Neurosymbolic Artificial Intelligence 1 (2024) 1–10
IOS Press

On the Potential of Logic and Reasoning in Neurosymbolic Systems using OWL-based Knowledge Graphs

A position paper

David Herron [a,*], Ernesto Jiménez-Ruiz [a,b] and Tillman Weyde [a]
[a] *City, University of London, UK*
E-mails: david.herron@city.ac.uk, ernesto.jimenez-ruiz@city.ac.uk, t.e.weyde@city.ac.uk
[b] *University of Oslo, Norway*

Abstract. Knowledge graphs feature ever more frequently as symbolic components in neurosymbolic research and systems. But even though a central concern of neurosymbolic AI is to combine neural learning with symbolic reasoning, relatively little neurosymbolic research focuses on leveraging the logical representation and reasoning capabilities of OWL-based knowledge graphs. The objective of this position paper is to inspire more neurosymbolic researchers to embrace the OWL and the Semantic Web by raising awareness of the benefits, capabilities, and applications of OWL-based knowledge graphs, particularly with respect to logical reasoning. We describe the ecosystem of open W3C standards-based resources available that support the adoption and use of OWL-based knowledge graphs; we describe tools that exist for engineering custom OWL ontologies tailored to particular research needs; we discuss the encoding of background KG knowledge in subsymbolic embedding spaces and various applications of this approach; we discuss and illustrate the reasoning capabilities of OWL-based knowledge graphs; and we describe several promising directions for research that focus on leveraging these reasoning capabilities. We also discuss the specialised resources needed to undertake research on OWL-based knowledge graphs in neurosymbolic systems. We use the example of NeSy4VRD, an image dataset with a custom-designed companion OWL ontology. The scarcity of this kind of resource should be addressed to accelerate research in this field.

Keywords: neurosymbolic, AI, deep learning, Semantic Web, OWL, ontologies, knowledge graphs, reasoning

1. Introduction

Following a long gestation spanning decades, neurosymbolic artificial intelligence (NeSy AI) has recently blossomed into a recognised subfield of AI. While neural and symbolic traditions of AI have been tribally rival, recently there is a vibrant diversity of approaches blending the two [1]. Prompted by analysis of the limitations of deep learning (in, *e.g.*, [2–6]), and despite the recent advances resulting from scaling up deep learning, as evidenced in large language models (LLMs), increasing numbers of researchers are drawn to NeSy AI. The shared motivation is to explore combinations of neural learning and symbolic knowledge representations in order to get the best of both worlds, in a shared belief that this is the best route for advancing AI towards artificial general intelligence.

Knowledge graphs (KGs) are representations of symbolic knowledge that conform to a graph model, where nodes are concepts and entities of interest and edges are relationships between them [7, 8]. As NeSy research has

*Corresponding author. E-mail: david.herron@city.ac.uk.

expanded, so has the frequency with which KGs feature as symbolic components in hybrid, NeSy systems [9]. The theme of 'deep deductive reasoning', where neural networks (NNs) are trained to reason over KGs, is progressively developing [10–12]. KGs can be particularly effective when data samples are expensive, difficult or impossible to obtain, so that there is a lack of data to train robust deep learning-based systems, as in few-shot and zero-shot learning scenarios [13–15].

The Web Ontology Language (OWL) [16, 17] is a key component of the Semantic Web technology stack [18, 19]. OWL ontologies (semantic schemas enriched with logic that describe domain knowledge symbolically) govern Semantic Web KGs by specifying what assertions of knowledge (types of triples) are admissible and inadmissible. Inference semantics (ontological rules) associated with OWL constructs permit reasoning algorithms to reason over OWL ontologies (and associated KG data, vast or tiny), both to infer new knowledge (new triples) and to enforce logical consistency constraints. With suitable ontology design, inference can be used as prediction and incremental (on-the-fly) reasoning can facilitate real-time interaction. In summary, OWL-based KGs can be used as symbolic deduction engines in NeSy systems. The NeSy system AlphaGeometry [20] combines an LLM with a symbolic deduction engine (Horn clause geometry and algebra rules, plus inference algorithms) to solve geometry problems. OWL-based KG technologies offer researchers the option to explore combining neural learning and symbolic reasoning in ways analogous to AlphaGeometry, by using OWL-based symbolic deduction during NN training and/or inference.

Given that combining neural learning with symbolic reasoning is central to NeSy AI, it is surprising how scant the literature is that explores applications of OWL reasoning in NeSy systems. A systematic mapping study of 476 recent papers that combine Semantic Web technologies with machine learning [21] reports that only 29 (about 6%) mention using semantic processing modules of some kind (where, by 'semantic', the study means symbolic knowledge representation). The dominant use cases for such modules relate to rulesets (learning, improving, applying) and to data enrichment. The study also finds that of these 29 papers, only 20 (about 4% of the total) mention using reasoning capabilities to infer knowledge. We consulted that study's companion Semantic Web and machine learning systems knowledge graph resource, SWeMLS-KG [22], to find those 4% of papers. Of the 17 we identified, we found only 5 to use OWL reasoning. Another recent survey and vision paper [23] reviews the role of KGs in machine learning, pointing out gaps and opportunities, and also observes that KG symbolic reasoning methods are under-explored and largely disregarded.

One factor explaining this under-exploration may be the cross-disciplinary nature of the endeavour. NeSy research with OWL-based KG reasoning requires researchers to be familiar not just with deep learning, KGs, and logic, but with Semantic Web technologies, especially OWL ontologies, as well. The authors of [23] point to the prevalence of huge public KGs and the perceived scalability limitations of symbolic reasoning methods in the face of such large KGs as an explanation. A third factor may be that the possibility of using OWL-based KG technologies to tailor symbolic deduction engines for NeSy systems has been under-recognised. After all, the Semantic Web was not conceived with such applications in mind.

The objective of this paper (which builds on [24]) is to argue the case for NeSy AI research using OWL-based KGs. We hope to inspire more NeSy research using OWL-based KGs by raising awareness of their benefits, capabilities, and flexible applications, especially with respect to reasoning. OWL-based KGs are exemplars of the symbolic knowledge representation and symbol manipulation and reasoning machinery that critiques of deep learning, such as in [2–6], advocate be incorporated in NeSy systems. By drawing upon illustrative examples from our own research in visual relationship detection as well as from the literature, and by describing promising research directions, we hope to convince readers of the potential of OWL-based KGs. We also discuss how to enable more NeSy research using OWL-based KGs through the creation of resources such as the recently contributed NeSy4VRD (Neurosymbolic AI for Visual Relationship Detection). NeSy4VRD addresses the scarcity of specialised resources required for such research: datasets that combine data for neural learning with companion OWL ontologies describing the domain of the data in order to support pertinent symbolic reasoning.

2. Benefits and capabilities of OWL-based KGs

In this section, we describe benefits and capabilities of OWL-based KGs. We illustrate capabilities by giving examples showing how and why OWL-based KGs can be utilised in NeSy systems.

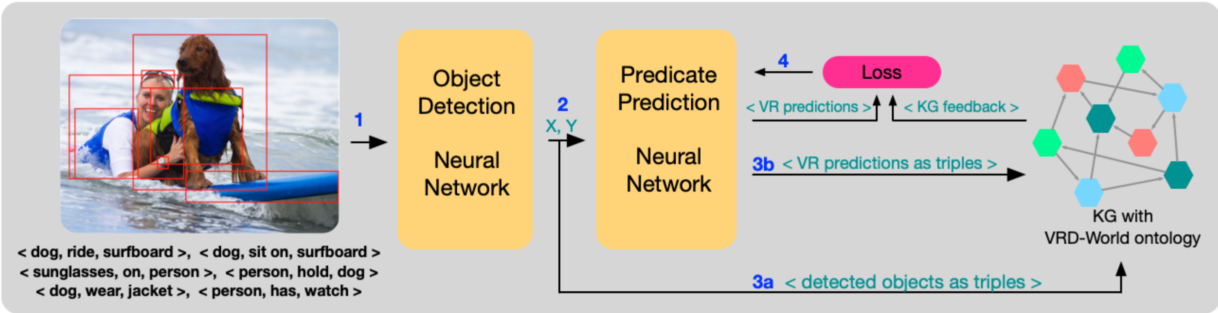

Fig. 1. An example neurosymbolic system architecture for detecting visual relationships in images. By using an OWL-based knowledge graph with an appropriate ontology as a symbolic deduction engine, feedback from OWL reasoning can influence loss to guide neural learning.

