# Provenance-Driven Nanopublications: Representing Source Lineage and Trust Networks for Multi-Source Assertions

Laura Menotti[1*], Stefano Marchesin[1], Fabio Giachelle[1], Gianmaria Silvello[1]

[1]Department of Information Engineering, University of Padova, Padova, Italy.

*Corresponding author(s). E-mail(s): laura.menotti@unipd.it;
Contributing authors: name.surname@unipd.it;

## Abstract

**Purpose:** Nanopublishing is a paradigm enabling the representation of scientific claims in a distinctive, identifiable, citable, and reusable format, i.e., as a named graph. This approach can be applied to sentences extracted from scientific publications or triples within a Knowledge Base (KB). This way, one can track the provenance of assertions derived from a specific publication or database. However, nanopublications do not natively support multi-source scientific claims generated by aggregating different bodies of knowledge.

**Methods:** This work extends the nanopublication model with *knowledge provenance*, capturing provenance information for assertions derived by an aggregation algorithm or a truth discovery process, e.g., an information extraction system aggregating several sources of knowledge to populate a Knowledge Base (KB). In these cases, provenance information cannot be attributed to a single source, but it is the result of an ensemble of evidence, that can comprehend supporting and conflicting pieces of evidence and truth values. Knowledge provenance is represented as a named graph following the PROV-K ontology, developed for the case. To show how knowledge provenance applies to a real-world scenario, we serialized gene expression-cancer associations generated by the Collaborative Oriented Relation Extraction (CORE) System. To demonstrate the value of trust relationships, we present a use case leveraging an existing scientific KB to construct a trust network employing three Large Language Model (LLM) agents. We analyzed the ability of LLMs to evaluate trustworthiness, exploiting techniques from KB accuracy estimation.

**Results:** We published **197, 511** assertions generated by the CORE system in the form of extended nanopublications with knowledge provenance. PROV-K also defines trust relationships between agents or between an agent and a proposition. Starting from these assertions, we leveraged external agents – namely, multiple LLMs – to assess their trusted truth value. Based on these values, we defined trust relationships between the agents and the facts, yielding an exemplar trust network comprising over 45,000 facts and four agents.

**Conclusion:** The *knowledge provenance* graph allows the tracking of provenance for each piece of evidence contributing to the support or refutation of an assertion. To capture the semantics of the newly presented graph, we define the PROV-K ontology, designed to represent provenance information for multi-source assertions. The two use cases serve as a template to show how to serialize extended nanopublications and showcase the trust relationships' capabilities.

**Keywords:** Nanopublications, Knowledge Provenance, Data Provenance, Knowledge Bases, Gene-Cancer Associations, Trust Relationships.

# 1 Introduction

Semantic Web technologies enhance data integration and interoperability, allowing access to information in a machine-readable format, i.e., Resource Description Framework (RDF). Given the high volume of publications, scientific evidence is usually extracted automatically from the literature and shared through Knowledge Bases (KBs) [1]. KBs are understood by machines and humans, making them suitable for various applications, such as Data Mining, Natural Language Processing, and Search and Recommendation [2].

In today's increasingly connected scientific landscape, KBs play a central role in enabling both humans and machines to discover, validate, and reason over scientific facts. A key aspect of building such scientific knowledge networks is extracting and integrating factual assertions from multiple, often diverse, scientific publications. This supports the aggregation of evidence and allows for creating a structured framework where each scientific claim can be supported or challenged by one or more sources. Ensuring that individual statements are independently accessible and traceable is essential for building a unified, trustworthy resource that supports transparent scientific discovery and reproducibility.

The current best approach to representing scientific assertions or facts in a machine-readable manner is through the use of nanopublications, as they enable the precise identification, representation, access, and citation of individual assertions [3, 4]. To the best of our knowledge, nanopublications are the only data model that supports publishing at the granularity of individual statements with such detailed context, enabling provenance traceability, interoperability, and precise citation. Nanopublications have gained widespread adoption, particularly within the life sciences domain [5–7]. A nanopublication consists of three distinct components, structured as separate named graphs: the assertion itself, its provenance, and associated metadata. This structure is purposefully designed to support the representation of one assertion from a single evidence source.

However, representing assertions derived from multiple sources using nanopublications is not currently feasible, as the information provenance they capture only identifies the origin of a represented fact. Indeed, the provenance graph in a nanopublication specifies only that a given assertion originates from a particular publication, including metadata such as the extraction method, the publisher, venue, and year of publication.

Consider, for instance, a gene–disease association stating that the gene `BRAF` is an `oncogene` for `mammal neoplasms`. Ten different publications may support this fact, while two other studies may provide refuting evidence. A standard nanopublication can only represent the fact as originating from a single source and lacks the capacity to express that the assertion is derived from multiple, potentially conflicting, sources.

In this paper, we focus on representing a large collection of scientific facts that result from multi-source assertions, which go beyond the expressive capabilities of the standard nanopublication model. As discussed earlier, while nanopublications provide a robust framework for encoding individual scientific assertions along with their metadata and source-specific information provenance [3, 4], they are inherently limited to representing assertions derived from a single source. This limitation makes it difficult to model more complex cases in which a scientific fact is supported by multiple, and possibly conflicting, pieces of evidence.

To address this challenge, we consider CoreKB, a large-scale knowledge discovery platform designed to manage and represent scientific assertions extracted from multiple sources [8].[1] CoreKB stores over 230, 000 gene expression–cancer associations, automatically generated by the Collaborative Oriented Relation Extraction (CORE) system [9]. These associations are derived by mining scientific literature and aggregating evidence from multiple documents to produce comprehensive, trustworthy scientific facts.

The CORE system identifies and extracts relevant statements from various articles, forming what is defined as a Gene–Cancer Status (GCS). A GCS is not based on a single claim from a single publication, but rather synthesized from multiple sources, potentially containing both supporting and refuting evidence. Aggregating this evidence enables the system to formulate a fact that reflects the most likely scientific consensus, thereby increasing its reliability. However, the standard

[1] https://gda.dei.unipd.it/

nanopublication model lacks the means to represent the detailed provenance of such aggregated knowledge, including the origins of each contributing assertion and their respective levels of support or conflict.

Therefore, it is necessary to extend the nanopublication model to track the knowledge provenance – the provenance of the scientific fact itself, as inferred from multiple supporting assertions. In this work, we focus on representing the knowledge provenance of a scientific fact, as derived from the collection of assertions that justify trusting that fact. To this end, we propose the extended nanopublication model, which introduces a fourth named graph dedicated specifically to capturing knowledge provenance. This extension remains fully compatible with the standard nanopublication model while enhancing it to support multi-source evidence aggregation and traceability.

***Contributions.***

This article builds upon and extends our prior work [10], with several contributions in the field.

First, we expand the nanopublication model to suit the needs of multi-source assertions by introducing an additional component to its structure called *knowledge provenance*. This novel graph represents the different sources that support or conflict with the given assertion. Additionally, one can include details about the reliability of each assertion, considering both the source of information and the extraction process. The proposed expansion is backward-compatible with the already-published nanopublications, as the *knowledge provenance* component is independent of the other components and is optional.

Similarly to the other components of a nanopublication, the *knowledge provenance* is modeled as an additional named graph referring to the newly developed PROV-K ontology to represent the provenance information of web assertions derived from multiple sources of evidence. Although we developed the ontology to represent the *knowledge provenance* component of nanopublications, the PROV-K ontology can be used in diverse contexts where one needs to represent provenance information of multi-source assertions. The PROV-K ontology is built upon the PROV Ontology (PROV-O) and is grounded in the literature defining knowledge provenance [11–13]. The PROV-K ontology is available in Zenodo [14], and its complete documentation is available at: https://prov-k.dei.unipd.it/ontology/.

Thirdly, we serialized 197,511 extended nanopublications representing GCS in CoreKB. Additionally, we integrated the extended nanopublications into the CoreKB platform to enable their visualization and download. We published the serialized nanopublications in Zenodo for everyone to download in bulk [15]. We released the source code for building the extended nanopublications as a template for future applications on different resources. The code can be accessed at the GitHub repository: https://github.com/mntlra/knowledgeProvenance.

Finally, we introduce a synthetic use case of the PROV-K that leverages trust relationships. To highlight their value, we construct a trust network comprising a subset of 45,649 GCSs generated by CORE and three Large Language Model (LLM) agents. This graph is available in Zenodo [16] and can be used for analytics and to assess the trustworthiness of each GCS, as evaluated by external agents not involved in its creation.

To summarize, our contributions include:

1. An extended nanopublication model expanded with *knowledge provenance* for multi-sourced assertions (Section 3);
2. The PROV-K ontology, representing provenance for multi-sourced assertions and describing their uncertainty (Section 3);
3. A large-scale application of the extended nanopublication model employing 197,511 facts from CoreKB, an in-use and large-scale knowledge discovery platform (Section 4);
4. An extension of the CoreKB platform to enable the visualization and download of each GCS as an extended nanopublication (Section 4);
5. An application of the PROV-K trust relationship exploiting 45,649 CoreKB GCSs to build a trust network, enabling fact discovery enriched with reliability scores provided by external agents (Section 5).

***Outline.***

The rest of this work is organized as follows. Section 2 introduces the original nanopublication model and describes previous efforts in data, information, and knowledge provenance. In addition, Section 2.2 describes the CORE system and CoreKB platform, two foundational components for the use cases. Section 3 illustrates the extended nanopublication model and defines the PROV-K ontology. Section 4 presents a large-scale application of the extended nanopublication model where the GCSs generated by the CORE system are serialized as extended nanopublications. Section 5 presents an application of trust relationships. Trust relationships are first derived from external agents, namely LLMs, and their accuracy is evaluated (Subsection 5.1). The resulting trust network for CoreKB is then serialized, and its utility is demonstrated through queries over the network (Subsection 5.2). Section 6 draws some final remarks and concludes the paper.

# 2 Background

## 2.1 Related Work

***Nanopublication model.***

Since its introduction, the nanopublication model has been designed to facilitate data integration and exchange, improve the accessibility and comprehension of scientific statements, and enable citations in the granularity of individual claims [3, 4]. In this conceptual framework, a scientific publication can be divided into single statements or assertions, with each assertion encapsulated in a distinct nanopublication containing all pertinent information about that specific claim. The nanopublication model uses Semantic Web technologies to represent scientific claims in a distinctive, identifiable, citable, and reusable format.

Notably, nanopublications utilize named graphs [17], which extend the foundational semantics of RDF by employing *quads* instead of triples. From a technical viewpoint, a nanopublication is a named graph that comprises four basic components; each represented as a named graph itself: (i) the *assertion graph*, containing the scientific assertion; (ii) the *provenance graph*, containing information about where the assertion comes from and how it has been defined; (iii) the *publication info graph*, containing all the metadata of the nanopublication, such as who curated it and when it was created; (iv) the *head* graph, which defines the nanopublication and connects all the other components together. Figure 1 shows an example of a nanopublication representing a CoreKB gene expression-cancer association.[2] All prefixes used in the figure are defined in Section 2.3. The *assertion graph* consists of the GCS itself in RDF format. The *publication information* graph represents the metadata of the nanopublication, such as its subject, license, and data used. The *provenance graph* describes how the assertion was derived and who generated it.

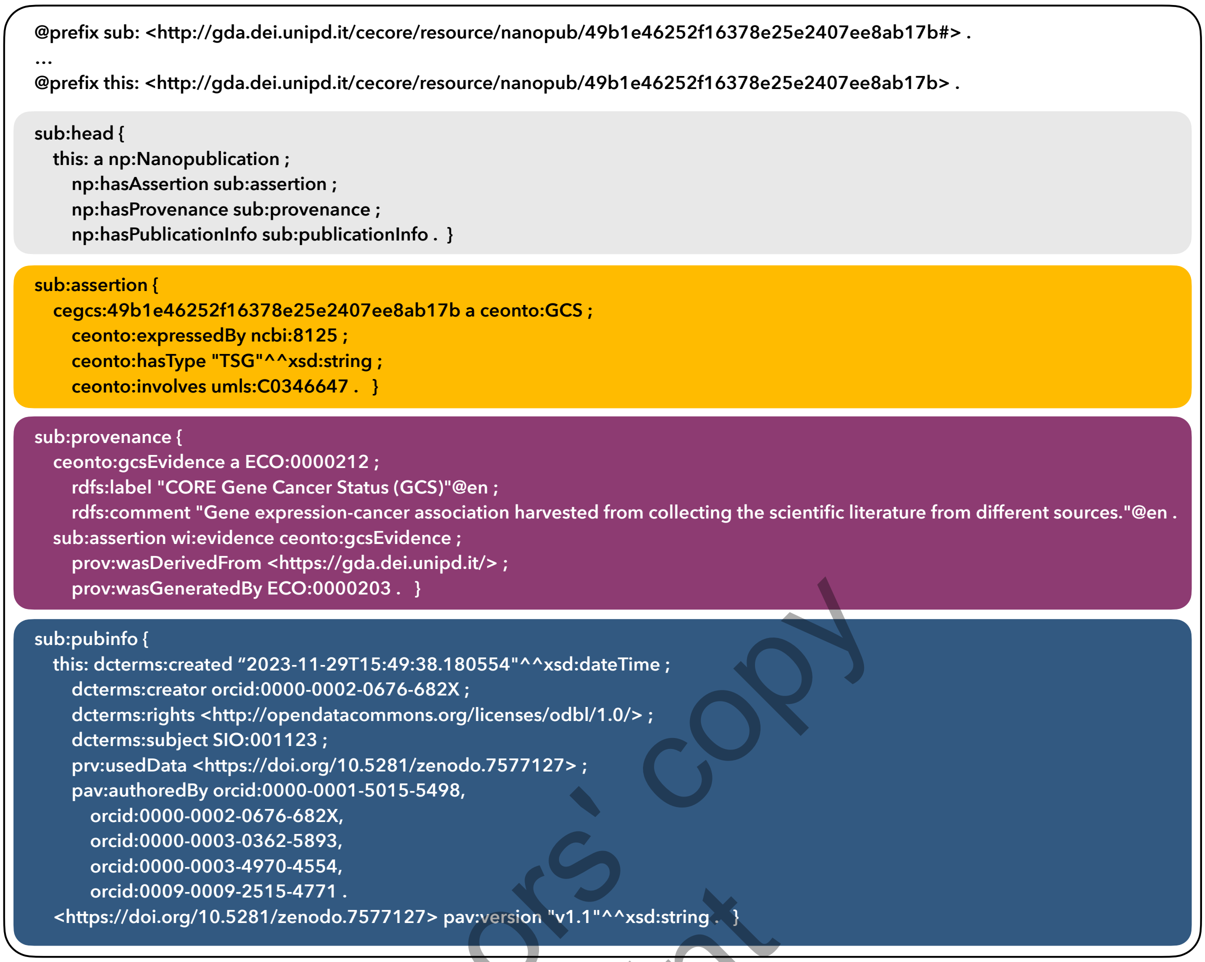
More in detail, the RDF graph in Figure 1 describes a nanopublication whose base URL is `.../49b1e46252f16378e25e2407ee8ab17b`. In the head of the nanopublication, identified by the prefix `sub:head`, the resource is declared to be an instance of `np:Nanopublication` and is linked to three primary components: the assertion, the provenance, and the publication information. These components are referenced by `sub:assertion`, `sub:provenance`, and `sub:publicationInfo`, respectively.

