Semantic Web 15 (2024) 1307–1331
DOI 10.3233/SW-233324
IOS Press

A neuro-symbolic system over knowledge graphs for link prediction

Ariam Rivas [a,b,c,*], Diego Collarana [f,g], Maria Torrente [d,e] and Maria-Esther Vidal [a,b,c]

[a] *Leibniz University of Hannover, Germany*
E-mail: ariam.rivas@tib.eu
[b] *TIB Leibniz Information Centre for Science and Technology, Germany*
E-mail: maria.vidal@tib.eu
[c] *L3S Research Centre, Germany*
[d] *Department of Medical Oncology, Puerta de Hierro-Majadahonda University Hospital, 28222 Madrid, Spain*
E-mail: mtorrente80@gmail.com
[e] *Faculty of Health Sciences, Francisco de Vitoria University, 28223 Madrid, Spain*
[f] *Fraunhofer Institute for Intelligent Analysis and Information Systems, Dresden, Germany*
E-mail: diego.collarana.vargas@iais.fraunhofer.de
[g] *Universidad Privada Boliviana, Cochabamba, Bolivia*

Editors: Monireh Ebrahimi, IBM, USA; Pascal Hitzler, Kansas State University, USA; Md Kamruzzaman Sarker, University of Hartford, USA; Daria Stepanova, Bosch Center for AI, Germany
Solicited reviews: Abhilekha Dalal, Kansas State University, USA; one anonymous reviewer

Abstract. Neuro-Symbolic Artificial Intelligence (AI) focuses on integrating symbolic and sub-symbolic systems to enhance the performance and explainability of predictive models. Symbolic and sub-symbolic approaches differ fundamentally in how they represent data and make use of data features to reach conclusions. Neuro-symbolic systems have recently received significant attention in the scientific community. However, despite efforts in neural-symbolic integration, symbolic processing can still be better exploited, mainly when these hybrid approaches are defined on top of knowledge graphs. This work is built on the statement that knowledge graphs can naturally represent the convergence between data and their contextual meaning (i.e., knowledge). We propose a hybrid system that resorts to symbolic reasoning, expressed as a deductive database, to augment the contextual meaning of entities in a knowledge graph, thus, improving the performance of link prediction implemented using knowledge graph embedding (KGE) models. An entity context is defined as the ego network of the entity in a knowledge graph. Given a link prediction task, the proposed approach deduces new RDF triples in the ego networks of the entities corresponding to the heads and tails of the prediction task on the knowledge graph (KG). Since knowledge graphs may be incomplete and sparse, the facts deduced by the symbolic system not only reduce sparsity but also make explicit meaningful relations among the entities that compose an entity ego network. As a proof of concept, our approach is applied over a KG for lung cancer to predict treatment effectiveness. The empirical results put the deduction power of deductive databases into perspective. They indicate that making explicit deduced relationships in the ego networks empowers all the studied KGE models to generate more accurate links.

Keywords: Neuro-symbolic artificial intelligence, deductive systems, knowledge graph embeddings, drug-drug interactions

[*] Corresponding author. E-mail: ariam.rivas@tib.eu.

1. Introduction

Neuro-Symbolic Artificial Intelligence is a research field that combines symbolic and sub-symbolic AI models [3, 8,35]. The symbolic models refer to AI approaches based on handling explicit symbols to conduct reasoning and support explainability. On the other hand, AI sub-symbolic systems are based on statistical and probabilistic learning from data mining and neural network models. Symbolic and sub-symbolic systems differ in how they represent and manage data to perform reasoning and prediction. As a result, they aim at solving complementary tasks whose integration has the potential to empower prediction with reasoning supported by symbolic formal frameworks [3,15].

Neuro-symbolic integration aims to bridge the gap between symbolic and sub-symbolic systems; it resorts to translation algorithms to align symbolic to sub-symbolic representations and improve performance [3,8,38]. However, integrating neuro-symbolic into real-world applications is a challenging task. Even in controlled environments, neuro-symbolic integration may not be completed performed [14]. For instance, Fernlund et al. [11] describe systems that use machine learning to learn relations from expert observations. While these systems are successful in learning, they lack the expressive power of symbolic systems. Another example of neuro-symbolic systems combining connectionist inductive learning and logic programming to solve the problems in the molecular biology and power plant fault diagnosis [9]. Furthermore, Karpathy et al. [19] combine convolutional neural networks with bidirectional recurrent neural networks over sentences to recognize and label image regions. Despite these advances in neuro-symbolic AI integration, symbolic processing is not fully exploited, in particular, if reasoning methods are implemented on top of knowledge graphs [38].

Problem Statement and Proposed Solution: We tackle the problem of link prediction over knowledge graphs and propose an approach combining symbolic reasoning and sub-symbolic prediction. Our approach integrates a domain-agnostic symbolic system with knowledge graph embedding models. It resorts to symbolic reasoning to deduce relationships between entities that compose the ego network of the entities in a knowledge graph, where the ego network of an entity v is the set of edges connected to v in the knowledge graph. Thus, contextual knowledge, represented by ego networks, is enhanced, and the sparsity of knowledge graphs is reduced. Since the behavior of knowledge graph embedding models can be affected in sparse graphs [45], training these models with these enhanced ego networks increases the chances of predicting accurate links between entities associated with these networks. We apply our hybrid approach in the context of lung cancer. The symbolic system implements a deductive database to infer drug-drug interactions in lung cancer treatments. Complementary, the sub-symbolic system resorts to knowledge graph embedding models to predict the effectiveness of a lung cancer treatment. These models transform RDF triples representing treatments, their drugs, and interactions among these drugs into a low-dimensional continuous vector space that preserves the knowledge graph structure. The integration of both systems enables the prediction of a treatment's response, taking into account the potential effect that drug-drug interactions have on the effectiveness of the treatment.

Results: We assess the performance of the proposed neuro-symbolic system on a knowledge graph built from clinical records of lung cancer patients; it comprises treatments prescribed to these patients, the responses of these treatments, and the drugs that have been administrated. Additionally, this knowledge graph integrates information about the drug-drug interactions between the oncological and non-oncological drugs composing a lung cancer treatment. These drug-drug interactions have been extracted from DrugBank[1] following the named entity recognition, and linking techniques proposed by Sakor et al. [33]. The prediction task is defined in terms of predicting links between treatments (i.e., heads) and instances of a class representing the different types of lung cancer responses (i.e., tails). The link prediction task is implemented using eleven state-of-the-art KGE models. The experiments are executed following different configurations and baselines, with the goal of assessing the accuracy of our proposed neuro-symbolic system. Results of a 5-fold cross-validation process demonstrate that our integrated system improves the prediction accuracy of studied state-of-the-art KGE models. Moreover, the outcomes of this experimental study put the power of deductive databases into perspective, showing how they can empower the accuracy of link prediction tasks. More importantly, these results provide evidence of the paramount role of deductive reasoning and knowledge graph embedding models in predicting treatment response.

