Improving Accessibility and Reproducibility by Guiding Large Language Models

Florian Andreas Marwitz[1,2,*], Marcel Gehrke[1]

[1] Institute for Humanities-Centered Artifical Intelligence, University of Hamburg, Warburgstraße 28, 20354 Hamburg, Germany
[2] Centre for the Study of Manuscript Cultures (CSMC), University of Hamburg, Warburgstraße 26, 20354 Hamburg, Germany

Abstract

Research data repositories store numerous entries of research data, to among other advantages one goal is allowing to store us all data to reproduce experiments. Working with large corpora of texts is made significantly easier with chatbots using Large Language Models. However, Large Language Models are trained for general purposes and are not finetuned for the data originating from different kinds of projects. But the creators of such texts have an expert viewpoint on the data. Therefore, we propose to leverage the expert viewpoints of creators to obtain better answers from a chatbot. When creating an entry for the Research Data Repository, the creators have the possibility to add a so-called interpretation prompt. The interpretation prompt contains their expert viewpoint and be of any textual form to guide the Large Language Model to interpret the project-specific data. In particular, the interpretation prompt may contain instructions on how to reproduce experiments right inside the Large Language Model invocation. Afterward, the interpretation prompt is prepended to the query of the Large Language Model. In our examples, we show how the interpretation prompt helps to receive more tailored answers.

Keywords

Large Language Model, Research Data Management, Prompt Engineering, Reproducibility

1. Introduction

The goal of research data management is to keep any kind of research data accessible for other researchers. To that end, research data repositories (RDRs) aim at preserving files for a long period of time. From the viewpoint of a curious researcher, an RDR may contain a large number of entries in a variety of forms: Conference paper, spreadsheets, measurements, images, and so on. Finding the relevant entries for the own work is therefore challenging because of the vast amount and size of available data. The advent of Large Language Models (LLMs) has given rise to the integration of LLMs into RDRs for finding relevant works [1]. However, LLMs are trained for general purposes and understand the project-specific data in the RDR only in a generic way, while being perfectly capable of answering questions and comparing texts, images, or other types of media. However, the uploader of an RDR entry understands her uploaded data normally quite well and is an expert in her field of research. Thus, we propose that the uploader helps the general purpose LLM to provide more finetuned answers to the data at hand. The suggested interpretation can be thought of a prompt like "*The main contribution in this paper is Definition 4*" or a more vague one like "*This data is a time series measuring heartbeats. You can use an algorithm to classify heartbeat anomalies*". The uploader creates such an *interpretation prompt* during the creation of the RDR entry, bringing her expert viewpoint into textual form. Combining the interpretation prompt with the actual data from the RDR entry, a chatbot is now capable of giving a project-specific answer to the user in the way the uploader most likely intended it. Another concern for future usage of research data is that of reproducibility: Researchers publish their results such that others can reproduce these results. Since RDRs provide long-term access to data, RDRs are also used to provide data for reproducing results from research projects. The interpretation prompt can then be used

---

Large Language Models for Research Data Management?! (LLMs4RDM), Workshop at the Conference of the German Informatics Society, September 18, 2025, Potsdam, Germany

[*] Corresponding author.

florian.marwitz@uni-hamburg.de (F. A. Marwitz); marcel.gehrke@uni-hamburg.de (M. Gehrke)
0000-0002-9683-5250 (F. A. Marwitz); 0000-0001-9056-7673 (M. Gehrke)

© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR-WS.org/Vol-4140/paper3.pdf

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

published 2025-12-22

to tell the LLM what code or algorithm to execute and to compare the results. Using the interpretation prompt and the LLM, users can reproduce experiments without any technical effort.

Contribution We propose to include an *interpretation prompt* when uploading data into an RDR. LLMs using the RDR should include the interpretation prompt when answering questions about the corresponding entry. This way, the uploader uses her expert viewpoint to guide the LLM to understand the data better, making her work more accessible and citable. Moreover, the users of the RDR benefit from a service offering reproducibility and insightful answers instead of leaving the LLM to decide what is useful.

