanderer über dem Nebelmeer" by Caspar David Friedrich, which was created in 1818. You may look up the GND ID of the artist if you need to.

First, the supplied function is invoked with the painter's name. Subsequently, the instance in listing 4 is generated, alongside a description of how GPT-4 resolved prompt ambiguities:

GPT-4: Please note that:
- I've used `de` language tag with the title to specify the language is German.
- For the `odrl:hasPolicy`, I created a placeholder URI (`http://example.org/policy/12345`) as the policy detail wasn't specified.
- The date of production `1818-01-01` is set to January 1st, 1818, as the exact date wasn't provided. In this example, the time component is set to midnight. Adjust the date according to specific requirements or available data.

We generate the usage policy based on the *Open Digital Rights Language (ODRL)*[10], following the approach presented by Mustafa et al. [8], who supplement GPT-4 with a sophisticated description of the key parts of the ODRL ontology. We provide the following prompt:

User: Given the following instance, create an ODRL policy that allows the dataset to be used within Germany until 2024-05-10. ... omitted for conciseness ...

Listing 5 shows the resulting usage policy, preserving the policy's *Internationalized Resource Identifier (IRI)* from listing 4.

Finally, we start a new conversation with GPT-4 and ask it to, given the SHACL shapes, explain the created instance, together with its associated usage policy and get this result:

GPT-4: This set of information is essentially a structured way to describe a dataset, in this case, related to a digital representation of a painting. Here's the breakdown:
1. Dataset Identification: The data snippet describes a dataset named "Der Wanderer über dem Nebelmeer" which is German for "The Wanderer above the Sea of Fog." This dataset is identified on the web at ex:DerWandererÜberDemNebelmeer.
2. Title: The title of the dataset is given as "Der Wanderer über dem Nebelmeer," which aligns with the actual painting's title.
3. Policy Association: The dataset is associated with a policy indicated by the URI <http://example.org/policy/12345>. This policy defines how the dataset can be used or accessed.
4. Artist Information:
   - The dataset specifies an artist associated with it, identified under the system gndo:firstArtist.
   - This artist is a "Differentiated Person" meaning they are uniquely identified, not just a name but a specific individual.
   - The artist has a unique identifier "118535889," which helps in uniquely identifying and retrieving information about the artist from a database or registry.
5. Date of Production: The dataset includes a production date, 1818-01-01T00:00:00 formatted in the standard

[10] https://www.w3.org/TR/odrl-model/

```
@prefix sh: <http://www.w3.org/ns/shacl#> .
@prefix dcat: <http://www.w3.org/ns/dcat#> .
@prefix dct: <http://purl.org/dc/terms/> .
@prefix odrl: <http://www.w3.org/ns/odrl/2/> .

[] a sh:NodeShape ;
  sh:targetClass dcat:Dataset ;
  sh:property [
    sh:path dct:title ;
    sh:maxCount 1 ; ↩ sh:minCount 1 ;
    sh:message "A dcat:Dataset must have exactly one dct:title." ;
  ] ;
  sh:property [
    sh:path odrl:hasPolicy ;
    sh:maxCount 1 ; ↩ sh:minCount 1 ;
    sh:message "A dcat:Dataset must have exactly one odrl:hasPolicy." ;
  ] .
```

Listing 1: The base SHACL shapes.

```
... omitted for conciseness ...

  sh:property [
    sh:path gndo:firstArtist ;
    sh:class gndo:DifferentiatedPerson ;
    sh:maxCount 1 ; ↩ sh:minCount 1 ;
    sh:message "A digital representation of a painting must have exactly one gndo:firstArtist." ;
    sh:node [
      a sh:PropertyShape ;
      sh:path gndo:gndIdentifier ;
      ... omitted for conciseness ...
    ] ;
  ] ;
  sh:property [
    sh:path gndo:dateOfProduction ;
    sh:maxCount 1 ; ↩ sh:minCount 1 ;
    sh:datatype xsd:dateTime ;
    sh:message "A digital representation of a painting must have exactly one gndo:dateOfProduction of type xsd:dateTime." ;
  ] .
```