[1] https://go.drugbank.com

Contributions: This paper resorts to our previous work [29], where we propose a deductive system over knowledge graphs to formalize the process of drug-drug interactions. Built on these results, we present a hybrid approach combining symbolic reasoning expressed by deductive systems with the sub-symbolic expressiveness of KGE models to enhance prediction accuracy. In a nutshell, our novel contributions are:

1. A domain-agnostic approach able to empower the predictive performance of sub-symbolic systems with a deductive database system. The deductive system reduces data sparsity issues by inferring implicit relationships in a KG. Consequently, the sub-symbolic system, implemented by KGE models, better represents statements described in the KG into a low-dimensional continuous vector space.
2. An extensive evaluation of our neuro-symbolic system with state-of-the-art KGE models demonstrates the benefit of integrating deductive reasoning and sub-symbolic systems. The evaluation is performed on the problem of predicting the effectiveness of lung cancer treatments composed of multiple drugs, i.e., polypharmacy treatments.

The rest of the paper is structured as follows: Section 2 presents the preliminaries and a motivating example. Section 3 shows the proposed approach and illustrates its main features with a running example. Section 4 applies our hybrid method in the context of predicting the effectiveness of polypharmacy lung cancer treatments. Results of the empirical evaluation of our method are reported in Section 5. Section 6 analyses the state-of-the-art. Finally, we close with the conclusion and future work in Section 7.

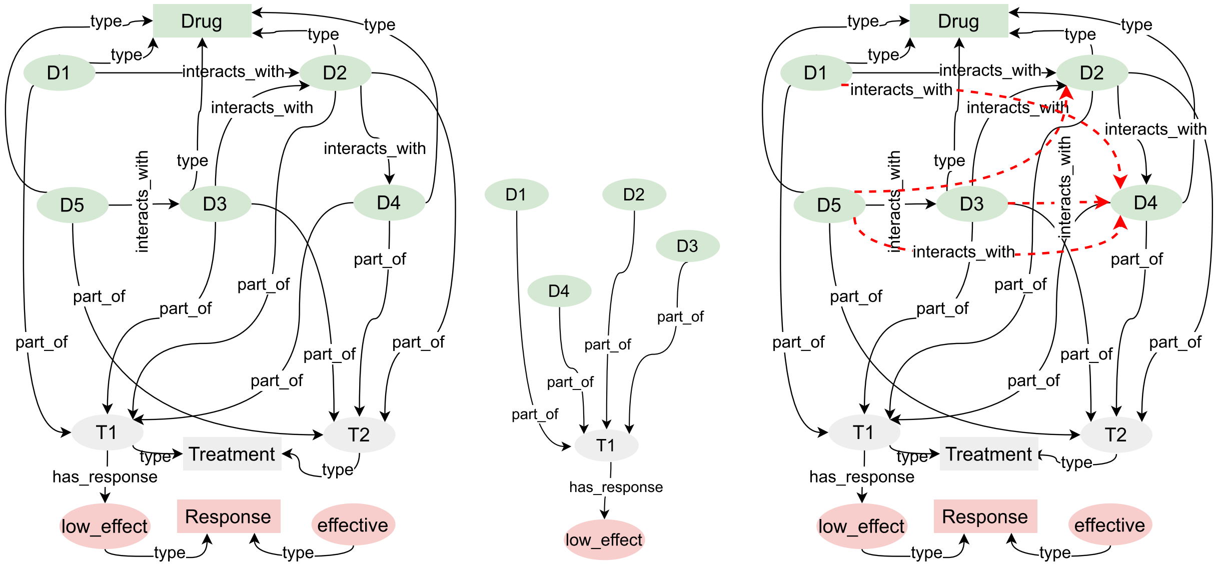
2. Preliminaries and motivation

Knowledge Graphs (KGs) are data structures converging data and knowledge as factual statements of a graph data model [13,16]. Formally, a knowledge graph is a 10-tuple $\mathcal{KG} = (V, E, L, C, I, D, R, \mathcal{N}, ego, \alpha)$, where:

- V is a set of nodes corresponding to concepts (e.g., classes and entities).
- $E \subseteq V \times L \times V$ is a set of edges representing relationships, i.e., triples (s, p, o), between concepts.
- L is a set of properties.
- C is a set of classes $C \subseteq V$.
- $I : V \to C$ is a function that maps each entity in V to a class C.
- $D : L \to C$ maps a property to the class that corresponds to the domain of the property.
- $R : L \to C$ maps each property to a class that corresponds to the range of the property.
- $\mathcal{N} : V \to 2^V$, where 2^V represents the power set of nodes V. $\mathcal{N}(v)$ defines the neighbors of the entity v, i.e., $\mathcal{N}(v) = \{v_i | (v, r, v_i) \in E \vee (v_i, r, v) \in E\}$.
- $ego : V \to 2^{V \times L \times V}$, the function $ego(.)$ represents ego networks in the knowledge graph. $ego(v)$ assigns to each concept in V the set of labeled edges, where v is in the subject or object position. $ego(v) = \{(u_1, r, u_2) | (u_1, r, u_2) \in E \wedge (u_1 = v \vee u_2 = v)\}$. The $ego(v)$ defines the ego network of the entity v.
- $\alpha : 2^V \to 2^{V \times L \times V}$. The function $\alpha(.)$ returns a set of triples between the pairs of elements in the input.If F is a set of entities in V, $\alpha(F) = \{(v_1, r, v_2) | (v_1, r, v_2) \in E \wedge v_1 \in F \wedge v_2 \in F\}$. The function $\alpha(.)$ returns the edges between pairs of entities in the input set F.

Figure 1(a) depicts a knowledge graph $\mathcal{KG}$, where the set of classes are represented by $C = \{Drug, Treatment, Response\}$. The class for each entity is represented by the function $I(.)$, e.g., the entity $T1$ belongs to the class *Treatment* and $I(T1) = Treatment$. For the property *has_response* $\in L$, the domain is defined by the function $D(has\_response) = Treatment$, while the range is $R(has\_response) = Response$. Figure 1(b) illustrates the ego network of the entity $T1$, where the neighbors of the entity $T1$ are defined by $\mathcal{N}(T1) = \{D1, D2, D3, D4, low\_effect\}$. Furthermore, the set of edges between pairs of entities in the set of neighbors of entity $T1$ is defined by $\alpha(\mathcal{N}(T1)) = \{(D1, interacts\_with, D2), (D2, interacts\_with, D4), (D3, interacts\_with, D2)\}$, where we can observe the three triples in Fig. 1(a). Note that although *low_effect* is in the ego network of the entity $T1$, this entity is not related to any other entity in this ego network.