Related Work Wilkinson et al. have proposed the FAIR principles in 2016, requesting that data should be findable, accessible, interoperable, and reusable [2]. RDRs help to comply with the FAIR principles by providing the means to upload data and keeping that data accessible for a long period of time [3]. LLMs change the way we work with data [4]: We can ask the model a question in natural language and receive an answer in natural language. LLMs are trained on a very large corpus of documents, like websites, codes and books, among other things. After training, the LLM is ready to use, but lacks newer information, which was not included in the training corpus. Retrieval-augmented generation (RAG) tries to fix this issue by searching for sentences, which appear relevant for the query, in a specified set of documents [5]. The found sentences are then added to the query. Another approach is to finetune the model on another, task-specific training corpus [6]. We go a step further by allowing the creator of a document to specify which parts of it should be understood in which way. Our work focuses on empowering the creator of data to employ prompt engineering [7] to help future users to correctly interpret their work using LLMs. Prompt engineering itself is the art of writing the query to the LLM in a way to get better answers.

Structure The remainder of this paper is structured as follows: We start with describing the setting in which we operate and continue with our concept of guiding LLMs by interpretation prompts in more detail. Next, we showcase on examples how the interpretation prompt changes the answer of an LLM. Then, we discuss the interpretation prompt in the light of the examples. We end with a conclusion.

2. Prerequisites

RDRs store numerous entries of different types, like data sets, conference publications, or books among other things. Users can upload their data through a form, which asks to input metadata like authorships alongside the actual files. We call the collection of metadata and files an *entry* in the RDR. The uploader is the user who has created the entry. When viewing the entry, other users can see the title, description, authorship information, and the uploaded files. For some file types, even an online preview might be available. If the user finds (parts of) the entry interesting, she might work with some files. For our paper, we assume that the RDR is equipped with some kind of LLM and that this LLM is sandboxed. The LLM is likely accessible through a chatbot interface and tasked with helping users to find useful entries or asking questions about entries. We call the user's question to the LLM a *query*. But only having an LLM at hand, does not fully leverage all the capabilities it offers. In the next section, we describe how the uploader of an entry can define how her entry should be understood or used by an LLM.

3. Bringing Expert Viewpoints to Large Language Models

We propose to accompany data in RDRs with a so-called *interpretation prompt*, which specifies how the data and how parts of it should be understood or used. In a nutshell, the uploader creates an interpretation prompt file containing instructions for the LLM how to interpret or use her data and when a user asks the LLM about that entry, the interpretation prompt is prepended to the query. We first describe the upload process and then the usage.

The uploader uses the upload form of the RDR. We extend this upload form with a new text field, in which the uploader can input the interpretation prompt. The RDR then stores the interpretation prompt in a file along the other files uploaded. The file is accessible to the users just as everything else in the entry.

The interpretation prompt itself may contain the viewpoint of the uploader regarding the uploaded data. This expert viewpoint can stem from expertise in the research area, describe the data set in more detail, be an annotated summary, or even be the result of running an algorithm. Furthermore, instructions on how to process the data can be included in the interpretation prompt to provide an automated execution for verifying reproducibility. In principal, the interpretation prompt can be anything textual, which the uploader finds important for the LLM to know about her entry. Thus, one may include terms like "*If the user asks for <X>, ...*" to include different guidings for different use cases.

Example. *An example for an interpretation prompt is: "This data set is a recording of heartbeats of the patient. The task for the patient was to rest on the couch and then move to a chair."*

When other users open the entry in the RDR, they can not only view the entry and preview files, but can also ask the LLM about the entry or the RDR in general. The interpretation prompt prepends the query and both are sent to the LLM with the answer being returned to the user. By prepending the interpretation prompt, the LLM can use the expert viewpoint provided in the interpretation prompt or reproduce some results. Moreover, the uploader benefits from her work may become more accessible and thereby citable. In the next section, we showcase the effect of interpretation prompts in examples.