Listing 3: The corrected shapes from listing 2.

```
@prefix sh: <http://www.w3.org/ns/shacl#> .
@prefix dcat: <http://www.w3.org/ns/dcat#> .
@prefix dct: <http://purl.org/dc/terms/> .
@prefix odrl: <http://www.w3.org/ns/odrl/2/> .
@prefix gndo: <http://d-nb.info/standards/elementset/gnd#> .

[] a sh:NodeShape ;
  sh:targetClass dcat:Dataset ;

  ... omitted for conciseness ...

  sh:property [
    sh:path gndo:firstArtist ;
    sh:class gndo:DifferentiatedPerson ;
    sh:maxCount 1 ; ↩ sh:minCount 1 ;
    sh:message "A digital representation of a painting must have exactly one gndo:firstArtist." ;
    sh:node [
      a sh:PropertyShape ;
      sh:path gndo:preferredNameForThePerson ;
      sh:maxCount 1 ; ↩ sh:minCount 1 ;
      sh:message "Each gndo:DifferentiatedPerson must have exactly one gndo:preferredNameForThePerson." ;
    ] ;
    sh:node [
      a sh:PropertyShape ;
      sh:path gndo:gndIdentifier ;
      sh:maxCount 1 ; ↩ sh:minCount 1 ;
      sh:message "Each gndo:DifferentiatedPerson must have exactly one gndo:gndIdentifier." ;
    ] ;
  ] ;
  sh:property [
    sh:path gndo:dateOfProduction ;
    sh:maxCount 1 ; ↩ sh:minCount 1 ;
    sh:datatype xsd:date ;
    sh:message "A digital representation of a painting must have exactly one gndo:dateOfProduction." ;
  ] .
```

Listing 2: The SHACL shapes with extensions related to paintings, by GPT-4.

```
@prefix dcat: <http://www.w3.org/ns/dcat#> .
@prefix dct: <http://purl.org/dc/terms/> .
@prefix odrl: <http://www.w3.org/ns/odrl/2/> .
@prefix gndo: <http://d-nb.info/standards/elementset/gnd#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix ex: <http://example.org/data/> .

ex:DerWandererÜberDemNebelmeer a dcat:Dataset ;
  dct:title "Der Wanderer über dem Nebelmeer"@de ;
  odrl:hasPolicy <http://example.org/policy/12345> ;
  gndo:firstArtist [
    a gndo:DifferentiatedPerson ;
    gndo:gndIdentifier "118535889" ;
  ] ;
  gndo:dateOfProduction "1818-01-01T00:00:00"^^xsd:dateTime .
```

Listing 4: The instance as created by GPT-4 based on the description prompt and the GND lookup.

```
@prefix odrl: <http://www.w3.org/ns/odrl/2/> .
@prefix ex: <http://example.org/data/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

ex:policy12345 a odrl:Policy ;
  odrl:permission [
    a odrl:Permission ;
    odrl:action odrl:use ;
    odrl:target ex:DerWandererÜberDemNebelmeer ;
    odrl:constraint [
      a odrl:Constraint ;
      odrl:leftOperand odrl:spatial ;
      odrl:operator odrl:eq ;
      odrl:rightOperand "DE"^^xsd:string ;
    ] ,
    [
      a odrl:Constraint ;
      odrl:leftOperand odrl:dateTime ;
      odrl:operator odrl:lteq ;
      odrl:rightOperand "2024-05-10T23:59:59"^^xsd:dateTime ;
    ] ;
  ] .
```

Listing 5: The policy generated for the instance in listing 4. The IRI has been preserved.

date-time format. This likely refers to the creation date of the painting.