The assertion component, contained within `sub:assertion` named graph, encapsulates the core scientific claim. Here, the assertion resource `cegcs:49b1e46252f16378e25e2407ee8ab17b` is established as an instance of `ceonto:GCS` (Gene Cancer Status). It is further specified that the assertion is *expressed by* `ncbi:8125` (ANP32A) and is assigned a type (given as the string “TSG”, i.e., Tumor Suppressor Gene) that refers to a specific biological category. Additionally, the assertion involves the disease denoted by `umls:C0346647` (malignant neoplasm of pancreas).

The provenance component, indicated by `sub:provenance`, documents the evidence that underpins the assertion. In this section, the resource `ceonto:gcsEvidence` is introduced, which is an instance of class `Combinatorial Evidence` from Evidence and Conclusion Ontology (ECO) [3] and is labeled as “CORE Gene Cancer Status (GCS)” in English. This label accompanies a comment explaining that the gene expression–cancer association was harvested by collecting scientific literature from diverse sources. The graph links the assertion to this evidence through the property `wi:evidence` and further provides that the evidence was derived from CoreKB (https://gda.dei.

[2] http://gda.dei.unipd.it/cecore/resource/GCS#49b1e46252f16378e25e2407ee8ab17b

[3] http://purl.obolibrary.org/obo/ECO_0000212



```
@prefix sub: <http://gda.dei.unipd.it/cecore/resource/nanopub/49b1e46252f16378e25e2407ee8ab17b#> .
...
@prefix this: <http://gda.dei.unipd.it/cecore/resource/nanopub/49b1e46252f16378e25e2407ee8ab17b> .

sub:head {
  this: a np:Nanopublication ;
    np:hasAssertion sub:assertion ;
    np:hasProvenance sub:provenance ;
    np:hasPublicationInfo sub:publicationInfo .  }

sub:assertion {
  cegcs:49b1e46252f16378e25e2407ee8ab17b a ceonto:GCS ;
    ceonto:expressedBy ncbi:8125 ;
    ceonto:hasType "TSG"^^xsd:string ;
    ceonto:involves umls:C0346647 .   }

sub:provenance {
  ceonto:gcsEvidence a ECO:0000212 ;
    rdfs:label "CORE Gene Cancer Status (GCS)"@en ;
    rdfs:comment "Gene expression-cancer association harvested from collecting the scientific literature from different sources."@en .
  sub:assertion wi:evidence ceonto:gcsEvidence ;
    prov:wasDerivedFrom <https://gda.dei.unipd.it/> ;
    prov:wasGeneratedBy ECO:0000203 .   }

sub:pubinfo {
  this: dcterms:created “2023-11-29T15:49:38.180554"^^xsd:dateTime ;
    dcterms:creator orcid:0000-0002-0676-682X ;
    dcterms:rights <http://opendatacommons.org/licenses/odbl/1.0/> ;
    dcterms:subject SIO:001123 ;
    prv:usedData <https://doi.org/10.5281/zenodo.7577127> ;
    pav:authoredBy orcid:0000-0001-5015-5498,
      orcid:0000-0002-0676-682X,
      orcid:0000-0003-0362-5893,
      orcid:0000-0003-4970-4554,
      orcid:0009-0009-2515-4771 .
  <https://doi.org/10.5281/zenodo.7577127> pav:version "v1.1"^^xsd:string .  }
```

**Figure 1** A nanopublication representing a scientific assertion from CoreKB serialized in TriG format. Each component of the nanopublication is highlighted with a different colour. In particular, the *head graph* is depicted in grey, the *assertion graph* in yellow, the *provenance graph* in purple, and the *publication info graph* in blue.

unipd.it/) and generated by the process identified with class `Automatic Assertion` from ECO [4].

Finally, the publication information component, denoted by `sub:pubinfo`, supplies detailed metadata about the nanopublication. It records the creation date and time as `2023-11-29T15:49:38.180554` and identifies the creator via the ORCID identifier `orcid:0000-0002-0676-682X`. The rights associated with the nanopublication are specified by referring to the Open Data Commons Open Database License (http://opendatacommons.org/licenses/odbl/1.0/). In addition, the subject of the nanopublication is classified according to `SIO:001123` (Gene-disease association linked with altered gene expression). The nanopublication also references external data used during its generation, linking to the dataset at https://doi.org/10.5281/zenodo.7577127, and records that this dataset (version “v1.1”) was authored collaboratively by several researchers (indicated by multiple ORCID identifiers).

In summary, this named graph provides a structured, machine-readable representation of a nanopublication that encapsulates a gene–cancer association assertion, details the evidence and provenance supporting the claim, and includes comprehensive publication metadata.

[4] http://purl.obolibrary.org/obo/ECO_0000203

### *Nanopublications in-use.*

The nanopublication model has been used to represent statements from different fields, especially in the life science domain. Chichester et al. [18] created nanopublications from scientific facts associated with more than 38K proteins stored in the neXtProt database.[5] This approach showed that using the nanopublication model for the neXtProt database eases access to its information and can be a useful tool for expanding biological research [5]. Queralt-Rosinach et al. [7] published the contents of the DisGeNET database as nanopublications to provide a Linked Data resource.[6] Waagmeester et al., in [6], described their endeavors in converting WikiPathways, an online collaborative pathway resource, into nanopublications.[7] Vogt et al. [19] employed nanopublications to organize knowledge graphs into semantically meaningful representation units. In this case, nanopublications enhance FAIRness by supporting the FAIR Guiding Principles by creating machine-actionable and semantically interoperable data and metadata. Furthermore, nanopublications offer a structured approach to making statements about statements, which is essential for accurately documenting provenance and attribution. They also introduce modularity into knowledge graphs, enabling more flexible data management and facilitating graph alignment. In [20], nanopublications were used not only to represent scientific claims but also to model the entire publication process, including submissions, peer reviews, author responses, and editorial decisions. This was achieved through a field study involving formalization papers, where participants formalized existing claims and submitted them for review. The results demonstrated both the technical and practical viability of using nanopublications for scholarly communication.

Overall, there are more than 10M nanopublications publicly accessible worldwide [21]. Representing data as nanopublications enhances data-intensive science and allows fact discovery by exploiting machine-readable information [22]. Concerning the aggregation of multiple nanopublications, Bucur et al. [23] propose an approach in which nanopublications represent snippets of scientific articles related to the same publication are interlinked, utilizing properties such as *refersTo*. Although the unifying model proposed in [23] is relevant to our study, it still does not consider the reliability of an assertion and the supporting or conflicting relationships between pieces of information. The concept of nanopublications has already been expanded in [24]. Here, the *assertion graph* has been extended to account for English sentences representing textual scientific claims following a semantic scheme called AIDA (Atomic, Independent, Declarative, Absolute). However, we are interested in machine-readable representations like the nanopublication model.

Thus, in the era of truth discovery algorithms and automatic information extraction, the nanopublication model fails to represent data reliability and the provenance of assertions constituted by an ensemble of contrasting and supporting evidence. In this regard, Clark et al. [25] formalized the micropublication model, which represents empirical evidence beyond statement-based models like nanopublications. The proposed model offers a representation of biomedical evidence with particular interest in building claim networks and their lineage. Although related, the main difference between nanopublications and micropublications is that the latter are tailored for biological processes, including methods and materials specifications, discussion and commentary, and reproducibility and verifiability in research. Although our use case pertains to biomedical information, the objective of this study is to model knowledge provenance for assertions in a domain-agnostic manner. Consequently, the nanopublication model is better aligned with our goals than the domain-specific micropublication model. In addition, the micropublication model represents the claim of a statement in a textual form, like for AIDA nanopublications [24].

### *A Hierarchy of Provenance.*

The Data–Information–Knowledge–Wisdom (DIKW) pyramid is a widely recognized model to represent information and knowledge within management systems [26]. It describes the processes involved in the data transformation, from a piece of data to the wisdom embedded in it. Each step adds value to the final results, starting from raw *Data*, where one can extract *Information*, to *Wisdom*, that is the application of

[5] https://www.nextprot.org/

[6] https://www.disgenet.com

[7] https://github.com/wikipathways/nanopublications

*Knowledge* acquired from the information block. We establish a connection between the DIKW pyramid and provenance. In earlier studies, the initial level, known as *Data Provenance*, has received extensive attention within databases. Its primary emphasis lies in tracing the data lineage in response to a query [27, 28]. In this context, provenance encompasses the origin and the pathway through which a specific piece of data was introduced into the given database. Over the years, various conceptualizations of provenance have been proposed and explored, such as "why-provenance," "where-provenance," and "how-provenance" [27, 28].

The second level concerns *Information Provenance*, which represents the provenance of assertions inferred from data. This is embedded in the provenance graph of the nanopublication model and has been studied in the context of the Semantic Web. Provenance on the Semantic Web comprises metadata representing the creation and publication of resources. The PROV Ontology (PROV-O) provides a formal language to encode provenance information in a machine-readable format.[8] It is based on the PROV Data Model (PROV-DM) and the Open Provenance Model (OPM) [29].[9] While extensive in scope, the PROV-O models provenance as in the provenance graph of the nanopublication model; therefore, it lacks the representations for supporting and contradicting evidence and reliability scores.

The third level, called *Knowledge Provenance*, is the focus of this work, and it has been studied in different works by Fox and Huang [12, 13, 30, 31]. Knowledge Provenance (KP) has been proposed to create an approach to annotate the reliability of information extracted from web sources based on who created the assertion, how much the creator can be trusted, and what the information depends on. Little work has been done towards this end, and it mostly focuses on providing a taxonomy of four levels of provenance based on the certainty degree of each assertion [30]. Below, we describe each KP level in detail.

*Static KP (Level 1)* describes assertions for which the truth value does not change over time [30]. In Static KP, the truth value can be "True", "False", or "Unknown", where the latter is used to handle uncertain information. The *Static Knowledge Provenance Ontology* defines a taxonomy of proposition types and a set of axioms allowing the development of a reasoner to assess truth values based on different cases. Although the ontology has been formalized in [11], it is not available as a resource.

*Dynamic KP (Level 2)* allows the validity of information to change over time, with the introduction of timestamps and time intervals to model when the truth value was assigned and the validity period of an assertion [12].

*Uncertainty-oriented KP (Level 3)* considers truth values and relationships that are uncertain, e.g., a web assertion that *may* be attributed to a given source. In this case, every assertion also has an "assigned certainty degree", i.e., the probability distribution of the truth value given by the creator of the assertion, and a "certainty degree", which is evaluated by the requester [13].

*Judgment-based KP (Level 4)* investigates the case in which provenance is supported by social processes, e.g., truth propagation in social networks [31].

The fourth level, *Wisdom Provenance*, focuses on keeping track of the provenance of the wisdom inferred from knowledge or applications exploiting such knowledge, which is still unexplored in the literature.