An ideal knowledge graph. An ideal knowledge graph is a knowledge graph $\mathcal{KG}' = (V, E', L, C, I, D, R, \mathcal{N}, ego, \alpha)$ that contains all the true existing relations between entities in V. The Closed World Assumption (CWA) is assumed on $\mathcal{KG}'$, i.e., what is unknown to be true in $\mathcal{KG}'$ it is false.



(a) Knowledge Graph $\mathcal{KG}$ (b) Ego network of entity $T1$ (c) Knowledge Graph $\mathcal{KG}$ applying DS

Fig. 1. **Example knowledge graph.** Fig. 1(a) shows a KG with three classes, five green entities belonging to class *Drug*, two gray entities belonging to class *Treatment*, and two red entities belonging to class *Response*. Figure 1(b) illustrates the ego network for the entity $T1$, where the entities $D1$, $D2$, $D3$, $D4$, and *low_effect* are the neighbors of $T1$. Figure 1(c) shows the KG resulting from DS (deductive system). The red arrows represent the new deduced links in the ego network $ego(.)$.

An actual knowledge graph. An actual knowledge graph $\mathcal{KG} = (V, E, L, C, I, D, R, \mathcal{N}, ego, \alpha)$ is a knowledge graph that follows the assumption Open World Assumption (OWA), i.e., what is not known to be true is just unknown and may be true.

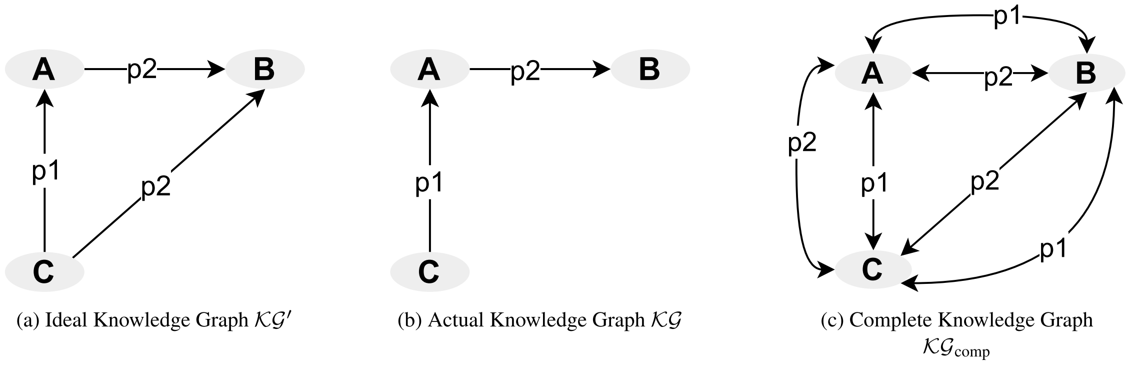
A complete knowledge graph. A complete knowledge graph $\mathcal{KG}_{\text{comp}} = (V, E_{comp}, L, C, I, D, R\mathcal{N}, ego, \alpha)$ is a knowledge graph, which includes a relation for each possible combination of entities in V. Note that all relationships in $\mathcal{KG}_{\text{comp}}$ are not necessarily declared as true (w.r.t. domain knowledge).

A knowledge graph $\mathcal{KG}$ may only contain a portion of the edges represented in $\mathcal{KG}'$, i.e., $E \subseteq E'$. $\mathcal{KG}$ represents those relations that are known but it is not necessarily complete. On the other hand, since $\mathcal{KG}_{\text{comp}}$ is a complete knowledge graph, $E \subseteq E' \subseteq E_{\text{comp}}$. The set of missing edges in $\mathcal{KG}$ is defined as $\Delta(E', E) = E' - E$, i.e., it is the set of relations existing in the ideal knowledge graph $\mathcal{KG}'$ that are not represented in $\mathcal{KG}$. Figure 2 illustrates three knowledge graphs. Figure 2(a) is an ideal knowledge graph that states that only three relationships are true. The actual knowledge graph, presented in Fig. 2(b), is incomplete and only includes two relationships; $(C, p2, B)$ is unknown and is not part of the current knowledge graph. Figure 2(c) illustrates a complete knowledge graph, with a relation for each combination of entities in V and properties in L. All the possible relationships are included in this graph.

An **abstract target prediction** over a knowledge graph $\mathcal{KG}$ is defined in terms of a tuple $\tau = \langle \mathcal{KG}, r, prediction, DS, KGE \rangle$:

- $\mathcal{KG}$ is a knowledge graph $\mathcal{KG} = (V, E, L, C, I, D, R, ego, \mathcal{N}, \alpha)$.
- r represents a prediction property, $r \in L$.
- *prediction* indicates the head or the tail of triples to predict. A tail prediction of triples $\langle h, r, t\rangle$ is the process of finding t for the incomplete triple $\langle h, r, ?\rangle$, head predictions can be defined analogously.
- DS is a deductive database system over $\mathcal{KG}$.
- KGE is a knowledge graph embedding model over $\mathcal{KG}$.

The deductive system DS derives new facts from inference rules and facts stored in a database [26]; it is expressed as a set of extensional and intensional rules in Datalog. A Datalog rule corresponds to a Horn clause [7], $L_1, \ldots, L_n \Rightarrow L_0$, where each L_i is a literal of the form $p_i(t_1, \ldots, tk_i)$. P_i is a predicate symbol and t_j are terms.



(a) Ideal Knowledge Graph $\mathcal{KG}'$ (b) Actual Knowledge Graph $\mathcal{KG}$ (c) Complete Knowledge Graph $\mathcal{KG}_{\text{comp}}$

Fig. 2. Example of actual, ideal, and complete knowledge graph.

