Journal Title
XX(X):1–18
©The Author(s) 2016
Reprints and permission:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

From Scientific Variables to Knowledge Graphs: The I-ADOPT Benchmark

Barbara Magagna[1,2], Arvin Rastegar[3], Esteban Gonzalez[4], Cristian Berrío[5], Stuart Chalk[6], José Manuel Gómez-Pérez[5], Christof Lorenz[3], Saurav Kumar[7,8] and Daniel Garijo[4]

Abstract

With the adoption of the Findable, Accessible, Interoperable and Reusable (FAIR) principles for data by researchers, an increasing amount of datasets have been made available online, supporting research investigations. In order to ease dataset interoperability, the I-ADOPT framework has been proposed by the scientific community as a means to capture the subtleties and nuance of scientific variables in a structured manner. However, creating machine readable variable representations requires significant expertise and manual effort, given the wealth of variable types in use by different communities. In this paper we explore the use of Large Language Models (LLMs) to aid addressing this manual step. We propose the I-ADOPT Benchmark, an expert annotated corpus and task designed to measure the performance of LLMs in the different stages of automatically creating a machine readable scientific variable. Our corpus includes more than 100 scientific variables as structured knowledge graphs, and our results show that even models of large size (32B) struggle in creating these representations accurately ($< 50\%$ F1 score).

Keywords

Scientific variables, knowledge graphs, I-ADOPT, benchmark, LLM

Introduction

The Findable, Accessible, Interoperable, and Reusable principles for data (Wilkinson et al., 2016) have made datasets first class citizens in scientific research in order to support the findings reported in research publications. Datasets (and other digital objects such as software (Chue Hong et al., 2022)) are now demanded by journals[1], conferences[2] and funding bodies alike.[3]

FAIR compliance in research is therefore increasingly expected at a large scale. However, applying the FAIR principles remains a challenging task, requiring effort and time by researchers. In addition, making data FAIR does not always ensure interoperability, given the large number of formats and domain-specific practices used by scientific communities.

Achieving true interoperability between datasets requires deliberate implementation efforts and alignment across their variable representations. However, creating unambiguous, machine-readable representations of the scientific variables present in a dataset is not trivial, as variables may need qualifiers and constraints that are key for data integration. For example, the variable *'Systolic blood pressure"*, which measures the pressure in the arteries when the heart beats and pumps blood, can be decomposed in the following description components: *pressure* as the main `property`, the *systolic* state as a `constraint` on *pressure*, *blood* as the `object of interest` and *human* as the context to define in which body it was measured.

The I-ADOPT ontology[4] is a Research Data Alliance initiative designed as a potential solution to address scientific variable representation. In I-ADOPT each variable is represented as a knowledge graph, providing a specification to systematically represent variables in a structured and domain-agnostic way while also ensuring sufficient precision for interpretation by both humans and machines.

However, modeling scientific variables as I-ADOPT variables is not a trivial process for two main reasons: (i) researchers may have differing (but correct) perspectives when representing similar concepts and restrictions over a variable, and (ii) the process requires time to agree on a common representation, along with expertise on the Semantic Web technologies (RDF, OWL) and domain expertise on the variables themselves. Therefore, scaling up variable generation requires semi-automated services for

[01] GO FAIR Foundation (GFF), Leiden, Netherlands
[2] Semantics, Cybersecurity and Services, University of Twente, NL
[3] Karlsruhe Institute of Technology (KIT), Institute of Meteorology and Climate Research and Atmospheric Environmental Research (IMKIFU), Germany
[4] Ontology Engineering Group, Universidad Politécnica de Madrid, Madrid, Spain
[5] Language Technology Research Lab, Expert.ai, Madrid, Spain
[6] University of North Florida: Jacksonville, FL, US
[7] IISPV, Departament d' Enginyeria Quimica, Universitat Rovira i Virgili, Av. Països Catalans 26, 43007 Tarragona, Catalonia, Spain
[8] German Federal Institute for Risk Assessment (BfR), Berlin, Germany
[0]

Corresponding author:
Barbara Magagna, GO FAIR Foundation, Poortgebouw Noord, Rijnsburgerweg 10, 2333 AA Leiden, NL.

[0] Email: barbara@gofair.foundation

[1] https://journals.plos.org/plosone/s/data-availability

[2] https://kdd2025.kdd.org/call-for-artifact-badging/

[3] https://sfdora.org/read/

[4] https://w3id.org/iadopt/ont/

generating variable candidates, along with a human-in-the-loop approach for their validation.

In order to address this issue, in this paper we assess the viability of Large Language Models (LLMs) for automated scientific variable generation following the I-ADOPT framework. LLMs have shown promise in knowledge representation tasks, such as the creation of structured representations from text (Mo et al., 2025) or the generation of ontologies from competency questions (Saeedizade & Blomqvist, 2024). Therefore, our goal is to leverage LLMs to generate machine-readable variables starting from the variable definition in text. The main contributions of this paper are:

- A corpus composed of 102 I-ADOPT machine-readable variables in domains ranging from Physical and Chemical Sciences to Social sciences. The corpus is validated by domain experts and built using a specific methodology, in order to become a ground truth to assess automated systems for variable representation. The corpus is available online (Magagna and Chalk, 2025).
- A comparative study of the performance of Large Language Models (ranging from 3 to 32 billion parameters, with both open and closed models) easily deployable at a large scale. All the evaluation details are available online (Rastegar et al., 2025).
- A detailed analysis of the errors introduced by LLMs, identifying the parts of the variables that are particularly difficult to represent.

The remainder of the paper is structured as follows. The Related Work section provides an overview of existing approaches for representing scientific variables in a structured manner, along with efforts to automatically convert them into knowledge graphs. Next, the Background: The I-ADOPT Framework section provides more information on the model used in this paper. The The I-ADOPT Corpus describes how we designed the gold standard, while the section I-ADOPT Benchmark: Evaluation of automated variable decomposition using LLMs describes the benchmark design and evaluation results. Finally, sections Discussion and Conclusions outline the existing challenges faced in the benchmark, as well as areas of future work.

Related Work

This section provides an overview of available frameworks for representing scientific variables, together with common methods for transforming these variables into Knowledge Graphs.

A Brief History on Scientific Variable Representation

Measurements, observations, and simulations are fundamental in scientific research. Proper interpretation of their results requires knowledge about contextual aspects, including what, how, when, and where the data acquisition took place. A number of standards have been designed to address this need, such as OBOE (Madin et al., 2007), Semantic Sensor Network ontology (SSNO) (Haller et al., 2018) and SAREF (J. Moreira et al., 2020). In its current version[5], the SSN Ontology is modularised and includes several modules that provide explicit alignments with Observations, Measurements, and Samples (OMS, also known as ISO 19156:2023) (Open Geospatial Consortium, 2023), PROV (Moreau & Groth, 2022), SAREF (J. L. Moreira et al., 2017), DOLCE (Borgo et al., 2022), and IDO[6], supporting interoperable use across communities. Via its core model SOSA Observation, SSNO defines an observation as an act of observing a single Property of a single Feature of Interest. In cases where direct observations are not possible, Samples are used as proxies. However, this model alone is insufficient to provide consistent and reusable representations of observed properties across domains. The Climate and Forecast Metadata (CF) Conventions (Eaton et al., 2003) addresses this challenge by using standard names to consistently describe variables captured in NetCDF data files. While widely adopted, these descriptions are not FAIR, as they lack formal, machine-interpretable semantics.

The Scientific Variable Ontology (SVO) Framework (Stoica & Peckham, 2019) goes a step further by providing formal, machine-readable rules for composing complex concepts from elementary ones and is used for semantic mediation within the interdisciplinary MINT framework (Gil et al., 2021)[7]. More recently, ML Commons Croissant (Akhtar et al., 2024) has emerged as a standard for describing dataset used in machine learning training. Croissant provides a lightweight, machine-readable skeleton that can be easily integrated into ML pipelines. However, it currently lacks the deep, domain-specific semantics needed for cross-disciplinary reuse. Croissant relies on schema.org[8] to describe datasets, and schema:measuredVariable offers a hook for systematically linking to richer contextual representations of variables.

To leverage these complementary strenghts while addressing existing gaps, representants of SVO, CF and terminology providers initiated a collaboration under the umbrella of RDA as the I-ADOPT Working Group[9] to define a common, lightweight approach for representing variables that is understandable by both humans and machines, while fully aligning with the FAIR principles. Endorsed by RDA in 2022, the I-ADOPT recommendations (Magagna et al., 2022) are meanwhile discussed as candidate Open Geospatial Consortium (OGC) standard to complement the OGC Observations, Measurements and Samples (OMS) Specification (Open Geospatial Consortium, 2023) to provide a rich and FAIR definition of the observable property concept in the OMS/SOSA model.

The initiative is widely supported since its beginning by environmental research infrastructures (Magagna et al., 2021) such as the eLTER RI (Integrated European Long-Term Ecosystem, Critical Zone and Socio-Ecological

[5] https://w3c.github.io/sdw-sosa-ssn/ssn/, version 2023

[6] https://rds-staging.posccaesar.org/ido/

[7] http://mint-project.info/

[8] https://schema.org/

[9] https://www.rd-alliance.org/groups/interoperable-descriptions-observable-property-terminology-wg-i-adopt-wg/activity/

Research Infrastructure)[10] with its Envthes vocabulary[11], LifeWatch ERIC[12], ACTRIS (Aerosols, Clouds, and Trace Gases Research Infrastructure)[13] with its ACTRIS vocabulary[14] and the OZCAR Resarch Infrastructure with its vocabulary[15] as discussed in (Coussot et al., 2024), by data centers like PANGAEA (Diepenbroek et al., 2017), Australia's Terrestrial Ecosystem Research Network TERN[16] and the British Oceanography Centre[17], and by terminology providers such as the OntoPortal Alliance (Jonquet et al., 2025) and NERC Vocabulary Server (NVS)[18].

Automated Extraction of Scientific Variables from text

Generating structured knowledge representations from unstructured text has been extensively studied in the knowledge graph (KG) community. This task involves extracting entities, identifying relationships, and organizing information according to predefined schemas or ontologies. A survey by Zhong et al. (2023) summarizes 300 methods for automatic KG construction based on the three steps required to generate a KG: knowledge acquisition, refinement, and evolution. These methods focus not only on the extraction of entities but also on the extraction of the relations between them. A collection of KG databases is provided to evaluate the methods. This survey highlights the heterogeneity of the available solutions.

