# AI Systems Ontology: Project Report

Author:
Leonardo Magliolo

Course:
Semantic Web 2024/2025

University of Turin – Department of Computer Science

# 1 Motivations

The ontology designed for artificial-intelligence systems represents an advanced knowledge-representation tool, essential in both academic and applied contexts. Its formalisation in OWL/SKOS makes it possible to model unambiguously key concepts—such as "model," "learning method," and "application domain"—and the relationships among them, facilitating comparative analysis of scientific publications and experimental datasets.

From a research perspective, the ontology enables:

- **Competency questions** formalised and translatable into optimised SPARQL queries, ensuring reproducibility and domain coverage;
- Automatic inference via OWL reasoners, revealing implicit relations among entities and supporting the discovery of patterns not immediately evident in the original data;
- Semantic information retrieval: rather than string matching, researchers can query the graph using conceptual similarity or semantic paths, improving relevance and reducing noise.

Moreover, the ontology is designed for integration with NLP methods, allowing automatic enrichment of the knowledge graph with entities and relationships extracted from scientific texts. This enables hybrid workflows in which NLP module outputs are mapped directly into RDF triples, accelerating construction and updates of the knowledge base.

Finally, alignment with domain ontologies ensures reusability and interoperability: different research groups can extend or integrate new knowledge without rebuilding the conceptual structure from scratch, fostering collaboration and benchmarking on shared datasets.

In short, the proposed AI-oriented ontology provides a rigorous research infrastructure in which ontological formalisms, semantic inference and NLP integration together enhance both the quality of bibliometric analyses and the discovery of new scientific correlations.

# 2 Requirements for Ontology Creation

## 2.1 General Purpose

The **AI Systems Ontology** was conceived to provide a rigorous **semantic representation** of key AI artefacts (papers, models, tasks, datasets, methods, metrics), reducing terminological ambiguity and making formal relationships among entities explicit. This formal encoding—implemented in OWL/SKOS with namespace `http://example.org/ai-ontology#`—gu

- **Expressiveness** in describing complex classes (e.g. `DeepLearningModel`, `HybridModel`) and their structural and logical properties;
- **Inferability**, through OWL reasoners, to extract implicit relationships (e.g. implications between methods and evaluation metrics);
- **Interoperability**, via alignments (`owl:equivalentClass`, `rdfs:subClassOf`) with sector ontologies.

## 2.2 Specific Tasks and Usage Context

The ontology primarily supports three research and validation tasks:

1. **Semantic querying** – querying a knowledge graph with SPARQL to retrieve structured information (e.g. list of papers that `employModel` with a certain `TrainingAlgorithm`);
2. **Reference and verification** – validating experimental assumptions via competency questions translated into reproducible SPARQL queries (e.g. counting `SymbolicModel` instances and verifying alignment with external resources);
3. **Extraction and population** – automatic integration of entities and relationships extracted from scientific texts (NLP modules for NER and relation extraction) to update the graph's A-Box dynamically.

The application context is academic and industrial research, where data scientists and knowledge engineers require a semantic infrastructure for comparative analysis, workflow auditing and standardised report production.

## 2.3 User Types

- **AI researchers:** need tools to formulate hypotheses, run complex queries and compare methodological approaches;
- **Knowledge engineers and ontologists:** need to extend, align and validate ontological structures following OWL best practices;
- **Data curators and bibliometric analysts:** use semantic encoding to create performance metrics, trend visualisations and support peer-review and compliance processes.

# 3 Domain Description

The **AI Systems Ontology** focuses on a semantic representation of AI research, with particular attention to the main scientific artefacts (papers, authors, conferences/journals) and the technical elements (tasks, methods, models, datasets, evaluation metrics) that frame them. Specifically, the domain covers:

- Papers and their metadata (title, publication date, authors, venue) as primary knowledge units;
- Authors and venues (conferences and journals) to trace collaboration networks and publication standards;
- Tasks (e.g. Computer Vision, Natural Language Processing), methods (classic vs. deep learning), models (symbolic, hybrid, neural) and related `TrainingAlgorithm` to describe applied techniques;
- Datasets and evaluation metrics (accuracy, F1-score, BLEU, etc.) to characterise experimentation and performance benchmarks.

This semantic structure stems from the analysis of existing taxonomies such as the *InnoGraph AI Taxonomy*, which integrates data from Wikipedia, ACM CCS, CSO and other sources to offer a holistic view of AI topics. The domain was further refined by aligning to sector ontologies (e.g. CSO for research areas) and OWL modelling patterns that ensure logical coherence and inferability.

The *InnoGraph Artificial Intelligence Taxonomy* served as a central reference in designing the *AI Systems Ontology.* In particular, it inspired:

- Defining the domain's breadth, including methods, models, tasks, datasets, metrics, publications, authors and application areas;
- Integrating multiple external sources through SKOS annotations, ensuring interoperability with CSO, Wikipedia/CPC and other consolidated taxonomies;
- Segmenting methods and models according to a similar subdivision (symbolic, traditional ML, deep learning) and handling datasets and evaluations with inverse properties, replicating the InnoGraph scheme.

In short, InnoGraph's broad, multi-source, semantic approach guided each design decision, making its taxonomy a continuous inspiration for the ontology's structure and contents.

# 4 Competency Questions

- **Directional retrieval**
  - Which entities (papers, models, authors, tasks, etc.) satisfy a given semantic relationship (e.g. "employ a certain model" or "are published in a conference")?
- **Quantitative analysis**
  - What is the numerical distribution of entities grouped by a property of interest (e.g. number of papers per task, per year, per author per venue)?
- **Coverage verification**
  - Which entities in the domain are not linked to other relevant entities (e.g. unused datasets, models not employed, tasks not addressed)?
- **Composite patterns**
  - Which combinations of classes and properties frequently recur in documents or experimental graphs (e.g. model–training-algorithm pairs, method–evaluation-metric pairs)?
- **External alignment**
  - How do internal entities (papers, authors, models, datasets) compare and integrate with external reference resources (e.g. conferences, repositories)?

# 5 Domain Documentation

## 5.1 Conceptual Structure of the Domain

The AI research domain can be described as a galaxy of entities and relationships reflecting the scientific innovation life-cycle. Fundamental elements of this ecosystem include:

- **People:** researchers, paper authors, patent inventors, software developers, university lecturers and project leaders;
- **Organisations:** universities, research centres, companies, start-ups, funding agencies, consortia and professional associations;
- **Scientific resources:** articles, book chapters, theses, conference presentations, datasets and published AI models;
- **Techniques, models and algorithms:** algorithms, architectures, learning paradigms (machine learning, deep learning, reinforcement learning), specific methods (e.g. attention, regularisation), state-of-the-art models (e.g. GPT-4, BERT, EfficientNet, ResNet);
- **Tasks and problems:** classification, regression, text generation, image recognition, segmentation, question answering, etc.;
- **Application domains:** vertical domains such as medicine, precision agriculture, finance, Industry 4.0, autonomous transport, energy, legaltech, robotics, etc.;
- **Software and infrastructure:** frameworks (TensorFlow, PyTorch, Keras), supporting libraries, model-sharing platforms (HuggingFace), distributed-computing environments;
- **Evaluation and benchmarking:** test datasets, leaderboards, competitions, standardised metrics for performance and robustness.

## 5.2 Concrete Example of Entities and Relationships

A real example illustrating domain complexity is a scientific article published on arXiv, associated with a model available on GitHub, validated via a benchmark on a public dataset (e.g. ImageNet) and cited in a challenge (such as the ImageNet Large Scale Visual Recognition Challenge). In this example, the following relationships stand out:

- The author is affiliated with a university and has released the code on GitHub;
- The model was trained on a public dataset and the results are compared on an international leaderboard;
- The article is referenced by other works and may be patented or further developed by companies.

## 5.3 Documentary Sources and Reference Taxonomies

- **Wikipedia and DBpedia:** articles, categories and glossaries covering techniques, applications, datasets, notable figures and AI history. Outlines and thematic lists help quickly navigate the main AI areas;
- **arXiv:** preprint portal where almost all recent literature is published, with thematic categories (cs.AI, cs.LG, cs.CV, etc.) reflecting the scientific domain's subdivision;
- **PapersWithCode:** portal linking publications, source code, datasets and benchmarking results;
- **ACM CCS, Computer Science Ontology (CSO):** classification systems and taxonomies reflecting computer-science and AI research structure;
- **Patent Classification Systems (e.g. Cooperative Patent Classification, CPC):** classification systems that frame AI inventions with respect to their industrial and technological application domains.