A term is either a constant or a variable. The right-hand side of a Datalog clause is the head, and the left-hand side is its body. Clauses with an empty body represent facts. A Datalog program P must satisfy the following safety conditions; each fact of P is ground, and each variable that occurs in the head of a rule of P must also occur in the body of the same rule. A rule is safe if all its variables are bounded, where any variable appearing as an argument in a body predicate is bounded. Datalog considers two sets of clauses: a set of ground facts called the Extensional Database (EDB) and a Datalog program P called the Intensional Database (IDB). The predicates in the EDB and IDB are divided into two disjoint sets, EDB predicates, which occur in the EDB, and the IDB predicates, which occur in IDB. The head predicate of each clause in P is an IDB predicate, and the EDB predicate can occur in the body of the rule. If C_1 and C_2 are the domain and range of r respectively, then EDB comprises ground facts of the form: $p(s, o)$ where the triple $(s, p, o) \in ego(v) \cup \alpha(\mathcal{N}(v))$, and $I(v) \in \{C_1, C_2\}$. The EDB in our *DS* contains ground facts from the ego networks and their neighbors. Given a prediction property, $r = has\_response$ we know the domain $D(has\_response) = Treatment$ and range $R(has\_response) = Response$. Figure 1(a) shows entities of type *Treatment* and entities of type *Response* for the domain and range of the property *has_response*, respectively. The EDB comprises all the ground facts defined by the ego networks: $ego(T1)$, $ego(T2)$, $ego(low\_effect)$, and $ego(effective)$, and their neighbors $\alpha(\mathcal{N}(T1))$, $\alpha(\mathcal{N}(T2))$, $\alpha(\mathcal{N}(T low\_effect))$, and $\alpha(\mathcal{N}(effective))$, where entities $T1$ and $T2$ belong to class *Treatment*, and *low_effect* and *effective* belong to the class *Response*.

An example of EDB is the set of facts $\{interacts\_with(D1, D2), interacts\_with(D2, D4)\}$, where the property $interacts\_with \in L$ and the entities $\{D1, D2, D4\} \subseteq V$. The predicate *interacts_with* represents interactions between two drugs. Let $P(1)$ be a Datalog program (IDB) containing the following clauses:

$$\begin{aligned} &rule1\ interactsWith(A, X) \Rightarrow inferredInteraction(A, X). \\ &rule2\ inferredInteraction(B, X), interactsWith(A, B) \Rightarrow inferredInteraction(A, X). \end{aligned} \tag{1}$$

The predicate $inferredInteraction(A, X)$ is an IDB predicate, and $interactsWith(A, X)$ is an EDB predicate. Rule *rule*2 states that exist an *inferred_interaction* between drug A and X, if there is another drug B which interacts with A with the predicate *interacts_with*, and there is an *inferred_interaction* from B to X. The evaluation results of *rule*2 is $\{inferred\_interaction(D1, D4)\}$, shown in Fig. 1(c) with a red arrow.

KGE model learns vector representation (i.e., KG embeddings) in a low dimensional continuous vector space for entities $v \in V$ and relations $e \in E$ in a $\mathcal{KG}$. *KGE* model exploits the $\mathcal{KG}$ structure to predict new relations in E. The *KGE* model resorts to a scoring function ϕ to estimate the plausibility of the vector representation of a triple, where higher ϕ values yield higher plausibility [31]. Link prediction is performed by identifying which vector representation of an entity provides the best values of the scoring function ϕ. These entities are added to the incomplete triples as heads or tails. If $prediction = tail$, then the link prediction task is the process of finding t as the best scoring tail for the incomplete triple $\langle h, r, ?\rangle$:

$$\underset{t \in V}{\operatorname{argmax}}\, \phi(h, r, t).$$

If $prediction = head$, it can be defined analogously. The state of the art of KGE methods may be negatively impacted by the data sparsity issue, i.e., ground facts that can be used as positive samples to guide KGE training represent only a minor portion. The proposed deductive database system for abstract target prediction alleviates the data sparsity issue by enhancing links in the ego network $ego(v)$, which are managed as new ground facts.

Suppose the abstract target prediction is defined for the current knowledge graph $\mathcal{KG}$ presented in Fig. 1(a) where the prediction property is $r = has\_response$, and the prediction corresponds to the tail, i.e., $prediction = tail$. The link prediction task predicts incomplete triples $\langle h, r, ?\rangle$, where the head h represents entities of class *Treatment*, i.e., entities h in V such that $I(h) = Treatment$, and the relation is $r = has\_response$.

2.1. Motivating example

We motivate our work in healthcare, specifically for predicting polypharmacy treatment response. Polypharmacy is the concurrent use of multiple drugs in treatments, and it is a standard procedure to treat severe diseases, e.g., lung cancer. Polypharmacy is a topic of concern due to the increasing number of unknown drug-drug interactions (DDIs) that may affect the response to medical treatment. Pharmacokinetics is a type of DDIs, i.e., *the course of a drug in the body*. Pharmacokinetics DDIs alter a drug's absorption, distribution, metabolism, or excretion. For example, an increase in absorption will increase the object drug's bioavailability and vice versa. If a DDI affects the object's drug distribution, the drug transport by plasma proteins is altered. Moreover, a drug's therapeutic efficacy and toxicity are affected when a pharmacokinetics DDI alters the object's drug metabolism. Lastly, if the excretion of an object drug is reduced, the drug's elimination half-life will be increased. Notice that the pharmacokinetic interactions can be encoded in a symbolic system.