A more recent survey, Choi and Jung (2025) explores KG generation across three core dimensions: Extraction, Learning Paradigm, and Evaluation Methodology. Extraction involves the processes used to collect and transform raw data into structured information. Learning refers to the use of ML techniques to identify relational patterns within KGs. Evaluation examines the frameworks and metrics employed to assess KG quality. This survey reviews more than 4000 papers related to KGs and shows a growing interest in multimodal and domain-specific extraction approaches.

In the Extraction dimension, we find concepts such as Named Entity Recognition (NER) and Relation Extraction (RE). NER employs advanced models, including BERT-based architectures (Devlin et al., 2019), Bi-LSTM (Graves, 2012), CRF (Lafferty et al., 2001), and graph-driven approaches, to identify entities and align them with predefined ontologies. RE uses dependency parsing, semantic feature modeling, and attention mechanisms to identify subject–relation–object triples from unstructured text. Furthermore, multimodal and domain-adapted extraction techniques incorporate heterogeneous data sources such as text, images, and sensor inputs to improve the accuracy and relevance of knowledge extraction.

KG learning comprises a diverse set of methods designed to support link prediction, relational inference, and structured data analysis. Early embedding models such as TransEBordes et al., 2013 provide efficient representations of entities and relations, while GNN-based approaches capture complex, heterogeneous graph interactions for tasks like node classification and link prediction. Transformer-based models further integrate textual and structural information through self-attention mechanisms, enhancing relational consistency by combining graph topology with logical constraints.

The emergence of generative models has encouraged their use to generate triples directly. In generating triples, we face three challenges: (i) producing a correct decomposition of the entities identified in the text, (ii) generating triples that conform to a given ontology, and (iii) performing entity linking, that is, associating each entity with a specific vocabulary concept. Each of these challenges represents a step in the KG generation pipeline and can introduce errors that may propagate.

I-ADOPT variable generation shares similarities with KG construction—both require parsing text, extracting semantic components, and mapping them to ontological structures. However, I-ADOPT presents unique challenges: variables can be correctly modeled in multiple ways, components must be precisely typed (property, constraint, matrix, etc.), and domain expertise is required to interpret scientific terminology. These characteristics make I-ADOPT variable generation a specialized structured extraction task that tests LLMs' ability to perform fine-grained semantic decomposition within constrained frameworks.

As for the evaluation of text to KG approaches, existing datasets such as WEBNLG (Gardent et al., 2017) and NYT (Riedel et al., 2010) provide mainly triples for different domains, but they do not include an ontology-based schema. That is, they lack a formal declaration of entities and relations.

Other comprehensive evaluation frameworks such as Text2Bench (Mihindukulasooriya et al., 2023) and OSKG (Wang & Iwaihara, 2025) exist. Text2Bench includes three main metrics: Accuracy of the facts extraction, ontology conformance, and hallucinations. Fact Extraction Accuracy assesses how well the language model (LLM) captures factual information by comparing its output triples to ground truth triples using precision, recall, and F1 score, where higher values indicate better performance. Ontology conformance (OC) measures the proportion of LLM-generated triples that adhere to the input ontology, considering a triple conforming if its relation matches one of the ontology's canonical relations. This metric can be extended to validate domain, range or other axioms. Finally, hallucination metrics quantify non-sensical or unfaithful output through subject (SH), relation (RH) and object (OH) hallucination rates, determined by checking whether the elements of each triple are present in the source sentence or ontology.

The OSKGC framework provides a benchmark for constructing KGs from text based on an ontology schema. Within this framework, different prompts are defined for each step of the process: Joint Extraction, Entity Recognition, Entity Typing, and Relation Extraction. The authors propose an evaluation metric called Structural

[10] https://elter-ri.eu/

[11] https://vocabs.lter-europe.net/EnvThes/en/

[12] https://www.lifewatch.eu/

[13] https://www.actris.eu/

[14] https://vocabulary.actris.nilu.no/skosmos/actris$_{v}ocab/en/$

[15] https://in-situ.theia-land.fr/skosmos/theia$_{o}zcar_{t}hesaurus/en/$

[16] https://www.tern.org.au/

[17] https://www.bodc.ac.uk/

[18] https://vocab.nerc.ac.uk/search$_{n}vs/$

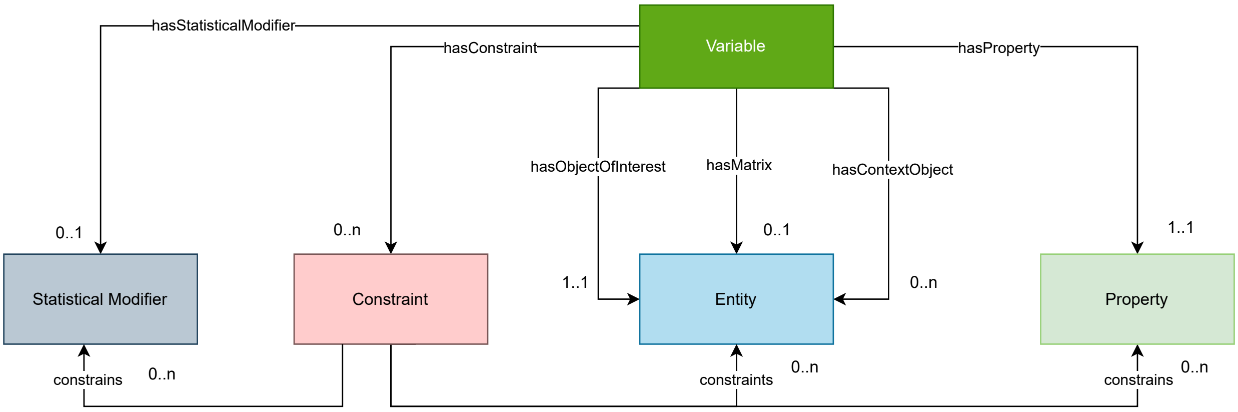

Figure 1. Core classes of the I-ADOPT ontology

Similarity (SS) to measure the degree of alignment between the schema of the constructed knowledge graph and the predefined ontology.

As can be observed in the analyzed frameworks, there is a lack of corpora focused on the representation of scientific variables, as well as a lack of benchmarks designed to evaluate the ability of decomposition, rather than the generation of triples. This gap highlights the necessity of resources such as the I-ADOPT corpus and its associated benchmark, which enable a systematic evaluation for the semantic decomposition of scientific variables.

Background: The I-ADOPT Framework

In our work, we use the I-ADOPT Framework to create machine-readable representations of scientific variables. According to I-ADOPT, a variable should explicitly capture the context needed to understand what the values of the digital object it represents mean. Information that needs to be preserved and carried with both the data and the metadata throughout their lifecycle. Current practices often contrast with this requirement by providing sloppy annotations that reduce the information to either the property or the measured phenomenon, or by providing only abbreviations or community-specific notations. This hinders the interoperable data reuse, as it remains unclear whether the data can be integrated for aggregation or analysis purposes. The objective, initially focused on environmental research, has been extended to specify a lingua franca that allows for a domain-agnostic representation that is understandable by both humans and machines, while also enabling the contextualization and accuracy required by individual scientific domains.

In I-ADOPT (see Figure 1), a Variable is understood as a compound concept consisting of at least one entity having the role of the ObjectofInterest and its Property and additional entities providing metadata, like the embedding medium or the body in which it is contained, as the Matrix or other relevant information as the ContextObject. The entity playing the role of ObjectOfInterest can be either an object or a process being observed. All entities used in the definition of the variable can be constrained to provide precision about their condition, state, or limitation. I-ADOPT focuses on *what* has been observed, measured, or simulated, independently of the exact geographical position, timestamp and the method applied. As a consequence, also the unit of measurement is omitted, as the same variable can be expressed in various units, making the variable concept reusable in different settings.

The framework was tested by semantic modelers during two organized challenges in 2024[19] based on a common set of 30 variables from various domains. This effort revealed some weaknesses of the model, including multiple possible representations as well as the inability to capture some complex scenarios. The challenges were followed by two modeling workshops in 2025[20] involving 25 experts in oceanography, atmospheric composition, biodiversity, ecosystem research and ontology engineering to address the shortcomings and discuss solutions. As a result, the ontology was refined, and its version 1.1.0 provides capabilities to describe fluxes and complex systems. Systems can be either symmetric (when entities have the same roles as parts of the system) or asymmetric (with entities having either the role of numerator/denominator, normally used for ratios like concentrations (see its application in Figure 2 or source/target used for representing flows in the matrix slot). The extension now also includes statistical modifiers to represent aggregations and makes it possible to apply constraints on all description components allowing for more flexible modeling (see Figure 1).

I-ADOPT is about providing triple statements to enrich the description of a scientific variable. The starting point is a human-readable description provided by a user. This might be a single sentence or an abstract from a paper. Applying I-ADOPT for the definition of a variable involves the splitting of the description into atomic information units, identifying their roles in the description and subsequently annotating them using semantic concepts from community-agreed terminologies.