Real-world schema and data examples:

```
Wikipedia Infobox
| Model name       = GPT-4
| Type             = Large Language Model
| Developer        = OpenAI
| Year             = 2023
| Applications     = Text generation, automatic translation, question answering
| Training dataset = Common Crawl, Wikipedia, BooksCorpus

PapersWithCode dataset
Task: Image Classification
Dataset: ImageNet
Model: ResNet-50
Score: Top-1 Accuracy 76.0%
Framework: PyTorch
Paper: "Deep Residual Learning for Image Recognition"
Repository: https://github.com/KaimingHe/deep-residual-networks

Extract from arXiv (cs.AI)
Title: "A General Language Assistant as a Laboratory for Alignment"
Authors: John Smith, Jane Doe
Categories: cs.AI, cs.CL
Abstract: "We introduce a new framework for language model alignment con human prefere
Link: https://arxiv.org/abs/2302.12345

Problem-class table (schematic example)
Macro area     Sub-category             Example Task                   Typical dataset
Computer Vision Image Classification    Object identification          ImageNet, CIFAR-1
Natural Language Question Answering     Free-form Q&A                  SQuAD, NaturalQA
Robotics        Motion Planning         Robotic navigation             KITTI, Waymo
Bioinformatics  Protein Structure       Protein-structure prediction   AlphaFold DB
```

## 5.4 Reference Standards and Models

The domain constantly refers to multiple taxonomies, classifications and reference systems that guarantee interoperability, comparability of results and terminological clarity. Relevant examples:

- **ACM Computing Classification System:** classifies scientific literature according to coded thematic areas;
- **CSO (Computer Science Ontology):** taxonomy automatically updated from scientific literature;
- **CPC (Cooperative Patent Classification):** detailed classification of patented inventions by their technological and application nature;
- **EuroSciVoc:** European thesaurus classifying research projects by scientific and engineering areas, including AI.

These systems allow coherent mapping of domain entities, facilitating alignment and data federation among heterogeneous sources.

## 5.5 Variety and Depth of the Domain

AI scientific research is characterised by marked heterogeneity: boundaries between subdisciplines are blurred and new interdisciplinary areas often emerge (e.g. AI for neuroscience, ethical AI, explainable AI, generative AI). At the same time, the depth of specialisations allows very strong verticalisation: from theory to practice, from general algorithms to micro-tasks on specific data.

The continuous evolution of methodologies, the emergence of new paradigms (e.g. foundation models, few-shot learning, hybrid symbolic-connectionist AI) and the intense scientific output require structured and shared tools for representing, organising and searching domain knowledge.

## 5.6 Real-World Example for the A-Box

**Use case:** A user wants to discover which AI models are most used in medical diagnostics on radiological images.

**Data example:**

- Task: Medical Image Classification
- Dataset: ChestX-ray14
- Model: DenseNet-121
- Lead author: Dr. Jane Doe
- Organisation: Stanford University
- Reference paper: "CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning"
- Score: AUC 0.84

This record, modelled as an individual in the ontology's A-Box, reflects the variety of entities and relationships making up the domain and the need to relate tasks, datasets, models, authors, organisations and results.

# 6 OntOlogy Pitfall Scanner (OOPS)

OOPS reports an empty checklist.

# 7 Visualisation

Below are three key visualisations illustrating, respectively, the class taxonomy, the relational structure around the `Model` class and the relational structure around the `Paper` class. The images are saved in the `images` folder.

## 7.1 Class Taxonomy

Figure 1 shows the class hierarchy starting from `owl:Thing` as the root. At the top are the most general domain classes (e.g. `ApplicationArea`, `Author`, `Conference`, `Dataset`, `Evaluation`, `Journal`), followed by the subclasses related to `Method` and `Model`. Specifically:

- `Method` is subdivided into `HybridMethod`, `MachineLearningMethod` and `RuleBasedMethod`. `MachineLearningMethod` further branches into `DeepLearningMethod` and `TraditionalMLMethod`, whereas `RuleBasedMethod` has the subclass `SymbolicMethod`.
- `Model` has analogous subdivisions: `HybridModel`, `MachineLearningModel` and `RuleBasedModel`. `MachineLearningModel` splits into `DeepLearningModel` and `TraditionalMLModel`, while `RuleBasedModel` has the subclass `SymbolicModel`.
- Leaf classes also include `Paper`, `Repository`, `Task` and `TrainingAlgorithm`.

This taxonomy helps understand how the main entities are organised into coherent hierarchical areas, facilitating OWL reasoners for inference on subsets and logical consistency.

## 7.2 Relational Structure of the `Model` Class

In Figure 2, the `Model` class is at the centre of the graph, with:

- Nodes corresponding to its direct subclasses (`HybridModel`, `MachineLearningModel`, `RuleBasedModel`), each linked to its subclasses (`DeepLearningModel`, `TraditionalMLModel`, `SymbolicModel`);
- The main object properties: e.g. `employsModel` (connecting a `Paper` to a particular `Model`) and its inverse `isModelEmployedBy`;
- Disjointness relationships (annotated as `owl:disjointWith`) separating semantically incompatible classes (e.g. `Model` is disjoint from `Conference`, `Dataset`, `Evaluation`, `Journal`, `Paper`, `Task`, `Author`, `Repository`, `Method`, `Thing` and `Nothing`).

This visualisation highlights where `Model` sits within the ontology, showing both the subclass tree and the main relational properties to other domain classes.

## 7.3 Relational Structure of the Paper Class

Figure 3 shows the `Paper` class at the centre, with:

- Example instances (e.g. `Paper1`, `Paper2`, `Paper3`, . . . ) connected via `rdf:type` to the `Paper` class;
- Relationships to domain classes (`Author`, `Model`, `Task`, `Conference`, `Dataset`, `Evaluation`, `Method`), each marked as `owl:disjointWith` relative to `Paper`, underlining that a `Paper` instance cannot simultaneously belong to any of these other classes;
- Arrows labelled with `predicate` and `type` illustrating the properties used (e.g. `hasTitle`, `hasAuthor`, `addressesTask`, `employsModel`, `publishedIn`, `hasDate`, `usesDataset`, `usesEvaluationMetric`).

This relational map quickly conveys how scientific articles (`Paper`) connect to all other ontology elements and which properties semantically affect graph consistency (through disjointness).

| **Subject** | **Predicate** | **Object** |
|---|---|---|
| `:Author_AliceSmith` | `a` | `:Author` |
| `:Author_AliceSmith` | `:hasTitle` | “Alice Smith” |
| `:Author_BobRossi` | `a` | `:Author` |
| `:Author_BobRossi` | `:hasTitle` | “Bob Rossi” |
| `:ComputerVisionClass` | `a` | `:ApplicationArea` |
| `:ComputerVisionClass` | `rdfs:label` | “Computer Vision” |
| `:Task_ImageClassification` | `a` | `:Task` |
| `:Task_ImageClassification` | `:hasTitle` | “Image Classification” |
| `:Task_ImageClassification` | `:hasApplicationArea` | `:ComputerVisionClass` |
| `:SGD` | `a` | `:TrainingAlgorithm` |
| `:SGD` | `:hasTitle` | “SGD” |
| `:ResNet50` | `a` | `:DeepLearningModel` |
| `:ResNet50` | `:usesTrainingAlgorithm` | `:SGD` |
| `:Paper1` | `a` | `:Paper` |
| `:Paper1` | `:hasTitle` | “Efficient ResNet50 for Image Classification” |
| `:Paper1` | `:hasAuthor` | `:Author_AliceSmith, :Author_BobRossi` |
| `:Paper1` | `:publishedIn` | `:CVPR2023` |
| `:Paper1` | `:hasDate` | “2023-06-20”8sd:date |
| `:Paper1` | `:addressesTask` | `:Task_ImageClassification` |
| `:Paper1` | `:employsModel` | `:ResNet50` |
| `:Paper1` | `:usesDataset` | `:ImageNet` |
| `:Paper1` | `:hasEvaluation` | `:Accuracy, :Precision, :Recall` |