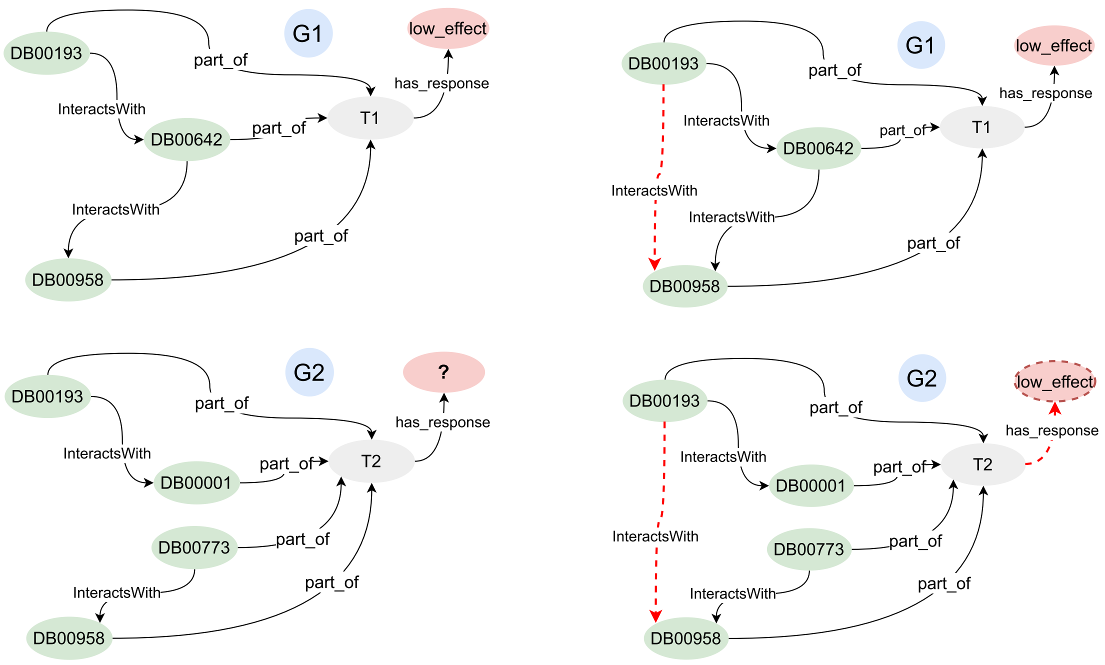
Figure 3(a) shows two polypharmacy oncological treatments encoded in RDF. We extract the known DDIs between the drugs of these treatments from DrugBank. However, polypharmacy therapies produce unforeseen DDIs due to drug interactions in the treatment. Since DDIs affect the effectiveness of a treatment, there is a great interest in uncovering these DDIs. Figure 3(b) depicts an ideal RDF graph where all the existing relations are explicitly represented. Dotted red arrows represent DDI between the drugs `DB00193` and `DB00958` that are generated as the result of DDIs among drugs in the treatment. A Datalog program represents the rules that state when these DDIs are produced between the drugs administrated in a treatment. The extensional database corresponds to facts representing explicit relationships; in our case, these facts are extracted from DrugBank. The intensional database corresponds to intensional rules that define all the combinations of DDIs that may produce new DDIs; they allow for deducing implicit DDIs in a treatment. The DDI between `DB00193` and `DB00958` increases the information of treatments `T1` and `T2`, enabling both treatments to share more relationships. Then, a sub-symbolic system, e.g., a knowledge graph embedding model, can explore these enhanced relationships and make a more accurate prediction of the treatment response by employing the deduced DDIs. For example, the geometric model *TransH* places `T1` and `T2` nearby in the embedding space after deducing DDIs and predicts the therapeutic response of `T2`. As a result, this neuro-symbolic system enhances treatment information by identifying drug combinations whose interactions may affect treatment effectiveness. We propose an approach that resorts to symbolic reasoning implemented by a Datalog database and stage-of-the-art KGE models; it deduces DDIs within a treatment. Then, the KGE model embeds all the knowledge in the graph and predicts treatment responses. Although we depict the method in the context of treatment effectiveness, this approach is domain-agnostic and could be applied to any other link prediction task.

3. Proposed symbolic and sub-symbolic system

3.1. Problem statement

Given an actual knowledge graph $\mathcal{KG} = (V, E, L, C, I, D, R, ego, \mathcal{N}, \alpha)$ and its corresponding ideal knowledge graph $\mathcal{KG}' = (V, E', L, C, I, D, R, ego, \mathcal{N}, \alpha)$. Given an abstract target prediction over an actual knowledge graph $\mathcal{KG}$, $\tau = \langle \mathcal{KG}, r, prediction, DS, KGE\rangle$, we tackle the *problem of predicting relationships over* $\mathcal{KG}$.

Given a relation, $e \in \Delta(E_{\text{comp}}, E)$ (i.e., the set of missing edges in $\mathcal{KG}$), the problem of predicting relationships consists of determining whether $e \in E'$, i.e., if a relation e corresponds to an existing relation in the ideal knowledge



(a) **Oncological Treatment**

(b) **Predicting Response of Oncological Treatment**

Fig. 3. **Motivating example.** Fig. 3(a) shows two polypharmacy oncological treatments, *T1* and *T2*, represented in RDF. The drugs *DB00193*, *DB00642*, and *DB00958* are part of *T1*, and the drug-drug interactions are represented by the property *InteractsWith*. The therapeutic response of *T1* is annotated as *low_effect* by the property *has_response*, while the therapeutic response of *T2* is unknown. Figure 3(b) depicts the ideal RDF graph, where a symbolic system generates a new DDI between *DB00193* and *DB00958*. Ideally, a sub-symbolic system detects that both treatments are similar and predicts the effectiveness of *T2* as low effective.

graph $\mathcal{KG}'$. We are interested in finding the maximal set of relationships or edges E_a that belongs to the ideal $\mathcal{KG}'$, i.e., find a set E_a that corresponds to a solution of the following optimization problem:

$$\underset{E_a \subseteq E_{comp}}{\operatorname{argmax}} \left| E_a \cap E' \right|.$$

3.2. *Proposed solution*

Our proposed solution resorts to a symbolic system implemented by a deductive database to enhance the predictive precision of the link prediction task solved by knowledge graph embedding models. The approach assumes that a link prediction problem is defined in terms of an abstract target prediction $\tau = \langle \mathcal{KG}, r, \mathit{prediction}, \mathit{DS}, \mathit{KGE} \rangle$ over a knowledge graph $\mathcal{KG} = (V, E, L, C, I, D, R, \mathit{ego}, \mathcal{N}, \alpha)$.

A Symbolic System: Deductive system *DS* corresponds to the deductive databases where the EDB comprises ground facts of the form: $p(s, o)$, where the triple $\langle s, p, o \rangle \in \mathit{ego}(v) \cup \alpha(\mathcal{N}(v))$, $I(v) \in \{C_1, C_2\}$, $C_1 = D(r)$, and $C_2 = R(r)$. The variables C_1 and C_2 represent the domain and range of the property r, respectively. The IDB contains rules that allow deducing new relationships in the ego network $\mathit{ego}(v)$. The computational method executed to empower the ego networks $\mathit{ego}(v)$ is built on the results of deductive databases to compute the minimal model of the deductive database [7]. The minimal model corresponds to the instantiations of IDB predicates. This minimal model is defined in terms of the fixed-point assignment $\sigma^{\mathit{ego}(.)}_{\text{MINFIX}}$, that deduces relationships between entities v_i and v_j in the neighbors $\mathcal{N}(.)$. The minimal model for *DS* can be computed in polynomial time in the overall size of the ego network $\mathit{ego}(v)$ and the neighbors $\alpha(\mathcal{N}(v))$ for all the entities v where $I(v) \in \{C_1, C_2\}$, $C_1 = D(r)$, and $C_2 = R(r)$.