2.1. *Open standards and reusable resources*

The Web Ontology Language (OWL) [16, 17] and OWL-based knowledge graphs (KGs) [7, 8] are key components of the W3C open standards ecosystem of the Semantic Web (SW) [18, 19, 25, 26]. Open standards facilitate interoperability and promote development of reusable, often free, software resources that make it easy to work with OWL-based KGs. Amongst the many such resources are: *(i)* public SW KGs like DBpedia [27], Wikidata [28] and Yago [29]; *(ii)* public repositories of curated OWL ontologies like BioPortal [30] and OBO Foundry [31] in the biomedical domain; *(iii)* RDF stores like GraphDB (not open, but has free version) [32] and RDFox (not open, but has free academic license) [33]; and *(iv)* efficient OWL reasoners like HermiT [34], Pellet [35], RDFox and ELK [36].

2.2. *Custom ontologies and custom KGs*

Reusing state-of-the-art ontologies and/or public KGs is a good practice option. But researchers can also design their own custom, domain-specific OWL ontologies tailored to their unique needs and use them to govern and enable reasoning within custom OWL-based KGs. Custom ontologies can be aligned with publicly available ontologies to enhance interoperability [37].

This is the approach taken for visual relationship detection in images in the design of a custom OWL ontology, called VRD-World [38]. This ontology describes the domain of the everyday images of the VRD dataset [39], as reflected in the object classes and relationships referred to in the `(subject, predicate, object)` visual relationships annotated for the images. As depicted in Figure 1, the VRD-World ontology can govern a custom KG in the hybrid NeSy systems with which we explore using symbolic reasoning to guide neural learning. While designing the VRD-World ontology guidance was taken from the large literature on ontology engineering [*e.g.*, 40–43]. The ontology was specified using the free ontology editor Protégé [44], taking advantage of free Protégé plug-in utilities designed to support ontology development, such as ontology debuggers. Many machine learning tools exist to support various different aspects of ontology development such as, for example, concept learners (see [45]).

We designed two versions of a class hierarchy for our ontology. One version is entirely custom designed and represents the broad range of everyday object classes of the VRD dataset (`person`, `dog`, `jacket`, `surfboard`, *etc.*) feature exclusively as leaf nodes. In the other version, the VRD object classes were first aligned with matching classes in Wikidata [28], and a small number of subsumption paths were selected for each such that this class hierarchy represents a faithful, tractable subset of the Wikidata class hierarchy.

2.3. *KG embeddings, KG completion and knowledge injection*

KGs (of all kinds) have inspired a large amount of NeSy research into encoding KG symbolic background knowledge into vectors as *KG embeddings*. The embeddings preserve semantic similarity and reflect this similarity by proximity within the embedding vector space [45–50]. The primary application area of KG embeddings so far has been tasks relating to *KG completion*: *link prediction* (relating individuals in a KG) or *type prediction* (classifying individuals in a KG). Regardless of the model used to generate the embeddings (of which there are many), these link and type prediction problems are typically cast as neural classification problems, where the embedded KG knowledge is used for training and methods exploiting the proximity principle are applied.

Like all KGs, OWL-based KGs are readily used in NeSy research that leverages KG embeddings. OWL2Vec* [51] is one embedding model designed for this purpose. Notice, though, that these applications of KG embeddings focus on leveraging KG symbolic background knowledge only. So even if the KG in question is OWL-based, its reasoning capabilities are generally not employed in these applications.

Link inference and *type inference* by logical reasoning are, however, the bread and butter of OWL reasoners. When an OWL reasoner infers the knowledge that is entailed by the inference semantics of a governing OWL ontology in the presence of KG data, it completes the KG by introducing new, explicit (knowledge) triples that were previously implicit (materialisation). The logical soundness of these inferences is guaranteed, whereas embedding-based KG completion is approximate and potentially error prone. The extent to which the KG is extended (completed) is commensurate with the richness of the inference semantics of the governing ontology and the nature of the KG data present at the time of materialisation. Our point is that OWL-based KGs can add important value in any NeSy task associated with KG completion. OWL reasoning can be used to complete a KG automatically, as far as possible, and then NeSy KG completion (NN emulated reasoning) can be used for special cases that the OWL ontology in question does not address or that OWL cannot address in general.

As discussed in [52], the phrase 'knowledge injection' is used with different meanings. It can be used to refer to the injection of knowledge in symbolic form, such as in Logic Tensor Networks (LTN) [53, 54], with its *Real Logic* axioms that are woven into loss functions. It can also refer to the injection of knowledge represented in subsymbolic form, *e.g*., as embeddings. One subcategory of subsymbolic knowledge injection is the use of KG embeddings (*e.g*., [55]) as domain knowledge supplements to primary training data. The hypothesis here is that 'data + knowledge' can enhance deep learning. This is what the authors of [56] mean when they speak of knowledge injection. In [57] this area is called knowledge-infused learning. Much of the research in this area focuses on language models. Using KG entity linking techniques, concepts and entities mentioned in text are matched with corresponding KG entities. The (pre-computed) embeddings of matched KG entities are then looked-up and injected into language models (typically during fine-tuning), often deep within transformer self-attention blocks. For example, [58–61] explore variations on this theme and all report performance improvements from injecting KG embeddings as background knowledge.

The efficacy of this approach to KG knowledge injection has come into question, however. Having examined several established language model knowledge injection frameworks and repeated the published experiments, the authors of [56] conclude that whatever is injected has an effect indistinguishable from that of Gaussian noise. They surmise that fine-tuning does not permit pre-trained language models sufficient opportunity to disentangle and assimilate the latent knowledge in the injected embeddings.

We suspect that the prevailing paradigm of matching text to KG entities, and injecting the corresponding embeddings, may limit the use of KG knowledge. KGs express knowledge relationally and KG embedding spaces attempt to represent that relational knowledge subsymbolically. It is implicit in the relations between pairs and clusters of vectors in the embedding space, and in how clusters are distributed across the space. There may be little usable knowledge residing in individual embeddings considered in isolation (*i.e*., single points in space). Discordance between heterogeneous text and KG embedding spaces may also be a factor contributing to the findings in [56]. We suggest that new paradigms crafted to facilitate the harvesting and injection of relational knowledge warrant exploration.

Advances in subsymbolic KG knowledge injection will in part be driven by advances in KG embedding models. As discussed in papers such as [23, 62, 63], KG embedding models currently capture only a portion of the rich semantics of OWL ontologies. Some embedding models focus on capturing lexical and syntactic patterns, such as entity/word correlations; others focus on capturing aspects of logical relationships, such as hierarchical structure; integrating numeric literals remains a challenge. Research into methods that can embed more of OWL's logical expressiveness and logical relationships, and that can embed the multiple aspects mentioned here jointly, remains preliminary (*e.g*., [64]).

With any application of KG embeddings—KG completion, knowledge injection deep into NNs, or otherwise—the reasoning capabilities of OWL-based KGs may potentially add important value. It seems intuitive that fully materialised OWL-based KGs, where everything implicit has been made explicit, contain more embeddable knowledge and lead to embedding spaces that better reflect the totalities of symbolic domain knowledge entailed by KGs. However, more embeddable knowledge (more triples) does not necessarily lead to better performance in downstream tasks that consume KG embeddings. Studies such as [65] show that downstream performance may even decline. Nevertheless, the authors of [23] share this intuition and call for extensive research to study the potential of materialised KGs in KG embeddings, so that guidance can evolve around whether, when, and why to use materialisation.

2.4. OWL-based KG reasoning, rules, and symbolic deduction engines

Despite recent successes, large language models are notorious for their lack of reliability in reasoning. In contrast, the reliability of OWL reasoning is guaranteed because it is grounded in formal Description Logics (DLs). DLs are decidable fragments of first-order logic with strong connections to set theory [66–69]. A prominent example is *SROIQ* [70], the highly expressive DL that underlies that latest version of OWL, OWL 2.

OWL reasoning can be described in terms of two broad categories. One category involves inferring new knowledge (*i.e.*, to introduce new, explicit triples into a KG). The other category involves checking the logical consistency of ontologies and KGs. This includes enforcing consistency by refusing insertion of triples into a KG that would lead to inconsistency. Both categories of reasoning are commonly used for debugging during OWL ontology development [8, 71]. What appears to be less well recognised is that both categories of OWL reasoning can also be leveraged in NeSy systems. Indeed, as depicted in Figure 1, OWL-based KG technologies can be used to assemble symbolic deduction engines designed to perform system-specific OWL reasoning on-call and on-the-fly. Figure 1 shows feedback from OWL reasoning being used to guide neural learning by influencing the calculation of loss.

Depending on the application, the KGs of such symbolic deduction engines may need to contain only minimal data at any one moment. As long as a governing OWL ontology is present, fast OWL reasoning can proceed against small numbers of inserted data triples, and these might be deleted just as rapidly once the momentary reasoning service has been performed and the feedback delivered. OWL reasoning in such symbolic deduction engines can be leveraged not just during NN training but during inference as well. For instance, predictions of visual relationships generated at inference time can be inserted into a KG in order to verify their semantic validity. Ones found to be semantically invalid (*i.e.*, that lead to logical inconsistency) can be filtered from the set of predictions.