4. Demonstration of Interpretation Prompts

Interpretation prompts are designed to help the LLM understand the data better. In this section, we showcase how the interpretation prompt alters the answer of the LLM and highlight how people might conceive the answer better. Throughout this section, we use GPT-4.1 [8] as LLM. We shorten the answers of the LLM to exemplary parts. We start with an analysis of a simple synthetic data set where the interpretation prompt tells the LLM more about the experiment behind the data set. We continue with a summary task, where we intend a specific direction of summary. Next, the LLM is tasked with explaining a knowledge representation. We end with reproducing an experiment via the LLM.

4.1. Synthetic Heartbeat Data

We have generated a synthetic heartbeat data set with the LLM. The heartbeats in the data set are based around 60 and go up to around 100 for a small continuous section of time steps, which mimic a movement of a patient. We intend that the LLM recognizes the increased activity and matches it to the movement of the patient. Thus, we ask the LLM to summarize the data set and report the key findings. Without interpretation prompt, the LLM correctly recognizes the section of increased activity and suggests an interval of stress or activity. But, it also returns a rather extensive statistical analysis. Listing 1 shows the shortened answer of the LLM.

Listing 1: Answer for analyzing artificial heartbeat data without using an interpretation prompt. The answer is shortened to exemplary parts.

```
## Summary of Heartbeat Data
...
### Descriptive Statistics
| Statistic | Value |
| :-- | :-- |
| Count | 40 |
| Mean | 71.08 |
| Standard Deviation | 18.42 |
| Minimum | 55 |
| 25th Percentile | 58 |
| Median | 63 |
| 75th Percentile | 95 |
| Maximum | 105 |
### Key Findings
...
### Notable Patterns
...
### Interpretation
- The dataset likely reflects a period of rest, a short interval of increased activity or stress,
    and a return to baseline.
- The sudden increase at time point 15 could indicate the onset of physical activity, emotional
    stress, or another stimulus affecting heart rate.
```

Afterward, we add the following interpretation prompt: "*The data set is a recording of the heartbeat of a female patient with her smartwatch. You can use any algorithms for such series to find interesting facts yourself. However, the task was to rest on a couch and then move to a chair. Please keep your answer short and precise and do not write about unnecessary facts, only the most important things.*" Now, the LLM matches the interval of increased heartbeats to the movement. Moreover, it keeps the answer short and reports only the relevant facts.

Listing 2: Answer for analyzing artificial heartbeat data using an interpretation prompt. The answer is shortened to exemplary parts.

```
### Heartbeat Data Summary
#### Key Findings
...
#### Notable Pattern
- **Average heart rate:** 71 bpm
- **Minimum heart rate:** 55 bpm
- **Maximum heart rate:** 105 bpm
#### Interpretation
- The sharp increase in heart rate at measurement 15 indicates the moment the patient moved from
    the couch to the chair.
- The elevated heart rate persists for several measurements, suggesting a physiological response
    to the movement before returning to baseline.
#### Table: Heart Rate Change Point
...
**Conclusion:** The data clearly shows a resting phase, a distinct increase in heart rate due to
    movement, and a return to baseline, matching the described activity.
```

4.2. Guiding the Summary of Linear Algebra

In this test, we use the Wikipedia article on linear algebra as a dummy RDR entry. We have used the Wikipedia article as of July 8th, 2025 [9]. We intend to learn about linear algebra with a focus on vectors and matrices, on which everything else is build upon. When only telling the LLM to summarize the article and report the key findings, it summarizes each section of the Wikipedia article seemingly independently. Moreover, the concept of *vectors* is not really explained, only indirect via the vector spaces. Listing 3 shows the shortened answer.