Figure 2 shows the application of I-ADOPT for a biochemistry example, representing the concentration of a contaminant originating from the degradation of a pesticide in the body of a mollusc. Using this example, we try

[19] https://i-adopt.github.io/challenge.html
[20] https://i-adopt.github.io/workshops.html

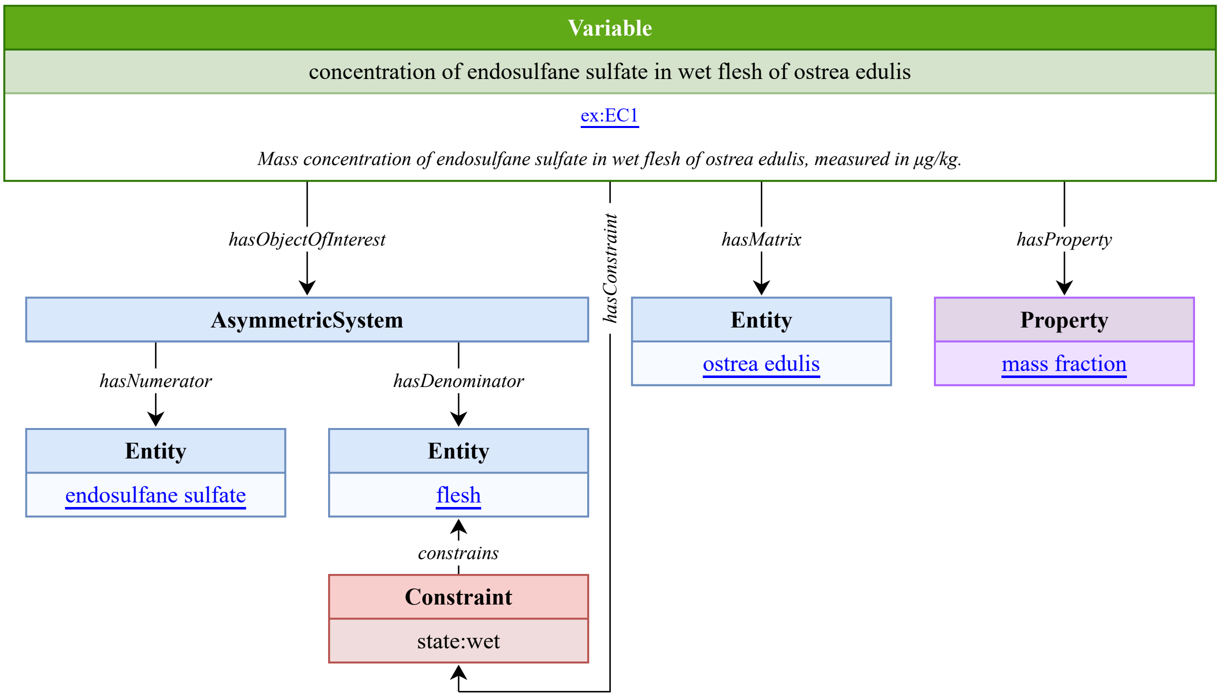

Figure 2. Example of a machine-readable variable[21] applying the I-ADOPT framework[22]

to illustrate how the description of a variable can be 'I-ADOPTed'.

1. Clarify the meaning of the provided definition. In the example given, the definition is: 'Mass concentration of endosulfane sulfate in wet flesh of ostrea edulis, measured in $\mu g\,kg^{-1}$'. The unit $\mu g\,kg^{-1}$ indicates that this is a mass fraction in strict metrology terms. Mass concentrations are defined as mass per volume, but sometimes they are also used for indicating mass concentrations per unit mass. The clarified sentence results in: 'Mass fraction of endosulfane sulfate in wet flesh of ostrea edulis, measured in $\mu g\,kg^{-1}$'.
2. Exclude information that is not relevant for the variable description. Here we need to exclude the second part of the sentence, which refers to the unit. The cleaned sentence reads like this: 'Mass fraction of endosulfane sulfate in wet flesh of ostrea edulis'.
3. Split the clarified sentence into atomic units of information, ensuring to keep the original meaning of the terms: mass fraction, endosulane sulfate, wet, flesh, and ostrea edulis.
4. Assign a specific role to atomic units in the description, selecting the appropriate predicates in the RDF statement: `iop:hasProperty` refers to the object *mass fraction*, `iop:hasObjectOfInterest` points to an asymmetric system of two entities where `iop:hasNumerator` links to the substance *endosulfane sulfate* and `iop:hasDenominator` to *flesh*, `iop:hasMatrix` refers to the species *ostrea edulis*, in which the substance was determined. Lastly, `iop:hasConstraint` is applied to the flesh: *wet* to define the state.
5. Annotate each description with a semantic concept using a chosen community terminology (QUDT (Quantities, Units, Dimensions and Data Types Ontologies) is a reference ontology providing standardized definitions for quantities and units of measure. NERC vocabularies are often used for marine biochemstry variables): *mass fraction*: qudt:MassFraction[23], *endosulfane sulfate*: s27:CS003625[24]; *flesh*: s12:S1214[25]; *ostrea edulis*: MS7472[26]; *wet*: pato:PATO_0001823[27].
6. Provide a correct syntax for constraints, which should be typed (here using the type state):

```
iop:hasConstraint
    [ a iop:Constraint, pato:PATO0001823
     ;
      rdfs:label "state: wet" ;
      iop:constrains s12:S1214 ;
    ] .
```

This process, typically conducted by a semantic expert in collaboration with a domain expert, is time-consuming. Leveraging an LLM-enabled service to support this process would allow researchers to document their variables in a FAIR-compliant and machine-readable way immediately upon dataset availability, without requiring the assistance of semantic experts.

[23] http://qudt.org/vocab/quantitykind/MassFraction
[24] http://vocab.nerc.ac.uk/collection/S27/current/CS003625/
[25] http://vocab.nerc.ac.uk/collection/S12/current/S1214/
[26] http://vocab.nerc.ac.uk/collection/P21/current/MS7472/
[27] http://purl.obolibrary.org/obo/PATO0001823

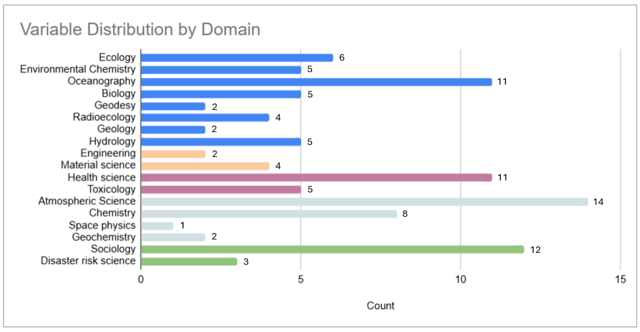

Figure 3. Variable distribution by domain of the I-ADOPT Corpus

The I-ADOPT Corpus

The I-ADOPT Corpus includes 102 scientific variables in multiple domains, represented in machine-readable manner and curated by experts. This section outlines the methodology and overview of the corpus, explaining the process and rationale of its inception.

Corpus Development Methodology

To create a domain-independent corpus that extends beyond the original focus on environmental domains, variables from a range of different areas were selected. First, a semantic expert redefined the thirty challenge variables according to the extended I-ADOPT ontology. We then added variables from different domains, collaborating with diverse communities: the EuroGOOS High Frequency Radar Working Group[28], PARC (Chemical Risk Assessment)[29], the Atmospheric Composition Standard Names Group[30], eLTER RI, Climate Change Adaptation (Horizon Europe project FAIR2Adapt[31]).

In total, we collected 102 variables from these areas (also compare to Figure 3:

- Physical and Chemical Sciences (25)
- Earth and Environmental Sciences (40)
- Life and Health Sciences (16)
- Engineering and Technology (6)
- Social and Risk Sciences (15)

The procedure for creating the corpus involved the following steps:

1. The participating communities suggested variables from their domains and provided detailed definitions. Complex variables were prioritized, in order to assess the I-ADOPT framework's representation capabilities
2. A semantic expert modeled the variables following a systematic design process
3. 15 domain experts evaluated the variable decompositions following an evaluation scheme
4. A semantic expert refined the variable models following suggestions provided by domain experts until a consensus was reached while keeping aligned with the discovered design patterns

Systematic design process. To ensure consistency in modeling all variables, design patterns were developed by identifying recurring structures in the decomposition and the analysis of the frequency of combinations among the associated description components. The modeling approach was guided by the principle of delivering generic, domain-independent representations via the I-ADOPT framework, alongside detailed, community-specific descriptions achieved through typed constraints and design patterns co-developed with domain experts in each field. The number of patterns applied was minimized during development to favor simple representations while still allowing for complex ones when needed. To ensure alignment with the I-ADOPT framework, the I-ADOPT Visualizer (Schindler, 2025b) was used, which only renders upon successful validation according to the I-ADOPT SHACL rules[32] . Moreover, the Visualizer allows for interactive modification of graphical elements and immediate adjustments in the turtle file, making the modeling process more intuitive and user-friendly. Finally, the description components were annotated using Wikidata concepts wherever possible, and only in cases of missing coverage, concepts from other terminologies were used. Constraint annotations were omitted for the same reason.

Once modeled, the turtle files for the variable representations were published on a dedicated GitHub repository (https://w3id.org/iadopt/corpus). The corpus variables are presented in a browsable catalog (Schindler, 2025a) that illustrates their decomposition to facilitate human readability. Each variable page is linked to its underlying turtle file and to GitHub issues, where the modeling was discussed with the domain experts involved in the evaluation process. As a result of additional online meetings, not all interactions are formally documented.

Expert evaluation scheme. For the assessment of the variable representations, 15 invited domain experts (acknowledged in Magagna and Chalk, 2025 were asked, after a basic introduction to the basic I-ADOPT rules, to evaluate the representations of the variables by rating the following questions using these options: yes, fully / yes, partially / no, barely / not at all / I am not able to answer the question):

- Correct: Is the representation capturing the variable correctly?
- Generic: Are the entity labels for Object of Interest and Matrix generic enough to be reusable for other variable decompositions?
- Complete: Is the representation explicit and comprehensive enough to cover the relevant parts of the variable description?
- Concise: Does the representation of the variable include only I-ADOPT relevant descriptions without redundancies in terms of methods and units, etc?
- Understandable: Is the representation, including its decomposition into atomic elements, easy to understand?

[28] https://eurogoos.eu/task-teams/high-frequency-radar/
[29] https://www.eu-parc.eu/
[30] https://ui.adsabs.harvard.edu/abs/2024AGUFMA41L.1731S
[31] https://fair2adapt-eosc.eu/
[32] https://github.com/SirkoS/iadopt-schema/blob/main/shacl/iadopt.sh.ttl

Table 1. Evaluation responses to questions assessing the fulfillment of modeling criteria

	Fully	Partially	Barely	Not at all	Not able to answer
Correct	46	44	6	1	5
Generic	76	25	0	0	1
Complete	69	27	3	0	3
Concise	78	14	2	0	8
Understandable	65	26	0	0	11

Table 1 shows the results of our evaluators. For variables for which the evaluators were unable to answer, two other domain experts were asked. In addition, the evaluators were requested to suggest changes to improve the representation when they considered it useful. 34 variables were fully accepted without changes and 77 required adjustments. These suggestions were incorporated, where feasible, in the refinement process and re-discussed in GitHub issues until a consensus aligned with the design principles could be achieved. This evaluation process demonstrated that correct decomposition of the variable description requires the involvement of domain experts because original descriptions may contain implicit knowledge and, therefore, can be misinterpreted.

Corpus Overview

The I-ADOPT Corpus consists of 102 expert-curated I-ADOPT variables spanning different scientific domains, with an uneven distribution. Earth and Environmental Science variables predominate, reflecting the disciplinary focus of the original contributing community to the I-ADOPT Working Group.

Table 2. Required adjustments to variable descriptions prior to decomposition

	Percentage of 102 variables
Refined definition	46.32
Added Matrix	26.32
Added Constraint	9.47
Parts omitted	23.16

We summarize two structural characteristics that strongly influence the decomposition difficulty and evaluation.