Before we discuss OWL reasoning in the literature, we introduce some basic examples. Link inference is driven mainly by the inference semantics associated with the characteristics (*e.g.*, symmetry, transitivity) and relationships (*e.g.*, inverses, subproperties, equivalent properties) declared for the object properties of an OWL ontology. Suppose a property `beside` is declared to be symmetric and a property `over` is declared to have as inverse the property `under`. In this case, given triples `(a, beside, b)` and `(c, over, d)`, OWL reasoning infers `(b, beside, a)` and `(d, under, c)`. Type inference (subsumption reasoning) is driven mainly by the class hierarchy of an ontology with the transitive property `rdfs:subClassOf`. If the ontology declares `(E, rdfs:subClassOf, F)` and data asserts `(e, rdf:type, E)`, OWL reasoning infers `(e, rdf:type, F)`.

The authors of [72] leverage OWL type inference in a more elaborate way so that a tutoring system can react intelligently in response to interactions with human learners. A custom OWL ontology models the tutoring system domain and contains descriptions of classes that correspond to tutoring system actions. Data regarding learner interactions with the system are progressively loaded into the system's KG. The OWL reasoner HermiT reasons over the KG to infer new knowledge based on the learner interaction data. In the process, each learner interaction is classified as belonging to one of the tutoring action classes, resulting in inferred triples such as, say, `(learnerX-interactionN, rdf:type, GiveEncouragement)`. The inferred classification triples are interpreted as predictions of the best next action for the tutoring system to take. The mechanism works because OWL allows classes to be described in terms of the characteristics possessed by their instances. This class description capability is a defining feature of Description Logics.

OWL reasoning can be extended by accompanying OWL ontologies with complementary rules. Such rules are constructed with reference to the classes and properties defined in the ontology, and they typically infer new knowledge (new triples). Various rule technologies for extending OWL exist, such as SPARQL rules [73], SWRL (Semantic Web Rule Language) rules [74], and (more commonly, today) Datalog rules [75–77].

One example in the literature, [78], uses a custom OWL ontology that describes dietary and physical activity domains and healthy lifestyle behaviours, supplemented with SPARQL rules representing unhealthy lifestyle behaviours, in a digital healthcare NeSy system. User diet and activity data are loaded into the system's KG. The RDFpro tool [79] drives the reasoning and rule execution. If a SPARQL rule is satisfied, an unhealthy behaviour has been detected, and the rule infers (creates) instances of rule violations in the KG. The rule violations are then rendered into natural language to encourage healthier user behaviours. The authors of [80] use a custom OWL ontology supplemented with SWRL rules as part of a system for monitoring vineyards. Data gathered by a wireless sensor network measuring micro climate conditions around a vineyard are funnelled into the system's KG. The Pellet OWL reasoner reasons over the KG to infer new triples that are interpreted as predictions of risk of particular diseases and pests. The authors of [81] use a custom OWL ontology supplemented with SWRL rules as part of a NeSy system for recognising surgical processes for robot-assisted surgery. A CNN recognises the current surgical workflow step; an LSTM RNN predicts the next surgical workflow step; and by reasoning over these inputs in the presence of the ontology and SWRL rules, the Pellet OWL reasoner infers supplementary surgical context information, such as the surgical phase, surgical instruments used, and actions to be taken.

The basis of the logical knowledge representation formalism for combining OWL ontologies with Datalog rules is defined in [82]. This formalism permits rules that refer to ontology vocabulary to be layered on top of ontologies, and it permits logic programming algorithms to reason efficiently over large ontologies. In the VRD dataset setting, a Datalog-like rule describing when it is reasonable to infer the visual relationship `(x, wear, y)` might be represented as

```
wear(x,y) := WearCapableThing(x), WearableThing(y), ir(y,x) > 0.8
```

In the body of this rule, the first two conditions rely on leveraging OWL type inference; the third condition extends OWL's capabilities by evaluating the spatial relationship between the bounding boxes of two objects. Suppose an object detection NN has predicted `(x, rdf:type, Dog)` and `(y, rdf:type, Jacket)`. OWL reasoning will infer the membership of these individuals in all of the parent classes of `Dog` and `Jacket` and, in so doing, determine whether or not `(x, rdf:type, WearCapableThing)` and `(y, rdf:type, WearableThing)`. If, during reasoning, both of these conditions turn out to be satisfied, plus the bounding box for `y` is mostly enclosed within the bounding box for `x` (as measured by function `ir()`), then the body of the Datalog rule is satisfied and the head of the rule is inferred: `(x, wear, y)`. We intend to explore the effect of such rules using tools such as RDFox, which translates OWL axioms into Datalog rules and performs all of its OWL reasoning using Datalog reasoning algorithms. This allows for seamless blending of reasoning over the OWL 2 RL profile with reasoning over supplementary Datalog rules that extend OWL reasoning.

3. Promising research directions

Here we describe some application areas where the potential for leveraging OWL-based KGs and OWL reasoning in NeSy systems looks promising.

3.1. Using OWL reasoning to enhance annotations and strengthen weak labelling

OWL reasoning has been used to enhance annotations. The authors of [83] use a small ontology crafted using a fuzzy DL, and a reasoning engine that uses a tableaux algorithm for doing *SHOIQ* DL reasoning (which is virtually OWL reasoning), to semantically enhance the annotations of images. Their knowledge-based framework detects (infers) what they call implicit, 'semantic context' concepts. For example, the co-occurrence of 'sea' and 'tree' in an image might lead to the inference of the context concept 'beach', which enhances the semantic interpretation of the image. The authors of [84] use a custom OWL ontology (based on a subset of WordNet), supplemented with SWRL rules (learned from data), to enhance video annotations. The SWRL rules infer instances of new, higher-level, composite concepts relevant to videos. For example, a video containing furniture, lamps, computers, etc., might be inferred to be ‘indoors’.

The VRD dataset we use in our research has weak labelling in the sense that it is not exhaustively annotated, either in terms of objects or relationships, and the visual relationship annotations of its images are sparse and arbitrary. The supervision they provide during machine learning is partial and inconsistent across images. OWL reasoning can mitigate weak labelling by augmenting it and making it more consistent. The properties of our VRD-World ontology (which correspond to predicates in annotated visual relationships) are rich in characteristics and relationships that carry inference semantics for link prediction (see Section 2.4). Our experiments with OWL reasoning over VRD-World have shown that the average number of annotated visual relationships per VRD training image increases by a factor of 2.5. Supplementing the ontology with Datalog rules that extend OWL reasoning is expected to yield further augmentation.

The augmentation of the ground-truth annotations for each image could, in theory, be performed in real-time within a NN training loop, but the same reasoning and augmentation would be performed repeatedly, each time each image is re-encountered in successive training epochs. So, in our case, it is more efficient computationally to do the annotation augmentation once, upfront, by materialising a KG containing the annotations of all images and saving the augmented annotations to a file. But the option of using OWL reasoning to perform annotation augmentation on-the-fly, is worthy of note because settings may arise where this facility is advantageous. Either way, augmented, denser and more consistent annotations are likely to provide a less noisy loss signal for neural learning.

The examples just discussed share the notion of using OWL reasoning to infer plausible annotations in the absence of explicit annotations. This notion has broad application. Within supervised learning, it may apply to many datasets (like the VRD dataset) that are not exhaustively annotated. It is also relevant in semi-supervised learning (where some examples are labelled, others not), and potentially in unsupervised learning problems as well. Further, it may be valuable in *k*-shot learning scenarios, supervised or otherwise. For example, in the VRD dataset, zero-shot cases arise where particular visual relationship types, *e.g.*, `(person, ride, elephant)`, exist in both training and test images but where only the test instances have been annotated. If, during training, OWL reasoning infers that predictions of such relationships are plausible, then, despite the absence of ground-truth annotations, we can avoid penalising loss and thereby increase the likelihood that the trained NN will predict such relationships when it encounters them in test images.

3.2. *Enabling neural networks to emulate OWL reasoning*

One approach to NeSy AI involves equipping NNs with background knowledge and learning biases by introducing structural extensions to their architectures that represent strong priors. An example of this approach is [85]. As part of our research, we have explored this approach to NeSy by considering the feasibility of transferring OWL-based KG reasoning capabilities to NNs in the form of structural extensions representing strong priors. In the process of doing so, we have developed a technique (as a proof-of-concept) for representing the class hierarchy of an OWL ontology as a structural extension to the architecture of an object detection classification NN. This structural extension can equip an object detection NN with the ability to precisely emulate the subsumption reasoning (type inference) capabilities of an OWL-based KG such that it can return not only the predicted class of a detected object (*e.g.*, Dog) but all of its parent classes as well (*e.g.*, Carnivore, Mammal, Animal, LivingThing, etc., as per the class hierarchy of the OWL ontology in question).