Listing 3: Answer for a summary of the Wikipedia article on linear algebra without using an interpretation prompt. The answer is shortened to exemplary parts.

```
...
## Key Concepts
### 1. **Historical Development**
...
### 2. **Vector Spaces**
...
### 3. **Linear Maps and Transformations**
...
### 4. **Matrices**
- **Representation**: Matrices represent linear maps and facilitate explicit calculations in
    finite-dimensional spaces.
```

```
- **Matrix Operations**: Multiplication, similarity, and elementary operations are central for
    solving systems of equations and understanding transformations.
- **Gaussian Elimination**: Core algorithm for solving linear systems and finding matrix
    properties.
### 5. **Determinants and Eigenvalues**
...
### 6. **Duality and Inner Product Spaces**
...
### 7. **Applications**
...
### 8. **Extensions and Generalizations**
...
## Key Findings
- **Central Role in Mathematics**: Linear algebra is foundational across nearly all mathematical
    disciplines and essential in modern geometry, analysis, and applied sciences.
- **Computational Importance**: Efficient algorithms for matrix operations have been developed and
    optimized for modern hardware, making linear algebra indispensable in scientific computing.
- **Geometric and Algebraic Unification**: The field bridges algebraic structures with geometric
    intuition, enabling powerful methods for modeling and problem-solving.
- **Broad Applicability**: Linear algebra is crucial in engineering, physics, computer science,
    economics, and beyond, providing tools for modeling, analysis, and optimization in complex
    systems.
- **Abstraction and Generalization**: The evolution from solving linear systems to abstract vector
    spaces and modules illustrates the power and flexibility of linear algebraic thinking.
## Conclusion
**Linear algebra** is a core mathematical discipline with deep theoretical foundations and vast
    practical applications. Its concepts - vector spaces, linear maps, matrices, determinants, and
    eigenvalues - form the backbone of modern mathematics, science, and engineering, enabling the
    modeling, analysis, and solution of a wide array of problems.
```

For the next query, we use the following interpretation prompt: "*When talking about this article, keep the math short and precise. Emphasize the role of vectors and matrices and open room for applications. Let other parts play a minor role.*" Listing 4 shows the shortened answer using the interpretation prompt. The answer reports on vectors and matrices, mentions other concepts only briefly and seems to provide a much more coherent answer. Therefore, the answer is significantly better and tailored to our expectations.

Listing 4: Answer for a summary of the Wikipedia article on linear algebra using an interpretation prompt. The answer is shortened to exemplary parts.

```
...
### Vectors
- **Vectors** are elements of a vector space, typically represented as ordered lists of numbers.
- Operations on vectors include addition and scalar multiplication, following specific axioms (
    associativity, commutativity, identity, and inverses).
- Vectors are used to model quantities with both magnitude and direction, such as forces or
    velocities.
### Matrices
- **Matrices** are rectangular arrays of numbers that represent linear maps between vector spaces.
- Matrix operations (addition, multiplication, inversion) mirror the composition and
    transformation of vectors.
- Matrices are essential for representing and solving systems of linear equations, encoding
    geometric transformations, and more.
### Key Mathematical Concepts (Kept Short)
...
## Applications
...
## Other Aspects (Briefly)
- **Subspaces, span, and basis:** Describe how sets of vectors generate spaces and how to find
    minimal generating sets.
- **Dual spaces and inner products:** Extend the theory to functionals and geometric notions like
    length and angle.
- **Extensions:** Module theory, multilinear algebra, and topological vector spaces generalize the
```

```
    core ideas.
## Key Takeaways
- **Vectors and matrices are the backbone** of linear algebra, enabling the modeling and solution
    of a vast array of problems.
- **Linear algebra is indispensable** for modern science, engineering, and technology, providing
    both theoretical insight and practical computational tools.
- Other advanced topics enrich the field but are secondary to the central role of vectors and
    matrices.
```