Table 1: Compact example of RDF triples represented at the ontological level

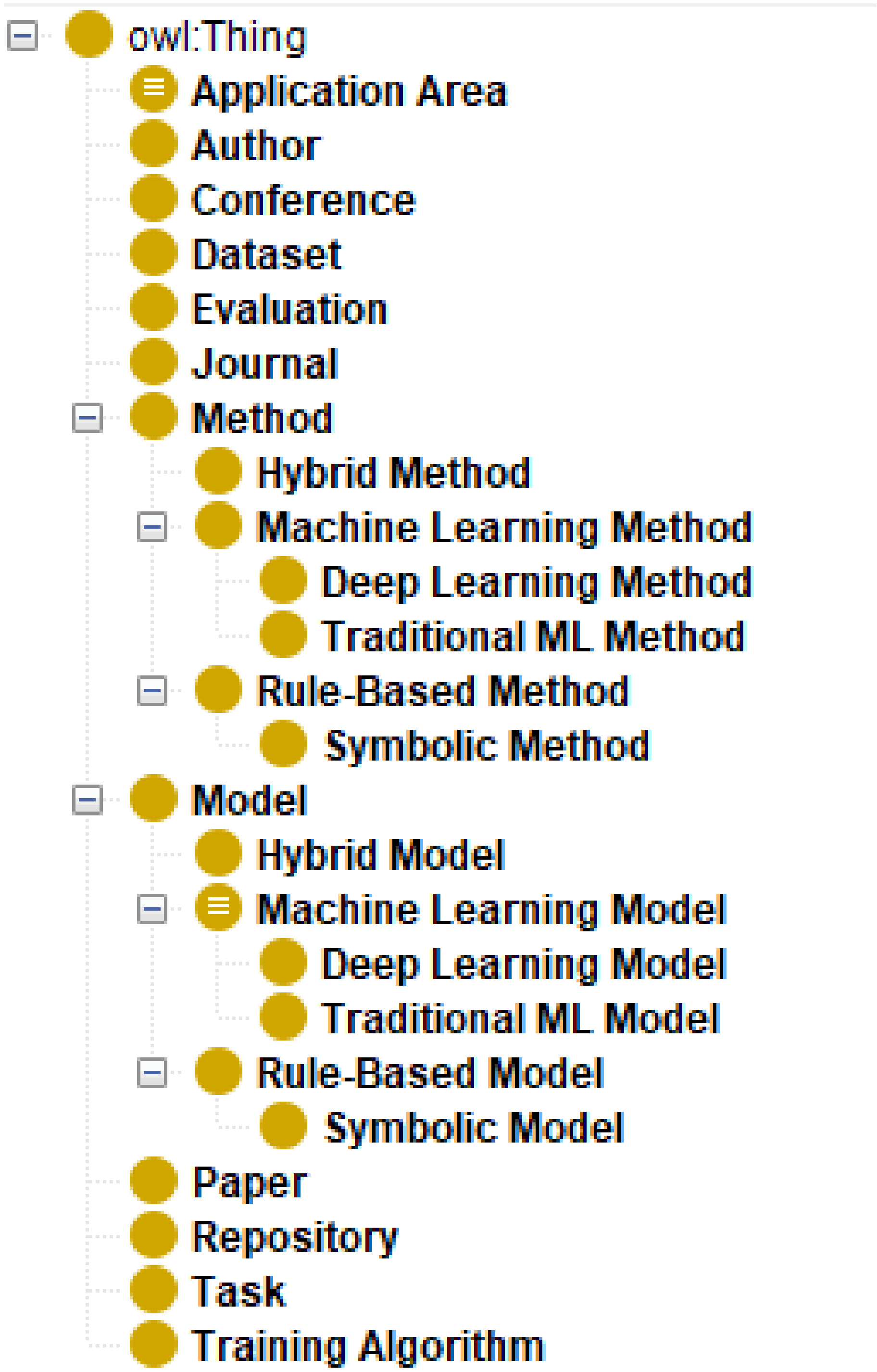


Figure 1: Taxonomy of the main classes in the AI Systems Ontology.

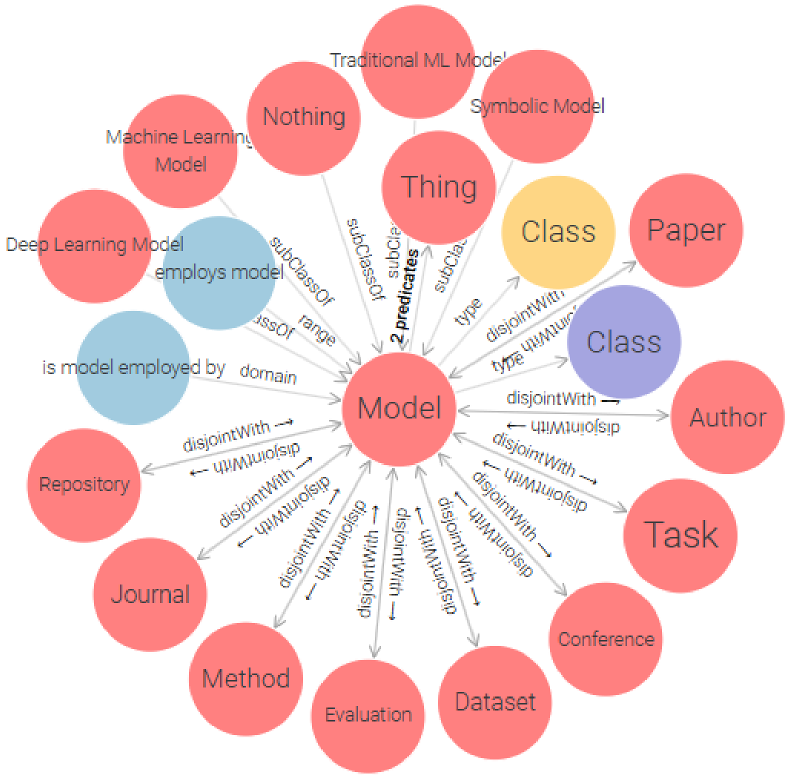


Figure 2: Graph of the main relationships involving the `Model` class.


Figure 3: Graph of the main relationships involving the `Paper` class, including some example instances.