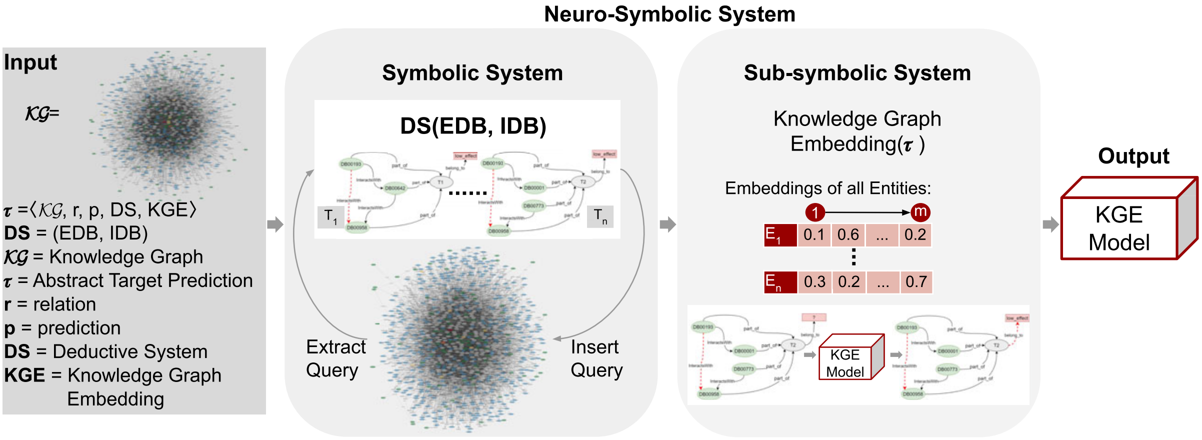


Fig. 4. **Approach**. The input is a knowledge graph ($\mathcal{KG}$), an abstract target prediction τ, and a deductive system, and returns a KGE model. The symbolic system is implemented by a deductive system $DS(EDB, IDB)$ that deduces new relationships in the ego network $ego(v)$ and between their neighbors $\alpha(\mathcal{N}(v))$. Then, the sub-symbolic system implemented by a KGE model employs the $\mathcal{KG}$ with the deduced new relationships to predict incomplete triples. KGE solves the abstract target prediction τ for the relation r and the *prediction* head or tail.

A Sub-symbolic System: A model to learn Knowledge Graph Embeddings solves the abstract target prediction τ over $\mathcal{KG}$ for the relation r and the *prediction* head or tail. The sub-symbolic system predicts incomplete triples of the way $\langle h, r, ?\rangle$ if $prediction = tail$ and $\langle ?, r, t\rangle$ if $prediction = head$.

The Integration of Symbolic and Sub-symbolic Systems: The ego network $ego(v)$ and the edges between their neighbors $\alpha(\mathcal{N}(v))$ are extended with explicit relationships among entities in the neighbors $\mathcal{N}(v)$ by the deductive system DS. As a result, the symbolic system implemented by DS alleviates the data sparsity issues in $\mathcal{KG}$ that may negatively affect the learning of the KGE in the abstract target prediction τ.

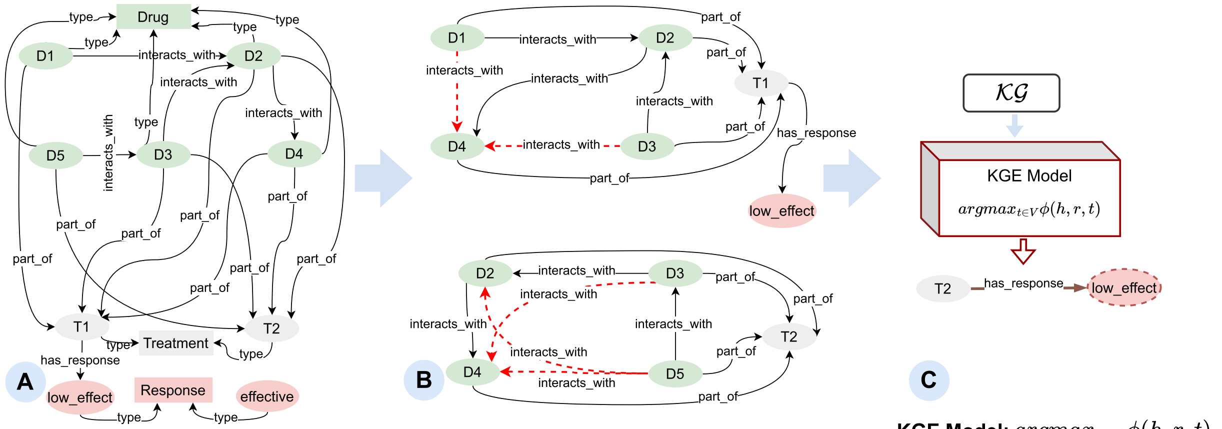
3.3. The symbolic and sub-symbolic system architecture

Figure 4 depicts the architecture that implements the proposed approach. The architecture receives a knowledge graph $\mathcal{KG} = (V, E, L, C, I, D, R, ego, \mathcal{N}, \alpha)$ and an abstract target prediction $\tau = \langle \mathcal{KG}, r, prediction, DS, KGE\rangle$, where $\mathcal{KG}$ is the knowledge graph, r is a property, *prediction* represents the head or tail of triples to predict, DS is the deductive system, and KGE is the knowledge graph embedding. The architecture returns a learned model of embeddings. These embeddings are used to solve the target prediction task defined by τ.

The architecture is composed of two main steps. First, the relationships implicitly defined by the deductive system are deduced by means of a Datalog program. Second, once $\mathcal{KG}$ is augmented with new deduced relationships, KGE learns a latent representation of entities and properties of $\mathcal{KG}$ in a low-dimensional space. The architecture is agnostic of the method to learn the embeddings. Moreover, our approach is domain-agnostic. For example, it can be applied in the context of Industry 4.0 to discover relations between standards and thus solve interoperability issues between standardization frameworks [27,28].

3.4. Abstract target prediction task. Running example

Albeit illustrated in the context of treatment response, the proposed method is domain-agnostic. It only requires the definition of the deductive system to enhance the relationships in the ego network of the entities v where $I(v) \in \{C_1, C_2\}$, $C_1 = D(r)$, and $C_2 = R(r)$. The variables C_1 and C_2 represent the domain and range of the property r, respectively. Figure 5 illustrates the proposed steps to enhance the predictive performance by knowledge graph embedding models. The $\mathcal{KG}$ shown in Fig. 5**(A)** is the same as in Fig. 1(a). Assuming we receive as input the abstract target prediction $\tau = \langle \mathcal{KG}, r, prediction, DS, KGE\rangle$, where the $\mathcal{KG}$ is represented in Fig. 5**(A)**, the property is $r = has\_response$, the $prediction = tail$, DS is the deductive system, and KGE is the KGE algorithm. The EDB of the DS comprises all the ground facts of the form: $p(s, o)$, where the triple $(s, p, o) \in ego(v) \cup \alpha(\mathcal{N}(v))$,



Original Knowledge Graph $\mathcal{KG}$: contains five green entities belonging to class Drug, two gray entities belonging to class Treatment, and two red entities belonging to class Response.