4.3. Focusing on Relevant Probability Relations

We test the interpretation prompt for explaining a knowledge representation in this subsection. We use the famous *Smokers and friends* data set [10]. The data set models that friends most likely share the same behavior regarding drinking and smoking, e.g., if one friend smokes, the other probably smokes, too. The task is to explain the knowledge representation, and we intend that the key fact of shared behavior is stated explicitly and prominently. We use two variants of the data set: The first is a parameterized factor graph [11] modeling the smoking and drinking relations, where we add noise to the probabilities [12]. The second variant is a Markov logic network containing weighted rules of the same model, which were simplified to have only a few rules using CoFE [12]. In this test, the first variant corresponds to the query without an interpretation prompt as it is the larger model and would possibly be present in an RDR entry. The simplified second variant resembles the query with interpretation prompt as we have simplified the knowledge representation.

When just letting the LLM explain the Smokers and friends data set present as a factor graph, the LLM describes the representation in great detail and reports the findings. In particular, it mentions the important fact of shared behavior, but among other findings and not prominently. Listing 5 shows the shortened answer for the prompt "*Please explain the following parfactor graph and summarize its key findings*", which is followed by the factor graph, which we omit for clarity. When presenting the second variant, i.e., the simplified model focusing on the important relation, the LLM highlights the key fact better and, especially, reports it as the final conclusion and not listed among other findings of low significance. Listing 6 shows the shortened answer for the prompt "*Please explain the following Markov Logic Network and summarize its key findings*", which is followed by the Markov logic network, which we again omit for clarity.

Listing 5: Answer for explaining the Smokers and friends data set given as a parameterized factor graph. The answer is shortened to exemplary parts.

```
## Explanation of the Parfactor Graph
...
### Types and Individuals
...
### Random Variables
...
### Parfactors
...
#### 1. Friendship and Smoking
...
#### 2. Friendship and Drinking
...
### Queries
...
## Key Findings
- **Relational Dependencies:** The graph models how personal habits (smoking, drinking) are
    correlated with friendships. Specifically, it encodes that the likelihood of two people being
    friends depends on whether they share similar habits.
- **Potential Values:** Higher potential values indicate more probable configurations. For example
    , the highest value in both tables is when certain combinations of habits align, suggesting
    that shared habits (both smoke or both drink) may make friendship more likely.
- **Symmetry:** The model treats all individuals equally (no person-specific parameters), focusing
    on pairwise relationships and shared behaviors.
```

Listing 6: Answer for explaining the Smokers and friends data set given as a Markov logic network. The answer is shortened to exemplary parts.

```
...
### Interpreting the Formulas
...
#### Smoking-Related Formulas
...
#### Drinking-Related Formulas
...
## Summary Table
| Formula | Weight | Meaning | Strength |
| :-- | :-- | :-- | :-- |
| friends(P, Q) ^ smokes(P) ^ smokes(Q) | 2.03 | Friends tend to both smoke | Strong positive |
| !friends(P, Q) v !smokes(P) v !smokes(Q) | 0.07 | Friends rarely both smoke | Very weak |
| friends(P, Q) ^ drinks(P) ^ drinks(Q) | 2.10 | Friends tend to both drink | Strong positive |
| !friends(P, Q) v !drinks(P) v !drinks(Q) | 0.05 | Friends rarely both drink | Very weak |
## Key Findings
- **Homophily in Habits:** There is a strong tendency for friends to share similar habits - if one
     smokes or drinks, their friends are also likely to do so.
- **Weak Contradictory Evidence:** The negative rules (suggesting friends do not share habits)
    have negligible weights and thus little effect on the overall model.
- **Interpretation:** The MLN encodes that friendship is a strong predictor of shared smoking and
    drinking behavior, with little evidence to the contrary.
## Conclusion
This Markov Logic Network robustly models the idea that **friends are likely to share smoking and
    drinking habits**, with the positive associations being much stronger than any negative or
    contradictory patterns.
```

```
## Summary Table: Parfactor Structure
| Parfactor | Variables | Potential Values (rounded) | Interpretation |
| :-- | :-- | :-- | :-- |
| 1 | friends(P,Q), smokes(P), smokes(Q) | 7.62, 1.33, 0.43, 0.82, 1.51, 1.49, 0.74, 1.22 |
    Friendship depends on both smoking status |
| 2 | friends(P,Q), drinks(P), drinks(Q) | 8.17, 0.66, 0.45, 0.80, 1.23, 1.79, 1.39, 1.01 |
    Friendship depends on both drinking status |
## Interpretation
...
```