# 8 The Ontology in Turtle Format

```
<?xml version="1.0"?>
<rdf:RDF xmlns="http://example.org/ai-ontology#"
     xml:base="http://example.org/ai-ontology"
     xmlns:dc="http://purl.org/dc/elements/1.1/"
     xmlns:owl="http://www.w3.org/2002/07/owl#"
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:xml="http://www.w3.org/XML/1998/namespace"
     xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
     xmlns:skos="http://www.w3.org/2004/02/skos/core#"
     xmlns:vann="http://purl.org/vann/"
     xmlns:dcterms="http://purl.org/dc/terms/">
    <owl:Ontology rdf:about="http://example.org/ai-ontology">
        <dc:creator>Leonardo Magliolo</dc:creator>
        <dc:date>2024-03-25</dc:date>
        <dc:description xml:lang="en">Ontology modeling AI systems, methods, models, t
        <dc:rights>CC BY 4.0</dc:rights>
        <dc:title xml:lang="en">Ontology of AI Systems</dc:title>
        <dcterms:license rdf:resource="https://creativecommons.org/licenses/by/4.0/"/>
        <vann:preferredNamespacePrefix>ai</vann:preferredNamespacePrefix>
        <vann:preferredNamespaceUri>http://example.org/ai-ontology#</vann:preferredNam
        <rdfs:label xml:lang="en">AI Systems Ontology</rdfs:label>
    </owl:Ontology>



    <!--
    ///////////////////////////////////////////////////////////////////////////////////////
    //
    // Annotation properties
    //
    ///////////////////////////////////////////////////////////////////////////////////////
     -->




    <!-- http://purl.org/dc/elements/1.1/creator -->

    <owl:AnnotationProperty rdf:about="http://purl.org/dc/elements/1.1/creator"/>



    <!-- http://purl.org/dc/elements/1.1/date -->

    <owl:AnnotationProperty rdf:about="http://purl.org/dc/elements/1.1/date"/>
```

```
<!-- http://purl.org/dc/elements/1.1/description -->

<owl:AnnotationProperty rdf:about="http://purl.org/dc/elements/1.1/description"/>



<!-- http://purl.org/dc/elements/1.1/rights -->

<owl:AnnotationProperty rdf:about="http://purl.org/dc/elements/1.1/rights"/>



<!-- http://purl.org/dc/elements/1.1/title -->

<owl:AnnotationProperty rdf:about="http://purl.org/dc/elements/1.1/title"/>



<!-- http://purl.org/dc/terms/license -->

<owl:AnnotationProperty rdf:about="http://purl.org/dc/terms/license"/>



<!-- http://purl.org/vann/preferredNamespacePrefix -->

<owl:AnnotationProperty rdf:about="http://purl.org/vann/preferredNamespacePrefix"/



<!-- http://purl.org/vann/preferredNamespaceUri -->

<owl:AnnotationProperty rdf:about="http://purl.org/vann/preferredNamespaceUri"/>



<!-- http://www.w3.org/2004/02/skos/core#broadMatch -->

<owl:AnnotationProperty rdf:about="http://www.w3.org/2004/02/skos/core#broadMatch"



<!-- http://www.w3.org/2004/02/skos/core#closeMatch -->

<owl:AnnotationProperty rdf:about="http://www.w3.org/2004/02/skos/core#closeMatch"
```

```
<!-- http://www.w3.org/2004/02/skos/core#exactMatch -->

<owl:AnnotationProperty rdf:about="http://www.w3.org/2004/02/skos/core#exactMatch"



<!-- http://www.w3.org/2004/02/skos/core#narrowMatch -->

<owl:AnnotationProperty rdf:about="http://www.w3.org/2004/02/skos/core#narrowMatch



<!--
///////////////////////////////////////////////////////////////////////////////////////
//
// Datatypes
//
///////////////////////////////////////////////////////////////////////////////////////
 -->




<!-- http://www.w3.org/2001/XMLSchema#date -->

<rdfs:Datatype rdf:about="http://www.w3.org/2001/XMLSchema#date"/>



<!--
///////////////////////////////////////////////////////////////////////////////////////
//
// Object Properties
//
///////////////////////////////////////////////////////////////////////////////////////
 -->




<!-- http://example.org/ai-ontology#addressesTask -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#addressesTask">
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#isTaskAddressedBy"
    <rdfs:domain>
```

```xml
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://example.org/ai-ontology#Model"/
                <rdf:Description rdf:about="http://example.org/ai-ontology#Paper"/
            </owl:unionOf>
        </owl:Class>
    </rdfs:domain>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Task"/>
    <rdfs:comment xml:lang="en">Relates a paper or model to the task it addresses
    <rdfs:label xml:lang="en">addresses task</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#applicationAreaOf -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#applicationAreaOf">
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#hasApplicationArea
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#ApplicationArea"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Task"/>
    <rdfs:comment xml:lang="en">Relates an application area to the tasks that belo
    <rdfs:label xml:lang="en">application area of</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#authored -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#authored">
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#hasAuthor"/>
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Author"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Paper"/>
    <rdfs:comment xml:lang="en">Relates an author to the papers they wrote. (Inver
    <rdfs:label xml:lang="en">authored</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#coAuthorOf -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#coAuthorOf">
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#SymmetricProperty"/>
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Author"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Author"/>
    <owl:propertyChainAxiom rdf:parseType="Collection">
        <rdf:Description rdf:about="http://example.org/ai-ontology#authored"/>
        <rdf:Description rdf:about="http://example.org/ai-ontology#hasAuthor"/>
    </owl:propertyChainAxiom>
```

```
    <rdfs:comment xml:lang="en">Two authors are co-authors if they have written tl
    <rdfs:label xml:lang="en">co-author of</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#employsModel -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#employsModel">
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#isModelEmployedBy'
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Paper"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Model"/>
    <rdfs:comment xml:lang="en">Relates a paper to the model it employs.</rdfs:cor
    <rdfs:label xml:lang="en">employs model</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#hasApplicationArea -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#hasApplicationArea":
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Task"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#ApplicationArea"/>
    <rdfs:comment xml:lang="en">Relates a task to its application area.</rdfs:comr
    <rdfs:label xml:lang="en">has application area</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#hasAuthor -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#hasAuthor">
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Paper"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Author"/>
    <rdfs:comment xml:lang="en">Relates a paper to its author(s).</rdfs:comment>
    <rdfs:label xml:lang="en">has author</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#hasAuthorList -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#hasAuthorList">
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#isAuthorListOf"/>
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Paper"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#AuthorList"/>
    <rdfs:comment xml:lang="en">Relates a paper to its ordered list of authors.</r
    <rdfs:label xml:lang="en">has author list</rdfs:label>
```

```
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#hasEvaluation -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#hasEvaluation">
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#isEvaluationOf"/>
    <rdfs:domain>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://example.org/ai-ontology#Model"
                <rdf:Description rdf:about="http://example.org/ai-ontology#Paper"
            </owl:unionOf>
        </owl:Class>
    </rdfs:domain>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Evaluation"/>
    <rdfs:comment xml:lang="en">Relates a paper or model to its evaluation metric
    <rdfs:label xml:lang="en">has evaluation</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#hasFirstNode -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#hasFirstNode">
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#isFirstNodeOf"/>
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#AuthorList"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#AuthorListNode"/>
    <rdfs:comment xml:lang="en">Relates an author list to its first node.</rdfs:c
    <rdfs:label xml:lang="en">has first node</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#hasListAuthor -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#hasListAuthor">
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#isListAuthorOf"/>
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#AuthorListNode"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Author"/>
    <rdfs:comment xml:lang="en">Relates a node in the list to the corresponding a
    <rdfs:label xml:lang="en">has list author</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#hasMethod -->
```

```
<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#hasMethod">
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#isMethodOfModel"/>
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Model"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Method"/>
    <rdfs:comment xml:lang="en">Relates a model to the method it implements.</rdfs
    <rdfs:label xml:lang="en">has method</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#hasNextNode -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#hasNextNode">
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#isPreviousNodeOf"/
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#TransitiveProperty"/>
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#AuthorListNode"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#AuthorListNode"/>
    <rdfs:comment xml:lang="en">Relates a node to the next node in the list.</rdfs
    <rdfs:label xml:lang="en">has next node</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#hostsDataset -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#hostsDataset">
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#isHostedBy"/>
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Repository"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Dataset"/>
    <rdfs:comment xml:lang="en">Relates a repository to the datasets it hosts.</rd
    <rdfs:label xml:lang="en">hosts dataset</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#isAuthorListOf -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#isAuthorListOf">
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#AuthorList"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Paper"/>
    <rdfs:comment xml:lang="en">Relates an author list to the paper that owns it
    <rdfs:label xml:lang="en">is author list of</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#isDatasetUsedBy -->
```

```
<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#isDatasetUsedBy">
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#usesDataset"/>
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Dataset"/>
    <rdfs:range>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://example.org/ai-ontology#Model"/
                <rdf:Description rdf:about="http://example.org/ai-ontology#Paper"/
            </owl:unionOf>
        </owl:Class>
    </rdfs:range>
    <rdfs:comment xml:lang="en">Relates a dataset to the papers or models that rel
    <rdfs:label xml:lang="en">is dataset used by</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#isEvaluationOf -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#isEvaluationOf">
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Evaluation"/>
    <rdfs:range>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://example.org/ai-ontology#Model"/
                <rdf:Description rdf:about="http://example.org/ai-ontology#Paper"/
            </owl:unionOf>
        </owl:Class>
    </rdfs:range>
    <rdfs:comment xml:lang="en">Relates an evaluation metric to the paper or model
    <rdfs:label xml:lang="en">is evaluation of</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#isFirstNodeOf -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#isFirstNodeOf">
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#AuthorListNode"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#AuthorList"/>
    <rdfs:comment xml:lang="en">Relates an author list node to the list it is the
    <rdfs:label xml:lang="en">is first node of</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#isHostedBy -->
```

```
<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#isHostedBy">