6. Policy Details:
   - The dataset is associated with a specific usage policy (ex:policy12345), which is an object defining permissions related to the dataset.
   - This policy allows the action of use on the dataset but has constraints:
     – It can only be used within Germany ("DE"), indicated by the spatial constraint.
     – It must be used before 2024-05-10T23:59:59, as indicated by the dateTime constraint.

This structured format helps machines and applications understand and process detailed metadata about digital resources, ensuring they're used and shared according to specified rules and providing essential context such as creation details and usage rights.

This is mostly a valid description of the instance. However, some important details get lost, including the reference to the source of the object's id, the GND.

In summary, we have shown that GPT-4 can help with some of the key tasks for preparing data for a FAIR exchange using dataspaces. The example was intentionally simple, and the LLM outputs had some issues. Still, it shows the general applicability of LLMs in this domain. In the next section, we propose a research agenda for using open LLMs in real dataspace settings, in combination with KGs to address some of the LLMs' intrinsic limitations.

3. Research Agenda

As Fernandez et al. [9] point out, LLMs have drastically changed the capabilities of *Natural Language Processing* systems as they can generate text and code of viable quality based on a natural language interface–with a better understanding of the underlying data's semantics. We are convinced that LLMs can aid many tasks in dataspaces, enhancing their adoption and FAIR-ness. In this section, we provide a research agenda, with research questions across multiple areas related to dataspaces that we consider pivotal for assessing how LLMs can benefit dataspaces.

Interactive and Automated Systems LLMs can be used in conversational settings and as prompt-based generators, raising the question of how assisting systems should be designed: interactive or as automated background services. We believe that this depends on the context.

One example of an automated system could be a dataspace metadata broker like the *Federated Catalogue* in Gaia-X. The stored *Self-Descriptions* in RDF can automatically be enhanced with an explanation by a specifically prompted LLM that can access the ontologies underlying the catalogue, see *Integration of Knowledge and Correctness*. Besides, mapping user data schemas to common standards can be automated using LLMs, fostering FAIRness [10].

In recent unpublished work, we use an interactive method to create Self-Descriptions with a natural language interface, leveraging conventional technology: Users provide attributes conversationally, and can ask questions. The system auto-corrects inputs using known facts and generates the self-description with standard RDF software, thus avoiding syntax errors.

For interactive approaches, it will be crucial to determine how the system has to be designed to be user-friendly. For automated approaches, ways have to be found to deal with unexpected output from the LLM, i.e., both hallucinations and unmatched output format requirements. For the latter, token sampling based on formal grammars [11] is an interesting research direction.

Adaptation: Prompt Engineering and Fine-Tuning When it comes to adapting a model to a new context like dataspaces and new tasks, two options are possible: The first technique is called *Prompt Engineering* and describes crafting specific prompts to obtain the desired output. Typically, there is additional context or example outputs given in the prompt that the LLM can use in addition to the knowledge encoded in its parameters. Another approach for adaption is *fine-tuning* where knowledge and desired output format are directly encoded into the parameters of the LLM, efficiently possible with adapter parameters [12]. For this, a dataset has to be carefully crafted, and enough data has to be available.

Both approaches differ cost- and time-wise. Fine-tuning incurs an initial additional cost for constructing the dataset and performing the resource-intense fine-tuning process. Prompt Engineering does not add cost in the beginning apart from building the prompt but requires more time during inference to process the sophisticated, context-enhanced prompts. Both approaches are worth investigating in the context of dataspaces: Few-shot prompting can be used to obtain the desired results when data scarcity or restrictive usage policies prohibit fine-tuning. However, fine-tuning is of particular interest in generating structured data, e.g., in JSON or Turtle, and can be required to make smaller LLMs adhere to the syntax.

Integration of Knowledge and Correctness Due to the intrinsic limitations of LLMs, i.e., hallucination, expensive updates, and lack of provenance, extensive research is carried out on interfacing them with KGs [13]. This is possible based on a more sophisticated variant of Prompt Engineering, *Retrieval Augmented Generation (RAG)* [6]. In RAG, the prompt is encoded using an embedding model and matched against the embeddings of knowledge in the form of documents saved in vector stores. The most similar results are then injected into the prompt. KGs can also be encoded into embeddings, making the knowledge stored in them accessible for LLMs. For outputs generated from injected KG data, the source part of the graph can be given as provenance. This way, users can even verify the results.