4.4. Reproducing Experiments

Listing 6 shows the better tailored answer for explaining the Smokers and friends data set. Naturally, we can also achieve this result by telling the LLM to "*focus on the relations smoking and drinking*". However, when having a large knowledge representation at hand, it is tedious to write down the important relations manually. Instead, we aim at using the LLM to use an algorithm to automatically focus on the important relations. CoFE is an algorithm for extracting simplified rules from knowledge representations [12]. The steps are to cluster the probabilities in each factor and extract rules for each cluster, which are simplified. Effectively, we reproduce a research experiment by running CoFE on a noised version of the Smokers and friends data set. To perform the reproduction by using an LLM, we give the LLM access to an implementation of the CoFE algorithm. We still use the interpretation to tell the LLM how to interpret the input and how to output the result. After finetuning our interpretation prompt, we have finally managed to get the LLM to correctly reproduce our experiment by running CoFE on the provided input. Listing 7 shows the (shortened) answer for the following interpretation prompt: "*Attached you find the CoFE paper and some CoFE evaluation scripts. The *.java files are for extraction and minimization, refer to them instead of transform.jar. Please review how CoFE works (mapping, extraction, minimization) and then run CoFE on the provided noised.blog file and output the minimized rules. Use an epsilon of 0.3 and a dbscan epsilon of 2 with two minimum samples. For MultiArrayPotential, the first* *entry stands for all variables set to true. Please output the resulting minified MLN, which should contain rules for each parfactor.”*

Listing 7: Shortened answer for running the CoFE algorithm on a noised Smokers and friends data set. While the final output contains rounded versions, the ln terms mentioned before are correct.

```
## How CoFE Works
...
## Running CoFE on the Provided Example
...
### Step 1: Mapping Potentials
...
#### Parfactor 1: (friends(P,Q), smokes(P), smokes(Q))
...
#### Parfactor 2: (friends(P,Q), drinks(P), drinks(Q))
...
### Step 2 + 3: Extraction \& Minimization
...
#### Parfactor 1 (friends, smokes):
**MLN Rules:**
- **Weight**: ln(7.62) ~ 2.03
**Formula**: friends(P, Q) ^ smokes(P) ^ smokes(Q)
- **Weight**: ln(1.08) ~ 0.08
**Formula**: !friends(P, Q) v !smokes(P) v !smokes(Q)
#### Parfactor 2 (friends, drinks):
**MLN Rules:**
- **Weight**: ln(8.17) ~ 2.10
**Formula**: friends(P, Q) ^ drinks(P) ^ drinks(Q)
- **Weight**: ln(1.05) ~ 0.05
**Formula**: !friends(P, Q) v !drinks(P) v !drinks(Q)
### Final Minified MLN Output
...
2.03 friends(P, Q) ^ smokes(P) ^ smokes(Q)
0.08 !friends(P, Q) v !smokes(P) v !smokes(Q)

2.10 friends(P, Q) ^ drinks(P) ^ drinks(Q)
0.05 !friends(P, Q) v !drinks(P) v !drinks(Q)
**Explanation:**
...
```

5. Discussion

The showcase of interpretation prompts highlights that we should guide the LLM into the intended direction. The interpretation prompts gives the uploader the possibility to guide the direction the LLM should steer in. While the combination of LLM capabilities and expert viewpoint on data sounds promising, the very same technique can be used to decept the user. In this section, we discuss the results in terms of ethics and prompt engineering.