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Dataset"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Repository"/>
    <rdfs:comment xml:lang="en">Relates a dataset to the repository that hosts it
    <rdfs:label xml:lang="en">is hosted by</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#isListAuthorOf -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#isListAuthorOf">
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Author"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#AuthorListNode"/>
    <rdfs:comment xml:lang="en">Relates a list node to the author it represents (
    <rdfs:label xml:lang="en">is list author of</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#isMethodOfModel -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#isMethodOfModel">
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Method"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Model"/>
    <rdfs:comment xml:lang="en">Relates a method to the model that implements it.
    <rdfs:label xml:lang="en">is method of model</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#isMethodUsedBy -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#isMethodUsedBy">
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#usesMethod"/>
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Method"/>
    <rdfs:range>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://example.org/ai-ontology#Model"
                <rdf:Description rdf:about="http://example.org/ai-ontology#Paper"
            </owl:unionOf>
        </owl:Class>
    </rdfs:range>
    <rdfs:comment xml:lang="en">Relates a method to the papers or models that empl
    <rdfs:label xml:lang="en">is method used by</rdfs:label>
</owl:ObjectProperty>
```

```
<!-- http://example.org/ai-ontology#isModelEmployedBy -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#isModelEmployedBy">
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Model"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Paper"/>
    <rdfs:comment xml:lang="en">Relates a model to the papers that employ it. (Inv
    <rdfs:label xml:lang="en">is model employed by</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#isPreviousNodeOf -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#isPreviousNodeOf">
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#TransitiveProperty"/>
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#AuthorListNode"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#AuthorListNode"/>
    <rdfs:comment xml:lang="en">Relates a node to the node that precedes it in the
    <rdfs:label xml:lang="en">is previous node of</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#isTaskAddressedBy -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#isTaskAddressedBy">
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Task"/>
    <rdfs:range>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://example.org/ai-ontology#Model"/
                <rdf:Description rdf:about="http://example.org/ai-ontology#Paper"/
            </owl:unionOf>
        </owl:Class>
    </rdfs:range>
    <rdfs:comment xml:lang="en">Relates a task to the papers or models that addres
    <rdfs:label xml:lang="en">is task addressed by</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#isTrainingAlgorithmOf -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#isTrainingAlgorithmO
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#usesTrainingAlgori
```

```xml
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#TrainingAlgorithm"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Model"/>
    <rdfs:comment xml:lang="en">Relates a training algorithm to the model it is us
    <rdfs:label xml:lang="en">is training algorithm of</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#publishedIn -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#publishedIn">
    <owl:inverseOf rdf:resource="http://example.org/ai-ontology#publishes"/>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Paper"/>
    <rdfs:range>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://example.org/ai-ontology#Confere
                <rdf:Description rdf:about="http://example.org/ai-ontology#Journal
            </owl:unionOf>
        </owl:Class>
    </rdfs:range>
    <rdfs:comment xml:lang="en">Relates a paper to its publication venue.</rdfs:co
    <rdfs:label xml:lang="en">published in</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#publishes -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#publishes">
    <rdfs:domain>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://example.org/ai-ontology#Confere
                <rdf:Description rdf:about="http://example.org/ai-ontology#Journal
            </owl:unionOf>
        </owl:Class>
    </rdfs:domain>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Paper"/>
    <rdfs:comment xml:lang="en">Relates a conference or journal to the papers publ
    <rdfs:label xml:lang="en">publishes</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#usesDataset -->
```

```
<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#usesDataset">
    <rdfs:domain>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://example.org/ai-ontology#Model",
                <rdf:Description rdf:about="http://example.org/ai-ontology#Paper",
            </owl:unionOf>
        </owl:Class>
    </rdfs:domain>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Dataset"/>
    <rdfs:comment xml:lang="en">Relates a paper or model to the dataset it uses.</
    <rdfs:label xml:lang="en">uses dataset</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#usesMethod -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#usesMethod">
    <rdfs:domain>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://example.org/ai-ontology#Model",
                <rdf:Description rdf:about="http://example.org/ai-ontology#Paper",
            </owl:unionOf>
        </owl:Class>
    </rdfs:domain>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#Method"/>
    <rdfs:comment xml:lang="en">Relates a paper or model to the method it uses.</r
    <rdfs:label xml:lang="en">uses method</rdfs:label>
</owl:ObjectProperty>



<!-- http://example.org/ai-ontology#usesTrainingAlgorithm -->

<owl:ObjectProperty rdf:about="http://example.org/ai-ontology#usesTrainingAlgorit
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Model"/>
    <rdfs:range rdf:resource="http://example.org/ai-ontology#TrainingAlgorithm"/>
    <rdfs:comment xml:lang="en">Relates a model to the training algorithm it uses
    <rdfs:label xml:lang="en">uses training algorithm</rdfs:label>
</owl:ObjectProperty>



<!--
//////////////////////////////////////////////////////////////////////////////////
//
```

```
// Data properties
//
///////////////////////////////////////////////////////////////////////////////////
 -->




<!-- http://example.org/ai-ontology#hasDOI -->

<owl:DatatypeProperty rdf:about="http://example.org/ai-ontology#hasDOI">
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Paper"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:comment xml:lang="en">Each paper has at most one DOI.</rdfs:comment>
    <rdfs:label xml:lang="en">has DOI</rdfs:label>
</owl:DatatypeProperty>



<!-- http://example.org/ai-ontology#hasDate -->

<owl:DatatypeProperty rdf:about="http://example.org/ai-ontology#hasDate">
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:domain>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://example.org/ai-ontology#Confere
                <rdf:Description rdf:about="http://example.org/ai-ontology#Dataset
                <rdf:Description rdf:about="http://example.org/ai-ontology#Evaluat
                <rdf:Description rdf:about="http://example.org/ai-ontology#Journal
                <rdf:Description rdf:about="http://example.org/ai-ontology#Paper"/
            </owl:unionOf>
        </owl:Class>
    </rdfs:domain>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#date"/>
    <rdfs:comment xml:lang="en">Date of publication/event.</rdfs:comment>
    <rdfs:label xml:lang="en">has date</rdfs:label>
</owl:DatatypeProperty>



<!-- http://example.org/ai-ontology#hasName -->

<owl:DatatypeProperty rdf:about="http://example.org/ai-ontology#hasName">
    <rdfs:domain rdf:resource="http://example.org/ai-ontology#Author"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:comment xml:lang="en">Nome completo dell&apos;autore.</rdfs:comment>
```

```
    <rdfs:label xml:lang="en">has name</rdfs:label>
</owl:DatatypeProperty>



<!-- http://example.org/ai-ontology#hasTitle -->

<owl:DatatypeProperty rdf:about="http://example.org/ai-ontology#hasTitle">
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#FunctionalProperty"/>
    <rdfs:domain>
        <owl:Class>
            <owl:unionOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://example.org/ai-ontology#Dataset
                <rdf:Description rdf:about="http://example.org/ai-ontology#Evaluat
                <rdf:Description rdf:about="http://example.org/ai-ontology#Method'
                <rdf:Description rdf:about="http://example.org/ai-ontology#Model",
                <rdf:Description rdf:about="http://example.org/ai-ontology#Paper",
                <rdf:Description rdf:about="http://example.org/ai-ontology#Reposit
                <rdf:Description rdf:about="http://example.org/ai-ontology#Task"/>
            </owl:unionOf>
        </owl:Class>
    </rdfs:domain>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
    <rdfs:comment xml:lang="en">Title of entity.</rdfs:comment>
    <rdfs:label xml:lang="en">has title</rdfs:label>
</owl:DatatypeProperty>



<!--
///////////////////////////////////////////////////////////////////////////////
//
// Classes
//
///////////////////////////////////////////////////////////////////////////////
 -->




<!-- http://example.org/ai-ontology#ApplicationArea -->

<owl:Class rdf:about="http://example.org/ai-ontology#ApplicationArea">
    <owl:equivalentClass>
        <owl:Class>
            <owl:oneOf rdf:parseType="Collection">
                <rdf:Description rdf:about="http://example.org/ai-ontology#Compute
                <rdf:Description rdf:about="http://example.org/ai-ontology#Natural
```

```
            </owl:oneOf>
        </owl:Class>
    </owl:equivalentClass>
    <rdfs:comment xml:lang="en">
        Enumerazione delle possibili aree applicative (es. CV, NLP).
    </rdfs:comment>
    <rdfs:label xml:lang="en">Application Area</rdfs:label>
</owl:Class>



<!-- http://example.org/ai-ontology#Author -->

<owl:Class rdf:about="http://example.org/ai-ontology#Author">
    <rdfs:comment xml:lang="en">Author of papers.</rdfs:comment>
    <rdfs:label xml:lang="en">Author</rdfs:label>
    <skos:broadMatch rdf:resource="http://xmlns.com/foaf/0.1/Person"/>
</owl:Class>



<!-- http://example.org/ai-ontology#AuthorList -->

<owl:Class rdf:about="http://example.org/ai-ontology#AuthorList">
    <rdfs:comment xml:lang="en">A list of authors in a specific order.</rdfs:comme
    <rdfs:label xml:lang="en">Author List</rdfs:label>
</owl:Class>



<!-- http://example.org/ai-ontology#AuthorListNode -->

<owl:Class rdf:about="http://example.org/ai-ontology#AuthorListNode">
    <rdfs:comment xml:lang="en">A node in the ordered author list.</rdfs:comment>
    <rdfs:label xml:lang="en">Author List Node</rdfs:label>
</owl:Class>



<!-- http://example.org/ai-ontology#ComputerVision -->

<owl:Class rdf:about="http://example.org/ai-ontology#ComputerVision">
    <rdfs:subClassOf rdf:resource="http://example.org/ai-ontology#ApplicationArea
</owl:Class>



<!-- http://example.org/ai-ontology#Conference -->
```

```
<owl:Class rdf:about="http://example.org/ai-ontology#Conference">
    <rdfs:comment xml:lang="en">Academic conference.</rdfs:comment>
    <rdfs:label xml:lang="en">Conference</rdfs:label>
    <skos:exactMatch rdf:resource="http://swrc.ontoware.org/ontology#Conference"/>
</owl:Class>



<!-- http://example.org/ai-ontology#Dataset -->

<owl:Class rdf:about="http://example.org/ai-ontology#Dataset">
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="http://example.org/ai-ontology#hasTitle
            <owl:minQualifiedCardinality rdf:datatype="http://www.w3.org/2001/XML
            <owl:onDataRange rdf:resource="http://www.w3.org/2001/XMLSchema#strin
        </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:comment xml:lang="en">Dataset used in AI.</rdfs:comment>
    <rdfs:label xml:lang="en">Dataset</rdfs:label>
    <skos:exactMatch rdf:resource="http://schema.org/Dataset"/>
</owl:Class>



<!-- http://example.org/ai-ontology#DeepLearningMethod -->

<owl:Class rdf:about="http://example.org/ai-ontology#DeepLearningMethod">
    <rdfs:subClassOf rdf:resource="http://example.org/ai-ontology#MachineLearning
    <rdfs:comment xml:lang="en">A method based on deep learning techniques.</rdfs
    <rdfs:label xml:lang="en">Deep Learning Method</rdfs:label>
</owl:Class>



<!-- http://example.org/ai-ontology#DeepLearningModel -->

<owl:Class rdf:about="http://example.org/ai-ontology#DeepLearningModel">
    <rdfs:subClassOf rdf:resource="http://example.org/ai-ontology#MachineLearning