Since both NNs and OWL-based KGs (including class hierarchies of OWL ontologies) can be viewed as being directed graphs, we looked to graph theory for inspiration as to how to effect a transfer of subsumption reasoning capability from OWL-based KGs to NNs. One basic result from graph theory is that a graph with n nodes can have its structure encoded in an *adjacency matrix*, where a 1 indicates that two nodes are connected by an edge. An adjacency matrix can readily encode the structure of the directed graph of an OWL class hierarchy, where OWL classes are the nodes and the `rdfs:subClassOf` links between OWL classes are the (directed) edges. It turns out that the adjacency matrix of the graph of the transitive closure of an OWL class hierarchy fully encodes all of the subsumption reasoning capability of an OWL-based KG.

By materialising the OWL ontology containing the class hierarchy of interest, OWL reasoning infers the entire transitive closure of the class hierarchy and, in the process, makes all implicit `rdfs:subClassOf` axioms explicit. The explicit `rdfs:subClassOf` axioms can then be extracted (*e.g.*, via SPARQL query) to build the adjacency matrix for the graph of the transitive closure of the OWL class hierarchy. This adjacency matrix can then be used as the (fixed) weight matrix for one extra linear layer (with no activation function) of an object detection NN, enabling normal forward-pass computation to generate (pseudo-infer) the parent classes of the classes of all detected objects.

The idea of transferring OWL-based KG reasoning capabilities to NNs by leveraging results from graph theory, as illustrated by the example just discussed, is ripe for further exploration. One candidate avenue is to explore adding more learnable layers to an object detection NN following the class generalisation layer (as just described) so that learning can proceed driven by generalised class predictions. Another avenue lies in recognising that the solution for transferring subsumption reasoning capabilities to NNs works because the `rdfs:subClassOf` property is *transitive*. However, not only subclass relationships can be transitive in OWL ontologies, so the technique described may be applied to transitive properties in general. Similarly, adjacency matrices for graphs (or subgraphs) of OWL-based KGs can, in theory, be constructed in relation to any given object property of an OWL ontology. Suppose an object property called `beside` that is declared to be *symmetric*, such that if `(:A :beside :B)`, an OWL reasoner can infer that `(:B :beside :A)`. An adjacency matrix encoding these relations will itself be symmetric, and may potentially be capable of being leveraged in the subsymbolic space of a NN to encode the inference semantics associated with a symmetric object property of an OWL ontology, so as to enable the NN to replicate the symmetry-related reasoning of an OWL-based KG. Finally, suppose an OWL object property called `over` that is declared in an OWL ontology to have as inverse the property called `under`, such that if `(:A :over :B)`, an OWL reasoner can infer that `(:B :under :A)`. The adjacency matrices for OWL-based KG data relating to these two properties are the transpose of one another, so only one of them is needed to enable the inference semantics associated with object properties that are inverses of one another to be encoded and replicated in a subsymbolic space.

3.3. Using OWL reasoning for applying logical constraints

Much NeSy research explores using background knowledge expressed in first-order logic, propositional logic, or logic programming as constraints to guide neural learning, often by manipulating loss to encourage constraint satisfaction. Examples are *(i)* the NN training framework LTN which allows fuzzy, first-order *Real Logic* knowledge axioms (constraints) to be defined over training data; *(ii)* the set of propositional logic constraints specified for the ROAD-R dataset [86]; and *(iii)* the (Prolog) rules defined in [87]. OWL-based KG technologies can be used for the same purpose. OWL reasoners can infer new knowledge as well as check and enforce a KG's logical consistency. Both of these capabilities can be leveraged to enable OWL-based KGs to participate in research associated with the *logical constraints* approach to NeSy AI.

First consider the capability for enforcing KG logical consistency. KG technologies that do on-the-fly reasoning can enforce logical consistency at the point of triple insertion: insertions that would make the KG inconsistent are rejected. Such rejections are signals that one or more ontological constraints have been violated. This information can be used to penalise loss. The Protégé editor has facilities and plug-ins that provide explanations for ontology inconsistencies. The OWL reasoner Pellet can also provide explanations to an extent.

For a concrete example, we discuss the use of domain and range restrictions as logical constraints in connection with the VRD dataset. We describe how these have been used in the context of LTN and then compare that approach with how they can be used in OWL ontologies. In [88], negative domain and negative range LTN *Real Logic* axioms (constraints) are used to train binary classifiers for the predicates of the VRD dataset. The VRD dataset has 100 object classes, so to train a binary classifier for predicate `wear`, close to 100 negative domain LTN axioms would have been required, such as

$$\forall xy\ wear(x,y) \rightarrow \neg Laptop(x) \quad \forall xy\ wear(x,y) \rightarrow \neg Sofa(x) \quad \forall xy\ wear(x,y) \rightarrow \neg Tree(x) \quad \ldots,$$

to express the knowledge that the data classes on the right of these implications cannot wear things. Similarly, close to 100 negative range LTN axioms would have been required, such as

$$\forall xy\ wear(x,y) \rightarrow \neg Table(y) \quad \forall xy\ wear(x,y) \rightarrow \neg Car(y) \quad \forall xy\ wear(x,y) \rightarrow \neg Oven(y) \quad \ldots,$$

to express the knowledge that the data classes on the right of these implications are not wearable things.

OWL can express equivalent logical constraints, and can do so more concisely. The VRD-World ontology can express the equivalent of the close to 100 negative domain constraints by defining the (disjoint) classes `WearCapableThing` and `WearIncapableThing` in its class hierarchy, and by declaring that the domain of object property `wear` is restricted to members of the class `WearCapableThing`, with the OWL axiom

```
vrd:wear rdfs:domain vrd:WearCapableThing .
```

Similarly, the logical constraint equivalent of the close to 100 negative range LTN *Real Logic* axioms can be established by defining the (disjoint) classes `WearableThing` and `NonWearableThing` in the class hierarchy of VRD-World, and by declaring that the range of object property `wear` is restricted to members of the class `WearableThing`, with the OWL axiom

```
vrd:wear rdfs:range vrd:WearableThing .
```

Figure 1 shows how an OWL-based KG with an appropriate ontology (such as VRD-World) can be used, in the guise of a symbolic deduction engine, to leverage ontological rules as logical constraints to guide neural learning. Suppose the Object Detection neural network predicts that `objectX` is a dog and `objectY` is a surfboard. If the multi-class, multi-label Predicate Prediction neural network shows a tendency to predict a visual relationship such as `(dog, wear, surfboard)`, the RDF triples representing this prediction

```
vrd:objectX  rdf:type  vrd:Dog
vrd:objectY  rdf:type  vrd:Surfboard
vrd:objectX  vrd:wear  vrd:objectY
```

can be inserted into the KG for evaluation. Type inference (subsumption reasoning) will infer that `:objectX` is a `WearCapableThing` (*i.e.*, in VRD-World, dogs can wear things), and that `:objectY` (a surfboard) is a `NonWearableThing`, not a `WearableThing`. OWL reasoning will detect that the range restriction of property `wear` has been violated and signal a logical inconsistency. This feedback can be used to penalise loss to help the Predicate Prediction NN learn to avoid predicting visual relationships that are semantically invalid.

In addition to illustrating that OWL-based KGs can emulate the logical constraints approach to NeSy AI, this example also illustrates an important advantage possessed by OWL-based KGs over other approaches to using logical constraints. The research in [88] shows that the logical constraints approach to NeSy AI is exposed to the risk of combinatorial explosion, where the number of constraints requiring expression grows too rapidly with the number of classes in the dataset. Almost 200 LTN `Real Logic` axioms would have been needed in relation to just one VRD predicate, `wear`. And about 30 of the 70 VRD predicates admit domain and/or range restrictions of some kind. Indeed, [88] reports implementing a "tractable sample" only of the LTN *Real Logic* axioms implied by the negative domain/range constraints training strategy selected for the experiments. In contrast, once an appropriate class hierarchy is defined, expressing powerful domain and range restrictions in OWL is easy.

This comparative advantage possessed by OWL for expressing background knowledge (and logical constraints) concisely is reinforced by considering a different example. The autonomous vehicle driving videos and annotated bounding boxes of the ROAD-R dataset [86] are accompanied by 243 manually specified propositional logic constraints that define the permissible combinations of labels for 10 agent classes, 19 agent action classes, and 12 agent location classes. Amongst the 243 logic constraints, 45 have a format such as $\neg\text{Car} \vee \neg\text{Bus}$ (meaning "an agent cannot be a car and a bus at the same time"), or $\neg\text{RedTL} \vee \neg\text{GreenTL}$ (meaning "a traffic light cannot be red and green at the same time"). Collectively, these 45 constraints express mutual exclusiveness between pairwise combinations of the 10 agent classes. Precise counterparts of these 45 propositional constraints can be represented in OWL with just two axioms, such as:

```
:DisjointAgents rdf:type owl:AllDisjointClasses .
:DisjointAgents owl:members (:Car :Bus :Motorbike :Pedestrian ... ) .
```

Similarly, 66 of the ROAD-R propositional constraints express pairwise mutual exclusiveness amongst the 12 agent location classes. Counterparts of these can be represented in OWL using just two more such axioms.