Constraints. A variable is counted as having constraints if it includes a `hasConstraint` relation, regardless of the number of individual constraints. In the corpus, 85 variables contain `hasConstraint`, and 16 do not. Among the constrained variables, 29 contain exactly one constraint.

Matrices. A variable is counted as having a matrix if it contains a `hasMatrix` relation, whether represented as a simple entity or as a system of entities. In total, 61 variables contain `hasMatrix`, and 40 do not. Among the variables with matrices, 34 are represented via a simple entity and 27 use systems (26 asymmetric and 1 symmetric system).

These counts provide context for interpreting per-component results and error patterns, particularly for `Constraint` and `Entities` having the role of Matrix (as used in Figure 2, which are more difficult to infer reliably from short natural-language definitions.

Of the 102 variables, only 21 could be decomposed directly. In fact, many variables had to be interpreted before they could be decomposed. This included refinements of the definition, additions of matrices or constraints, or omissions of parts in the definition in case it included I-ADOPT irrelevant content, such as methods or instruments. In Table 2 the percentage of variables that require adjustments according to these criteria is provided (units are not counted as omissions, as they are required for proper interpretation of the property).

Overall, the I-ADOPT Corpus can serve as a gold standard for benchmarking analyses. It spans multiple domains, extending beyond its original focus on environmental domains, and has been reviewed by domain experts.

I-ADOPT Benchmark: Evaluation of automated variable decomposition using LLMs

Variable decomposition and linking is a time consuming manual step requiring assistance and validation by experts. In order to aid this process, this section explores how large language models (LLMs) may be used to automatically decompose scientific variable definitions into structured representations aligned with the I-ADOPT ontology.

In I-ADOPT, a variable is described through a set of *slots*, where each slot corresponds to a specific description component, such as Property, ObjectOfInterest, Matrix, or Constraint—that captures a distinct semantic aspect of the variable. Automated variable interpretation therefore consists of identifying and populating these slots from natural-language definitions and, where applicable, linking their values to concepts in controlled vocabularies.

This section describes the benchmark methodology used to evaluate this process. We introduce the decomposition and linking tasks, outline the input and output assumptions, describe the prompting strategy and JSON-based interaction format, and present the evaluation protocol and experimental setup. Results are reported using aggregated metrics as well as per-slot analyses, reflecting performance at the level of individual I-ADOPT description components.

All benchmark code, prompts, schemas, and evaluation scripts are archived and publicly available (Rastegar et al., 2025).

Benchmark development methodology

Definition of tasks. Given a scientific variable expressed as an uncurated natural-language definition, the primary task assigned to the LLM is to generate a structured I-ADOPT representation by populating the corresponding slots. Each slot is filled with a textual or structured value inferred directly from the definition text, and each variable is processed independently.

In addition to slot population, a secondary task consists of linking the extracted slot values to entities from external controlled vocabularies. This linking step supports semantic interoperability but is evaluated separately from the decomposition task.

Although the I-ADOPT ontology includes the `ContextObject` role, this slot is rarely populated in the corpus variables and is therefore excluded from the quantitative results reported in results subsection. The role is retained in the methodology description for completeness but is omitted from result tables due to sparsity.

Input and output assumptions. In the current version of the benchmark evaluation variable descriptions are assumed to be complete, i.e., each LLM is instructed to rely *exclusively* on the provided variable definition text and to refrain from introducing information that is not explicitly stated. This restriction is imposed to minimize hallucination and to enforce ontology alignment at the schema level rather than through implicit background knowledge. As a result, mismatches are expected when the expert-curated ground truth relies on domain knowledge or preferred formulations that are not literally present in the definition text.

For each run, the model receives: (i) a fixed set of prompt rules corresponding to one of the prompt variants described in the following prompt variants paragraph, (ii) a JSON Schema derived from the I-ADOPT ontology, (iii) zero, one, three, or five example decompositions depending on the shot setting, and (iv) the target variable definition to be decomposed. The model does *not* receive ontology serializations (TTL), SHACL constraints, or controlled vocabulary identifiers.

The model is required to output a *single* JSON object that exactly conforms to the provided schema. If a slot cannot be populated based solely on the definition text, the model is instructed to leave the slot empty (using an empty string or an empty list, as appropriate) rather than attempting to infer or guess missing information. This design choice ensures that the evaluation reflects extraction fidelity rather than implicit reasoning.

We use JSON as an intermediate interaction and serialization format for variable decomposition because prior work has shown that large language models can effectively generate structured outputs when constrained by explicit schemas and fixed output formats (Shorten et al., 2024).

In this workflow, JSON serves as a schema-aligned interface between the natural-language variable definition and the ontology-based representation defined by I-ADOPT. The explicit slot structure enables automatic validation of model outputs, ensures consistency across prompt variants and models, and supports fine-grained, slot-level evaluation of decomposition performance.

Importantly, JSON is used solely as an interaction and evaluation format for the language model. It does not replace or reinterpret the underlying ontology semantics. All generated representations are ultimately converted back into ontology-aligned structures for comparison with the expert-curated corpus.

Prompt variants. We evaluate three prompt variants that differ in how they structure the decomposition task and guide the extraction of I-ADOPT description components:

- `strict_minimal`: a minimal instruction set that emphasizes extraction strictly from explicitly stated text. The model is instructed to leave slots empty when the required information is not clearly supported by the variable definition.
- `constraint_decision_tree`: a stepwise prompt structure that enforces an explicit extraction order. In this variant, constraint extraction is deferred until after core components have been identified, and constraints are explicitly required to reference previously extracted slots.
- `matrix_decision_tree`: a decision-oriented prompt that prioritizes distinguishing matrix-related phrases (e.g., materials or media) from conditions that should instead be modeled as constraints.

These prompt variants are designed to test how different levels of structural guidance affect decomposition quality, particularly for components that are frequently ambiguous in natural-language definitions, such as matrices and constraints.

All prompts are handcrafted and iteratively refined using a small set of explicit, expert-informed heuristics derived from the I-ADOPT modeling guidelines. These heuristics include enforcing a fixed extraction order, restricting extraction to information explicitly stated in the variable definition, distinguishing matrix-like contexts from constraints, and requiring empty outputs when a component cannot be supported by the text. Aside from the variable definition itself and the number of example decompositions provided (shot setting), prompts are fixed, task-agnostic, and reused unchanged across all variables and model configurations. The complete prompt templates used in the experiments are provided in Appendix section .

Schema-driven validation and retries. Generated JSON outputs are validated against the JSON Schema provided in Appendix section . If an output is not schema-valid, the request is retried, and the same prompt is reissued to the model, without modification, up to a maximum of three attempts. Retries occur *only* due to schema non-conformance (not because an output is low-quality but valid). In practice, retries are rarely required. Only schema-valid JSON outputs are retained for evaluation. Valid LLM JSON outputs are compared with the corpus representation.

Component-based evaluation. Evaluation is performed at the level of I-ADOPT components, where a *component* corresponds to one role in the description of a variable (e.g., Property, ObjectOfInterest, Matrix, Constraint). For each variable, the model-generated representation is compared to the expert-curated gold standard (in the corpus) separately for each component.

Systems are treated as representations occupying a slot, rather than as standalone entity, in accordance with the I-ADOPT ontology. Mandatory components are `hasProperty` and `hasObjectOfInterest`. All other components are evaluated only when applicable.

The **ObjectOfInterest** and **Matrix** slots may be represented either as **simple entities** or as **system entities**. A system can be **asymmetric**, when the entities involved have different roles, or **symmetric**, when the entities involved have the same role. When these slots are represented as systems, their evaluation follows system-specific rules described below. Otherwise, they are evaluated using string comparison.

Simple and system-valued components. For components such as `hasProperty`, `hasContextObject`, and `hasStatisticalModifier`, evaluation is based on string comparison between the gold standard and the generated representation:

- a correct match yields a full True Positive (TP = 1.0),
- an incorrect value yields a full False Positive (FP = 1.0),
- missing values yield a False Negative (FN = 1.0),
- correctly omitted non-applicable components yield a True Negative (TN = 1.0).

Constraint components. Constraints are represented as instances of `hasConstraint`, each associated with a human-readable label and linked via the predicate `constrains` to the component it restricts.

For evaluation, constraints are compared based on their label and constrained target. The order of constraints is ignored. Each constraint contributes a fractional score based on the proportion of correctly matched elements.

Asymmetric systems. Asymmetric systems may occur as representations in the ObjectOfInterest or Matrix slots. When present, they are evaluated by comparing all required parts (e.g., `hasSource` and `hasTarget`, or `hasNumerator` and `hasDenominator`). Order is not ignored for asymmetric systems. Each part contributes equally to the final score, and partial correctness results in fractional TP and FP values.

Symmetric systems. Symmetric systems may occur as representations in the ObjectOfInterest or Matrix slots. They are evaluated based on their `hasPart` elements. The order of parts is ignored. Each part contributes equally to the total score.

Structural mismatches. If the slot specifies a system representation and the generated output provides a simple entity representation (or vice versa), the prediction is treated as a structural mismatch. In such cases, the component is evaluated as incorrect, even if partial textual overlap exists, because the representation structure is semantically significant in I-ADOPT.

Exact and close matching. Two matching strategies are used:

- **Exact match**: ignores capitalization and leading/trailing whitespace. If the normalized strings are identical, similarity is 1.0. Otherwise, it is 0.0.
- **Close match**: uses cosine similarity between sentence embeddings (model `all-MiniLM-L6-v2`). A similarity score of 0.8 or higher is considered a match.

The embedding model is chosen as a lightweight and widely used sentence encoder that provides stable semantic representations and is suitable for large-scale, reproducible evaluation (Galli et al., 2024). A similarity score of 0.8 or higher is considered a match, reflecting a conservative threshold intended to capture clear semantic equivalence while avoiding overly permissive matches.

Metric computation. For each component, we compute True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN). Fractional values are permitted for composite representations such as constraints and systems. Precision, Recall, and F1-score are computed per component and aggregated over all test variables.

These metrics are widely used for evaluating structured extraction and classification tasks and are standard in knowledge graph construction and information extraction evaluations (Bhatt et al., 2024).

Linking to controlled vocabularies. After schema validation and component-level evaluation, generated variable representations are semantically enriched by linking each slot defined by the I-ADOPT ontology—such as `hasProperty`, `hasObjectOfInterest`, `hasMatrix`, or `hasConstraint`—to persistent identifiers from controlled vocabularies. This step ensures interoperability and supports FAIR principles by enabling machine-readable, domain-agnostic representations.