    <rdfs:comment xml:lang="en">Deep learning neural networks.</rdfs:comment>
    <rdfs:label xml:lang="en">Deep Learning Model</rdfs:label>
    <skos:broadMatch rdf:resource="https://cso.kmi.open.ac.uk/topics/neural_netwo
</owl:Class>



<!-- http://example.org/ai-ontology#Evaluation -->
```

```
<owl:Class rdf:about="http://example.org/ai-ontology#Evaluation">
    <rdfs:comment xml:lang="en">Evaluation metric.</rdfs:comment>
    <rdfs:label xml:lang="en">Evaluation</rdfs:label>
</owl:Class>



<!-- http://example.org/ai-ontology#HybridMethod -->

<owl:Class rdf:about="http://example.org/ai-ontology#HybridMethod">
    <rdfs:subClassOf rdf:resource="http://example.org/ai-ontology#Method"/>
    <rdfs:comment xml:lang="en">A method integrating symbolic and sub-symbolic ap
    <rdfs:label xml:lang="en">Hybrid Method</rdfs:label>
</owl:Class>



<!-- http://example.org/ai-ontology#HybridModel -->

<owl:Class rdf:about="http://example.org/ai-ontology#HybridModel">
    <rdfs:subClassOf rdf:resource="http://example.org/ai-ontology#Model"/>
    <rdfs:comment xml:lang="en">Hybrid AI models integrating symbolic and sub-sym
    <rdfs:label xml:lang="en">Hybrid Model</rdfs:label>
    <skos:exactMatch rdf:resource="https://cso.kmi.open.ac.uk/topics/neurosymbolic
</owl:Class>



<!-- http://example.org/ai-ontology#Journal -->

<owl:Class rdf:about="http://example.org/ai-ontology#Journal">
    <rdfs:comment xml:lang="en">Academic journal.</rdfs:comment>
    <rdfs:label xml:lang="en">Journal</rdfs:label>
    <skos:exactMatch rdf:resource="http://swrc.ontoware.org/ontology#Journal"/>
</owl:Class>



<!-- http://example.org/ai-ontology#MachineLearningMethod -->

<owl:Class rdf:about="http://example.org/ai-ontology#MachineLearningMethod">
    <rdfs:subClassOf rdf:resource="http://example.org/ai-ontology#Method"/>
    <rdfs:comment xml:lang="en">A method employing machine learning techniques.</
    <rdfs:label xml:lang="en">Machine Learning Method</rdfs:label>
</owl:Class>
```

```
<!-- http://example.org/ai-ontology#MachineLearningModel -->

<owl:Class rdf:about="http://example.org/ai-ontology#MachineLearningModel">
    <owl:equivalentClass>
        <owl:Restriction>
            <owl:onProperty rdf:resource="http://example.org/ai-ontology#usesTrai
            <owl:someValuesFrom rdf:resource="http://example.org/ai-ontology#Trai
        </owl:Restriction>
    </owl:equivalentClass>
    <rdfs:subClassOf rdf:resource="http://example.org/ai-ontology#Model"/>
    <rdfs:comment xml:lang="en">ML model trained by algorithms.</rdfs:comment>
    <rdfs:label xml:lang="en">Machine Learning Model</rdfs:label>
    <skos:exactMatch rdf:resource="https://cso.kmi.open.ac.uk/topics/machine_lear
</owl:Class>



<!-- http://example.org/ai-ontology#Method -->

<owl:Class rdf:about="http://example.org/ai-ontology#Method">
    <rdfs:comment xml:lang="en">AI method used.</rdfs:comment>
    <rdfs:label xml:lang="en">Method</rdfs:label>
</owl:Class>



<!-- http://example.org/ai-ontology#Model -->

<owl:Class rdf:about="http://example.org/ai-ontology#Model">
    <rdfs:comment xml:lang="en">Generic AI model.</rdfs:comment>
    <rdfs:label xml:lang="en">Model</rdfs:label>
</owl:Class>



<!-- http://example.org/ai-ontology#NaturalLanguageProcessing -->

<owl:Class rdf:about="http://example.org/ai-ontology#NaturalLanguageProcessing">
    <rdfs:subClassOf rdf:resource="http://example.org/ai-ontology#ApplicationArea
</owl:Class>



<!-- http://example.org/ai-ontology#Paper -->

<owl:Class rdf:about="http://example.org/ai-ontology#Paper">
    <rdfs:subClassOf>
```

```
        <owl:Restriction>
            <owl:onProperty rdf:resource="http://example.org/ai-ontology#hasAuthor
            <owl:minQualifiedCardinality rdf:datatype="http://www.w3.org/2001/XMLS
            <owl:onClass rdf:resource="http://example.org/ai-ontology#Author"/>
        </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:comment xml:lang="en">Research paper.</rdfs:comment>
    <rdfs:label xml:lang="en">Paper</rdfs:label>
    <skos:exactMatch rdf:resource="http://purl.org/ontology/bibo/AcademicArticle"/
    <skos:narrowMatch rdf:resource="http://purl.org/ontology/bibo/Document"/>
</owl:Class>



<!-- http://example.org/ai-ontology#Repository -->

<owl:Class rdf:about="http://example.org/ai-ontology#Repository">
    <rdfs:comment xml:lang="en">Repository hosting resources.</rdfs:comment>
    <rdfs:label xml:lang="en">Repository</rdfs:label>
</owl:Class>



<!-- http://example.org/ai-ontology#RuleBasedMethod -->

<owl:Class rdf:about="http://example.org/ai-ontology#RuleBasedMethod">
    <rdfs:subClassOf rdf:resource="http://example.org/ai-ontology#Method"/>
    <rdfs:comment xml:lang="en">A method based on explicit rules.</rdfs:comment>
    <rdfs:label xml:lang="en">Rule-Based Method</rdfs:label>
</owl:Class>



<!-- http://example.org/ai-ontology#RuleBasedModel -->

<owl:Class rdf:about="http://example.org/ai-ontology#RuleBasedModel">
    <rdfs:subClassOf rdf:resource="http://example.org/ai-ontology#Model"/>
    <rdfs:comment xml:lang="en">AI model based on explicit rules.</rdfs:comment>
    <rdfs:label xml:lang="en">Rule-Based Model</rdfs:label>
    <skos:narrowMatch rdf:resource="https://cso.kmi.open.ac.uk/topics/expert_knowl
</owl:Class>



<!-- http://example.org/ai-ontology#SymbolicMethod -->

<owl:Class rdf:about="http://example.org/ai-ontology#SymbolicMethod">
    <rdfs:subClassOf rdf:resource="http://example.org/ai-ontology#RuleBasedMethod"
```

```
    <rdfs:comment xml:lang="en">A method using symbolic approaches.</rdfs:comment>
    <rdfs:label xml:lang="en">Symbolic Method</rdfs:label>
</owl:Class>



<!-- http://example.org/ai-ontology#SymbolicModel -->

<owl:Class rdf:about="http://example.org/ai-ontology#SymbolicModel">
    <rdfs:subClassOf rdf:resource="http://example.org/ai-ontology#RuleBasedModel"/
    <rdfs:comment xml:lang="en">Symbolic AI models.</rdfs:comment>
    <rdfs:label xml:lang="en">Symbolic Model</rdfs:label>
    <skos:narrowMatch rdf:resource="https://cso.kmi.open.ac.uk/topics/knowledge_re
</owl:Class>



<!-- http://example.org/ai-ontology#Task -->

<owl:Class rdf:about="http://example.org/ai-ontology#Task">
    <rdfs:comment xml:lang="en">AI task addressed.</rdfs:comment>
    <rdfs:label xml:lang="en">Task</rdfs:label>
</owl:Class>



<!-- http://example.org/ai-ontology#TraditionalMLMethod -->

<owl:Class rdf:about="http://example.org/ai-ontology#TraditionalMLMethod">
    <rdfs:subClassOf rdf:resource="http://example.org/ai-ontology#MachineLearningM
    <rdfs:comment xml:lang="en">A method based on traditional machine learning (e
    <rdfs:label xml:lang="en">Traditional ML Method</rdfs:label>
    <skos:narrowMatch rdf:resource="https://cso.kmi.open.ac.uk/topics/machine_lear
</owl:Class>



<!-- http://example.org/ai-ontology#TraditionalMLModel -->

<owl:Class rdf:about="http://example.org/ai-ontology#TraditionalMLModel">
    <rdfs:subClassOf rdf:resource="http://example.org/ai-ontology#MachineLearningM
    <rdfs:comment xml:lang="en">Traditional ML algorithms.</rdfs:comment>
    <rdfs:label xml:lang="en">Traditional ML Model</rdfs:label>
    <skos:narrowMatch rdf:resource="https://cso.kmi.open.ac.uk/topics/machine_lear
</owl:Class>
```

```
<!-- http://example.org/ai-ontology#TrainingAlgorithm -->

<owl:Class rdf:about="http://example.org/ai-ontology#TrainingAlgorithm">
    <rdfs:comment xml:lang="en">Algorithm for training models.</rdfs:comment>
    <rdfs:label xml:lang="en">Training Algorithm</rdfs:label>
</owl:Class>



<!--
///////////////////////////////////////////////////////////////////////////////
//
// Individuals
//
///////////////////////////////////////////////////////////////////////////////
 -->




<!-- http://example.org/ai-ontology#ComputerVision -->

<owl:NamedIndividual rdf:about="http://example.org/ai-ontology#ComputerVision"/>



<!-- http://example.org/ai-ontology#ComputerVisionClass -->

<owl:NamedIndividual rdf:about="http://example.org/ai-ontology#ComputerVisionClas
    <rdf:type rdf:resource="http://example.org/ai-ontology#ApplicationArea"/>
    <rdfs:label xml:lang="en">Computer Vision</rdfs:label>
</owl:NamedIndividual>



<!-- http://example.org/ai-ontology#NaturalLanguageProcessing -->

<owl:NamedIndividual rdf:about="http://example.org/ai-ontology#NaturalLanguagePro



<!-- http://example.org/ai-ontology#NaturalLanguageProcessingClass -->

<owl:NamedIndividual rdf:about="http://example.org/ai-ontology#NaturalLanguagePro
    <rdf:type rdf:resource="http://example.org/ai-ontology#ApplicationArea"/>
    <rdfs:label xml:lang="en">Natural Language Processing</rdfs:label>
</owl:NamedIndividual>
```

```
<!--
///////////////////////////////////////////////////////////////////////////////
//
// Annotations
//
///////////////////////////////////////////////////////////////////////////////
 -->

<rdf:Description rdf:about="http://example.org/ai-ontology#ComputerVision">
    <rdfs:comment xml:lang="en">Subset di ApplicationArea per Computer Vision.</r
    <rdfs:label xml:lang="en">Computer Vision</rdfs:label>
</rdf:Description>
<rdf:Description rdf:about="http://example.org/ai-ontology#NaturalLanguageProcess
    <rdfs:comment xml:lang="en">Subset di ApplicationArea per NLP.</rdfs:comment>
    <rdfs:label xml:lang="en">Natural Language Processing</rdfs:label>
</rdf:Description>
<rdf:Description rdf:about="https://cso.kmi.open.ac.uk/topics/expert_knowledge">
    <rdfs:comment xml:lang="en">Computer Science Ontology concept for the topic &a
    <rdfs:label xml:lang="en">Expert Knowledge (CSO)</rdfs:label>
</rdf:Description>
<rdf:Description rdf:about="https://cso.kmi.open.ac.uk/topics/knowledge_representa
    <rdfs:comment xml:lang="en">Computer Science Ontology concept for the topic &a
    <rdfs:label xml:lang="en">Knowledge Representation (CSO)</rdfs:label>
</rdf:Description>
<rdf:Description rdf:about="https://cso.kmi.open.ac.uk/topics/machine_learning">
    <rdfs:comment xml:lang="en">Computer Science Ontology concept for the topic &a
    <rdfs:label xml:lang="en">Machine Learning (CSO)</rdfs:label>
</rdf:Description>
<rdf:Description rdf:about="https://cso.kmi.open.ac.uk/topics/neural_networks">
    <rdfs:comment xml:lang="en">Computer Science Ontology concept for the topic &a
    <rdfs:label xml:lang="en">Neural Networks (CSO)</rdfs:label>
</rdf:Description>
<rdf:Description rdf:about="https://cso.kmi.open.ac.uk/topics/neurosymbolic_ai">
    <rdfs:comment xml:lang="en">Computer Science Ontology concept for the topic &a
    <rdfs:label xml:lang="en">Neuro-Symbolic AI (CSO)</rdfs:label>
</rdf:Description>



<!--
///////////////////////////////////////////////////////////////////////////////
//
// General axioms
//
///////////////////////////////////////////////////////////////////////////////
 -->
```

```
    <rdf:Description>
        <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AllDisjointClasses"/>
        <owl:members rdf:parseType="Collection">
            <rdf:Description rdf:about="http://example.org/ai-ontology#Author"/>
            <rdf:Description rdf:about="http://example.org/ai-ontology#Conference"/>
            <rdf:Description rdf:about="http://example.org/ai-ontology#Dataset"/>
            <rdf:Description rdf:about="http://example.org/ai-ontology#Evaluation"/>
            <rdf:Description rdf:about="http://example.org/ai-ontology#Journal"/>
            <rdf:Description rdf:about="http://example.org/ai-ontology#Method"/>
            <rdf:Description rdf:about="http://example.org/ai-ontology#Model"/>
            <rdf:Description rdf:about="http://example.org/ai-ontology#Paper"/>
            <rdf:Description rdf:about="http://example.org/ai-ontology#Repository"/>
            <rdf:Description rdf:about="http://example.org/ai-ontology#Task"/>
        </owl:members>
    </rdf:Description>
    <rdf:Description>
        <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AllDisjointClasses"/>
        <owl:members rdf:parseType="Collection">
            <rdf:Description rdf:about="http://example.org/ai-ontology#DeepLearningMet
            <rdf:Description rdf:about="http://example.org/ai-ontology#HybridMethod"/>
            <rdf:Description rdf:about="http://example.org/ai-ontology#SymbolicMethod"
            <rdf:Description rdf:about="http://example.org/ai-ontology#TraditionalMLMe
        </owl:members>
    </rdf:Description>
    <rdf:Description>
        <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AllDisjointClasses"/>
        <owl:members rdf:parseType="Collection">
            <rdf:Description rdf:about="http://example.org/ai-ontology#DeepLearningMod
            <rdf:Description rdf:about="http://example.org/ai-ontology#HybridModel"/>
            <rdf:Description rdf:about="http://example.org/ai-ontology#SymbolicModel"/
            <rdf:Description rdf:about="http://example.org/ai-ontology#TraditionalMLMo
        </owl:members>
    </rdf:Description>
</rdf:RDF>



<!-- Generated by the OWL API (version 4.5.29.2024-05-13T12:11:03Z) https://github.com
```