More broadly, Pan et al. [14] present a roadmap summarizing the possibilities for interfacing both technologies. The integration of KGs and LLMs is pivotal, especially in the context of FAIR dataspaces, as the metadata exchange in dataspaces is KG-based, especially in IDS and Gaia-X. Exploiting them as a source of knowledge can make LLMs more reliable for assisting dataspace tasks. Besides, open alternatives to OpenAI's function calling, like agent settings, are worth investigating for interfacing with data sources, e.g. for PID retrieval.

Open Models for Data Sovereignty In our example in section 2, we used the highly advanced but proprietary LLM GPT-4 by OpenAI to demonstrate general applicability. While valid for examples, two main reasons make using GPT-4 in dataspaces problematic: First, its proprietary nature creates a reliance on the user for the availability of OpenAI's API. Software built upon this API becomes immediately useless once availability is not given. As the effectiveness of prompts is often highly specific to the LLM at hand, there is a significant vendor lock-in. Besides, data sovereignty is one of the core selling points of dataspaces, i.e., the data stays in the participants' infrastructures and is exchanged peer-to-peer. This contradicts the widespread adoption of closed models like GPT-4 for dataspaces, which require sending sensitive data to a third-party service. For GDPR-relevant data, this can even be illegal. For these reasons, freely available LLMs should be preferred, although their performance does not yet match that of GPT-4. Multiple models with different strengths and vastly differing sizes have emerged from the small *Phi-2*[11] (2.7B param.) to large models like the aforementioned *Mixtral-8x7B* (46.7B param.). For this to be feasible, we propose to investigate the following research questions:

- Given an application domain and a specific LLM use case: What is a reasonable tradeoff between model size (i.e., inference cost), fine-tuning effort, model performance and safety?
- How can synergies between dataspace participants be used to fine-tune models?
- Where is using GPT-4 appropriate and its benefits outweigh the downsides?
- How can dataspace participants be empowered to perform inference on the edge with equal or less (personnel) cost compared to the hosted OpenAI API?

Efficiency And Latency Energy efficiency and latency are major aspects for self-hosted LLMs and out of control for closed models like GPT-4. Since the widespread interest in LLMs has grown, performance has drastically improved, e.g. with *QLoRA* [12] for fine-tuning. Quantization, i.e. representing the LLM parameters with fewer than the canonical 16 bits, strongly reduces the required amount of RAM and the inference latency, allowing for larger models on less powerful hardware, even CPUs, at often marginally reduced performance [15].

Subsequent research in the context of dataspaces will focus on how those scientific and engineering advances can be translated into dataspace-related tools. Specifically, we believe it is crucial to determine how the dataspace participants' acceptance of LLM-based tools is affected by latency, energy consumption, and the balance of latency and output quality.

Safety LLMs generate text based on their internal parameters, optimized on large amounts of text, cf. section 1. Therefore, the outputs reflect what the model has "seen" during training, leading to the adoption of biases in the used data–which is problematic for marginalized groups in particular, in-depth analyzed by Bender et al. [16]. This is especially critical because, for most LLMs, the underlying training data is not openly available for review.

It is pivotal for our research to keep safety in mind and be careful while developing and deploying LLMs for dataspace applications. As an overarching consideration for our research, awareness of this topic and the potential problems must be addressed. Unified data-model life-cycles are needed to ensure the provenance and trust in the data used to train LLMs [17]. Guardrails must be implemented to avoid problematic LLM generations being displayed to users or used in automatic systems. However, this must be balanced with preserving the system's core functionality. The integration of factual knowledge with provenance using KGs aims at this. To ensure safety, the actual dataspace participants must be involved during research, e.g., through user studies, to detect problematic cases. Besides ethical safety, measures have to be applied as well for securely handling input and output, especially when code is executed.