For each slot, we take the textual value produced during decomposition and query the **Wikidata Search API** (`wbsearchentities`[33]) using that term as the search string. The API returns a list of candidate entities with:

- **label**: the entity's preferred name in Wikidata,
- **description**: a short textual description provided by Wikidata.

We use the default API behavior without limiting the number of candidates beyond what the API returns.

Since the goal of this paper is not to propose a state-of-the-art entity linker but to provide an efficient baseline in terms of computational cost and performance, we first compare a naive approach—selecting the top result returned by Wikidata—with an improved baseline that leverages reranking. Specifically, we explore whether incorporating the variable definition as contextual information can enhance candidate selection. For this purpose, we evaluate two cross-encoder models: a widely used lightweight option (`cross-encoder/ms-marco-MiniLM-L6-v2`) and a more advanced reranking model based on the smallest Qwen3 variant (`Qwen3-Reranker-0.6B`). For the first model, we use a simple prompt template: `Definition of "{term}" in context: "{context}" + label: "{label}", description: "{description}"`, whereas for the second model we adopt the following prompt adapted to the Qwen3 architecture:

```
<|im_start|>system
Judge whether the Document meets the
    requirements based on the Query and the
    Instruct provided. Note that the answer
    can only be "yes" or "no".
<|im_end|>
<|im_start|>user
<Instruct>: Given a web search query,
    retrieve relevant passages that answer
    the query
<Query>: Definition of "{term}" in context:
    "{context}"
```

[33] https://www.wikidata.org/w/api.php?action=help&modules=wbsearchentities

```
<Document>: label: "{label}", description:
    "{description}"
<|im_end|>
<|im_start|>assistant
<think>

</think>
```

Here:

- `term` = the slot value from the decomposition,
- `context` = the full variable definition,
- `label`, `description` = values returned by Wikidata for the candidate entity.

The model returns a binary decision internally converted into a confidence score. Candidates are ranked by this score, and the top candidate is selected. No type checks or additional filtering are applied.

To evaluate entity linking independently, we use the ground truth variable decomposition as the starting point, removing the Wikidata links before performing entity linking and reserving them as ground truth for this specific task. As a metric, we report the average accuracy per variable. When computing accuracy, only cases where the ground truth entity is linked to Wikidata are considered, and a prediction is counted as correct only if the predicted link exactly matches the expected one.

Results

This subsection reports the results of evaluating multiple LLM configurations on the I-ADOPT corpus. Performance is assessed using aggregated Exact and Close matching metrics across all description roles, complemented by per-slot analyses that reveal differences in component-level behavior.

Evaluation split and comparability across shot settings. To ensure comparability across prompting strategies, a fixed set of five variables is reserved as the few-shot example pool and excluded from evaluation in *all* runs, including the 0-shot setting. Consequently, every experimental configuration is evaluated on the same set of 97 variables (102 total variables minus the five held-out examples), guaranteeing fair comparison across different shot settings.

Shot settings. We evaluate 0-shot, 1-shot, 3-shot, and 5-shot prompting strategies, where the number of example decompositions included in the prompt corresponds to the shot setting. The results indicate that the optimal number of shots is model-dependent. While larger models such as Qwen-32B benefit from additional examples and achieve their best performance under 5-shot prompting, most smaller models reach peak performance in the 0–1 shot regime. This suggests that few-shot prompting is not universally beneficial and that its effectiveness depends on model capacity and robustness.

Model selection. The evaluated models represent a selected subset of language models chosen to reflect realistic deployment scenarios for automated variable decomposition services. Selection criteria prioritized open-source instruction-tuned models that can be executed on institutional HPC or on-premise infrastructure, ensuring that the benchmark remains reproducible and practically applicable in typical research environments. Models were therefore chosen to span different parameter scales while remaining feasible for local deployment. In addition, a lightweight proprietary model (GPT-4o-mini) was included as a reference point to contextualize the performance of open-source models against a commonly used closed alternative.

Grid search strategy. A grid search is used to identify well-performing configurations. The following parameters are explored:

- model choice (open-source and proprietary),
- temperature,
- prompt instructions (`strict_minimal`, `constraint_decision_tree`, `matrix_decision_tree`),
- number of shots (0, 1, 3, 5).

The search is performed in two stages. First, the set of example variables is fixed while varying models, temperatures, prompts, and number of shots. Once these parameters are selected, the parameter selection is held constant, and the example variable set that is given to the model changes to find the best set of examples.

Aggregated results across description roles. Table 3 summarizes the best-performing configuration for each evaluated model using aggregated Exact and Close Precision, Recall, and F1-score across all I-ADOPT description roles.

Overall, Qwen-32B (Team, 2025) shows the highest scores among the evaluated models, reaching an $\mathrm{F1}_{exact}$ of 0.45 and an $\mathrm{F1}_{close}$ of 0.46 under the reported configuration. This configuration combines a larger model size with few-shot prompting, which coincides with higher aggregated performance in this benchmark setting.

Among the smaller open-source models, **Qwen-8B** and **LLaMA-3-8B** achieve comparable results, particularly in low-shot configurations. In contrast, **GPT-4o-mini**, included as a lightweight proprietary reference model, attains lower aggregated scores in this evaluation, highlighting performance differences across model families under identical schema constraints.

Across all models, Close matching yields consistently higher scores than Exact matching. This reflects cases where generated slot values are semantically similar to the corpus annotations but differ at the lexical level.

Effect of prompt variants and shot settings. Table 4 reports the effect of the three prompt variants across different shot settings for Qwen-32B, selected as a representative large model. Results are reported using Exact and Close F1-scores aggregated across all evaluated description roles. This analysis illustrates how prompt structure and the number of provided examples affect decomposition performance. Qwen-32B is chosen due to its strong overall performance and stable behavior across configurations.

Overall, performance improves consistently as the number of shots increases across all prompt variants, confirming the benefit of providing example decompositions. The `strict_minimal` prompt achieves the highest overall performance, reaching the best Exact and Close F1-scores in the 5-shot setting. The constraint_decision_tree based

Table 3. Aggregated Exact and Close Precision, Recall, and F1-score across all I-ADOPT description roles. For each model, the best-performing configuration is reported.

Model	Type	Prompt	Shots	Temp	$\mathbf{P}_{exact}$	$\mathbf{R}_{exact}$	$\mathbf{F1}_{exact}$	$\mathbf{P}_{close}$	$\mathbf{R}_{close}$	$\mathbf{F1}_{close}$
Qwen-32B	Open-source (large)	strict_minimal	5	0.5	0.29	0.49	0.45	0.33	0.52	0.46
Qwen-8B	Open-source (small)	strict_minimal	0	0.5	0.22	0.43	0.38	0.26	0.46	0.42
LLaMA-3-8B	Open-source (small)	constraint_tree	1	0.0	0.26	0.34	0.36	0.29	0.36	0.40
Mistral-7B	Open-source (small)	strict_minimal	0	0.0	0.21	0.35	0.29	0.24	0.38	0.32
GPT-4o-mini	Proprietary	strict_minimal	0	0.5	0.19	0.23	0.23	0.22	0.26	0.25

prompts show competitive results, particularly in the 3-shot and 5-shot configurations, but do not surpass the strict_minimal variant. Differences between prompt variants are most pronounced in the 0-shot setting, where more structured prompts provide slightly better Close F1-scores, while these differences narrow as more examples are introduced.

Per-slot performance analysis. Table 5 reports per-slot Exact and Close F1-scores for the main I-ADOPT description roles evaluated in this study, namely Property, ObjectOfInterest, Matrix, and Constraint. These slots represent the core semantic components required to characterize scientific variables and occur sufficiently often in the benchmark to support quantitative analysis.

The ObjectOfInterest and Matrix slots are evaluated at the slot level regardless of whether their representations are simple entities or system entities. Differences in internal representation structure (simple vs. system) are handled internally by the evaluation rules described in the methodology subsection and are not reflected as separate result categories. The ContextObject role is excluded from this analysis due to its low frequency in the benchmark.

Both models perform best on the Property component, achieving high recall and the highest F1-scores, indicating that core physical or chemical properties are reliably extracted from variable definitions. Performance decreases for ObjectOfInterest, and drops further for Matrix and Constraint, which remain the most challenging components. The larger Qwen3-32B model consistently outperforms Qwen3-8B across all components, with the largest gains observed for ObjectOfInterest and Constraint. Across all slots, recall is generally higher than precision, suggesting that models tend to over-generate candidate components rather than miss relevant ones.

Errors Analysis. Table 6 shows the summary of the error analysis we performed on our results. Although *ContextObject* is excluded from the main performance tables due to sparsity, we include it here for completeness in hallucination/conformance diagnostics.

We have observed that some representations in the variable decomposition exhibit very low values; therefore, we decided to study the predicted values and the ground truth values in greater depth. We use three main metrics:

- Hallucination. This metric measures whether the model is overly verbose; that is, whether it generates non-empty predictions when the corresponding slot in the benchmark is empty.
- Ground Truth (GT) conformance. A metric that evaluates whether, in the ground truth, annotators use text that is not present in the definition slot. This indicates whether the model needs to include new text or not.
- Predictive conformance (PRED). A metric that evaluates whether, in the predictive results, the model uses text that is not present in the definition slot.

We observe that the impact of hallucination on the predicted results is minimal for most slots, except for `hasMatrix`, where it reaches approximately 20%. In this case, the model tends to include values even when they are not present in the ground truth, suggesting that it struggles to handle this field appropriately. Hallucination levels are consistent across all prompts, with no significant variation.

Regarding text conformance in GT, we observe that for most fields, the text used in the annotations is not present in the field of definition of the ground truth. This complicates the decomposition task as the model must introduce additional text that is not included in the input provided to it. An interesting case is `hasMatrix`: although this field exhibits higher hallucination rates than the others, it also shows high text conformance, which at first glance may appear contradictory.

The results indicate that conformance in the predicted values is higher than in the ground truth, suggesting that the model predominantly relies on text present in the definition. This behavior is consistent with the analyzed prompts and helps explain the overall low scores.

Linking to controlled vocabularies. Table 7 provides a summary of the entity linking results. We observe that the naïve model alone achieves relatively strong performance, and that applying reranking further improves this baseline. In particular, the Qwen3-based model delivers

Table 4. Effect of prompt variants and number of shots on Qwen-32B performance. Exact and Close F1-scores are aggregated across all I-ADOPT description roles.