# User-Interaction Flow

## 1. General Description

The interface targets two user types:

- **Researcher/Student:** wants to search papers, models and tasks within the ontology;

- **Administrator/Curator:** has permission to add new items (authors, papers, models, tasks) and modify their metadata.

Main *use cases*:

1. **Search papers by author or task:** the user enters an author name or task title and receives a list of matching papers;
2. **Hierarchical browsing:** the user explores application areas (Computer Vision, NLP, etc.) to view corresponding tasks and models;
3. **Data entry/editing (Administrator only):** add a new paper with associated author, date, task and model.

---

# Interface Schema (Text-Only Mock-Up)

## 1. Login Screen (Administrator only)

```
+-------------------------------------------------------+
|                    AI Systems Ontology                |
|-------------------------------------------------------|
|  [ University Logo ]                                   |
|                                                       |
|   +-------------------+    +----------------------+   |
|   | Username:       |    | (input field)       |   |
|   +-------------------+    +----------------------+   |
|                                                       |
|   +-------------------+    +----------------------+   |
|   | Password:       |    | (input field)       |   |
|   +-------------------+    +----------------------+   |
|                                                       |
|   [     Login button     ]    [  Cancel ]             |
+-------------------------------------------------------+
```

## 2. Main Menu (Regular User)

```
+-------------------------------------------------------+
|                    AI Systems Ontology                |
|-------------------------------------------------------|
|  Welcome, Researcher!                                  |
|                                                       |
|  [1] Search Papers by Author                           |
|  [2] Search Papers by Task                             |
|  [3] Explore Application Areas                         |
|  [4] Logout                                            |
|                                                       |
|  Select an option [1-4]: [   ]                         |
+-------------------------------------------------------+
```

## 3. Search Interface (By Author or Task)

```
+-------------------------------------------------------+
|                     Paper Search                      |
|-------------------------------------------------------|
|  Enter Author Name: [__________________________]      |
|                                                       |
|  Or                                              »
|                                                       |
|  Select Task: [Dropdown with all Tasks]               |
|                                                       |
|  [   Search   ]    [   Back   ]                       |
+-------------------------------------------------------+
```