## 2.2 The CORE System

CORE is a Knowledge Base Construction (KBC) system based on the combination of ML-based models and domain-expert feedback [9]. CORE harvests text from the literature, identifies sentences containing pairs of relevant entities, and extracts fine-grained aspects from them to generate gene expression-cancer associations that can be published as facts – defined as Gene–Cancer Status (GCS). The sentence-level annotations performed by CORE comprise three different aspects: Change of Gene Expression (CGE), which represents the change of the gene expression level (`up`, `down`, `not informative`); Change of Cancer Status (CCS), which represents the change of cancer status (`progression`, `regression`, `not informative`); and Gene-Cancer Interaction (GCI), which indicates the interaction occurring between CGE and CCS (`causality`,

[8] http://www.w3.org/TR/2013/REC-prov-o-20130430/
[9] https://www.w3.org/TR/prov-dm/

`correlation, not informative`). For each fact, CORE aggregates the probabilities of these three aspects to assign a probability to three mutually exclusive gene classes: `oncogene`, `tumor suppressor gene`, and `biomarker`. Subsequently, the facts are validated by a two-stage reliability test. For each GCS, CORE verifies that the fact has sufficient evidence (sufficiency condition) and that mutually exclusive classes are not similarly probable (consistency condition). That is, the system assesses the degree of contradictory evidence. A fact generated by CORE passes the sufficiency checks if the probability of CCS and GCI being not informative is below a threshold value $\alpha$ set to 0.7. The consistency test instead checks whether the difference between the probabilities of the fact being classified with the two gene classes with the highest likelihood is bigger than a threshold value $\beta$ set to 0.4. The values for $\alpha$ and $\beta$ were set empirically. Facts classified as unreliable are fed back to domain experts for manual annotation in an active learning fashion. For technical details and the evaluation of the CORE system, we resort the interested reader to [9].

***CoreKB Web Platform.***

The data extracted by CORE is then ingested by CoreKB [8],[10] a web platform to search and explore each GCS. CoreKB contains information about 23,879 genes and 11,530 diseases for a total of more than 230,000 fine-grained facts supported by 1,037,845 sentences from 251,038 research articles. Figure 2 shows the GCS card displayed by the CoreKB platform for the example presented in Figure 1. Each GCS comprises information about the gene and disease involved, which are identified by the National Center for Biotechnology Information (NCBI) Gene IDs and the Unified Medical Language System (UMLS) Concept Unique Identifiers (CUIs) respectively, together with the assigned gene class. In addition, each GCS is linked to the sentences supporting the fact, i.e., identifying the same gene class, and those conflicting with it. For each sentence, provenance information includes the PubMed ID of the article from which the sentence has been extracted and the year of publication of such an article. CoreKB includes reliable and unreliable facts classified during the two-stage reliability test by CORE. Based on the specific unmet condition, unreliable facts are further divided into unreliable facts due to insufficient evidence or due to low consensus (contrasting evidence).

## 2.3 Recurring Prefixes

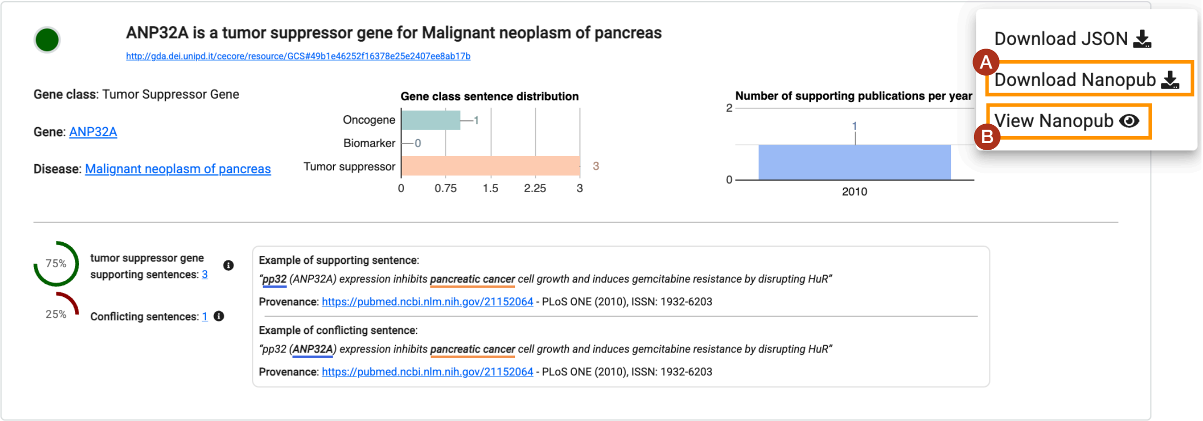
In the following, we provide the list of prefixes that are used in the figures and examples.

| | |
|---|---|
| **cegcs**: | http://gda.dei.unipd.it/cecore/resource/GCS# |
| **ceonto**: | http://gda.dei.unipd.it/cecore/ontology/ |
| **cesent**: | http://gda.dei.unipd.it/cecore/resource/Sentence# |
| **corekp**: | http://gda.dei.unipd.it/cecore/resource/nanopub/PROV-K/ |
| **dc**: | http://purl.org/dc/elements/1.1/ |
| **dcterms**: | http://purl.org/dc/terms/ |
| **ECO**: | http://purl.obolibrary.org/obo/ECO_ |
| **ncbi**: | https://www.ncbi.nlm.nih.gov/gene/ |
| **np**: | http://www.nanopub.org/nschema# |
| **orcid**: | https://orcid.org/ |
| **pav**: | http://purl.org/pav/ |
| **pk**: | https://w3id.org/PROV-K/ontology/schema/ |
| **prov**: | http://www.w3.org/ns/prov# |
| **prv**: | http://purl.org/net/provenance/ns# |
| **rdfs**: | http://www.w3.org/2000/01/rdf-schema# |
| **SIO**: | http://semanticscience.org/resource/SIO_ |
| **umls**: | http://linkedlifedata.com/resource/umls/id/ |
| **wi**: | http://purl.org/ontology/wi/core# |
| **xsd**: | http://www.w3.org/2001/XMLSchema# |

# 3 Extended Nanopublication Model

Motivated by the discussion above, we introduce a novel component to the original nanopublication model called *knowledge provenance graph*. The *knowledge provenance graph* represents provenance information for multi-sourced assertions, tracking the supporting or conflicting evidence

[10] https://gda.dei.unipd.it/



ANP32A is a tumor suppressor gene for Malignant neoplasm of pancreas

http://gda.dei.unipd.it/cecore/resource/GCS#49b1e46252f16378e25e2407ee8ab17b

**Gene class:** Tumor Suppressor Gene

**Gene:** ANP32A

**Disease:** Malignant neoplasm of pancreas

tumor suppressor gene supporting sentences: 3

Conflicting sentences: 1

**Example of supporting sentence:**
*"pp32 (ANP32A) expression inhibits pancreatic cancer cell growth and induces gemcitabine resistance by disrupting HuR"*
**Provenance:** https://pubmed.ncbi.nlm.nih.gov/21152064 - PLoS ONE (2010), ISSN: 1932-6203

**Example of conflicting sentence:**
*"pp32 (ANP32A) expression inhibits pancreatic cancer cell growth and induces gemcitabine resistance by disrupting HuR"*
**Provenance:** https://pubmed.ncbi.nlm.nih.gov/21152064 - PLoS ONE (2010), ISSN: 1932-6203

**Figure 2** Landing page for the GCS `49b1e46252f16378e25e2407ee8ab17b` displayed by CoreKB platform. Each GCS card comprises information about the gene and cancer labels, the gene class and its distribution across the associated publications, and statistics about the number of supporting and conflicting evidence. Each GCS can be downloaded in JSON format and its representation as an extended nanopublication can be downloaded in TriG syntax (A), or visualized (B).

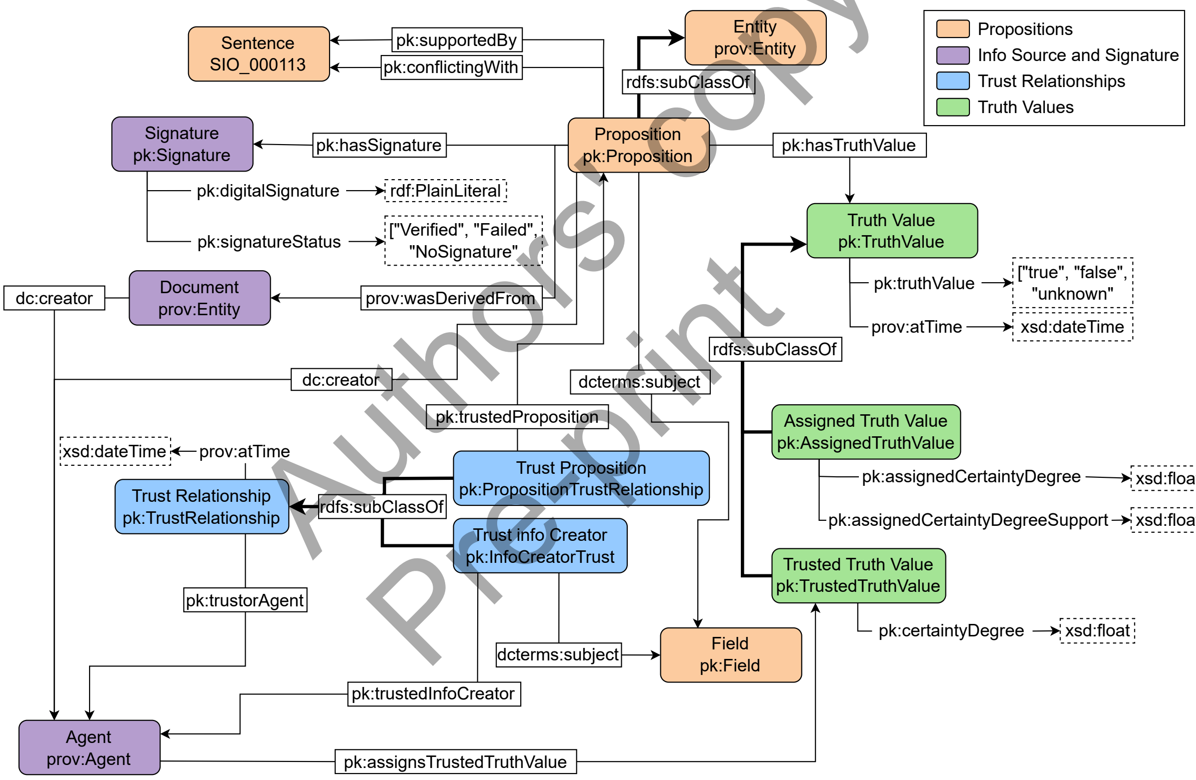


**Figure 3** The PROV-K ontology. We classify the PROV-K ontology into four main areas: Propositions (displayed in orange), Digital Signature and Information Sources (depicted in purple), Trust Relationships (pictured in blue), and Truth Value (represented in green).

exploited to build an assertion. In addition, the graph can include information about the reliability of the assertion, which the creator of the assertion or external agents can assign. Our approach does not change the other components of the original nanopublication model, but defines an optional additional component.

Like the other modules of a nanopublication, the *knowledge provenance graph* is a named graph that can be instantiated according to an ontology. To this end, we define the PROV-K ontology [14], an extension of PROV-O to represent provenance information of assertions derived from multiple sources using an aggregation algorithm.[11] The PROV-K follows the theoretical guidelines provided by the Static KP ontology [11] while integrating some elements from the Dynamic KP, such as timestamps for truth values and trust relationships [12]. We also include the concepts of *assigned certainty degree* and *certainty degree*, originally defined in the Uncertainty-oriented KP [13]. In fact, representing truth values as probability distributions instead of string values – as done in Static KP – is more practical in real-world scenarios, where most assertions are generated by automated probabilistic models. Additionally, this solution enables the classification of assertions as either reliable or unreliable, while explicitly representing the conditions that an assertion fails to satisfy. In the following, we describe the PROV-K ontology – illustrated in Figure 3 – structured around four main areas: Propositions, Digital Signature and Information Sources, Trust Relationships, and Truth Value.