Shots	strict_minimal		constraint_decision_tree		matrix_decision_tree	
	$\mathbf{F1}_{exact}$	$\mathbf{F1}_{close}$	$\mathbf{F1}_{exact}$	$\mathbf{F1}_{close}$	$\mathbf{F1}_{exact}$	$\mathbf{F1}_{close}$
0-shot	0.27	0.31	0.24	0.27	0.30	0.35
1-shot	0.39	0.43	0.36	0.39	0.36	0.39
3-shot	0.40	0.42	0.40	0.45	0.40	0.44
5-shot	**0.45**	**0.46**	0.42	0.44	0.42	0.44

Table 5. Per-slot Exact and Close Precision, Recall, and F1-scores for the best-performing large and small models. Both Qwen3-8B and Qwen3-32B are evaluated using the `strict_minimal` prompt with temperature 0.5. Qwen3-8B uses 0-shot prompting and Qwen3-32B uses 5-shot prompting, corresponding to their best-performing configurations reported in Table 3.

I-ADOPT Component	Qwen3-8B						Qwen3-32B					
	$\mathbf{P}_{exact}$	$\mathbf{R}_{exact}$	$\mathbf{F1}_{exact}$	$\mathbf{P}_{close}$	$\mathbf{R}_{close}$	$\mathbf{F1}_{close}$	$\mathbf{P}_{exact}$	$\mathbf{R}_{exact}$	$\mathbf{F1}_{exact}$	$\mathbf{P}_{close}$	$\mathbf{R}_{close}$	$\mathbf{F1}_{close}$
Property	0.38	1.00	0.55	0.38	1.00	**0.55**	0.51	1.00	0.68	0.52	1.00	**0.69**
ObjectOfInterest	0.13	0.86	0.22	0.15	0.88	**0.26**	0.22	1.00	0.36	0.24	1.00	**0.39**
Matrix	0.14	0.21	0.17	0.18	0.26	**0.21**	0.20	0.29	0.24	0.22	0.31	**0.26**
Constraint	0.15	0.11	0.13	0.32	0.20	**0.25**	0.30	0.28	0.29	0.43	0.35	**0.39**

Table 6. Error Analysis of the Qwen3-32B Model Results. Hall. denotes hallucination, GT. denotes text conformance in the GT. and PRED. denotes text performance in the predictive values. StatMod = hasStatisticalModifiers, ObjInt = hasObjectOfInterest, CtxObj = hasContextObject, Constr = hasConstraints.

	strict_minimal			matrix_decision_tree			constraint_decision_tree		
Field	Hall.	GT.	PRED.	Hall.	GT.	PRED.	Hall.	GT.	PRED.
StatMod	0.01	–	–	0.01	–	–	0.02	–	–
Prop	0.00	0.59	0.78	0.00	0.59	0.81	0.00	0.59	0.84
ObjInt	0.00	0.38	0.85	0.00	0.38	0.88	0.00	0.38	0.92
Matrix	**0.19**	**0.92**	**0.98**	**0.20**	**0.92**	**0.99**	**0.19**	**0.92**	**0.98**
CtxObj	0.00	–	–	0.02	–	–	0.01	–	–
Constr	0.00	0.32	0.7	0.00	0.32	0.65	0.00	0.32	0.72

an improvement of approximately ten percentage points in accuracy compared to the naïve approach.

Table 7. Results of entity linking evaluation: average accuracy per variable for cases with Wikidata ground truth links.

Model	URI accuracy
naïve	0.656
cross-encoder/ms-marco-MiniLM-L6-v2	0.716
tomaarsen/Qwen3-Reranker-0.6B-seq-cls	0.759

Expert assessment

Using the best-performing configuration—Qwen3-32B with the `strict_minimal` prompt, 5-shot prompting, and temperature 0.5, as identified by the highest aggregated Exact and Close F1-scores against the benchmark, expert semantic analysis was conducted on the generated decompositions. The semantic correctness of alternative outputs, as compared to the gold standard, can ultimately be evaluated only by a semantic expert who contributed to the creation of the I-ADOPT Corpus. The following grades were assigned (see table 8):

- Good, if the decomposition is identical to the corpus, allowing for synonymous terms
- Correct, if the decomposition can be mapped to the corpus
- Weak, if the decomposition includes correct components, but fails to capture some important aspects
- Wrong, if the decomposition misses essential aspects or is syntactically invalid

Allowed mappings include:

- unconstrained `Entity` versus `Entity` plus `Constraint`
- `ObjectOfInterest` plus `Matrix` versus `AsymmetricSystem` (hasNumerator/hasDenominator) for the `ObjectOfInterest` for representing ratios

A weak grade is assigned when aspects of the variable description that are relevant for the correct interpretation of the variable are omitted, while the representation still correctly captures the `Property` and the `ObjectOfInterest`.

Essential aspects refer to concepts related to `Property` and the `ObjectOfInterest`; if these are missing, the overall representation is considered wrong. Syntactically invalid representations were identified with respect to the constraints: in some cases, constraints applied to entities that were not detected in the decomposition, and in other cases, constraints were themselves constrained, which is not permitted by the I-ADOPT Ontology.

Table 8. Semantic expert analysis of the best-performing configuration—Qwen3-32B for all 102 variables, reported as percentages

	Percentage
Good	10.53
Acceptable	25.26
Weak	31.58
wrong	32.63

Because slightly different representations can be automatically mapped to the gold standard, good and acceptable assignments can be aggregated. The results show an approximately equal distribution between correct, weak, and wrong decompositions, with a slight predominance of correct ones. Good classification results are associated with variables that did not require any refinement of their definitions.

Discussion

Variable descriptions following the I-ADOPT ontology exhibit a key characteristic that affects their automatic generation: the current version of the ontology allows multiple valid representations for the same variable. I-ADOPT provides a minimal set of core classes that serve as a baseline for decomposing a variable description; however, the concepts used to annotate the components in the different slots are not constrained.

The richness of natural language and the diversity of terminologies used to describe variables therefore make the consistent use of concepts inherently difficult. As a result, automatic generation can only reflect this heterogeneity, although it has the potential to reduce subjectivity in the selection of appropriate terms.

For improved results, additional knowledge support should be provided to guide the decomposition process. Consistent construction of the I-ADOPT Corpus benefited from following recurring best decomposition practices, which can be formalized as reusable patterns. Since observations in different domains adhere to specific schemas, the resulting patterns are also to a large extent domain-specific.

Combining these patterns with decision trees to guide the process, LLMs are expected to perform better.

An additional challenge arises from implicit domain-expert knowledge, present in more than 46% of the variables (see Table 2), which is difficult for both non-experts and LLMs to capture. This knowledge has to be extracted in order to lead to a refined definition from which the decomposition should start. This process may reveal the need to introduce concepts for a matrix or additional constraints, and, in some cases, the omission of irrelevant information (like sensors and geographical positions). This explains the low precision values in the automatic metrics, since LLMs were instructed to stick to the definition as is.

Improvements were observed when refining the few-shot examples, particularly for the categories `hasProperty` and `hasObjectOfInterest`, where the additional contextual guidance helped the model produce more accurate variable representations.

Some slots, such as `hasMatrix` and `hasConstraints`, exhibited consistently low performance (see Table 5). The low performance related to the Matrix slot can be explained by the system representation for entities, as analysis of the semantic evaluation results shows that complex representations involving systems of entities are not captured naturally by the LLM without additional guidance. The system representation for entities was used only once in the ObjectOfInterest slot (29 occurrences in the corpus) and twice in the Matrix slot (7 occurrences in the corpus), with only one of these uses being correct. While ratios can be represented using simple Entities for `ObjectOfInterest` plus `Matrix`, flows cannot be represented without an asymmetric system at the `Matrix` slot, thus the flow concept is largely absent within the LLM-generated representations.

Recall, for both exact and close matches, is consistently higher than precision. Higher recall suggests that the number of false negative entities present in the corpus that are not detected by the model is relatively low. However, the models appear to be overly verbose, often generating incorrect values and thereby introducing noise into the decomposition. In particular, these effects persist even when the prompts explicitly instruct the models to avoid such behavior.

In addition, we did not observe significant differences (see Table 5) in performance between large and small models, with a difference of only about 4%. This raises the question of whether the use of smaller models is justified, especially when considering additional metrics such as processing time, memory consumption, and overall computational cost.

The `hasMatrix` slot exhibits a particular behavior: it shows higher hallucination rates than the other slots while still maintaining a high degree of text conformance with the variable definition. This finding appears to contradict the previous result, in which a higher text conformance was associated with improved precision. We hypothesize that this effect may be due to the specific characteristics of the `hasMatrix` field. As observed in the ground truth, `hasMatrix` is an ambiguous slot and may be subject to multiple interpretations. This ambiguity may result in lower performance, even when the model relies on the text present in the definition.

When comparing the two evaluations, it may appear surprising that the semantic evaluation yields lower values. The key difference is that the semantic evaluation was based on a complete LLM output assessment, rather than per individual slot.

In the automatic evaluation, each slot (e.g. Property, ObjectOfInterest, Matrix, Constraint) is scored independently, and precision, recall, and F1-scores are subsequently aggregated. As a result, a variable may still achieve a relatively high score even if one important slot is incorrect.

In contrast, the expert analysis evaluates each variable as an integrated whole. Classifications of Good or Acceptable indicate that the entire LLM-generated output is correct across all slots. Conversely, Weak or Wrong classifications are assigned whenever at least one critical slot (such as Property or ObjectOfInterest) is incorrect, even if the remaining slots are correct.

For this reason, the expert evaluation applies a substantially stricter criterion. The lower percentages observed in the expert evaluation compared to the automatic F1-scores are therefore expected. In the automatic evaluation, a variable with three correct slots out of four may still score well overall, whereas in the expert review a single incorrect but essential slot leads to a downgrade of the entire variable.

This difference in evaluation granularity also explains why per-component results are not directly comparable. For example, the automatic evaluation reports a hasProperty precision of approximately 38%, while the expert review yields an overall hasProperty accuracy of around 72%. These figures reflect different evaluation questions:

- Automatic metrics: How often is this individual slot correctly generated?
- Expert review: Is this variable acceptable for use as a whole?

Consequently, the expert analysis does not contradict the automatic evaluation, rather, it complements it by applying a human, end-to-end correctness criterion, which is necessarily more stringent.