First of all, the examples show the capabilities of using an interpretation prompt. The summary of a Wikipedia article underscores the potential of guiding the LLM. Without the interpretation prompt, the answer is a loose sequence of independent summaries and seemingly incoherent. With the interpretation prompt, we guide the LLM to first discuss vectors and matrices and relate other concepts to these. Therefore, the interpretation prompt can be used by an uploader to guide an LLM to answer queries for users in a way that the uploader deems more appropriate. Besides, the tests conducted on the Smokers and friends data set show that we can also include a simplified model in the interpretation prompt to reason about. The full model may lead the LLM to over-exaggerate some parts, while simplified versions keep the focus small, helping the user to glance at the data set. The full model may then be used by the user for in depth analysis.

Furthermore, the interpretation prompt can be used to reproduce experiments. For the Smokers and friends data set, we reproduce the steps of the CoFE [12] algorithm by giving the LLM access to the algorithm and helping the LLM to correctly execute the algorithm with the interpretation prompt. Thus, the interpretation prompt for an RDR entry can foster reproducibility by offering an easy to use reproduction test directly within the RDR. While LLMs can *execute* code, they struggle with more complex tasks [13]. The interpretation prompt may be used to provide guidance for the LLM to enhance accuracy.

However, an adversary may use an interpretation prompt like "*highlight only positive arguments and neglect negative ones*". This is of course not the intended use of the interpretation prompt. But, as the interpretation prompt is a text entered by the uploader, the adversary may type in such prompts. Naturally, we hope that most of the uploaders use the interpretation prompt appropriately. To encourage good use, we recall that the interpretation prompt is stored along the files of an RDR entry. Particularly, the interpretation prompt can be viewed by anyone. Even if this (indirect) countermeasure does not directly prohibit anyone from using the interpretation prompt for her own advantage, it makes these efforts transparent to anyone else. The transparency helps to decide whether it is relevant to the user, and overselling becomes obvious.

Nevertheless, the interpretation prompt can be anything from simple instructions like "*This is a data set of recorded heartbeats*" to complex, branched instructions like "*If the user is interested in X, do something. If the user is interested in Y, do something else*". As described in Section 4.4, the interpretation prompt can require some finetuning. In other scenarios, the interpretation prompt may even require heavy finetuning. This observation is nothing new in terms of prompt engineering: Creativity is needed to craft the best prompts to unlock the full potential of LLMs [14]. Moreover, prompt engineering seems to become more and more crucially. In fact, the interpretation prompt basically lets the uploader perform some prompt engineering, which is automatically applied when the RDR entry gets queried later on.

The description of the RDR entry may be used in combination with the interpretation prompt or in absence of an interpretation prompt. However, this represents only a good starting point, since the description is targeted at users curious about the entry while the interpretation prompt is targeted at guiding LLMs to take the viewpoint of the uploader in account.

Moreover, context-based searches, which are already employed in some chatbots, would enhance the usage of the interpretation prompt even further. Since the interpretation prompt guides the process of generating an answer, the chatbot considers the guidance when searching for other, relevant documents.

All in all, the uploader has the possibility of helping others to understand their data in the way the uploader would like to have the data used. The users benefit from expert viewpoints guiding the answer generation of the LLM.

6. Conclusion

An RDR preserves research data for a long period of time, accumulating numerous entries over time. When adding an LLM chatbot to the RDR, users get the possibility to ask questions about the entries. The creators of the entry are most likely experts and have additionally information, like which parts are more important than others. Thus, we propose to use this knowledge to help the LLM understand the entry better. To do so, we suggest the use of an additional interpretation prompt, which contains an expert viewpoint by the creators. The interpretation prompt is entered during the creation of the entry in the RDR. The interpretation prompt is prepended for the query sent to the LLM, such that the LLM can benefit from the expert knowledge. In our examples, we demonstrate that the usage of an interpretation prompts significantly alters the answer and directs it in the direction we intended and expected. Thus, creators may use the interpretation prompt to guide the LLM in the direction they intend to understand and use their data. Moreover, the interpretation prompt can be used for automated reproduction of research experiments. With our experiments, we demonstrate that interpretation prompts actually help the LLM to reproduce experiments and to be guided in the intended direction. Overall, the interpretation prompt helps to move from a general purpose trained LLM to a project-specific LLM invocation.