### *Proposition.*

The focal point of the ontology is the "Proposition", which is defined in [11] as "*the smallest piece of information to which provenance-related attributes may be ascribed*". For instance, in the context of nanopublishing, the *assertion graph* of the nanopublication model can be viewed as a proposition. Following the Static KP ontology, we also define a taxonomy of propositions, which classifies them as independent or dependent propositions based on whether their truth value is linked to another proposition. The taxonomy is reported in Figure 4. Since the PROV-K ontology extends PROV-O, we model propositions as subclasses of `prov:Entity`. Each proposition can be supported by or conflicting with other knowledge sources. To encompass this situation, we defined two object properties called `supportedBy` and `conflictingWith`, both with range class "Sentence" from the Semanticscience Integrated Ontology (SIO).[12]

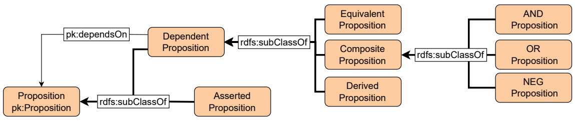


**Figure 4** The taxonomy of propositions modeled in the PROV-K ontology.

### *Information Source and Signature.*

To determine the source of information, we link each proposition to the "document" it belongs to via the object property `wasDerivedFrom` from PROV-O. The `Document` class is represented with class `Entity` from PROV-O to allow a proposition to belong to any entity, from a textual document to a dataset. One can define the creator of a proposition or a document with object property `creator` from the Dublin Core (DC) Metadata Items, with range class `Agent` from PROV-O. As for the class `Document`, we use `Agent` to allow any agent to be an information creator, from a real person to an automated model. We also represent the digital signature and signature status that can be assigned to a proposition. The digital signature is a string provided by a cryptographic mechanism that certifies the proposition and its information integrity. A proposition can have different signature status based on the presence or absence of the digital signature (Status "No Signature"). If the proposition has a digital signature, its status can be either "Verified", if the digital signature is successfully verified, or "Failed", if the verification failed. Note that the PROV-K ontology is a general resource designed to track provenance information for aggregated sources of evidence beyond the *knowledge provenance graph* of the nanopublication model. Thus, we defined the Signature class within the ontology rather than solely relying on the modeling provided by the nanopublication model. Nevertheless, when we apply such an ontology to the nanopublication model, we can represent digital signatures with the class "Nanopub Signature Element" from `nanopubx`.[13]

[11] https://www.w3.org/TR/prov-o/

[12] http://semanticscience.org/resource/SIO_000113

[13] http://purl.org/nanopub/x/NanopubSignatureElement

***Trust Relationships.***

We define two trust relationships, each corresponding to a different scenario. On the one hand, the class `InfoCreatorTrust` models a trust relationship between two agents within a specific field: one is the creator of a proposition, referred to as `trustedInfoCreator`, and the other is the trustor agent, denoted as `trustorAgent`. In the Static KP ontology, this relationship is formalized as "*the trustor agent* `a` *"trusts" information creator* `c` *in a specific knowledge field* `f`", where "trusts" means "`a` *believes any proposition created by* `c` *in field* `f` *to be true*". On the other hand, the class `PropositionTrustRelationship` represents a trust relationship between an agent (`trustorAgent`) and a specific proposition (`trustedProposition`). In this case, the relationship takes the form: "*Proposition* `x` *is trusted to be true by agent* `a`". For both trust relationships, we include a data property called `atTime` to indicate when the relationship was established. Note that the PROV-K ontology is designed to be extensible, allowing for the integration of more complex decision processes or use-case-specific trust relationships. New trust relationships can be introduced by defining them as subclasses of `TrustRelationship`. An example of the usage of trust relationships is described in Section 5.2.

***Truth Values.***

Previous work on KP distinguishes between two types of truth values, depending on who assigns them [11, 13]. The *assigned truth value* refers to the truth value assigned to a proposition by its creator, while the *trusted truth value* is determined by an external agent. Both types are modeled as subclasses of the general class `TruthValue`, which encapsulates information on when the value was assigned (or trusted) and its magnitude, in accordance with the Static KP definition. As mentioned earlier, in real-world scenarios, truth values are often best represented as probability distributions. To reflect this, we integrate concepts from the uncertainty-oriented KP and define the class `AssignedTruthValue` and the property `assignedCertaintyDegree`, indicating the probability that the proposition is true, as assessed by its creator. Similarly, for the `TrustedTruthValue` class, we define the property `certaintyDegree`, which represents the probability of the proposition being true according to an external agent.

Propositions can be classified as either reliable or unreliable based on case-specific conditions related to the assigned truth value. To capture this distinction, we introduce two subclasses of `AssignedTruthValue` representing reliable and unreliable propositions, illustrated in Figure 5. When a fact is considered unreliable, we can describe the reasons for its unreliability using the object property `unmetCondition`, which links to instances of the `ReliabilityCondition` class. Each condition is characterized by a score, a threshold, and a criterion describing how the condition is evaluated.

## 4 Extended Nanopublications in Practice: The CoreKB Use Case

This section presents a real-world use case built upon the CoreKB platform, describing how to represent CoreKB facts as extended nanopublications in order to represent all the information embedded in each GCS.

***Nanopublication Architecture.***

We apply the extended nanopublication model and the PROV-K ontology to publish all the facts in CoreKB as nanopublications enriched with the *knowledge provenance graph*. Such an effort represents a real-world use case for extended nanopublications and highlights a practical application of the PROV-K ontology. We follow the same methodology used to publish DisGeNET triples as nanopublications in [7]. Figure 6 shows the GCS in Figure 1 modeled with the extended nanopublication model and serialized in TriG format.[14] A nanopublication representing the facts generated by CORE consists of five named graphs: *head graph*, *assertion graph*, *provenance graph*, *publication information graph*, and *knowledge provenance graph*. The *head graph* defines the nanopublication and connects all the components. The *head graph*, *assertion graph*, *provenance graph*, and *publication information*

[14] The extended nanopublication can be visualized at: https://gda.dei.unipd.it/cecore/resource/nanopub/49b1e46252f16378e25e2407ee8ab17b/.

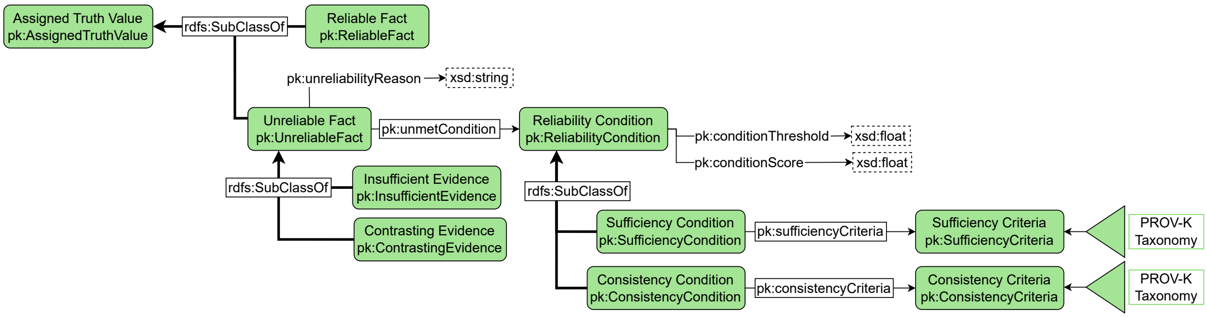
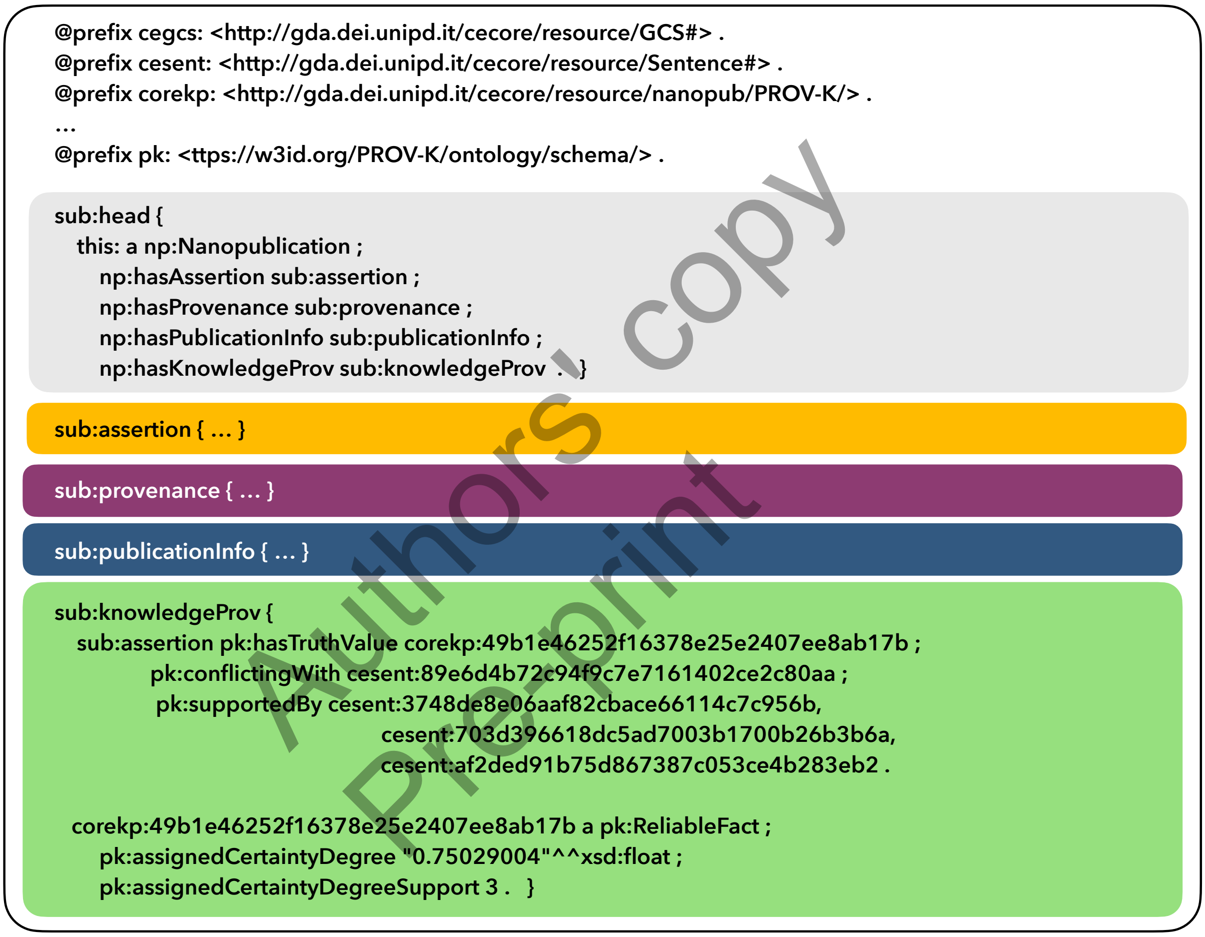


**Figure 5** Assigned truth value modeling in the PROV-K ontology.



```
@prefix cegcs: <http://gda.dei.unipd.it/cecore/resource/GCS#> .
@prefix cesent: <http://gda.dei.unipd.it/cecore/resource/Sentence#> .
@prefix corekp: <http://gda.dei.unipd.it/cecore/resource/nanopub/PROV-K/> .
…
@prefix pk: <ttps://w3id.org/PROV-K/ontology/schema/> .

sub:head {
  this: a np:Nanopublication ;
    np:hasAssertion sub:assertion ;
    np:hasProvenance sub:provenance ;
    np:hasPublicationInfo sub:publicationInfo ;
    np:hasKnowledgeProv sub:knowledgeProv  .  }

sub:assertion { … }

sub:provenance { … }

sub:publicationInfo { … }

sub:knowledgeProv {
  sub:assertion pk:hasTruthValue corekp:49b1e46252f16378e25e2407ee8ab17b ;
           pk:conflictingWith cesent:89e6d4b72c94f9c7e7161402ce2c80aa ;
            pk:supportedBy cesent:3748de8e06aaf82cbace66114c7c956b,
                                  cesent:703d396618dc5ad7003b1700b26b3b6a,
                                  cesent:af2ded91b75d867387c053ce4b283eb2 .

  corekp:49b1e46252f16378e25e2407ee8ab17b a pk:ReliableFact ;
    pk:assignedCertaintyDegree "0.75029004"^^xsd:float ;
    pk:assignedCertaintyDegreeSupport 3 .   }
```

**Figure 6** An extended nanopublication representing the fact from CoreKB shown in Figure 1. Due to space reasons, we only report the *head graph* (in grey) and the *knowledge provenance graph*, *knowledgeProv*, (in green). The other graphs are equal to those reported in Figure 1.

*graph* are detailed in Section 2.1. The *knowledge provenance graph* includes all supporting and conflicting sentences using object properties `supportedBy` and `conflictingWith` and information about the reliability of the GCS, represented by data property `assignedCertaintyDegree`. For instance, the GCS in Figure 6 is a reliable fact (instance of class `ReliableFact`). Its assigned certainty degree is 0.7, which represents the probability that the assertion is true assigned by its creator (CORE). Moreover, the GCS is supported by three sentences and conflicts with one.

### *Ontologies.*

The *assertion graph* is represented exploiting the ontology underlying KBs generated by CORE,[15] while the *provenance graph* relies on PROV-O.[16] The Provenance, Authoring, and Versioning (PAV) vocabulary [32] handles authorship and versioning, while the Provenance Vocabulary Core ontology Specification (PRV) [33] represents the description of the used datasets. The evidence annotation is described using the Weighted Evidence (WI) vocabulary,[17] which comprises the object property `evidence` to link the assertion to its evidence, and the ECO ontology.[18] The SIO ontology is exploited for the description of the topic of the nanopublications and the process used to build the assertion.[19] The *knowledge provenance graph* is serialized using the PROV-K ontology, which has been extended with the reliability conditions specific to the CORE system. A CoreKB fact may be deemed unreliable if it fails one of the two reliability criteria defined by the CORE system: one sufficiency criterion (with two conditions) and one consistency criterion. Thus, we define two classes called `kp:InsufficientEvidence` or `kp:ContrastingEvidence`, modeled as subclasses of `kp:ReliabilityCondition`, defined in the PROV-K ontology (see Section 3). Each criterion is represented as a named individual of type `skos:Concept` and classified as either `kp:SufficiencyCriteria` or `kp:ConsistencyCriteria`. We added one sufficiency criterion and one consistency criterion to represent the sufficiency and consistency conditions defined by the CORE system to classify reliable facts. A brief description of the two conditions is reported in Section 2.2; an in-depth analysis of the condition is presented in the CORE system reference paper [9].