The **target prediction** corresponds to links between entities of type Treatment and Response. Thus, the head of a target prediction is a treatment, while the tail is a response.

Enhancing links between entities in the ego network ego(T1) and ego(T2). Both ego networks contain the relations between the set of neighbors of T1 and T2 defined by $\alpha(\mathcal{N}(T1))$ and $\alpha(\mathcal{N}(T2))$.

The deductive system DS deduces new relationships in ego(T1) and ego(T2). The red arrows represent the deduced relationships. The deductive system enriches the $\mathcal{KG}$.

KGE Model: $argmax_{t\in V}\phi(h,r,t)$

KGE model infers missing links for each triple using the highest score of the function $\phi(h,r,t)$.

KGE model places T1 and T2 nearby in the embedding space after deducing new facts in their ego networks. The KGE model predicts the tail = low_effect for the missing link $\langle T2, has\_response, ?\rangle$

Fig. 5. **Running example**. Figure 5 illustrates the proposed steps to enhance the predictive performance of KGE models. **Step A**: given a KG and an abstract target prediction $\tau = \langle$*Treatment, has_response, Response*$\rangle$ the ego network $Ego_{\mathcal{G}}(v)$ is defined. **Step B**: illustrates the neighbors of the ego network $Ego_{\mathcal{G}}(T1)$ and $Ego_{\mathcal{G}}(T2)$ and a deductive system based on the head and tail of τ deduces new relationships to enhances the neighbors $\mathcal{N}_{\mathcal{G}}(Ego_{\mathcal{G}}(v))$. **Step C**: depicts a KGE model in which predictive performance is enhanced by symbolic reasoning. The relationships in $\mathcal{N}_{\mathcal{G}}(Ego_{\mathcal{G}}(v))$ improving the link prediction task in $\mathcal{G}|_{\tau}$.

```
PREFIX ex: <http://example/vocab/>
CONSTRUCT {?A <interacts_with> ?B} WHERE {
        ?A ex:part_of <T1> .
        ?B ex:part_of <T1> .
        ?A ex:interacts_with ?B. }
```

Listing 1. **SPARQL query to ground the extensional predicate** *interacts_with*(A, B)

$I(v) \in \{C_1, C_2\}$, $C_1 = D(has\_response)$, and $C_2 = R(has\_response)$. Then, the domain and range of the property $r = has\_response$ are *Treatment* and *Response*, respectively. In addition, the entity type for v in ego network $ego(v)$ is *Treatment* or *Response*. The entities *low_effect* and *effective* are of type *Response*, and $T1$ and $T2$ are entities of type *Treatment*.

The EDB comprises all the ground facts defined by the ego networks: $ego(T1)$, $ego(T2)$, $ego(low\_effect)$, and $ego(effective)$, and their neighbors $\alpha(\mathcal{N}(T1))$, $\alpha(\mathcal{N}(T2))$, $\alpha(\mathcal{N}(low\_effect))$, and $\alpha(\mathcal{N}(effective))$. Figure 5**(B)** shows the ego networks $ego(T1)$ and $ego(T2)$ with the set of edges between pairs of entities in the set of neighbors of entity $T1$ and $T2$ defined by $\alpha(\mathcal{N}(T1))$ and $\alpha(\mathcal{N}(T2))$, respectively. Then, *DS* deduces new relationships enhancing the links in the $ego(T1)$ and $ego(T2)$; red arrows represent the deduced relationships. Considering the Datalog program $P(1)$ as the IDB for *DS*, the facts *inferred_interaction*$(D1, D4)$, *inferred_interaction*$(D3, D4)$, *inferred_interaction*$(D5, D4)$, and *inferred_interaction*$(D5, D2)$ are deduced enhancing the ego network.

The SPARQL query in Listing 1 extracts the ego network $ego(T1)$ and the set of edges between pairs of entities in the set of neighbors of entity $T1$ defined as $\alpha(\mathcal{N}(T1))$. Listing 1 illustrates a CONSTRUCT query that returns RDF triples in the form of subject, predicate, and object and represents the ground facts of the EDB. The predicate represents the ground predicated in the EDB, the subject represents the first term of the ground predicated, and the object represents the second term.

```
PREFIX ex: <http://example/vocab/>
INSERT DATA {
        <A> ex:interacts_with <X>
        }
```

Listing 2. **SPARQL query to insert the deduced relationships from the intensional predicate** *inferred_interaction(A,X)*

The IDB described by the Datalog program $P(1)$ allows deducing new relationships and increasing the ego networks $ego(T1)$ and $ego(T2)$. The deduced relations are inserted into the $\mathcal{KG}$ through the SPARQL query in Listing 2. The deduced relationship e belongs to E', i.e., $e \in E'$.

Figure 5**(C)** illustrates a *KGE* model in which the symbolic system enhances predictive precision. The *DS* increases the relationships E in $\mathcal{KG}$, alleviating the data sparsity issues in $\mathcal{KG}$. Thus, the $\mathcal{KG}$ that contains new facts deduced by *DS* guides the *KGE* model, improving the link prediction task for $r = has\_response$ and $prediction = tail$. Figure 5**(C)** shows the link prediction task of finding t as the best scoring tail for the incomplete triple $(T2, has\_response, ?)$: $\text{argmax}_{t \in V}\ \phi(T2, has\_response, t)$. Treatment $T2$ is predicted to have a response *low_effect*(*T2, has_response, low_effect*), i.e., $T1$ and $T2$ are nearby in the embedding space after enhancing the $ego(T1)$ and $ego(T2)$ in $\mathcal{KG}$.

4. Use case: Prediction of polypharmacy treatment effectiveness

As a proof concept, we apply our neuro-symbolic approach to address the problem of predicting polypharmacy treatment effectiveness. We have implemented a deductive system on top of a Treatment Knowledge Graph ($\mathcal{KG}$). The technique aims to identify the combination of drugs whose interactions may affect the treatment's effectiveness. Then, the problem of predicting treatment effectiveness is modeled as a problem of link prediction between treatments and the responses: *low-effect* or *effective*.

4.1. Treatment knowledge graph creation

The P4-LUCAT consortium[2] collected heterogeneous data sources that comprise clinical records, drugs, and scientific publications and built a knowledge graph that provides an integrated view of these data. The KG is built with the aim of personalized medicine for Lung Cancer treatments. The treatments are extracted from Electronic Health Records (EHRs) from the Hospital Universitario Puerta del Hierro of Majadahonda of Madrid (HUPHM). Furthermore, the DDIs are extracted from DrugBank, in the approved category. The interactions' type and effect are extracted using named entity and linking methods implemented by Sakor et al. [34]. These methods have also been used to extract DDIs in covid-19 and lung cancer treatments [1,33,39]. Table 1 contains a summary of the number of annotations by classes in the Lung Cancer Knowledge Graph.