In fact, by making appropriate use of OWL's constructs for declaring domain and range restrictions, disjoint classes, disjoint properties, functional properties, and the like, it may well be possible to design an OWL ontology that emulates all of the 243 propositional logic constraints specified for the ROAD-R dataset. Doing so would, in theory, make it feasible to repeat the ROAD-R experiments in [86] using and OWL-based KG as a symbolic deduction engine instead of using the original propositional constraints with a SAT solver as a reasoning engine.

Now we consider how OWL reasoning's capability for inferring new knowledge can be leveraged in the context of using logical constraints to guide neural learning. An alternate strategy for using OWL to emulate the propositional logical constraints of the ROAD-R dataset is to employ the concept of *integrity constraints* described in [89]. Rather than checking and enforcing ontology (KG) logical consistency, integrity constraints employ Datalog rules that supplement an OWL ontology to represent logical constraints. Such constraint rules, if satisfied, infer explicit new triples into the KG which can then be queried and interpreted as signals of constraint violations. For example, the Datalog integrity constraint rule

```
Violation(X) :- Car(X), Bus(X)
```

declares that an agent cannot be both a car and a bus. Similarly, the ontology could drive the creation of rules like

```
Violation(X) :- TL(X), hasL(X, Y), hasL(X, Z), GreenL(Y), RedL(Z)
```

to establish that a traffic light (TL) cannot be red and green at the same time. Note that, unlike in the propositional case, Datalog rules provide additional granularity for describing cases in which an image contains more than one traffic light. This approach using integrity constraints could also be applied to the VRD case, with rules like

```
Violation(X) :- vrd:wear(X, Y), not vrd:WearCapableThing(X)
Violation(Y) :- vrd:wear(X, Y), not vrd:WearableThing(Y)
```

3.4. Integrating OWL-based KG reasoning with existing NeSy frameworks

OWL-based KG symbolic knowledge and deductive reasoning can be integrated with and leveraged by existing logic-based NeSy frameworks such as LTN. LTN functions that encapsulate interactions with OWL-based KGs can, in theory, participate in LTN *Real Logic* knowledge axioms used to train NNs. One precondition is that there is sufficient contextual information contained in LTN tensors (or otherwise) to permit RDF triples to be constructed and inserted into a KG to drive reasoning, and/or to enable KG queries to be formulated. The only other precondition is that the results of KG interactions can be mapped to fuzzy truth values in $[0, 1]$.

One application of this idea involves using OWL-based KG reasoning to manage the risk of combinatorial explosion (described in Section 3.3) to which the *logical constraints* approach to NeSy AI is exposed. A prime cause of exposure to this risk derives from the fact that logical constraints (as used by LTN and the ROAD-R dataset, for example) are restricted to being expressed in terms of the low-level, granular object classes present in data and their annotations. The option to express constraints more concisely, in terms of higher-level, more general classes, is not available. In contrast, OWL ontologies routinely possess rich class hierarchies that permit ontological rules to be defined in terms of high-level, general classes, which affords simplicity and parsimony.

To illustrate, consider again the research undertaken in [88], where the use of negative domain and range constraints leads to the need for an intractable number of LTN *Real Logic* knowledge axioms to be crafted. This time, however, suppose that we integrate interactions with an OWL-based KG (and OWL reasoning) into our LTN *Real Logic* knowledge axioms in order to map the granular classes present in the data to higher-level, more general classes defined in the class hierarchy of the VRD-World ontology. Using this strategy, we can imagine replacing the original (close to) 200 *negative* domain and range LTN constraints used to train a binary classifier for VRD predicate `wear` with just two *positive* LTN *Real Logic* knowledge axiom constraints, such as:

$$\forall xy\ wear(x,y) \rightarrow WearCapableThing(x) \qquad \forall xy\ wear(x,y) \rightarrow WearableThing(y)$$

For clarity, note that, unlike in OWL reasoning and Datalog rule reasoning where new knowledge is inferred, LTN *Real Logic* knowledge axioms do not infer anything, despite the fact that they can be expressed as logical implications. Instead, the degree to which LTN *Real Logic* knowledge axioms are satisfied is measured, and the extent to which they are not satisfied represents loss that drives neural learning.

A more compute-efficient implementation of the proposal just described is also feasible. In this setting, we only wish to exploit the type inference (subsumption reasoning) capabilities of OWL reasoning. But, as we saw in Section 3.2, for a given OWL ontology, these capabilities can be fully encoded in the *adjacency matrix* of the transitive closure of the ontology's class hierarchy. So instead of interacting with an OWL-based KG to use OWL reasoning to map granular classes to higher-level classes, we can instead use the *adjacency matrix* to do the mapping.

Integrating OWL-based KG reasoning with LTN in the manner just described is a specialised approach to blending OWL (a Description Logic) with (fuzzy) first-order logic (FOL). Another approach to blending OWL with FOL is to translate it into FOL. This leads to opportunities to extend OWL reasoning capabilities by augmenting OWL ontologies with supplementary FOL axioms (or ontologies) that express things OWL cannot. Tools that support this approach include *Hets* (The Heterogeneous Tool Set) [90] and *Gavel-OWL* [91]. The resulting integrated FOL ontologies (OWL-to-FOL axioms, plus supplementary FOL axioms) that such tools produce are reasoned over using established FOL Automated Theorem Provers (ATPs).

The 'translate OWL to FOL' strategy just described and the 'translate OWL to Datalog' strategy mentioned in Section 2.4 are instances of the same pattern: *(i)* translate OWL into logic space X; *(ii)* optionally extend OWL with supplementary knowledge expressible in logic space X; *(iii)* reason using the logical inference technology established for logic space X. Such strategies widen the window of opportunity for leveraging the available OWL resources in NeSy frameworks.

4. Enabling NeSy Research using OWL-based KGs with NeSy4VRD

Sections 2 and 3 focus on inspiring more NeSy research using OWL-based KGs by highlighting their benefits, capabilities, and applications, especially with respect to reasoning, and particularly in symbolic deduction engine settings. But inspiration alone may not be enough. To undertake NeSy research with OWL-based KGs and reasoning in a practical way, researchers need to also be enabled with appropriate dataset resources. Resources are needed that combine data for neural learning with strongly-aligned, companion OWL ontologies describing the domains of the data in order to support directly pertinent symbolic OWL reasoning. Such resources are scarce. We suspect this scarcity represents a silent barrier that inhibits NeSy research using OWL-based KGs that might otherwise be undertaken. As well as echoing our observations, [23] calls for a central repository for such specialised resources in order to simplify their discovery. A resource of this kind (one which belongs in such a repository), *NeSy4VRD* (Neurosymbolic AI for Visual Relationship Detection), was co-developed and published by the authors of this paper [38].

NeSy4VRD consists of the following components and services:

1. the images of the original VRD dataset [39] (distributed with permission from one of the principals associated with its creation) in order to make them publicly available once again;
2. quality-improved versions of the original VRD visual relationship annotations that have been comprehensively customised and extended to enable the engineering of a robust ontology;
3. a strongly-aligned, custom-designed companion OWL ontology, called VRD-World, that precisely describes the domain of the images and visual relationships;
4. sample Python code for loading the annotated visual relationships into a knowledge graph hosting the VRD-World ontology, and for extracting them from a knowledge graph and restoring them to their native format;
5. support for extensibility of the annotations (and, thereby, the ontology) in the form of *(a)* comprehensive Python code enabling deep but easy analysis of the images and their annotations, *(b)* a custom, text-based protocol for specifying annotation customisation instructions declaratively, and *(c)* a configurable, managed Python workflow for customising annotations in an automated, repeatable process;
6. comprehensive documentation describing *(a)* how to use the extensibility support infrastructure, *(b)* how to share annotation/ontology extensibility projects undertaken by researchers in pursuit of their private research interests, *(c)* how to reuse shared extensibility projects and use the NeSy4VRD workflow to compose them in novel combinations, and *(d)* how the ability to undertake, share, reuse and compose NeSy4VRD extensibility projects represents a new model of collaborative data annotation that we call Distributed Annotation Enhancement.

The NeSy4VRD dataset package (VRD images, quality-improved visual relationship annotations, and companion VRD-World OWL ontology) is distributed on Zenodo[1]. The NeSy4VRD extensibility support infrastructure and comprehensive documentation are available on GitHub[2].