**Table 1** Gene class distribution of the facts in CoreKB serialized as extended nanopublications.

| Class | # of Nanopublications |
|---|---|
| Biomarker | 107,830 |
| Oncogene | 35,821 |
| Tumor Suppressor Gene | 12,521 |
| Contrasting Evidence | 41,339 |

### *Implementation Details.*

To serialize extended nanopublications, we extend the Python package **`nanopub`** by adding the novel component.[20] To provide backward probability with the original nanopublication model, the provenance, publication information, and assertion graph are left untouched. Facts in CoreKB are serialized as extended nanopublications in TriG syntax through the Python package. The code can take as input two CSV files comprising the facts and the sentences supporting or conflicting with it, or one can provide a Turtle (.ttl) file comprising the CoreKB dump available in Zenodo [34]. The code for serializing the facts in CoreKB as extended nanopublications can also serve as a template for future applications on different resources.

### *Statistics and Visualization.*

CoreKB consists of $231,099$ GCSs classified as reliable facts, unreliable due to insufficient evidence, and unreliable due to low consensus (contrasting evidence). Since publishing unreliable facts due to insufficient evidence provides little to no information, we filter them out. As a result, we published $197,511$ GCSs from CoreKB as extended nanopublications, accounting for $156,172$ reliable facts and $41,339$ unreliable ones due to contrasting evidence. Table 1 shows the gene class distribution of the extended nanopublications. The extended nanopublications are also available in Zenodo [15]. We integrate the extended nanopublications into the CoreKB platforms to ease facts visualization. For each GCS, one can explore the serialized nanopublication by clicking on the eye icon placed in the drop-down list on the right side of the claim (see point B in Figure 2). The visualization depicts each component with a different color and displays

[15] http://gda.dei.unipd.it/cecore/ontology/
[16] http://www.w3.org/TR/prov-o/
[17] http://www.evidenceontology.org/
[18] https://ontobee.org/ontology/ECO
[19] http://sio.semanticscience.org/
[20] https://github.com/fair-workflows/nanopub

URIs redirected to a functioning website containing the description of the considered element. One can also download the extended nanopublication representing a specific GCS thanks to the download button (see point A in Figure 2).

# 5 Trust Relationships in Practice: The CoreKB Trust Network Use Case

Trust relationships build a network of trust between agents and assertions, which can be especially useful in social processes, for instance, to track how truth propagates through social networks. This network enables analysis of information flows, perceived reliability, and the spread of influence among agents. Such insights are valuable for detecting misinformation, assessing credibility, and modeling collective decision-making. In this section, we show how to build such a trust network considering CoreKB facts as a practical use case.

## 5.1 Exploiting Large Language Models

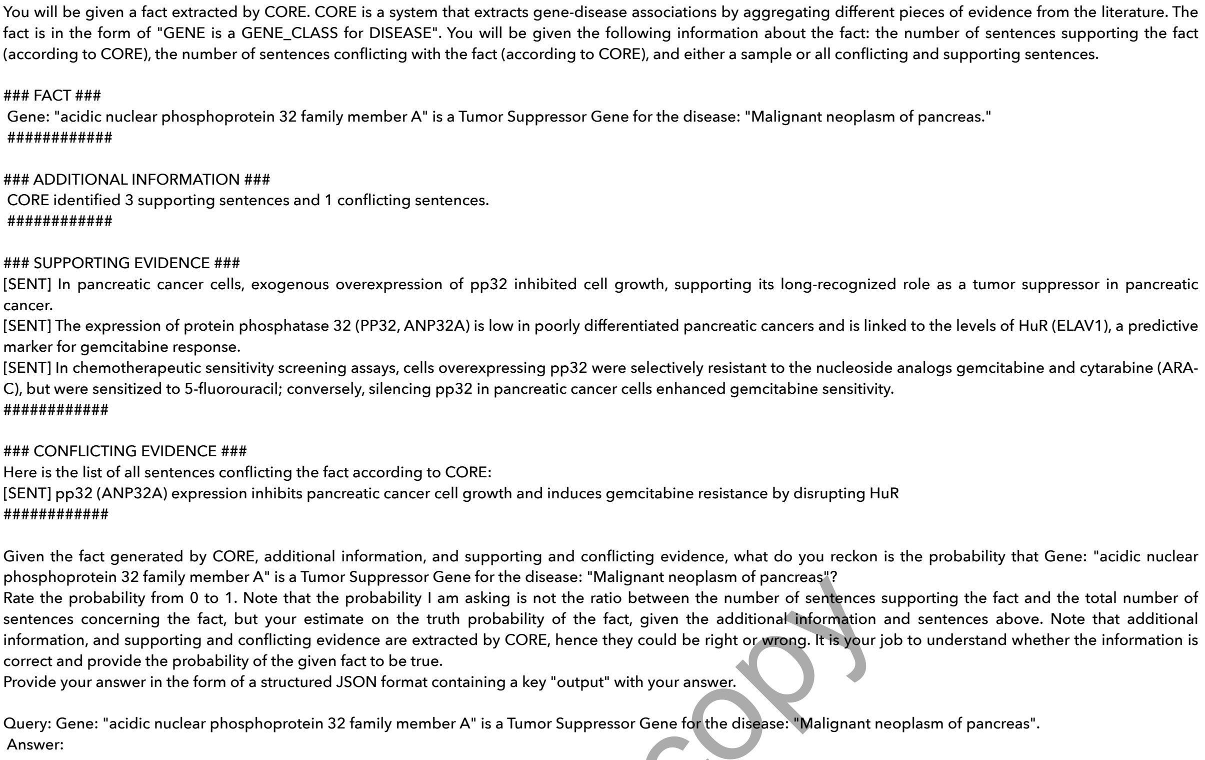
Different LLMs are employed as external agents to assign trusted truth values to each GCS, where the value represents the probability that a given fact is considered true by the respective LLM. In the following, we first describe the prompt-based approach used to obtain the scores. Then, we show with Figure 8 that LLMs can exhibit polarized or divergent assessments of the same facts (GCSs). To better assess their behavior in this setting, we conduct a rigorous evaluation of their quality, estimating accuracy scores with minimal human annotations while providing strong statistical guarantees through the use of Credible Intervals (CrIs). Finally, the trusted truth values from all LLMs are aggregated based on each LLM accuracy to simulate the behavior of a *crowd agent*, analogous to an agent relying on crowd-sourced annotations.

***Obtaining the scores.***

To assess the trusted truth value of each GCS, we leverage three LLMs: DeepSeek V3 [35], Meta Llama 3.1 405B-Instruct [36], and GPT-4o mini [37], a smaller variant of OpenAI's GPT-4o model [38]. We selected these three models primarily due to their availability and compatibility with our computational resources, which facilitated the integration into our experimental setup and enabled an efficient and reliable testing process. To run GPT-4o mini, we used the Azure OpenAI Batch API in the Azure AI Foundry Portal and set the LLM temperature to 0.75. To run DeepSeek V3 and Meta Llama 3.1 405B-Instruct, we developed two Python scripts that exploit the Azure AI Inference library. For both LLMs, we set `max_token` to $2,048$, `temperature` to 0.8, `top_p` to 0.1, and `presence_penalty` and `frequency_penalty` both to zero. All parameters were set empirically.

We exploit a zero-shot prompt-based approach that uses the same prompt for all LLMs to obtain the values. For each GCS, we provide the LLMs with the fact in textual form, the number of supporting and conflicting sentences identified by CORE, and the actual sentences divided between supporting and conflicting. Although the average number of supporting sentences per GCS is 2.3, some GCSs have as many as $2,588$ supporting sentences. To limit the length of the prompt, we insert all supporting sentences for a GCS if they are less than fifteen; otherwise, we only report the first fifteen. The same applies to conflicting sentences. A prompt template is used to incorporate GCS-specific information accordingly. To illustrate the procedure, we report the prompt for our recurring example GCS in Figure 7.

At this stage, only reliable GCSs are considered, as obtaining trusted truth values for facts already classified as unreliable by the CORE system would not provide additional insights. Due to budget constraints, the trusted truth values are obtained from the three LLMs for a subset of $45,649$ GCSs. Figure 8 reports the distribution of the trusted truth values for each LLM. Each LLM exhibits a different distribution of trusted truth values. Meta Llama 3.1 produces a widespread distribution across the probability range, whereas GPT-4o mini and DeepSeek V3 display more polarized distributions concentrated around specific (range of) values. In particular, DeepSeek V3 frequently assigns trusted truth value between 0.6 and 0.7, while GPT-4o mini distribution is sharply centered around 0.85. Notably, GPT-4o mini assigns a trusted truth value of 0.85 to $23,196$ GCSs, accounting for 51% of the entire dataset.



```
You will be given a fact extracted by CORE. CORE is a system that extracts gene-disease associations by aggregating different pieces of evidence from the literature. The fact is in the form of "GENE is a GENE_CLASS for DISEASE". You will be given the following information about the fact: the number of sentences supporting the fact (according to CORE), the number of sentences conflicting with the fact (according to CORE), and either a sample or all conflicting and supporting sentences.

### FACT ###
 Gene: "acidic nuclear phosphoprotein 32 family member A" is a Tumor Suppressor Gene for the disease: "Malignant neoplasm of pancreas."
 ############

### ADDITIONAL INFORMATION ###
 CORE identified 3 supporting sentences and 1 conflicting sentences.
 ############

### SUPPORTING EVIDENCE ###
[SENT] In pancreatic cancer cells, exogenous overexpression of pp32 inhibited cell growth, supporting its long-recognized role as a tumor suppressor in pancreatic cancer.
[SENT] The expression of protein phosphatase 32 (PP32, ANP32A) is low in poorly differentiated pancreatic cancers and is linked to the levels of HuR (ELAV1), a predictive marker for gemcitabine response.
[SENT] In chemotherapeutic sensitivity screening assays, cells overexpressing pp32 were selectively resistant to the nucleoside analogs gemcitabine and cytarabine (ARA-C), but were sensitized to 5-fluorouracil; conversely, silencing pp32 in pancreatic cancer cells enhanced gemcitabine sensitivity.
############

### CONFLICTING EVIDENCE ###
Here is the list of all sentences conflicting the fact according to CORE:
[SENT] pp32 (ANP32A) expression inhibits pancreatic cancer cell growth and induces gemcitabine resistance by disrupting HuR
############

Given the fact generated by CORE, additional information, and supporting and conflicting evidence, what do you reckon is the probability that Gene: "acidic nuclear phosphoprotein 32 family member A" is a Tumor Suppressor Gene for the disease: "Malignant neoplasm of pancreas"?
Rate the probability from 0 to 1. Note that the probability I am asking is not the ratio between the number of sentences supporting the fact and the total number of sentences concerning the fact, but your estimate on the truth probability of the fact, given the additional information and sentences above. Note that additional information, and supporting and conflicting evidence are extracted by CORE, hence they could be right or wrong. It is your job to understand whether the information is correct and provide the probability of the given fact to be true.
Provide your answer in the form of a structured JSON format containing a key "output" with your answer.

Query: Gene: "acidic nuclear phosphoprotein 32 family member A" is a Tumor Suppressor Gene for the disease: "Malignant neoplasm of pancreas".
 Answer:
```

**Figure 7** Prompt used to obtain the trusted truth value for GCS `49b1e46252f16378e25e2407ee8ab17b`. The same prompt is used for all LLMs.

This centralization observed in the distribution of DeepSeek V3 and GPT-4o mini suggests two possible explanations: either the GCSs have high reliability, or the LLMs – especially GPT-4o mini – tend to assign the same value systematically. To investigate this further, we conduct a manual evaluation to assess the accuracy of the LLMs in this setting.

***Evaluation.***

Auditing the accuracy of LLM-assigned truth values involves annotating *(fact, truth value)* pairs with correctness labels. This task closely aligns with the broader challenge of evaluating KB accuracy [39, 40]. However, two major challenges arise when working with large-scale KBs such as CoreKB. First, obtaining high-quality correctness labels is costly [39, 40]. Second, annotating every *(fact, truth value)* pair in a large-scale KB is infeasible [41], as real-world KBs often contain hundreds of thousands or even millions of facts. As such, cost-effective and scalable evaluation methods are essential.

Recent research has proposed efficient approaches for KB accuracy estimation [42], leveraging sampling strategies for efficient data collection, point estimators for accuracy assessment, and confidence or credible intervals to quantify the uncertainties inherent in the sampling procedure [40, 41, 43, 44]. Inspired by these advances, we adopt a similar methodology to evaluate the accuracy of LLMs in assigning truth values.