## 4. Search Results

```
+-------------------------------------------------------+
|                     Search Results                    |
|-------------------------------------------------------|
|  List of papers found:                                |
|                                                       |
|  1. "Efficient ResNet50 for Image Classification"     |
|     Authors: Alice Smith, Bob Rossi                   |
|     Date: 2023-06-20    Venue: CVPR2023               |
|                                                       |
|  2. "YOLOv5: Real-Time Object Detection"              |
|     Authors: Bob Rossi, Carla Bianchi                 |
|     Date: 2023-06-21    Venue: CVPR2023               |
|                                                       |
|  3. "Text Generation with Transformer Models"         |
|     Authors: Giovanni Esposito                        |
|     Date: 2023-05-20    Venue: NeurIPS2023            |
|                                                       |
|  ...                                                    |
|                                                       |
|  Enter paper number for details [   ]                 |
|  [  Back  ]                                           |
+-------------------------------------------------------+
```

## 5. Paper Detail View

```
+-------------------------------------------------------+
|                 Selected Paper Details                |
|-------------------------------------------------------|
|  Title: "Efficient ResNet50 for Image Classification"|
|  Authors: Alice Smith, Bob Rossi                      |
|  Date: 2023-06-20                                     |
|  Venue: CVPR2023                                      |
|  Task: Image Classification                           |
```

```
|  Model: ResNet50                                     |
|  Dataset: ImageNet                                   |
|  Metrics: Accuracy, Precision, Recall                |
|                                                      |
|  [  Back to Results  ]                               |
|  [  New Search       ]                               |
+------------------------------------------------------+
```

---

# Real-Data Interaction Example

## Scenario: Search Papers by Author

1. **Initial Screen (Regular User):**

```
+------------------------------------------------------+
|                   AI Systems Ontology                |
|------------------------------------------------------|
|  Welcome, Researcher!                                  |
|                                                        |
|  [1] Search Papers by Author                           |
|  [2] Search Papers by Task                             |
|  [3] Explore Application Areas                         |
|  [4] Logout                                            |
|                                                        |
|  Select an option [1-4]: [ 1 ]                         |
+------------------------------------------------------+
```

   The user types `1` and presses Enter.

2. **Search Interface (By Author):**

```
+------------------------------------------------------+
|                   Paper Search                       |
|------------------------------------------------------|
|  Enter Author Name: [ Giovanni Esposito ]            |
|                                                      |
|  [   Search   ]    [   Back   ]                      |
+------------------------------------------------------+
```

   The user enters `Giovanni Esposito` and chooses `Search`.

   *(Assuming the database contains a single author "Giovanni Esposito" and the following results:)*

   - **Paper003:** "Text Generation with Transformer Models" – 2023-05-20 – Venue: NeurIPS2023

- **Paper010:** "New Frontiers in Text Generation" – 2024-02-15 – Venue: NeurIPS2023

3. **Search Results:**

```
+-------------------------------------------------------+
|                    Search Results                     |
|-------------------------------------------------------|
|  Papers found for author "Giovanni Esposito":         |
|                                                       |
|  1. "New Frontiers in Text Generation"                |
|     Date: 2024-02-15    Venue: NeurIPS2023            |
|                                                       |
|  2. "Text Generation with Transformer Models"         |
|     Date: 2023-05-20    Venue: NeurIPS2023            |
|                                                       |
|  Enter paper number for details [ 1-2 ]               |
|  [  Back  ]                                           |
+-------------------------------------------------------+
```

The user types `1` to see the details of "New Frontiers in Text Generation".

4. **Paper Detail View:**

```
+-------------------------------------------------------+
|                 Selected Paper Details                |
|-------------------------------------------------------|
|  Title: "New Frontiers in Text Generation"            |
|  Authors: Giovanni Esposito, Anna Ferretti            |
|  Date: 2024-02-15                                     |
|  Venue: NeurIPS2023                                   |
|  Task: Text Generation                                |
|  Model: Transformer Model                             |
|  Dataset: WikiText-103                                |
|  Metrics: BLEU, ROUGE                                 |
|                                                       |
|  [  Back to Results  ]                                |
|  [  New Search       ]                                |
+-------------------------------------------------------+
```

The user can now go back to browse other papers or start a new search.

---

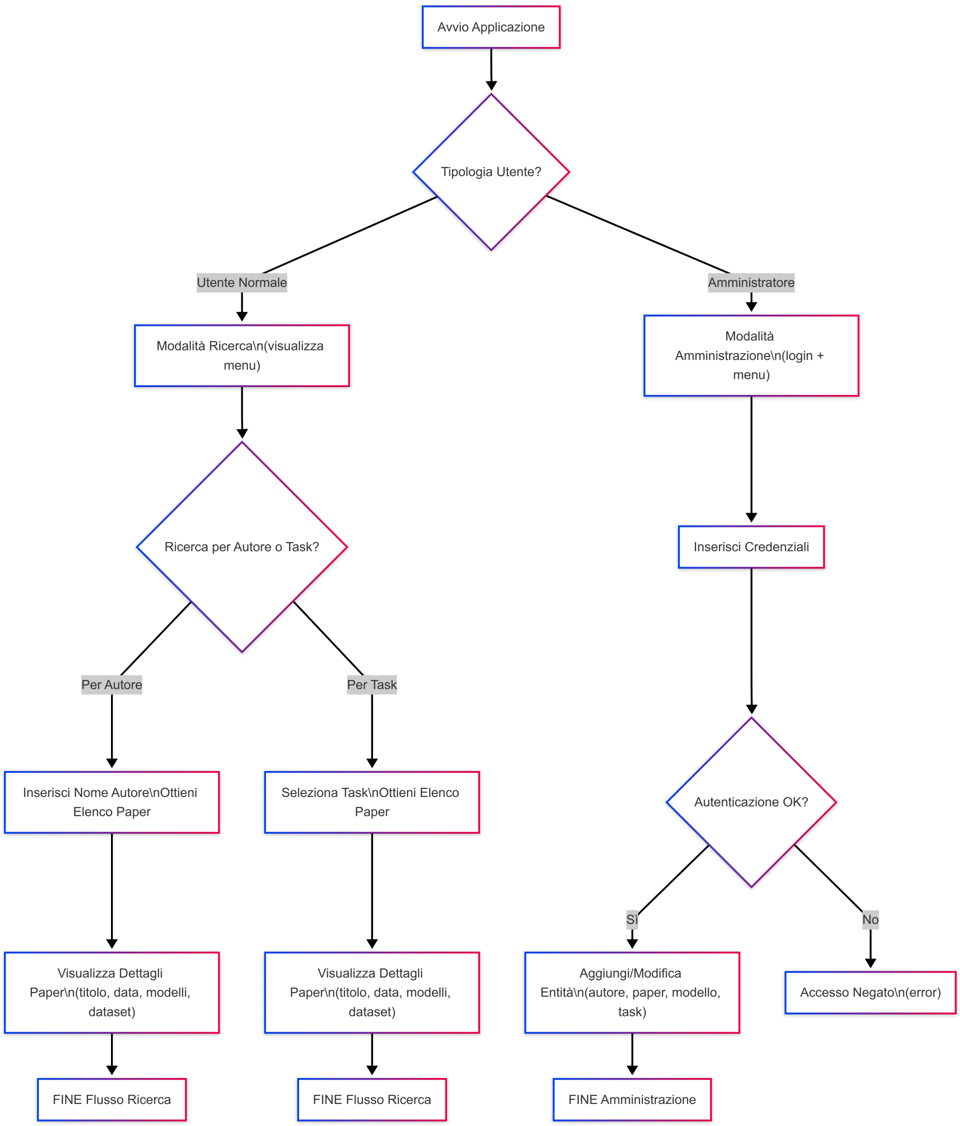


Figure 4: Flowchart illustrating a hypothetical user-interaction scheme

# Final Notes

- The **flow chart** clearly shows the separation between “Regular” and “Administrator” users and the logical paths of searching vs. data management;
- The **interface schema** offers simple textual mock-ups, sufficient to grasp the layout of fields and commands without graphical elements;
- The **real-data interaction example** (author “Giovanni Esposito”) demonstrates how the system sends SPARQL queries, receives results and displays them intuitively.

# References


- Ontotext. “The InnoGraph Artificial Intelligence Taxonomy.”
  https://www.ontotext.com/blog/the-innograph-artificial-intelligence-taxonomy/
- Computer Science Ontology (CSO).
  https://cso.kmi.open.ac.uk/home
- Wikipedia, “Artificial intelligence” (English edition).
  https://en.wikipedia.org/wiki/Artificial_intelligence
- Wikipedia, “Research” (English edition).
  https://en.wikipedia.org/wiki/Research