Acknowledgments

The research for this paper was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC 2176 'Understanding Written Artefacts: Material, Interaction and Transmission in Manuscript Cultures', project no. 390893796. The research was conducted within the scope of the Centre for the Study of Manuscript Cultures (CSMC) at Universität Hamburg.

Declaration on Generative AI

The authors have used GPT-4.1 for evaluating their approach of interpretation prompts. Apart from that, the authors have not employed any Generative AI tools for writing the paper.

References

[1] T. Asselborn, S. Melzer, S. Schiff, M. Bender, F. A. Marwitz, S. Aljoumani, S. Thiemann, K. Hirschler, R. Möller, Building sustainable information systems and transformer models on demand, Humanities and Social Sciences Communications 12 (2025) 1–15.

[2] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne, et al., The fair guiding principles for scientific data management and stewardship, Scientific data 3 (2016) 1–9.

[3] H. Peukert, S. Melzer, J. Jacob, I. Vogel, K. Wörner, S. Grünler, L. Brase, A. Heinrich, S. Thiemann, Das forschungsdatenzentrum der universität hamburg: Auf dem weg zu einem gemeinsamen verständnis im umgang mit forschungsdaten in den natur-, sozial-und geisteswissenschaften, Bausteine Forschungsdatenmanagement (2023).

[4] Y. Annepaka, P. Pakray, Large language models: A survey of their development, capabilities, and applications, Knowledge and Information Systems 67 (2025) 2967–3022. doi:`10.1007/s10115-024-02310-4`.

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

[6] D. Anisuzzaman, J. G. Malins, P. A. Friedman, Z. I. Attia, Fine-tuning large language models for specialized use cases, Mayo Clinic Proceedings: Digital Health 3 (2025).

[7] A. Bozkurt, Tell me your prompts and i will make them true: The alchemy of prompt engineering and generative ai, Open Praxis (2024). doi:`10.55982/openpraxis.16.2.661`.

[8] OpenAI, et al., Gpt-4 technical report, 2024. URL: https://arxiv.org/abs/2303.08774. `arXiv:2303.08774`.

[9] Wikipedia contributors, Linear algebra — Wikipedia, the free encyclopedia, https://en.wikipedia.org/w/index.php?title=Linear_algebra&oldid=1296750597, 2025. [Online; accessed 08-July-2025].

[10] G. Van den Broeck, N. Taghipour, W. Meert, J. Davis, L. De Raedt, Lifted probabilistic inference by first-order knowledge compilation, in: Proceedings of the Twenty-Second international joint conference on Artificial Intelligence, AAAI Press/International Joint Conferences on Artificial Intelligence, 2011, pp. 2178–2185.

[11] D. Koller, N. Friedman, Probabilistic graphical models: principles and techniques, MIT press, 2009.

[12] F. A. Marwitz, T. Braun, R. Möller, A first step towards even more sparse encodings of probability distributions, in: N. Katzouris, A. Artikis (Eds.), Inductive Logic Programming, Springer International Publishing, Cham, 2022, pp. 183–192.

[13] C. Lyu, L. Yan, R. Xing, W. Li, Y. Samih, T. Ji, L. Wang, Large language models as code executors: An exploratory study, arXiv preprint arXiv:2410.06667 (2024).

[14] G. Marvin, N. Hellen, D. Jjingo, J. Nakatumba-Nabende, Prompt engineering in large language

models, in: International conference on data intelligence and cognitive informatics, Springer, 2023, pp. 387–402.