Specifically, we adopt a sampling strategy that yields a representative subset of the population – here, the set of $45,649$ GCSs. As sampling method, we resort to Simple Random Sampling (SRS), which selects $n$ GCSs from the population without replacement. The accuracy of an LLM is then estimated using the sample proportion $\hat{\mu} = \frac{1}{n}\sum_{j=1}^{n} \mathbb{1}(f_j, p_j)$, where $\mathbb{1}(f_j, p_j)$ is an indicator function denoting whether the truth value predicted by the LLM for the $j$-th sampled GCS is correct. When used with SRS, the sample proportion represents an unbiased estimator [45] – i.e., $E[\hat{\mu}] = \mu$. In other words, the sample represents a

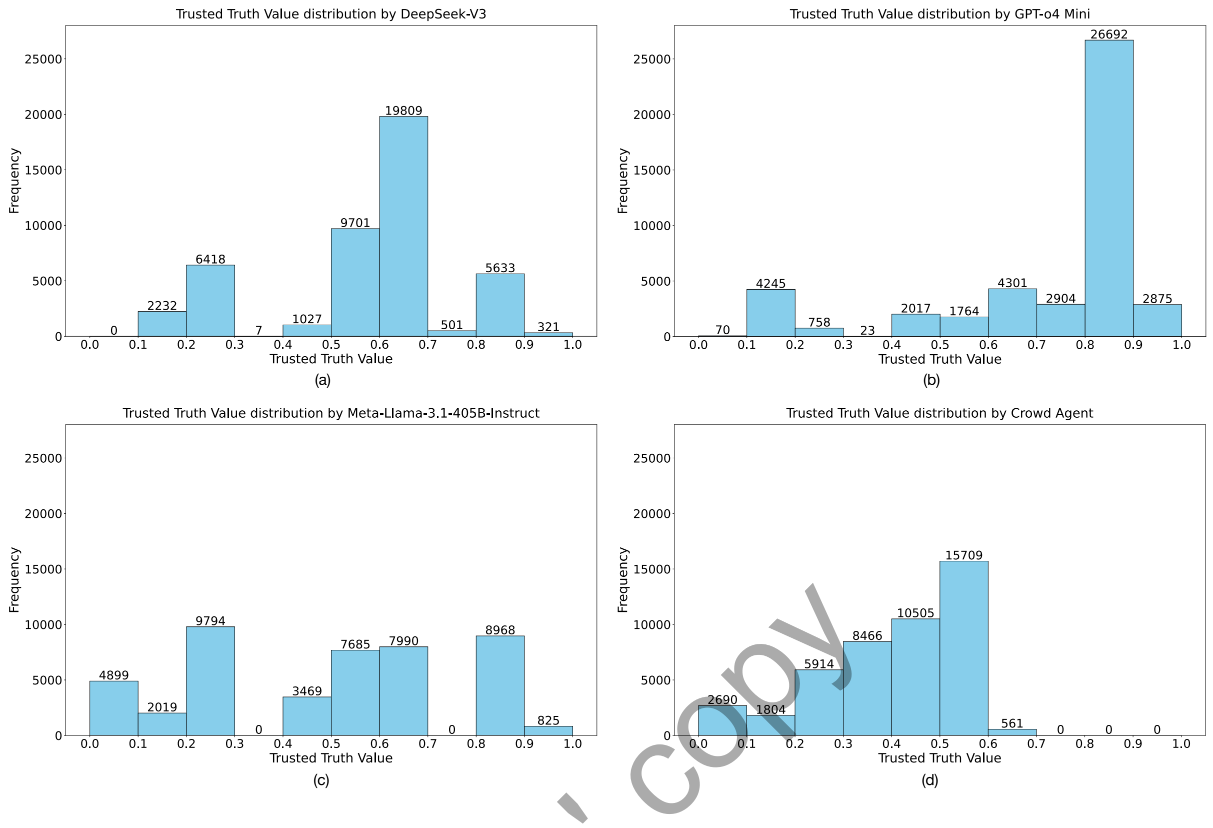


**Figure 8** Distribution of trusted truth values assigned by each individual LLM and by the aggregated crowd agent.

good approximation of the population, and the estimated accuracy reflects what would be obtained by evaluating the full dataset. Finally, to quantify the uncertainties associated with the estimate, we employ the *adaptive* Highest Posterior Density ($a$HPD) algorithm [44], which concurrently runs multiple $1-\alpha$ High Posterior Density (HPD) CrIs – based on different objective priors – competing to achieve the highest level of precision. Rooted in Bayesian statistics, HPD intervals define the most probable range of values for the parameter of interest – in this case, the accuracy – with a (posterior) probability of $1-\alpha$ [46].[21] In this way, the $a$HPD algorithm ensures reliability by dynamically selecting the most precise interval among the candidates based on the annotated sample.

It is worth noting that both the sample-based estimator and the $a$HPD algorithm operate independently of the population size. Thus, the evaluation procedure remains valid and scalable even for large KBs, making the approach particularly efficient at scale.

Following the above evaluation procedure, we draw a sample of 200 GCSs using SRS and manually annotate the correctness of the LLM-generated truth values. Each GCS receives a score of 1 if the assigned probability is deemed accurate, and 0 otherwise. For instance, in the example shown in Figure 1 representing the fact "*ANP32A is a tumor suppressor gene for malignant neoplasm of pancreas*", DeepSeek V3 and GPT-4o mini both assign a probability of 0.1, while Meta Llama 3.1 405B-Instruct assigns 0.8. Manual evaluation identifies the GCS as highly reliable, so only the Llama output is marked as correct. Finally, we compute accuracy estimates for each LLM and use the $a$HPD algorithm with $\alpha = 0.05$ to generate CrIs. Table 2 reports the resulting accuracy estimates along with their CrIs.

[21] These intervals operate directly in the probabilistic space, thereby avoiding the common interpretational issues associated with confidence intervals [47], emerging as well suited solutions for *one-shot* settings [44].

| LLM | Accuracy | CrI | CrI Width |
|---|---|---|---|
| DeepSeek V3 | 0.75 | $[0.68, 0.80]$ | 0.12 |
| GPT-4o mini | 0.58 | $[0.51, 0.65]$ | 0.14 |
| Meta Llama 3.1 405B | 0.76 | $[0.69, 0.81]$ | 0.12 |

**Table 2** LLMs accuracy in assigning trusted truth values. A sample of 200 GCSs is manually annotated to evaluate each LLM. The "Accuracy" column reports the estimated accuracy based on the sample. The "CrI" column presents the 95% CrI, derived using the $a$HPD algorithm, while "CrI Width" reports its width.

The results indicate that DeepSeek V3 and Meta Llama 3.1 405B are more reliable, with accuracy estimates around 0.75, compared to GPT-4o mini, which yields a lower estimate of 0.58. The reported intervals represent the 95% CrIs, reflecting the range within which the true accuracy is likely to fall. Notably, DeepSeek and Llama have narrower intervals (12% width) than GPT-4o mini (14%), indicating greater precision in their estimates. This improved precision arises from the higher accuracy scores, which reduce estimation uncertainty at the given sample size compared to GPT-4o mini. In contrast, GPT-4o mini requires a larger number of annotations to reach comparable precision, due to greater variability in its manual annotations – i.e., higher variance in the binary correctness labels.

LLMs tend to be polarized by overestimating or underestimating the correctness of factual statements and may exhibit performance levels comparable to those of low-quality human annotators [48]. The presented experiments align with prior research, providing additional evidence that LLMs overestimate factual correctness and are not yet suitable substitutes for human annotators. Indeed, several limitations hinder their effectiveness in verifying factual knowledge. LLMs struggle to represent accurately and reason over isolated and single statements [49]. Furthermore, their performance degrades when dealing with less popular entities, commonly referred to as torso and tail entities [50].

In addition, employing LLMs can introduce various forms of bias into their outputs. These biases may stem from multiple sources, including the phrasing of the textual instructions or the input provided [51, 52]. Notably, also the ordering of candidate responses has been shown to influence the model's predictions [53]. Such factors can lead to systematic preference patterns, undermining the objectivity and consistency expected in evaluation or fact-checking tasks.

### *The Crowd Agent.*

Building a trust network between agents and assertions is particularly valuable in Web scenarios, where content can be published freely and without verification. In this context, assertions can be verified through crowd-sourcing platforms, and agents can embody trusted truth values and trust relationships informed by the collective input of the crowd. To mimic this mechanism, we introduce a crowd agent by aggregating the trusted truth values from all LLMs, thereby creating a fourth agent. Moreover, given the varying quality of the considered LLMs, aggregating their outputs represents a potential solution for more robust results [48]. The aggregation is weighted by the estimated accuracy of each LLM, ensuring that models with higher reliability exert greater influence on the final score. Formally, the trusted truth value of the crowd agent is computed as $\frac{1}{n}\sum_{i=1}^{n} \mathrm{ttv}(\mathrm{LLM}_i) \cdot \mathrm{acc}(\mathrm{LLM}_i)$, where $n$ denotes the number of LLMs (three in our case), $\mathrm{ttv}(\mathrm{LLM}_i)$ is the trusted truth value assigned by the $i$-th LLM, and $\mathrm{acc}(\mathrm{LLM}_i)$ is its estimated accuracy. The formula $\mathrm{ttv}(\mathrm{LLM}_i) \cdot \mathrm{acc}(\mathrm{LLM}_i)$ provides an entropy-inspired estimate of the reliability of the given LLM. A related work in fact verification has similarly employed multiple LLMs to evaluate the same factual statement, leveraging entropy-based measures as indicators of cross-model agreement and overall factual consistency [54].

The distribution of trusted truth values generated by the crowd agent is shown in Figure 8(d). The distribution is skewed toward lower values compared to those of the LLMs. This shift arises from the definition of the aggregation formula used in the crowd model: since it involves multiplying probabilities bounded between 0 and 1, the scores tend to be small. Consequently, even moderately low trusted truth values can significantly lower the aggregated outcome. In this scenario, the maximum trusted truth value for all LLMs is 1.0. Applying the aggregation formula with $\mathrm{ttv}(\mathrm{LLM}_i) = 1$ for each model ($n = 3$), and substituting $\mathrm{acc}(\mathrm{LLM}_i)$ with the models' accuracies in

Table 2, the maximum trusted truth value achievable by the crowd agent is: $\frac{1}{3}(0.75 \cdot 1 + 0.58 \cdot 1 + 0.76 \cdot 1) = 0.7$. As a result, no GCS can attain a trusted truth value exceeding this threshold.

## 5.2 Building the Trust Network

Once we obtain the trusted truth values, we build a trust network using the PROV-K ontology. This section describes the process of serializing the trust network and inferring trust relationships between agents and each GCS. Following serialization, we show the utility of the resulting network by running a set of representative SPARQL queries over it.

***Trust Network Serialization.***

We exploit the PROV-K ontology to serialize the trust network. The PROV-K ontology represents provenance information of assertions derived from multiple sources and trust relationships. In the use case presented in Section 4, we used the PROV-K ontology to serialize a named graph as part of the extended nanopublication model. However, ontologies can be applied in a wide range of use cases. Here, we apply the PROV-K ontology to build an RDF graph representing each GCS, the trusted truth values assigned by each agent, and the trust relationships between assertions and agents.

Trust relationships between an agent and a proposition indicate that the agent trusts the proposition to be true. To ensure that only high-probability trust relationships are considered, we define them as follows:

> *An agent $A$ is said to trust a proposition $P$ if the trusted truth value assigned by $A$ to $P$ lies in the third quartile of $A$'s distribution of trusted truth values.*

We use the third quartile as a threshold to distinguish between trusted and untrusted propositions as it allows us to focus on the top 25% of trust values, i.e., those significantly higher than the majority, ensuring a conservative and reliable decision boundary. In practice, the resulting thresholds for establishing trust relationships are: 0.52 for the crowd agent, 0.7 for DeepSeek V3, 0.85 for GPT-4o mini, and 0.7 for Meta Llama 3.1 405-Instruct.

Figure 9 shows the serialization of the trust network for the GCS we used as a recurring example (i.e., `49b1e46252f16378e25e2407ee8ab17b`).

The trusted truth value obtained by each LLM is instantiated as a named individual belonging to the class `TrustedTruthValue`, which is linked to the GCS with object property `hasTruthValue` and to the agent with property `assignsTrustedTruthValue`. The data property `certaintyDegree` reports the trusted truth value. In this example, only Meta Llama 3.1 405-Instruct trusts the GCS. The trust relationship is instantiated as a named individual of type `PropositionTrustRelationship`, which is linked to the agent with object property `trustorAgent` and to the GCS with object property `trustedPropsition`.

***Trust Network Exploration.***

As a result, we serialize an RDF graph representing the trust network of the $45,649$ reliable GCSs assessed by LLMs and publish it in Zenodo [16]. This graph can be explored through SPARQL queries to provide useful network statistics. We show three example queries that can be run on the trust network.

Query 1 returns the trusted truth value assigned by the four agents for each GCS. This type of query highlights the utility of the trust network, which serves as a powerful tool for assessing the reliability of facts. By leveraging the structure of the trust network, one can identify facts that are widely accepted –indicating a high degree of trust for all agents – as well as contentious facts, where there is significant disagreement among agents regarding their truth value. This capability is essential for distinguishing between consensus-driven knowledge and areas of uncertainty or dispute. Table 3 reports a subset of the query result set. The first two GCSs have a high trusted truth value assigned by all agents, hence one can identify them as reliable. The third and fourth GCSs show disagreement between the models. In particular, GCS ...`846a66` has a high trusted truth value assigned by GPT-4o mini, DeepSeek V3 assigns 0.5, which can be interpreted as borderline, while Llama 3.1 405B-Instruct gives a low score. On the other hand, both GPT-4o mini and DeepSeek V3 assigns a very low reliability score to GCS ...`0ef38b` while Llama 3.1 405B-Instruct is almost certain the fact is true and assigns a score of 0.9. For the last two GCSs, all LLMs agree to a low truth score.