Conclusions

Unambiguously representing scientific variables in datasets to improve interoperability remains challenging. The I-ADOPT Framework, intended as a domain-independent lingua franca, must be inherently flexible while maintaining sufficient precision to capture domain-specific requirements. This approach faces two key issues: (i) interpretability, since a variable can have multiple semantically correct representations, and (ii) scalability in generating I-ADOPT variables.

This paper proposes a corpus of 102 annotated multi-domain variables that can be used for two purposes: (i) as a benchmark to evaluate AI models capable of automatically generating I-ADOPT variables and (ii) as a training corpus for both AI models but also for human experts who intend to apply the framework. In this paper, we focus on the first approach, comparing the performance of different LLMs and analyzing the errors produced.

On the basis of the experimental results and the discussion, the following conclusions can be drawn.

Variability in I-ADOPT representations is the main challenge

Multiple valid representations—arising from the inherent complexity of the observed natural phenomena and their interpretation by humans—pose a fundamental challenge for the automatic generation of I-ADOPT variables. This issue is illustrated by the differences between the evaluation performed using the corpus and the validation performed by the expert. The discrepancy in the results indicates that there are multiple valid solutions beyond those represented in the corpus.

Prompt engineering improves performance, but not across slots

There is a discrepancy in the results in the different slots. The best results are observed for *hasProperty* and *hasObjectOfInterest*, indicating a better understanding of these slots by the models analyzed.

I-ADOPT slots remain intrinsically difficult for language models

Some slots, such as *hasMatrix* and *hasConstraints*, exhibit very low values. This indicates that the LLMs analyzed do not fully understand what text should be included in these slots. This is a common result across the models and prompts analyzed.

High difference in precision and recall

A higher recall compared to precision indicates that the models rarely miss relevant entities but introduce incorrect values. This verbosity persists despite explicit instructions to avoid it, suggesting some structural bias not identified.

Implicit knowledge may penalize the results

When refinements were required in the definitions in the I-ADOPT Corpus, the models failed in the decomposition, as demonstrated in the error analysis. In this benchmark study, models were required to adhere to the definition of the variable, without introducing any interpretations. The results show that slots with high conformance, when using text from definitions, generally achieve better F1-scores, with the exception of *hasMatrix*. High conformance implies that models do not need to add information to slots that is not present in the definition, a task that would require a deeper understanding of the slot semantics.

The use of larger models does not significantly improve the results.

As demonstrated in the experiments, the performance gain obtained by using larger models (32B) compared to smaller models (8B) is below 4%, indicating that increasing model size alone has a limited impact on performance for this task and that more targeted methodological improvements may be required.

Ambiguity in the variable descriptions constrains automation

The results indicate that improving model performance will require not only better prompts, but also additional supporting instructions on how to interpret and streamline ambiguous variable descriptions and on how to use the different slots including system of entities. These might be provided by formalized patterns in combination with decision trees to train the models to capture the required semantics in the different slots more accurately.

Based on these conclusions, we propose the following research directions as next steps:

- The documentation of formalized patterns to reduce the ambiguity of I-ADOPT variables. These patterns will help researchers create more uniform I-ADOPT variables and can also be used by LLMs to produce better results.
- The creation of additional variables to be added to the corpus. This will help to train more accurate models. In addition, the corpus can serve as a reference database for researchers to create new variables and converge toward consistent solutions.
- The application of additional model families and sizes beyond those used in this study will be explored. In particular, we plan to investigate BERT based models, such as SciBERT, including their fine tuning to apply them to some specific slots.

Acknowledgement

Funded by the European Union (under Grant Agreement No. 101057262 FAIR2Adapt). Views and opinions expressed are, however, those of the author(s) only and do not necessarily reflect those of the European Union or the granting authority. Neither the European Union nor the granting authority can be held responsible for them.

This work was also supported by the *NFDI4Earth* initiative, part of the National Research Data Infrastructure (NFDI), funded by the German Research Foundation (DFG). The views and opinions expressed are those of the author(s) and do not necessarily reflect those of the DFG or NFDI4Earth.

The authors wish to acknowledge the core members of the Research Data Alliance I-ADOPT Working Group, as listed at https://i-adopt.github.io/, as well as the reviewers of the variables, whose contributions are documented in the I-ADOPT Corpus catalogue: https://w3id.org/iadopt/corpus/.

The authors acknowledge the use of ChatGPT (OpenAI) for assistance in improving the English language. The authors remain fully responsible for the content of the article.

References

Akhtar, M., Benjelloun, O., Conforti, C., Foschini, L., Giner-Miguelez, J., Gijsbers, P., Goswami, S., Jain, N., Karamousadakis, M., Kuchnik, M., et al. (2024). Croissant: A metadata format for ml-ready datasets. *Advances in Neural Information Processing Systems*, *37*, 82133–82148.

Bhatt, A., Vaghela, N., & Dudhia, K. (2024). Generating knowledge graphs from large language models: A comparative study of gpt-4, llama 2, and bert [Accessed: 2025-12-XX]. *arXiv preprint arXiv:2412.07412*. https://arxiv.org/pdf/2412.07412

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., & Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, *26*.

Borgo, S., Ferrario, R., Gangemi, A., Guarino, N., Masolo, C., Porello, D., Sanfilippo, E. M., & Vieu, L. (2022). Dolce: A descriptive ontology for linguistic and cognitive engineering. *Applied ontology*, *17*(1), 45–69.

Choi, S., & Jung, Y. (2025). Knowledge graph construction: Extraction, learning, and evaluation. *Applied Sciences*, *15*(7), 3727.

Chue Hong, N. P., Katz, D. S., Barker, M., Lamprecht, A.-L., Martinez, C., Psomopoulos, F. E., Harrow, J., Castro, L. J., Gruenpeter, M., Martinez, P. A., Honeyman, T., Struck, A., Lee, A., Loewe, A., van Werkhoven, B., Jones, C., Garijo, D., Plomp, E., Genova, F., . . . WG, R. F. (2022, May). Fair principles for research software (fair4rs principles). https://doi.org/10.15497/RDA00068

Coussot, C., Braud, I., Chaffard, V., Boudevillain, B., & Galle, S. (2024). Implementing a new research data alliance recommendation, the i-adopt framework, for the naming of environmental variables of continental surfaces. *Earth Science Informatics*, *17*(5), 4261–4277. https://doi.org/10.1007/s12145-024-01373-9

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, 4171–4186.

Diepenbroek, M., Schindler, U., Huber, R., Pesant, S., Stocker, M., Felden, J., Buss, M., & Weinrebe, M. (2017). Terminology supported archiving and publication of environmental science data in pangaea. *Journal of Biotechnology*, *261*, 177–186. https://doi.org/10.1016/j.jbiotec.2017.07.016

Eaton, B., Gregory, J., Drach, B., Taylor, K., Hankin, S., Caron, J., Signell, R., Bentley, P., Rappa, G., Höck, H., et al. (2003). Netcdf climate and forecast (cf) metadata conventions. *URL: http://cfconventions. org/Data/cf-conventions/cf-conventions-1.8/cf-conventions. pdf*.

Galli, C., Donos, N., & Calciolari, E. (2024). Performance of 4 pre-trained sentence transformer models in the semantic query of a systematic review dataset on peri-implantitis. *Information*, *15*(2). https://doi.org/10.3390/info15020068

Gardent, C., Shimorina, A., Narayan, S., & Perez-Beltrachini, L. (2017). The webnlg challenge: Generating text from rdf data. *10th International Conference on Natural Language Generation*, 124–133.

Gil, Y., Garijo, D., Khider, D., Knoblock, C. A., Ratnakar, V., Osorio, M., Vargas, H., Pham, M., Pujara, J., Shbita, B., Vu, B., Chiang, Y.-Y., Feldman, D., Lin, Y., Song, H., Kumar, V., Khandelwal, A., Steinbach, M., Tayal, K., . . . Shu, L. (2021). Artificial intelligence for modeling complex systems: Taming the complexity of expert models to improve decision making. *11*(2). https://doi.org/10.1145/3453172

Graves, A. (2012). Long short-term memory. *Supervised sequence labelling with recurrent neural networks*, 37–45.

Haller, A., Janowicz, K., Cox, S. J., Lefrançois, M., Taylor, K., Le Phuoc, D., Lieberman, J., García-Castro, R., Atkinson, R., & Stadler, C. (2018). The modular ssn ontology: A joint w3c and ogc standard specifying the semantics of sensors, observations, sampling, and actuation. *Semantic Web*, *10*(1), 9–32.

Jonquet, C., Bouazzouni, S., Alviset, G., Fiore, N., Karam, N., Kihal, B., Pierkot, C., Pulieri, M., & Rosati, I. (2025). Federated fair semantic artefacts discovery and search with ontoportal federation. *International Semantic Web Conference*, 434–450.

Lafferty, J., McCallum, A., & Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Madin, J., Bowers, S., Schildhauer, M., Krivov, S., Pennington, D., & Villa, F. (2007). An ontology for describing and synthesizing ecological observation data. *Ecological informatics*, *2*(3), 279–296.

Magagna, B., & Chalk, S. (2025). I-adopt/corpus: V1.0.3. https://doi.org/10.5281/ZENODO.18105306

Magagna, B., Moncoiffé, G., Devaraju, A., Stoica, M., Schindler, S., Pamment, A., Environment Agency Austria, Austria/University of Twente, NL, National Oceanography Centre/British Oceanographic Data Centre, UK, Terrestrial Ecosystem Research Network (TERN), University of

Queensland, Australia, University of Colorado, Boulder, USA, Institute of Data Science, German Aerospace Centre (DLR), Germany, & National Centre for Atmospheric Science/UKRI, UK. (2022). Interoperable descriptions of observable property terminologies (i-adopt) wg outputs and recommendations. https://doi.org/10.15497/RDA00071

Magagna, B., Rosati, I., Stoica, M., Schindler, S., Moncoiffe, G., Devaraju, A., Peters, J., & Huber, R. (2021). The i-adopt interoperability framework for fairer data descriptions of biodiversity. *Proceedings of the 3rd International Workshop on Semantics for Biodiversity (S4BioDiv 2021), 2969*. https://ceur-ws.org/Vol-2969/paper10-s4biodiv.pdf

Mihindukulasooriya, N., Tiwari, S., Enguix, C. F., & Lata, K. (2023). Text2kgbench: A benchmark for ontology-driven knowledge graph generation from text. In T. R. Payne, V. Presutti, G. Qi, M. Poveda-Villalón, G. Stoilos, L. Hollink, Z. Kaoudi, G. Cheng, & J. Li (Eds.), *The semantic web – iswc 2023* (pp. 247–265). Springer Nature Switzerland.