Acknowledgments

This work has been partially funded by the German Federal Ministry of Education and Research (BMBF) under grant number FAIRDS05, by the German Federal Ministry for Economic Affairs

[11] https://www.microsoft.com/en-us/research/blog/phi-2-the-surprising-power-of-small-language-models/

and Climate Action (BMWK) under grant number 01MJ22011B, by the German Federal Government Commissioner for Culture and the Media (BKM) under grant number 2522DIG012 and with financial support of Science Foundation Ireland (SFI) under grant number SFI/12/RC/2289_P2.

References

[1] S. Bader, J. Pullmann, C. Mader, S. Tramp, C. Quix, A. W. Müller, H. Akyürek, M. Böckmann, B. T. Imbusch, J. Lipp, S. Geisler, C. Lange, The International Data Spaces Information Model – An Ontology for Sovereign Exchange of Digital Content, in: ISWC, 2020.

[2] J. Theissen-Lipp, M. Kocher, C. Lange, S. Decker, A. Paulus, A. Pomp, E. Curry, Semantics in Dataspaces: Origin and Future Directions, WWW '23 Companion, 2023.

[3] B. Otto, M. Jarke, Designing a multi-sided data platform: findings from the International Data Spaces case, Electronic Markets 29 (2019).

[4] E. Curry, Future Research Directions for Dataspaces, Data Ecosystems, and Intelligent Systems, Springer International Publishing, Cham, 2020.

[5] S. Geisler, M.-E. Vidal, C. Cappiello, B. F. Lóscio, A. Gal, M. Jarke, M. Lenzerini, P. Missier, B. Otto, E. Paja, B. Pernici, J. Rehof, Knowledge-driven data ecosystems toward data transparency, Journal of Data and Information Quality 14 (2021).

[6] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, et al., Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, NeurIPS (2020).

[7] OpenAI, GPT-4 technical report, 2023. arXiv:2303.08774.

[8] D. M. Mustafa, B. T. Arnold, D. Collarana, A. Nadgeri, C. Quix, C. Lange, S. Decker, From Instructions to ODRL Usage Policies: An Ontology Guided Approach, under review (2024).

[9] R. C. Fernandez, A. J. Elmore, M. J. Franklin, S. Krishnan, C. Tan, How large language models will disrupt data management, Proceedings of the VLDB Endowment 16 (2023).

[10] S. S. Sundaram, M. A. Musen, Making Metadata More FAIR Using Large Language Models, 2023. arXiv:2307.13085.

[11] S. Geng, M. Josifoski, M. Peyrard, R. West, Grammar-constrained decoding for structured NLP tasks without finetuning, in: EMNLP, ACL, 2023.

[12] T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer, QLoRA: Efficient finetuning of quantized LLMs, in: NeurIPS, 2024.

[13] D. Collarana, M. Busch, C. Lange, Knowledge graph treatments for hallucinating large language models, ERCIM News 2024 (2024).

[14] S. Pan, L. Luo, Y. Wang, C. Chen, J. Wang, X. Wu, Unifying Large Language Models and Knowledge Graphs: A Roadmap, IEEE T-KDE (2024).

[15] H. Shen, H. Chang, B. Dong, Y. Luo, H. Meng, Efficient LLM Inference on CPUs, in: NeurIPS, 2023.

[16] E. M. Bender, T. Gebru, A. McMillan-Major, S. Shmitchell, On the dangers of stochastic parrots: Can language models be too big?, in: ACM FAcct, 2021.

[17] E. Curry, T. Zaarour, Y. Yang, M. Timilsina, M. Al-Qatf, R. Haque, Foundation Data Space Models: Bridging the Artificial Intelligence and Data Ecosystems (Vision Paper), in: 2023 IEEE International Conference on Big Data (BigData), IEEE, 2023.