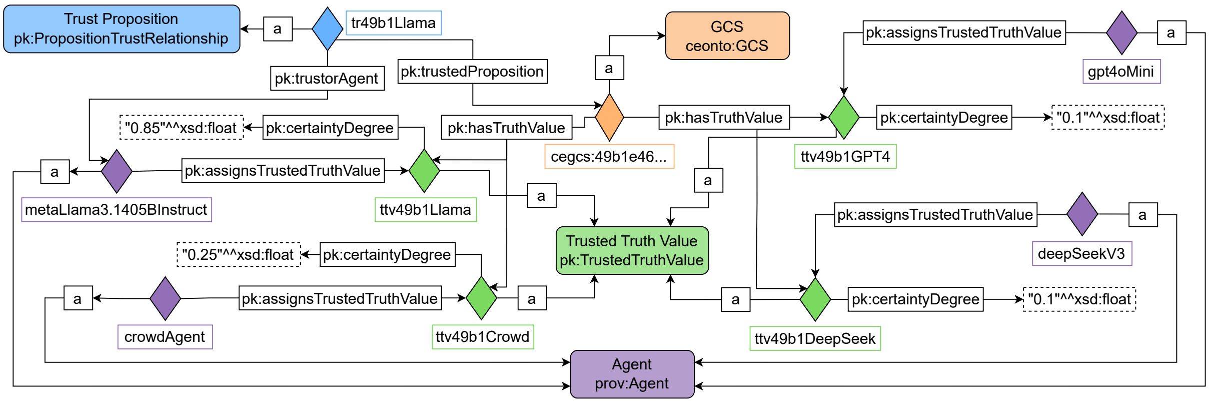


**Figure 9** Trust network for GCS `49b1e46252f16378e25e2407ee8ab17b`. Rhombuses represent named individuals, while rectangles are classes. Components in orange are related to propositions, those in purple are agents, green represents trusted truth values, and blue identifies trust relationships.

```
PREFIX ceonto: <http://gda.dei.unipd.it/cecore/
     ontology/>
PREFIX corekp: <http://gda.dei.unipd.it/cecore/
     resource/nanopub/PROV-K/>
PREFIX pk: <https://w3id.org/PROV-K/ontology/
     schema/>

SELECT ?GCS ?dseek ?gpt ?llama ?crowd
WHERE
{
    ?GCS a ceonto:GCS;
         pk:hasTruthValue ?ttv_ds;
         pk:hasTruthValue ?ttv_g;
         pk:hasTruthValue ?ttv_l;
         pk:hasTruthValue ?ttv_ca.
    ?ttv_ds a pk:TrustedTruthValue;
            pk:certaintyDegree ?dseek.
    corekp:deepSeekV3 pk:
         assignsTrustedTruthValue ?ttv_ds.
    ?ttv_g a pk:TrustedTruthValue;
           pk:certaintyDegree ?gpt.
    corekp:gpt4oMini pk:assignsTrustedTruthValue
          ?ttv_g.
    ?ttv_l a pk:TrustedTruthValue;
           pk:certaintyDegree ?llama.
    corekp:metaLlama3.1405BInstruct pk:
         assignsTrustedTruthValue ?ttv_l.
    ?ttv_ca a pk:TrustedTruthValue;
            pk:certaintyDegree ?crowd.
    corekp:crowdAgent pk:
         assignsTrustedTruthValue ?ttv_ca.
}
```

**Query 1** For each GCS, return its trusted truth value assed by the Crowd Agent.

| GCS | dseek | gpt | llama | crowd |
|---|---|---|---|---|
| cegcs:. . .545502 | 0.95 | 0.95 | 0.95 | 0.66 |
| cegcs:. . .49516a | 0.70 | 0.85 | 0.70 | 0.52 |
| ... | ... | ... | ... | ... |
| cegcs:. . .846a66 | 0.50 | 0.70 | 0.20 | 0.31 |
| cegcs:. . .0ef38b | 0.10 | 0.05 | 0.90 | 0.26 |
| ... | ... | ... | ... | ... |
| cegcs:. . .e89d41 | 0.10 | 0.10 | 0.20 | 0.10 |
| cegcs:. . .8a50c0 | 0.10 | 0.05 | 0.00 | 0.03 |

**Table 3** Result sample for Query 1: "For each GCS, return its trusted truth value assed by the Crowd Agent.". Columns are named as the selected variables in the query (the LLM label corresponds to its trusted truth value). To ease the visualization of the result set, we report a subset of GCSs and the last six digits ot the GCS URI.

Query 2 counts how many GCSs are trusted by each LLM. Table 4 reports the query result. Meta Llama 3.1 405B-Instruct is the LLM trusting the highest number of GCSs, i.e., 9, 793. The Crowd agent follows with 9, 124 trusted GCSs, DeepSeek V3 trusts 6, 455 GCSs, and GPT-4o mini only trusts 2, 875 GCSs. The lower number of trusted GCSs for GPT-4o mini can be attributed to the LLM assigning a truth score of 0.85 to 51% of the dataset, thereby compressing the score distribution and reducing the number of items in the third quartile.

```
PREFIX pk: <https://w3id.org/PROV-K/ontology/
     schema/>

SELECT ?agent (COUNT(?trustRel) AS ?trustedGCSs)
WHERE
{
    ?trustRel a pk:PropositionTrustRelationship;
              pk:trustorAgent ?agent.
}
GROUP BY ?agent
```

**Query 2** Count how many GCSs are trusted by each LLM.

**Table 4** Result sample for Query 2: "Count how many GCSs are trusted by each LLM". Columns are named as the selected variables in the query.

| Agent | Trusted GCSs |
|---|---|
| Crowd Agent | 9,124 |
| DeepSeek V3 | 6,455 |
| GPT-4o mini | 2,875 |
| Llama 3.1 405B | 9,793 |

**Table 5** Result sample for Query 3: "Trusted relationships frequency distribution. Count the number of GCSs associated with each distinct number of trust relationships". The first column is variable `trustRels`, while the second column represents `numGCS`.

| Trust Relationships | Number of GCS |
|---|---|
| 4 | 1,333 |
| 3 | 4,473 |
| 2 | 3,409 |
| 1 | 2,678 |

Query 3 computes the frequency of trusted relationships. For each possible number of trust relationships, we count how many GCSs are trusted by exactly that number of agents. Table 5 reports the query result. As we can see $1,333$ GCSs are trusted by all LLMs, $4,473$ GCSs are trusted by three LLMs, $3,409$ GCSs are trusted by two LLMs, while $2,678$ GCSs are trusted by one LLM. The remaining GCSs – i.e., $33,756$ – are not trusted by any LLM. Under the quartile threshold, we expected that the majority of the GCSs were not trusted by any LLM. Since the overlap across all four LLMs is naturally limited, and these universally trusted GCSs likely represent clear, unambiguous content.

```
PREFIX pk: <https://w3id.org/PROV-K/ontology/
    schema/>

SELECT ?trustRels (COUNT(?GCS) AS ?numGCS)
WHERE
{{
    SELECT DISTINCT ?GCS (COUNT(?tr) AS ?
        trustRels)
    WHERE
    {
        ?tr a pk:PropositionTrustRelationship;
            pk:trustedProposition ?GCS.
    }
    GROUP BY ?GCS
}}
GROUP BY ?trustRels
ORDER BY DESC(?trustRels)
```

**Query 3** Trusted relationships frequency. Count the number of GCS associated with each distinct number of trust relationships.

## 6 Conclusions

This work highlights the limitations of the standard nanopublication model when used to represent multi-source assertions supported by multiple, and possibly conflicting, pieces of evidence. To address this challenge, we proposed a novel component for the nanopublication model: the *knowledge provenance graph*. We defined the *knowledge provenance graph* as a named graph that tracks the provenance of each piece of information contributing to the support or refutation of an assertion. To capture the semantics of the *knowledge provenance graph*, we introduced the PROV-K ontology, an extension of the PROV-O ontology designed to represent provenance information for assertions resulting from multi-source aggregation. While the primary use of PROV-K is to support the serialization of the *knowledge provenance graph* within nanopublications, the ontology has been designed as a more general resource capable of modeling provenance for aggregated sources of evidence beyond the nanopublication context.

We applied the proposed model to $197,511$ facts in CoreKB, serializing them as extended nanopublications. The CoreKB facts show a critical limitation of the standard nanopublication model. Specifically, this model does not offer sufficient mechanisms to represent detailed provenance information when assertions are generated by aggregating multiple sources, as is the case in the CORE system. These nanopublications can be easily browsed and downloaded through the CoreKB platform. The serialization of CoreKB facts can also serve as a practical handbook for applying the extended nanopublication model to different resources.

Finally, we presented an additional use case for the PROV-K ontology by showcasing the role of trust relationships. Starting from CoreKB facts, we leveraged external agents – namely, multiple LLMs – to assess the trusted truth value of each GCS. Based on these values, we defined trust relationships between the agents and the facts, resulting in a trust network, i.e., an RDF graph, comprising over $45,000$ facts and four agents. The network can be queried to derive meaningful statistics, and the trust structure can be further analyzed to uncover additional insights. By examining the underlying trust structure, one can identify assertions that are broadly accepted as true or offer a deeper understanding of how trust propagates through indirect relationships. In addition, the trust network can serve as a valuable tool for detecting regions of uncertainty, i.e., areas within the graph where no clear consensus exists regarding the truth value of certain facts. Pinpointing these uncertain regions can enhance knowledge graph refinement by directing attention to areas that require further verification, thereby reducing the need for exhaustive manual annotation across the entire graph. The trust network can represent entropy-based fact assessments that evolve over time, which is crucial for capturing how trust in facts fluctuates with new evidence, shifting consensus, or changes in source credibility.

# Declarations

## Funding

This project has received funding from the HEREDITARY Project as part of the European Union's Horizon Europe research and innovation programme under grant agreement No GA 101137074. Views and opinions expressed are, however, those of the authors only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible.

## Competing Interests

The authors declare no competing interests.

## Ethics approval and consent to participate

Not applicable.

## Data Availability

The PROV-K ontology is published in Zenodo [14] and the ontology documentation is available at: https://prov-k.dei.unipd.it/ontology/.
All extended nanopublications derived from the facts in CoreKB are available in Zenodo [15].
The trust network consisting of trust relationships between external agents and CoreKB facts enriched with trusted truth values is published in Zenodo [16].

## Materials Availability

Not applicable.

## Code Availability

The source code to build nanopublications following the extended model and to represent facts in CoreKB as nanopublications is published in the GitHub repository at: https://github.com/mntlra/knowledgeProvenance.

## Author Contribution

L.M. designed and developed the ontology encoding, formalized the knowledge provenance in the context of nanopublications, serialized CoreKB facts as extended nanopublications, serialized the trust network, developed the prompt templates, and prepared the manuscript. S.M. designed and developed the CORE system, designed and conducted the LLM evaluation, and contributed to CoreKB, the formalization of the knowledge provenance, and the ontology development. F.G. designed and developed CoreKB, and wrote the code to visualize and download nanopublications from the platform. G.S. ideated the concept of knowledge provenance, coordinated the teamwork and contributed to all the different aspects of the work. All the authors contributed to the revision of the manuscript.

# References

[1] Weikum, G., Dong, X.L., Razniewski, S., Suchanek, F.M.: Machine Knowledge: Creation and Curation of Comprehensive Knowledge Bases. Found. Trends Databases **10**(2-4), 108–490 (2021) https://doi.org/10.1561/1900000064

[2] Dong, X.L.: Generations of Knowledge Graphs: The Crazy Ideas and the Business Impact. Proc. VLDB Endow. **16**(12),

4130–4137 (2023) https://doi.org/10.14778/3611540.3611636

[3] Groth, P., Gibson, A., Velterop, J.: The anatomy of a nanopublication. Inf. Serv. Use **30**(1-2), 51–56 (2010) https://doi.org/10.3233/ISU-2010-0613

[4] Fabris, E., Kuhn, T., Silvello, G.: A Framework for Citing Nanopublications. In: Proc. of the Digital Libraries for Open Knowledge - 23rd International Conference on Theory and Practice of Digital Libraries, TPDL 2019, Oslo, Norway, September 9-12, 2019. Lecture Notes in Computer Science, vol. 11799, pp. 70–83. Springer, Heidelberg, Germany (2019). https://doi.org/10.1007/978-3-030-30760-8_6

[5] Chichester, C., Gaudet, P., Karch, O., Groth, P., Lane, L., Bairoch, A., Mons, B., Loizou, A.: Querying neXtProt nanopublications and their value for insights on sequence variants and tissue expression. J. Web Semant. **29**, 3–11 (2014) https://doi.org/10.1016/j.websem.2014.05.001

[6] Waagmeester, A., Kutmon, M., Riutta, A., Miller, R.A., Willighagen, E.L., Evelo, C.T.A., Pico, A.R.: Using the Semantic Web for Rapid Integration of WikiPathways with Other Biological Online Data Resources. PLoS Comput. Biol. **12**(6) (2016) https://doi.org/10.1371/journal.pcbi.1004989

[7] Queralt-Rosinach, N., Kuhn, T., Chichester, C., Dumontier, M., Sanz, F., Furlong, L.I.: Publishing disgenet as nanopublications. Semantic Web **7**(5), 519–528 (2016) https://doi.org/10.3233/SW-150189

[8] Giachelle, F., Marchesin, S., Silvello, G., Alonso, O.: Searching for reliable facts over a medical knowledge base. In: Proc. of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2023), pp. 3205–3209. ACM, New York, NY, USA (2023). https://doi.org/10.1145/3539618.3591822

[9] Marchesin, S., Menotti, L., Giachelle, F., Silvello, G., Alonso, O.: Building a large gene expression-cancer knowledge base with limited human annotations. Database J. Biol. Databases Curation **2023** (2023) https://doi.org/10.1093/database/baad061