5. Conclusion

A central concern of NeSy AI research is to explore ways of combining neural learning with symbolic background knowledge and reasoning. OWL-based KGs are exemplars of symbolic knowledge representation and reasoning technology and machinery. They can do everything that general KGs can do in terms of representing symbolic knowledge and generating embeddings, plus they can perform sound deductive reasoning to both infer new knowledge and enforce logical consistency, and they can do so in the guise of symbolic deduction engines. Given these attractive features, OWL-based KGs warrant more research attention from the NeSy community than they have received. Their potential for contributing to NeSy AI is not being fully explored. By describing and illustrating their benefits, capabilities, and flexible applications, we have endeavoured to inspire more such research. By having contributed NeSy4VRD — a specialised and scarce dataset resource — to the NeSy community, we hope to have lowered barriers to entry and thereby enabled more such research. A recent overview of NeSy systems [92] reports success using an OWL-based KG to boost expert user satisfaction with large language model performance. Like us, the authors strongly advocate the use of KGs (general and OWL-based) as symbolic components in NeSy systems.

Acknowledgements

This work has been partially supported by the Academy of Medical Sciences Network Grant (Neurosymbolic AI for Medicine, NGR1\1857) and the project "XAI4SOC: Explainable Artificial Intelligence for Healthy Aging and Social Wellbeing" funded by the Agencia Estatal de Investigación (AEI), the Spanish Ministry of Science, Innovation and Universities and the European Social Funds (PID2021-123152OB-C22).

References

[1] M.K. Sarker, L. Zhou, A. Eberhart and P. Hitzler, Neuro-Symbolic Artificial Intelligence: Current Trends, *CoRR* (2021). https://arxiv.org/abs/2105.05330.
[2] F. Chollet, Deep Learning: Current Limits and What Lies Beyond Them, 2018. https://raais.co/speakers-2018.
[3] G. Marcus, Deep Learning: A Critical Appraisal, *CoRR* (2018). http://arxiv.org/abs/1801.00631.
[4] G. Marcus, Innateness, AlphaZero, and Artificial Intelligence, *CoRR* (2018). http://arxiv.org/abs/1801.05667.
[5] G. Marcus, The Next Decade in AI: Four Steps Towards Robust Artificial Intelligence, *CoRR* (2020). https://arxiv.org/abs/2002.06177.
[6] H. Kautz, The third ai summer: Aaai robert s. engelmore memorial lecture, Vol. 43, 2022, pp. 105–125.
[7] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. de Melo, C. Gutiérrez, S. Kirrane, J.E. Labra Gayo, R. Navigli, S. Neumaier, A.-C. Ngonga Ngomo, A. Polleres, S.M. Rashid, A. Rula, L. Schmelzeisen, J.F. Sequeda, S. Staab and A. Zimmermann, *Knowledge Graphs*, Synthesis Lectures on Data, Semantics, and Knowledge, Springer, 2021. https://kgbook.org/.
[8] M. Kejriwal, C.A. Knoblock and P. Szekely, *Knowledge Graphs: Fundamentals, Techniques, and Applications*, The MIT Press, 2021.
[9] P. Hitzler, Knowledge Graphs in Neuro-Symbolic AI, 2021, (Presentation slides). https://people.cs.ksu.edu/~hitzler/pub2/2021-10-AKBC.pdf.
[10] F. Bianchi and P. Hitzler, On the Capabilities of Logic Tensor Networks for Deductive Reasoning, in: *Proceedings of the AAAI-MAKE*, Vol. 2350, 2019.
[11] M. Ebrahimi, A. Eberhart, F. Bianchi and P. Hitzler, Towards Bridging the Neuro-Symbolic Gap: Deep Deductive Reasoners, *Appl. Intell.* **51**(9) (2021), 6326–6348.
[12] M. Ebrahimi, M.K. Sarker, F. Bianchi et al., Neuro-Symbolic Deductive Reasoning for Cross-Knowledge Graph Entailment, in: *Proceedings of the AAAI-MAKE*, Vol. 2846, 2021. http://ceur-ws.org/Vol-2846/paper8.pdf.
[13] Z. Chen, J. Chen et al., Zero-Shot Visual Question Answering using Knowledge Graph, *CoRR* (2021). https://arxiv.org/abs/2107.05348.

[1] https://doi.org/10.5281/zenodo.7916355
[2] https://github.com/djherron/NeSy4VRD

[14] Y. Geng, J. Chen, Z. Ye, Z. Yuan, W. Zhang and H. Chen, Explainable Zero-shot Learning via Attentive Graph Convolutional Network and Knowledge Graphs, *Semantic Web* **12**(5) (2021), 741–765. https://doi.org/10.3233/SW-210435.

[15] J. Chen, Y. Geng et al., Low-Resource Learning with Knowledge Graphs: A Comprehensive Survey, *CoRR* (2021). https://arxiv.org/abs/2112.10006.

[16] P. Hitzler, M. Krötzsch, B. Parsia, P.F. Patel-Schneider and S. Rudolph, OWL 2 Web Ontology Language Primer, W3C Recommendation, World Wide Web Consortium, 2012. http://www.w3.org/TR/owl2-primer/.

[17] B. Motik, P.F. Patel-Schneider, B. Parsia, C. Bock, A. Fokoue, P. Haase, R. Hoekstra, I. Horrocks, A. Ruttenberg, U. Sattler and M. Smith, OWL 2 Web Ontology Language: Structural Specification and Functional-Style Syntax, W3C Recommendation, World Wide Web Consortium, 2012. http://www.w3.org/2007/OWL/draft/owl2-syntax/.

[18] The Semantic Web Wiki, 2001. https://www.w3.org/2001/sw/wiki/Main_Page.

[19] The Semantic Web Stack, 2022. https://en.wikipedia.org/wiki/Semantic_Web_Stack.

[20] T.H. Trinh, Y. Wu, Q.V. Le, H. He and T. Luong, Solving olympiad geometry without human demonstrations, *Nature* **625** (2024), 476–482. https://doi.org/10.1038/s41586-023-06747-5.

[21] A. Breit, L. Waltersdorfer, F.J. Ekaputra, M. Sabou, A. Ekelhart, A. Iana, H. Paulheim, J. Portisch, A. Revenko, A.t. Teije and F. van Harmelen, Combining Machine Learning and Semantic Web: A Systematic Mapping Study, *ACM Computing Surveys* (2023). https://doi.org/10.1145/3586163.

[22] F.J. Ekaputra, M. Llugiqi, M. Sabou, A. Ekelhart, H. Paulheim, A. Breit, A. Revenko, L. Waltersdorfer, K.E. Farfar and S. Auer, Describing and Organizing Semantic Web and Machine Learning Systems in the SWeMLS-KG, in: *Proceedings of the Extended Semantic Web Conference (ESWC)*, Vol. 13870, Springer, 2023, pp. 372–389. https://doi.org/10.1007/978-3-031-33455-9_22.

[23] C. d'Amato, L. Mahon, P. Monnin and G. Stamou, Machine Learning and Knowledge Graphs: Existing Gaps and Future Research Challenges, *Transactions on Graph Data and Knowledge (TGDK)* **1**(1) (2023), 8:1–8:35. https://doi.org/10.4230/TGDK.1.1.8.

[24] D. Herron, E. Jiménez-Ruiz and T. Weyde, On the Benefits of OWL-based Knowledge Graphs for Neural-Symbolic Systems, in: *NeSy 2023: 17th International Workshop on Neural-Symbolic Learning and Reasoning, July 3–5, La Certosa di Pontignano, Siena, Italy*, CEUR Workshop Proceedings, Vol. 3432, CEUR-WS.org, 2023. https://ceur-ws.org/Vol-3432/paper28.pdf.

[25] T. Berners-Lee, J. Hendler and O. Lassila, The Semantic Web, *Scientific American* **284**(5) (2001), 34–43. https://www.jstor.org/stable/pdf/26059207.pdf.

[26] P. Hitzler, M. Krötzsch and S. Rudolph, *Foundations of Semantic Web Technologies*, 3rd edn, CRC Press, 2010.

[27] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P.N. Mendes, S. Hellmann, M. Morsey, P. van Kleef, S. Auer and C. Bizer, DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia., *Semantic Web* **6**(2) (2015), 167–195. http://dx.doi.org/10.3233/SW-140134.

[28] D. Vrandečić and M. Krötzsch, Wikidata: A Free Collaborative Knowledgebase, *Communications of the ACM* **57**(10) (2014), 78–85.

[29] T.P. Tanon, G. Weikum and F.M. Suchanek, YAGO 4: A Reason-able Knowledge Base, in: *The Semantic Web - 17th International Conference, ESWC 2020, Proceedings*, Lecture Notes in Computer Science, Vol. 12123, Springer, 2020, pp. 583–596. https://doi.org/10.1007/978-3-030-49461-2_34.

[30] P.L. Whetzel, N.F. Noy, N.H. Shah, P.R. Alexander, C. Nyulas, T. Tudorache and M.A. Musen, BioPortal: Enhanced Functionality via new Web Services from the National Center for Biomedical Ontology to Access and use Ontologies in Software Applications, *Nucleic Acids Research* **39**(suppl 2) (2011), W541–W545, https://bioportal.bioontology.org/.