Mo, B., Yu, K., Kazdan, J., Mpala, P., Yu, L., Kanatsoulis, C. I., & Koyejo, S. (2025). KGGen: Extracting knowledge graphs from plain text with language models. *The Thirty-ninth Annual Conference on Neural Information Processing Systems*. https://openreview.net/forum?id=YyhRJXxbpi

Moreau, L., & Groth, P. (2022). *Provenance: An introduction to prov*. Springer Nature.

Moreira, J., Pires, L. F., van Sinderen, M., Daniele, L., & Girod-Genet, M. (2020). Saref4health: Towards iot standard-based ontology-driven cardiac e-health systems (S. Borgo, P. Hitzler, & C. Shimizu, Eds.). *Applied Ontology*, *15*(3), 385–410. https://doi.org/10.3233/ao-200232

Moreira, J. L., Daniele, L. M., Ferreira Pires, L., van Sinderen, M. J., Wasielewska, K., Szmeja, P., Pawlowski, W., Ganzha, M., & Paprzycki, M. (2017). Towards IoT platforms' integration: Semantic translations between W3C SSN and ETSI SAREF. *Proceedings of the SEMANTiCS Conference 2017*. http://ceur-ws.org/Vol-2063/sisiot-paper3.pdf

Open Geospatial Consortium. (2023, May). *Ogc abstract specification topic 20: Observations, measurements and samples* (K. Schleidt & I. Rinne, Eds.; OGC Abstract Specification No. 20-082r4). Open Geospatial Consortium. http://www.opengis.net/doc/as/om/3.0

Rastegar, A., Magagna, B., Berrio, C., & Gonzalez, E. (2025). *Arvinrastegar/i-adopt-llm-based-service: Scientific variable benchmark* (Version V1.1-Experiment) [Software. Accessed: 2025-12-31]. Zenodo. https://doi.org/10.5281/zenodo.18108688

Riedel, S., Yao, L., & McCallum, A. (2010). Modeling relations and their mentions without labeled text. *Joint European conference on machine learning and knowledge discovery in databases*, 148–163.

Saeedizade, M. J., & Blomqvist, E. (2024). Navigating ontology development with large language models. *European Semantic Web Conference*, 143–161. https://doi.org/10.1007/978-3-031-60626-7_8

Schindler, S. (2025a). Sirkos/iadopt-catalogue: V0.1.0. https://doi.org/10.5281/ZENODO.18098560

Schindler, S. (2025b). Sirkos/iadopt-vis: V0.3.0. https://doi.org/10.5281/ZENODO.18097903

Shorten, C., Pierse, C., Smith, T. B., Cardenas, E., Sharma, A., Trengrove, J., & van Luijt, B. (2024). Structuredrag: Json response formatting with large language models. *arXiv preprint arXiv:2408.11061*. https://doi.org/10.48550/arXiv.2408.11061

Stoica, M., & Peckham, S. (2019). The scientific variables ontology: A blueprint for custom manual and automated creation and alignment of machine-interpretable qualitative and quantitative variable concepts. *Modeling the World's Systems Conference*.

Team, Q. (2025). Qwen3 technical report. https://arxiv.org/abs/2505.09388

Wang, D., & Iwaihara, M. (2025). Oskgc: A benchmark for ontology schema-based knowledge graph construction from text. *CEUR Workshop Proceedings*, *4041*.

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., da Silva Santos, L. B., Bourne, P. E., Bouwman, J., Brookes, A. J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C. T., Finkers, R., ... Mons, A. (2016). The fair guiding principles for scientific data management and stewardship. *Scientific Data*, *3*, 160018. https://doi.org/10.1038/sdata.2016.18

Zhong, L., Wu, J., Li, Q., Peng, H., & Wu, X. (2023). A comprehensive survey on automatic knowledge graph construction. *ACM Computing Surveys*, *56*(4), 1–62.

Appendix: Prompts used in the experiments

This appendix lists the prompt templates used for automated I-ADOPT variable decomposition. Prompts are identical across all variables and model configurations, except for the number of few-shot examples provided.

strict_minimal

```
Follow the JSON-Schema exactly. Do not
    infer or invent new concepts.

definition must be exactly the same string
    as provided.
comment = short summary of the definition.
    Do not add new ideas.

hasProperty = the main measurable property
    in the definition.
hasObjectOfInterest = the thing that has
    this property.
hasMatrix = the medium in which the object
    occurs. Never a method or location.

If a required key is not in the definition,
    output an empty string for it.

Output only the JSON object.

Additional rules:

 Only extract what is explicitly stated in
    the definition.
 hasProperty = the main measurable
    characteristic.
 hasObjectOfInterest = the thing that has
    that characteristic.
 hasMatrix = medium the object is in, only
    if directly stated.
 hasConstraint = only conditions explicitly
    stated.
 If unsure: leave it empty. Do not guess.
```

constraint_decision_tree

```
Follow the JSON-Schema exactly. Do not
    infer or invent new concepts.

definition must be exactly the same string
    as provided.
comment = short summary of the definition.
    Do not add new ideas.

hasProperty = the main measurable property
    in the definition.
hasObjectOfInterest = the thing that has
    this property.
hasMatrix = the medium in which the object
    occurs. Never a method or location.

If a required key is not in the definition,
    output an empty string for it.

Output only the JSON object.

Extraction order:

1. Copy definition exactly.
2. Extract hasProperty (main measurable
    characteristic).
3. Extract hasObjectOfInterest (entity with
    that property).
4. Extract hasMatrix only if the definition
    states a medium.
5. Extract hasConstraint last:
    Only explicit limiting phrases.
    label = short phrase
    on = EXACT string from hasProperty or
    an entity
6. Never paraphrase or introduce new
    concepts.
```

matrix_decision_tree

```
Follow the JSON-Schema exactly. Do not
    infer or invent new concepts.

definition must be exactly the same string
    as provided.
comment = short summary of the definition.
    Do not add new ideas.

hasProperty = the main measurable property
    in the definition.
hasObjectOfInterest = the thing that has
    this property.
hasMatrix = the medium in which the object
    occurs. Never a method or location.

If a required key is not in the definition,
    output an empty string for it.

Output only the JSON object.

Decision rules:

1. Identify hasProperty first.
2. Identify hasObjectOfInterest:
    the entity that carries the property.
3. Identify hasMatrix only if the
    definition clearly states
   the medium or material the object is
    inside.
4. If a phrase describes a condition/state,
    not a medium:
    put it in hasConstraint.
5. Never use methods, units, instruments,
    or locations.
```

JSON-SCHEMA

```
{
  "$schema": "https://json-schema.org/draft
    /2020-12/schema",
  "$id": "https://example.org/schemas/iadopt
    -variable.json",
  "title": "I-ADOPT Variable (Decomposed
    Form)",
  "description": "A single scientific
    variable structured according to the I-
    ADOPT framework. This compact JSON model
    is used for LLM benchmarking and
    represents: label, natural-language
    definition, cleaned comment, property,
    objects/entities, matrices, context
    objects, and constraints.",
  "type": "object",

  "required": ["label", "definition", "
    comment", "hasProperty", "
    hasObjectOfInterest"],

  "properties": {
    "label": {
      "type": "string",
      "description": "Human-readable name of
     the variable (rdfs:label)."
    },
    "definition": {
      "type": "string",
      "description": "Full natural-language
    description."
    },
    "comment": {
      "type": "string",
      "description": "Use a short summary of
     the definition. Do not add new concepts
    ."
    },
    "hasProperty": {
      "type": "string",
      "description": "The main measurable
    property in the definition."
    },
    "hasStatisticalModifier": {
      "type": "string",
      "description": "Optional statistical
    qualifier (e.g., 'maximum', 'minimum', '
    median')."
    },
    "hasObjectOfInterest": {
```

```
    "$ref": "#/$defs/entityOrSystem",
    "description": "The thing that has the
   property."
  },
  "hasMatrix": {
    "$ref": "#/$defs/entityOrSystem",
    "description": "Medium in which the
  object occurs. Not a process or location
  ."
  },
  "hasContextObject": {
    "$ref": "#/$defs/entityOrSystem",
    "description": "Optional contextual
  Entity or System that provides
  environmental or situational context (e.g
  ., ’air’, ’atmosphere’)."
  },
  "hasConstraint": {
    "type": "array",
    "description": "List of Constraints
  describing states, conditions, purity,
  normalization, or other limiting
  qualifiers and or quantifiers that apply
  to the Entity.",
    "items": {
      "type": "object",
      "required": ["label", "on"],
      "properties": {
        "label": {
          "type": "string",
          "description": "Short cleaned
  phrase describing the restriction (e.g.,
  ’dry’, ’purity 99.98%’, ’5.00 g sample’,
  ’per mol’)."
        },
        "on": {
          "type": "string",
          "description": "What the
  constraint applies to (The name of a
  Property or Entity in this variable, e.g.
   ’distance’, ’mass flux’, ’habitat patch
  ’, ’organism’)."
        }
      },
      "additionalProperties": false
    },
    "minItems": 1
  }
},

"additionalProperties": false,

"$defs": {
  "entityOrSystem": {
    "description": "An Entity or System
  involved in the variable. It may be a
  simple entity (e.g., ’hexanol’, ’air’, ’
  soil’) or a structured system (asymmetric
   or symmetric).",
    "oneOf": [
      { "type": "string",
      "description": "A simple entity
  label (e.g., ’air’, ’soil’, ’nitrogen’)."
    },
      {
        "$comment": "Asymmetric system,
  from A to B",
        "type": "object",
        "required": [
            "AsymmetricSystem",
            "hasSource",
            "hasTarget",
            "hasNumerator",
            "hasDenominator"
          ],
          "properties": {
            "AsymmetricSystem": { "type": "
   string" },
            "hasSource":        { "type": "
   string" },
            "hasTarget":        { "type": "
   string" }
          },
          "additionalProperties": false
        },
        {
          "$comment": "Symmetric system, A
   and B together form a system",
          "type": "object",
          "required": ["SymmetricSystem", "
   hasPart"],
          "properties": {
            "SymmetricSystem": { "type": "
   string" },
            "hasPart": {
              "type": "array",
              "items": { "type": "string" },
              "minItems": 1
            }
          },
          "additionalProperties": false
        }
      ]
    }
  }
}
```