[10] Giachelle, F., Marchesin, S., Menotti, L., Silvello, G.: Extending Nanopublications with Knowledge Provenance for Multi-Source Scientific Assertions. In: Proc. of the 21st Conference on Information and Research Science Connecting to Digital and Library Science (IRCDL 2025). CEUR-WS Proceedings, vol. 3937. CEUR-WS.org, Aachen, Germany (2025). https://ceur-ws.org/Vol-3937/paper10.pdf

[11] Fox, M.S., Huang, J.: An Ontology for Static Knowledge Provenance. In: Proc. of the Knowledge Sharing in the Integrated Enterprise - Interoperability Strategies for the Enterprise Architect, 2004 International Conference on Enterprise Integration and Modelling Technology (ICEIMT 2004), The 7th International Conference on Design of Information Infrastructure Systems for Manufacturing, (DIISM 2004). IFIP, vol. 183, pp. 203–213. Springer, Heidelberg, Germany (2004). https://doi.org/10.1007/0-387-29766-9_17

[12] Huang, J., Fox, M.S.: Dynamic Knowledge Provenance. In: Proc. of the Business Agents and Semantic Web Workshop, pp. 372–387. National Research Council of Canada, Canada (2004). http://www.eil.utoronto.ca/wp-content/uploads/km/papers/huang-nrc04.pdf

[13] Huang, J., Fox, M.S.: Uncertainty in Knowledge Provenance. In: Proc. of The Semantic Web: Research and Applications, First European Semantic Web Symposium (ESWS 2004). Lecture Notes in Computer Science, vol. 3053, pp. 372–387. Springer, Heidelberg, Germany (2004). https://doi.org/10.1007/978-3-540-25956-5_26

[14] Menotti, L., Marchesin, S., Silvello, G.: The PROV-K Ontology for tracking provenance of multi-sourced assertions. Zenodo (2025). https://doi.org/10.5281/zenodo.15187371

[15] Giachelle, F., Marchesin, S., Menotti, L.,

Silvello, G.: CORE Extended Nanopublications. Zenodo (2023). https://doi.org/10.5281/zenodo.10277210

[16] Menotti, L., Marchesin, S., Giachelle, F., Silvello, G.: CoreKB Trust Network. Zenodo (2025). https://doi.org/10.5281/zenodo.15748152

[17] Carroll, J.J., Bizer, C., Hayes, P.J., Stickler, P.: Named graphs. J. Web Semant. **3**(4), 247–267 (2005) https://doi.org/10.1016/j.websem.2005.09.001

[18] Chichester, C., Karch, O., Gaudet, P., Lane, L., Mons, B., Bairoch, A.: Converting neXtProt into Linked Data and nanopublications. Semantic Web **6**(2), 147–153 (2015) https://doi.org/10.3233/SW-140149

[19] Vogt, L., Kuhn, T., Hoehndorf, R.: Correction to: Semantic units: organizing knowledge graphs into semantically meaningful units of representation. J. Biomed. Semant. **15**(1), 10 (2024) https://doi.org/10.1186/S13326-024-00313-2

[20] Bucur, C.-I., Kuhn, T., Ceolin, D., Ossenbruggen, J.: Nanopublication-based semantic publishing and reviewing: a field study with formalization papers. PeerJ Comput. Sci. **9**, 1159 (2023) https://doi.org/10.7717/PEERJ-CS.1159

[21] Kuhn, T., Meroño-Peñuela, A., Malic, A., Poelen, J.H., Hurlbert, A.H., Ortiz, E.C., Furlong, L.I., Queralt-Rosinach, N., Chichester, C., Banda, J.M., Willighagen, E.L., Ehrhart, F., Evelo, C.T.A., Malas, T.B., Dumontier, M.: Nanopublications: A Growing Resource of Provenance-Centric Scientific Linked Data. In: Proc. of the 14th IEEE International Conference on e-Science (e-Science 2018), pp. 83–92. IEEE Computer Society, Washington, DC, USA (2018). https://doi.org/10.1109/eScience.2018.00024

[22] Mons, B., Haagen, H., Chichester, C., Hoen, P.-B.t., Dunnen, J.T., Ommen, G., Mulligen, E., Singh, B., Hooft, R., Roos, M., Hammond, J., Kiesel, B., Giardine, B., Velterop, J., Groth, P., Schultes, E.: The value of data. Nature Genetics **43**(4), 281–283 (2011) https://doi.org/10.1038/ng0411-281

[23] Bucur, C.-I., Kuhn, T., Ceolin, D.: A Unified Nanopublication Model for Effective and User-Friendly Access to the Elements of Scientific Publishing. In: Proc. of Knowledge Engineering and Knowledge Management (EKAW 2020). Lecture Notes in Computer Science, vol. 12387, pp. 104–119. Springer, Heidelberg, Germany (2020). https://doi.org/10.1007/978-3-030-61244-3_7

[24] Kuhn, T., Barbano, P.E., Nagy, M.L., Krauthammer, M.: Broadening the Scope of Nanopublications. In: Proc. of The Semantic Web: Semantics and Big Data (ESWC 2013). Lecture Notes in Computer Science, vol. 7882, pp. 487–501. Springer, Heidelberg, Germany (2013). https://doi.org/10.1007/978-3-642-38288-8_33

[25] Clark, T., Ciccarese, P., Goble, C.A.: Micropublications: a semantic model for claims, evidence, arguments and annotations in biomedical communications. J. Biomed. Semant. **5**, 28 (2014) https://doi.org/10.1186/2041-1480-5-28

[26] Rowley, J.E.: The wisdom hierarchy: representations of the DIKW hierarchy. J. Inf. Sci. **33**(2), 163–180 (2007) https://doi.org/10.1177/0165551506070706

[27] Buneman, P., Khanna, S., Tan, W.C.: Why and where: A characterization of data provenance. In: Proc. of the 8th International Conference on Database Theory (ICDT 2001). Lecture Notes in Computer Science, vol. 1973, pp. 316–330. Springer, Heidelberg, Germany (2001). https://doi.org/10.1007/3-540-44503-X_20

[28] Cheney, J., Chiticariu, L., Tan, W.C.: Provenance in Databases: Why, How, and Where. Found. Trends Databases **1**(4), 379–474 (2009) https://doi.org/10.1561/1900000006

[29] Moreau, L., Clifford, B., Freire, J., Futrelle, J., Gil, Y., Groth, P., Kwasnikowska, N., Miles, S., Missier, P., Myers, J., Plale, B., Simmhan, Y., Stephan, E.G., Bussche, J.V.:

The Open Provenance Model core specification (v1.1). Future Gener. Comput. Syst. **27**(6), 743–756 (2011) https://doi.org/10.1016/j.future.2010.07.005

[30] Fox, M.S., Huang, J.: Knowledge Provenance. In: Proc. of the Advances in Artificial Intelligence, 17th Conference of the Canadian Society for Computational Studies of Intelligence, (Canadian AI 2004). Lecture Notes in Computer Science, vol. 3060, pp. 517–523. Springer, Heidelberg, Germany (2004). https://doi.org/10.1007/978-3-540-24840-8_47

[31] Huang, J., Fox, M.S.: Trust Judgment in Knowledge Provenance. In: Proc. of the 16th International Workshop on Database and Expert Systems Applications (DEXA 2005), pp. 524–528. IEEE Computer Society, Washington, DC, USA (2005). https://doi.org/10.1109/DEXA.2005.193

[32] Ciccarese, P., Soiland-Reyes, S., Belhajjame, K., Gray, A.J.G., Goble, C.A., Clark, T.: PAV ontology: provenance, authoring and versioning. J. Biomed. Semant. **4**, 37 (2013) https://doi.org/10.1186/2041-1480-4-37

[33] Hartig, O., Zhao, J.: Publishing and consuming provenance metadata on the web of linked data. In: Proc. of the Provenance and Annotation of Data and Processes - Third International Provenance and Annotation Workshop (IPAW 2010). Lecture Notes in Computer Science, vol. 6378, pp. 78–90. Springer, Aachen, Germany (2010). https://doi.org/10.1007/978-3-642-17819-1_10

[34] Marchesin, S., Menotti, L., Silvello, G., Alonso, O.: CORE: Gene Expression-Cancer Knowledge Base. Zenodo (2023). https://doi.org/10.5281/zenodo.7577127

[35] DeepSeek-AI, Liu, A., et al.: DeepSeek-V3 Technical Report (2025). https://arxiv.org/abs/2412.19437

[36] Grattafiori, A. et al.: The Llama 3 Herd of Models (2024). https://arxiv.org/abs/2407.21783

[37] OpenAI: GPT-4o Mini: Advancing Cost-Efficient Intelligence. https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/. Accessed: 2025-06-26 (2024)

[38] OpenAI, Achiam, J., et al.: GPT-4 Technical Report (2024). https://arxiv.org/abs/2303.08774

[39] Ojha, P., Talukdar, P.: KGEval: Accuracy Estimation of Automatically Constructed Knowledge Graphs. In: Proc. of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017), pp. 1741–1750. Association for Computational Linguistics, Copenhagen, Denmark (2017). https://doi.org/10.18653/v1/d17-1183

[40] Gao, J., Li, X., Xu, Y.E., Sisman, B., Dong, X.L., Yang, J.: Efficient Knowledge Graph Accuracy Evaluation. Proc. VLDB Endow. **12**(11), 1679–1691 (2019) https://doi.org/10.14778/3342263.3342642

[41] Qi, Y., Zheng, W., Hong, L., Zou, L.: Evaluating Knowledge Graph Accuracy Powered by Optimized Human-Machine Collaboration. In: Proc. of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2022), pp. 1368–1378. ACM, New York, NY, USA (2022). https://doi.org/10.1145/3534678.3539233

[42] Xue, B., Zou, L.: Knowledge Graph Quality Management: A Comprehensive Survey. IEEE Trans. Knowl. Data Eng. **35**(5), 4969–4988 (2023) https://doi.org/10.1109/TKDE.2022.3150080

[43] Marchesin, S., Silvello, G.: Efficient and Reliable Estimation of Knowledge Graph Accuracy. Proc. VLDB Endow. **17**(9), 2392–2404 (2024) https://doi.org/10.14778/3665844.3665865

[44] Marchesin, S., Silvello, G.: Credible Intervals for Knowledge Graph Accuracy Estimation. Proc. ACM Manag. Data (SIGMOD) **3**(3) (2025) https://doi.org/10.1145/3725279

[45] Cochran, W.G.: Sampling Techniques, 3rd

Edition. John Wiley & Sons, New York, NY, USA (1977)

[46] Box, G.E.P., Tiao, G.C.: Bayesian Inference in Statistical Analysis. John Wiley & Sons, New York, NY, USA (2011)

[47] Morey, R.D., Hoekstra, R., Rouder, J.N., Lee, M.D., Wagenmakers, E.J.: The Fallacy of Placing Confidence in Confidence Intervals. Psychon. Bull. Rev. **23**, 103–123 (2016) https://doi.org/10.3758/s13423-015-0947-8

[48] Marchesin, S., Silvello, G., Alonso, O.: Utility-oriented knowledge graph accuracy estimation with limited annotations: A case study on dbpedia. Proceedings of the AAAI Conference on Human Computation and Crowdsourcing **12**(1), 105–114 (2024) https://doi.org/10.1609/hcomp.v12i1.31605

[49] Mruthyunjaya, V., Pezeshkpour, P., Hruschka, E., Bhutani, N.: Rethinking Language Models as Symbolic Knowledge Graphs (2023). https://arxiv.org/abs/2308.13676

[50] Sun, K., Xu, Y., Zha, H., Liu, Y., Dong, X.L.: Head-to-tail: How knowledgeable are large language models (LLMs)? A.K.A. will LLMs replace knowledge graphs? In: Duh, K., Gomez, H., Bethard, S. (eds.) Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pp. 311–325. Association for Computational Linguistics, Mexico City, Mexico (2024). https://doi.org/10.18653/v1/2024.naacl-long.18 . https://aclanthology.org/2024.naacl-long.18/

[51] Turpin, M., Michael, J., Perez, E., Bowman, S.R.: Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In: Thirty-seventh Conference on Neural Information Processing Systems (2023). https://openreview.net/forum?id=bzs4uPLXvi

[52] Dong, Q., Li, L., Dai, D., Zheng, C., Ma, J., Li, R., Xia, H., Xu, J., Wu, Z., Chang, B., Sun, X., Li, L., Sui, Z.: A survey on in-context learning. In: Al-Onaizan, Y., Bansal, M., Chen, Y.-N. (eds.) Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pp. 1107–1128. Association for Computational Linguistics, Miami, Florida, USA (2024). https://doi.org/10.18653/v1/2024.emnlp-main.64 . https://aclanthology.org/2024.emnlp-main.64/

[53] Wang, P., Li, L., Chen, L., Cai, Z., Zhu, D., Lin, B., Cao, Y., Kong, L., Liu, Q., Liu, T., Sui, Z.: Large language models are not fair evaluators. In: Ku, L.-W., Martins, A., Srikumar, V. (eds.) Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 9440–9450. Association for Computational Linguistics, Bangkok, Thailand (2024). https://doi.org/10.18653/v1/2024.acl-long.511 . https://aclanthology.org/2024.acl-long.511/

[54] Shami, F., Marchesin, S., Silvello, G.: Fact verification in knowledge graphs using llms. In: Proc. of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2025) (2025). To appear.