[31] R. Jackson, N. Matentzoglu, J.A. Overton, R. Vita, J.P. Balhoff, P.L. Buttigieg, S. Carbon, M. Courtot, A.D. Diehl, D.M. Dooley, W.D. Duncan, N.L. Harris, M.A. Haendel, S.E. Lewis, D.A. Natale, D. Osumi-Sutherland, A. Ruttenberg, L.M. Schriml, B. Smith, C.J. Stoeckert Jr., N.A. Vasilevsky, R.L. Walls, J. Zheng, C.J. Mungall and B. Peters, OBO Foundry in 2021: Operationalizing Open Data Principles to Evaluate Ontologies, *Database* **2021** (2021), http://obofoundry.org/. https://doi.org/10.1093/database/baab069.

[32] Ontotext GraphDB, 2023, A popular RDF store; https://www.ontotext.com/products/graphdb/. https://en.wikipedia.org/wiki/Ontotext_GraphDB.

[33] Y. Nenov, R. Piro, B. Motik, I. Horrocks, Z. Wu and J. Banerjee, RDFox: A Highly-Scalable RDF Store, in: *The Semantic Web - 14th International Conference, ISWC 2015, Proceedings*, Lecture Notes in Computer Science, Vol. 9367, Springer Verlag, 2015, pp. 3–20, https://www.oxfordsemantic.tech/product. https://ora.ox.ac.uk/objects/uuid:2a08b023-77be-431a-a08c-89b47381586a.

[34] B. Glimm, I. Horrocks, B. Motik, G. Stoilos and Z. Wang, HermiT: An OWL 2 Reasoner, *J. Autom. Reason.* **53**(3) (2014), 245–269. https://doi.org/10.1007/s10817-014-9305-1.

[35] E. Sirin, B. Parsia, B.C. Grau, A. Kalyanpur and Y. Katz, Pellet: A Practical OWL-DL Reasoner, *J. Web Semant.* **5**(2) (2007), 51–53. https://doi.org/10.1016/j.websem.2007.03.004.

[36] Y. Kazakov, M. Krötzsch and F. Simančík, Unchain My *EL* Reasoner, in: *Proceedings of the 24th International Workshop on Description Logics (DL'11)*, CEUR Workshop Proceedings, Vol. 745, CEUR-WS.org, 2011.

[37] J. Euzenat and P. Shvaiko, *Ontology Matching, Second Edition*, Springer, 2013. ISBN 978-3-642-38720-3.

[38] D. Herron, E. Jiménez-Ruiz, G. Tarroni and T. Weyde, NeSy4VRD: A Multifaceted Resource for Neurosymbolic AI Research using Knowledge Graphs in Visual Relationship Detection, *CoRR* (2023). http://arxiv.org/abs/2305.13258.

[39] C. Lu, R. Krishna, M. Bernstein and L. Fei-Fei, Visual Relationship Detection with Language Priors, in: *European Conference on Computer Vision*, Springer, 2016, pp. 852–869. https://cs.stanford.edu/people/ranjaykrishna/vrd/.

[40] N. Noy and D. McGuinness, Ontology Development 101: A Guide to Creating Your First Ontology, Technical Report, Knowledge Systems Laboratory, Stanford University, 2001. http://www.ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness.pdf.

[41] E.F. Kendall and D.L. McGuinness, *Ontology Engineering*, Synthesis Lectures on The Semantic Web: Theory and Technology, Morgan & Claypool Publishers, 2019.

[42] D. Allemang, J. Hendler and F. Gandon, *Semantic Web for the Working Ontologist*, 3rd edn, ACM Books, 2020.
[43] C.M. Keet, *An Introduction to Ontology Engineering*, 'v1.5' edn, 2020. https://people.cs.uct.ac.za/~mkeet/OEbook/.
[44] M.A. Musen, The Protégé Project: A Look Back and a Look Forward, *AI Matters* **1**(4) (2015), 4–12. doi:10.1145/2757001.2757003. https://protege.stanford.edu/.
[45] C. d'Amato, Machine Learning for the Semantic Web: Lessons learnt and next research directions, *Semantic Web* **11**(1) (2020), 195–203. https://doi.org/10.3233/SW-200388.
[46] M. Nickel, K. Murphy, V. Tresp and E. Gabrilovich, A Review of Relational Machine Learning for Knowledge Graphs, *CoRR* (2015). http://arxiv.org/abs/1503.00759.
[47] Y. Dai, S. Wang, N.N. Xiong and W. Guo, A Survey on Knowledge Graph Embedding: Approaches, Applications and Benchmarks, *Electronics* **9**(5) (2020). doi:10.3390/electronics9050750. https://www.mdpi.com/2079-9292/9/5/750.
[48] M. Wang, L. Qiu and X. Wang, A Survey on Knowledge Graph Embeddings for Link Prediction, *Symmetry* **13**(3) (2021). https://doi.org/10.3390/sym13030485.
[49] Z. Chen, Y. Wang, B. Zhao, J. Cheng, X. Zhao and Z. Duan, Knowledge Graph Completion: A Review, *IEEE Access* **8** (2020), 192435–192456. https://doi.org/10.1109/ACCESS.2020.3030076.
[50] A. Rossi, D. Barbosa, D. Firmani, A. Matinata and P. Merialdo, Knowledge Graph Embedding for Link Prediction: A Comparative Analysis, *ACM Trans. Knowl. Discov. Data* **15**(2) (2021), 14:1–14:49. https://doi.org/10.1145/3424672.
[51] J. Chen, P. Hu, E. Jiménez-Ruiz, O.M. Holter, D. Antonyrajah and I. Horrocks, OWL2Vec*: Embedding of OWL Ontologies, *Mach. Learn.* **110**(7) (2021), 1813–1845. https://doi.org/10.1007/s10994-021-05997-6.
[52] D. Buffelli and E. Tsamoura, Scalable Theory-Driven Regularization of Scene Graph Generation Models, in: *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37, 2023, pp. 6850–6859. https://ojs.aaai.org/index.php/AAAI/article/view/25839.
[53] L. Serafini and A.S. d'Avila Garcez, Logic Tensor Networks: Deep Learning and Logical Reasoning from Data and Knowledge, in: *Proceedings of the 11th International Workshop on Neural-Symbolic Learning and Reasoning (NeSy'16)*, CEUR Workshop Proceedings, Vol. 1768, CEUR-WS.org, 2016. http://ceur-ws.org/Vol-1768/NESY16_paper3.pdf.
[54] S. Badreddine, A. d'Avila Garcez, L. Serafini and M. Spranger, Logic Tensor Networks, *Artificial Intelligence* **303** (2022). https://doi.org/10.1016/j.artint.2021.103649.
[55] E.B. Myklebust, E. Jiménez-Ruiz, J. Chen, R. Wolf and K.E. Tollefsen, Prediction of adverse biological effects of chemicals using knowledge graph embeddings, *Semantic Web* **13**(3) (2022), 299–338. doi:10.3233/SW-222804.
[56] P. Fu, Y. Zhang, H. Wang, W. Qiu and J. Zhao, Revisiting the Knowledge Injection Frameworks, in: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, 2023, pp. 10983–10997. https://aclanthology.org/2023.emnlp-main.677.
[57] A.P. Sheth, M. Gaur, U. Kursuncu and R. Wickramarachchi, Shades of Knowledge-Infused Learning for Enhancing Deep Learning, *IEEE Internet Comput.* **23**(6) (2019), 54–63. https://doi.org/10.1109/MIC.2019.2960071.
[58] I. Yamada, A. Asai, H. Shindo, H. Takeda and Y. Matsumoto, LUKE: Deep Contextualized Entity Representations with Entity-aware Self-attention, in: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, 2020, pp. 6442–6454. https://doi.org/10.18653/v1/2020.emnlp-main.523.
[59] Z. Zhang, X. Han, Z. Liu, X. Jiang, M. Sun and Q. Liu, ERNIE: Enhanced Language Representation with Informative Entities, in: *Proceedings of the Conference of the Association for Computational Linguistics*, Association for Computational Linguistics, 2019, pp. 1441–1451. https://doi.org/10.18653/v1/p19-1139.
[60] M.E. Peters, M. Neumann, R.L.L. IV, R. Schwartz, V. Joshi, S. Singh and N.A. Smith, Knowledge Enhanced Contextual Word Representations, in: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, 2019, pp. 43–54. https://doi.org/10.18653/v1/D19-1005.
[61] K. Roy, Y. Zi, V. Narayanan, M. Gaur and A.P. Sheth, Knowledge-Infused Self Attention Transformers, *CoRR* (2023). https://doi.org/10.48550/arXiv.2306.13501.
[62] J. Chen, Y. He, Y. Geng, E. Jiménez-Ruiz, H. Dong and I. Horrocks, Contextual Semantic Embeddings for Ontology Subsumption Prediction, *World Wide Web (WWW)* **26**(5) (2023), 2569–2591. https://doi.org/10.1007/s11280-023-01169-9.
[63] R. Abboud, I. Ceylan, T. Lukasiewicz and T. Salvatori, BoxE: A Box Embedding Model for Knowledge Base Completion, in: *Advances in Neural Information Processing Systems*, Vol. 33, Curran Associates, Inc., 2020, pp. 9649–9661. https://proceedings.neurips.cc/paper_files/paper/2020/file/6dbbe6abe5f14af882ff977fc3f35501-Paper.pdf.
[64] M. Jackermeier, J. Chen and I. Horrocks, Dual Box Embeddings for the Description Logic $\mathcal{EL}$++, in: *WWW*, 2024.
[65] A. Iana and H. Paulheim, More is not Always Better: The Negative Impact of A-box Materialization on RDF2vec Knowledge Graph Embeddings, in: *Proceedings of the CIKM Workshops*, CEUR Workshop Proceedings, Vol. 2699, CEUR-WS.org, 2020. https://ceur-ws.org/Vol-2699/paper05.pdf.
[66] R. Brachman and H. Levesque, *Knowledge Representation and Reasoning*, Morgan Kaufman, 2004.
[67] F. Baader, D. Calvanese, D. McGuinness, D. Nardi and P.F. Patel-Schneider (eds), *The Description Logic Handbook*, 2nd edn, Cambridge University Press, 2007.
[68] F. Baader, I. Horrocks, C. Lutz and U. Sattler, *An Introduction to Description Logic*, Cambridge University Press, 2017.
[69] D. Nardi and R.J. Brachman, An Introduction to Description Logics, in: *The Description Logic Handbook: Theory, Implementation, and Applications*, F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi and P.F. Patel-Schneider, eds, Cambridge Univ. Press, 2003, pp. 1–40.
[70] I. Horrocks, O. Kutz and U. Sattler, The Even More Irresistible SROIQ, in: *Proceedings, Tenth International Conference on Principles of Knowledge Representation and Reasoning*, P. Doherty, J. Mylopoulos and C.A. Welty, eds, AAAI Press, 2006, pp. 57–67. http://www.aaai.org/Library/KR/2006/kr06-009.php.

[71] E. Jiménez-Ruiz, Logic-based Support for Ontology Development in Open Environments, PhD thesis, Jaume I University, Spain, 2010. http://hdl.handle.net/10803/10493.
[72] M. Chang, G. D'Aniello, M. Gaeta, F. Orciuoli, D.G. Sampson and C. Simonelli, Building Ontology-Driven Tutoring Models for Intelligent Tutoring Systems Using Data Mining, *IEEE Access* **8** (2020), 48151–48162. https://doi.org/10.1109/ACCESS.2020.2979281.
[73] S. Harris and A. Seaborne, SPARQL 1.1 Query Language, *W3C Recommendataion, W3C* (2013). https://www.w3.org/TR/sparql11-query/.
[74] I. Horrocks, P.F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof and M. Dean, SWRL: A Semantic Web Rule Language Combining OWL and RuleML, *W3C Member Submission, W3C* (2004). https://www.w3.org/submissions/SWRL/.
[75] S. Ceri, G. Gottlob and L. Tanca, What you Always Wanted to Know About Datalog (And Never Dared to Ask), *IEEE Trans. Knowl. Data Eng.* **1**(1) (1989), 146–166. https://doi.org/10.1109/69.43410.
[76] S. Abiteboul, R. Hull and V. Vianu, *Foundations of Databases*, Addison-Wesley, 1995. http://webdam.inria.fr/Alice/.
[77] T.J. Green, S.S. Huang, B.T. Loo and W. Zhou, Datalog and Recursive Query Processing, *Foundations and Trends in Databases* **5**(2) (2013), 105–195. https://doi.org/10.1561/1900000017.
[78] I. Donadello, M. Dragoni and C. Eccher, Persuasive Explanation of Reasoning Inferences on Dietary Data, in: *Proceedings of the 1st Workshop on Semantic Explainability, co-located with the 18th International Semantic Web Conference (ISWC 2019)*, CEUR Workshop Proceedings, Vol. 2465, CEUR-WS.org, 2019, pp. 46–61. https://ceur-ws.org/Vol-2465/semex_paper2.pdf.
[79] F. Corcoglioniti, M. Rospocher, M. Mostarda and M. Amadori, Processing Billions of RDF Triples on a Single Machine using Streaming and Sorting, in: *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, ACM, 2015, pp. 368–375. https://doi.org/10.1145/2695664.2695720.
[80] A. Mouakher, R. Belkaroui, A. Bertaux, O. Labbani, C. Hugol-Gential and C. Nicolle, An Ontology-Based Monitoring System in Vineyards of the Burgundy Region, in: *28th IEEE International Conference on Enabling Technologies*, IEEE, 2019, pp. 307–312. https://doi.org/10.1109/WETICE.2019.00070.
[81] H. Nakawala, R. Bianchi, L.E. Pescatori, O.D. Cobelli, G. Ferrigno and E.D. Momi, "Deep-Onto" network for surgical workflow and context recognition, *International Journal of Computer Assisted Radiology Surgery* **14**(4) (2019), 685–696. https://doi.org/10.1007/s11548-018-1882-8.
[82] B.N. Grosof, I. Horrocks, R. Volz and S. Decker, Description Logic Programs: Combining Logic Programs with Description Logic, in: *Proceedings of the Twelfth International World Wide Web Conference*, ACM, 2003, pp. 48–57. https://doi.org/10.1145/775152.775160.
[83] S.S. Edris, M. Zarka, W. Ouarda and A.M. Alimi, A Fuzzy Ontology driven Context Classification System using Large-Scale Image Recognition based on Deep CNN, in: *Sudan Conference on Computer Science and Info. Tech. (SCCSIT)*, IEEE, 2017, pp. 1–9.
[84] L. Ballan, M. Bertini, A.D. Bimbo and G. Serra, Video Annotation and Retrieval Using Ontologies and Rule Learning, *IEEE MultiMedia* **17**(4) (2010), 80–88. https://doi.org/10.1109/MMUL.2010.4.
[85] R. Kopparti and T. Weyde, Weight Priors for Learning Identity Relations, in: *Workshop Knowledge Representation & Reasoning Meets Machine Learning at NeurIPS*, 2019, pp. 8–14.
[86] E. Giunchiglia, M.C. Stoian, S. Khan, F. Cuzzolin and T. Lukasiewicz, ROAD-R: The Autonomous Driving Dataset with Logical Requirements, *Machine Learning* (2023). https://doi.org/10.1007/s10994-023-06322-z.
[87] C. Cornelio, J. Stuehmer, S.X. Hu and T. Hospedales, Learning Where and When to Reason in Neuro-Symbolic Inference, in: *International Conference on Learning Representations*, 2023. https://openreview.net/forum?id=en9V5F8PR-.
[88] I. Donadello and L. Serafini, Compensating Supervision Incompleteness with Prior Knowledge in Semantic Image Interpretation, in: *IJCNN Hungary, July 14-19*, IEEE, 2019, pp. 1–8. https://doi.org/10.1109/IJCNN.2019.8852413.
[89] E. Kharlamov, B.C. Grau, E. Jiménez-Ruiz, S. Lamparter, G. Mehdi, M. Ringsquandl, Y. Nenov, S. Grimm, M. Roshchin and I. Horrocks, Capturing Industrial Information Models with Ontologies and Constraints, in: *The Semantic Web - ISWC 2016 - 15th International Semantic Web Conference, Kobe, Japan, October 17-21, 2016, Proceedings, Part II*, P. Groth, E. Simperl, A.J.G. Gray, M. Sabou, M. Krötzsch, F. Lécué, F. Flöck and Y. Gil, eds, Lecture Notes in Computer Science, Vol. 9982, 2016, pp. 325–343. doi:10.1007/978-3-319-46547-0_30.
[90] T. Mossakowski, C. Maeder and K. Lüttich, The Heterogeneous Tool Set, Hets, in: *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, Vol. 4424, Springer, 2007, pp. 519–522. https://doi.org/10.1007/978-3-540-71209-1.
[91] S. Flügel, M. Glauer, F. Neuhaus and J. Hastings, When one Logic is Not Enough: Integrating First-order Annotations in OWL Ontologies, *Semantic Web* (2022). https://www.semantic-web-journal.net/system/files/swj3440.pdf.
[92] A.P. Sheth, K. Roy and M. Gaur, Neurosymbolic Artificial Intelligence (Why, What, and How), *IEEE Intell. Syst.* **38**(3) (2023), 56–62. https://doi.org/10.1109/MIS.2023.3268724.