Figure 6 describes a Lung Cancer patient in the Lung Cancer Knowledge Graph. The patient *P1* is in stage II and has surgery. Also, *P1* received treatment on *10.07.2020* with an effective therapeutic response. In that treatment, *P1* was treated with a combination of chemotherapy drugs and one non-oncological drug. Drug-Drug Interactions with the effect and the impact are reported.

The input $\mathcal{KG}$ in our use case contains 548 polypharmacy cancer treatments $\mathcal{T}$ extracted from lung cancer clinical records, with the therapeutic response from each of them and the known Drug-Drug Interactions. The therapeutic response is the target class and can be set to the value *low-effect* or *effective* treatment. The meaning of an *effective* treatment is because of a complete therapeutic response or stable disease. A *low-effect* treatment means a partial therapeutic response or disease progression. Figure 7 depicts a descriptive analysis of the treatment response according to the data extracted from the clinical records. Figure 7(a) shows the treatment response distribution, where there are 149 *effective* treatments and 399 *low-effect* treatments. Figure 7(b) and 7(c) present the histogram for the class *effective* and *low-effect*, respectively. We can observe that there are treatments with nine and ten drugs

[2] https://p4-lucat.eu/

Table 1

Summary of the lung cancer knowledge graph

Knowledge graph for lung cancer	Records
Lung Cancer Patients	1’242
Lung Cancer Drug	45
Chemotherapy Drug	7
Immunotherapy Drug	3
Antiangiogenic Drug	2
Tki Drug	5
Non Oncological Drug	41
Oncological Surgery	9
Tumor Stage	6
Publications	178’265
Drugs	8’453
Drug-Drug Interactions	1’550’586

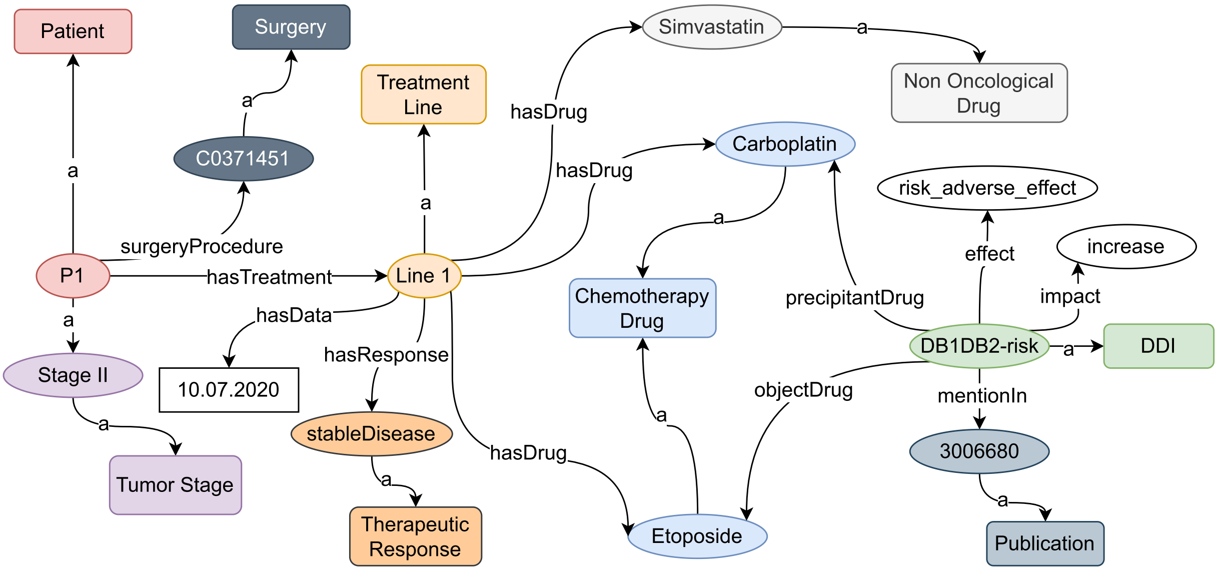
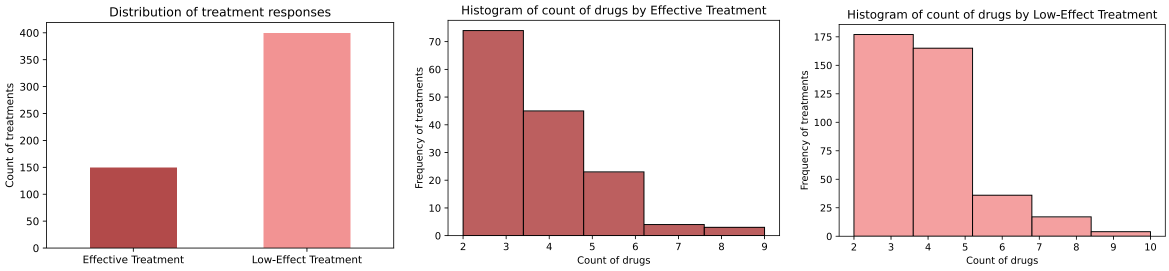


Fig. 6. **Representation of a patient in the lung cancer knowledge graph**.

in both treatments’ response classes. Also, the most *low-effect* treatments are composed of more drugs than *effective* treatments. The rate of drugs between five and ten can be explained by the fact that in patients with multiple comorbidities, multiple drugs are prescribed to treat the disease.

For each treatment, $t_i \in \mathcal{T}$, the DDIs and their effect are known from DrugBank [42]. Then, the treatments, the treatment response, the drugs, DDIs, and DDI effects for each treatment are managed in $\mathcal{KG}$. Figure 8 shows a portion of $\mathcal{KG}$. The node colors correspond to the type of entity, and the edges represent relationships amount the drugs grouped in a treatment. The polypharmacy treatment knowledge graph $\mathcal{KG} = (V, E, L, C, I, D, R, ego, \mathcal{N}, \alpha)$ is defined as follows:

- The types Drug, Treatment, DDI, Effect of DDI, and Treatment Response belong to *Classes*.
- Drugs, Treatments, DDIs, Effect of DDI, and Treatment Response are represented as instances of V.
- Edges in E that belong to $V \times V$ represent relations about drugs into a treatment.
- Properties *ex:has_response*, *part_of*, *ex:precipitant_drug*, *ex:object_drug*, *ex:effect*, *ex:impact*, and *ex:hasInteraction* correspond to labels in L.



(a) **Distribution of classes** (b) **Histogram Effective Treatment** (c) **Histogram Low-Effect Treatment**

Fig. 7. **Descriptive analysis of the treatment responses.**

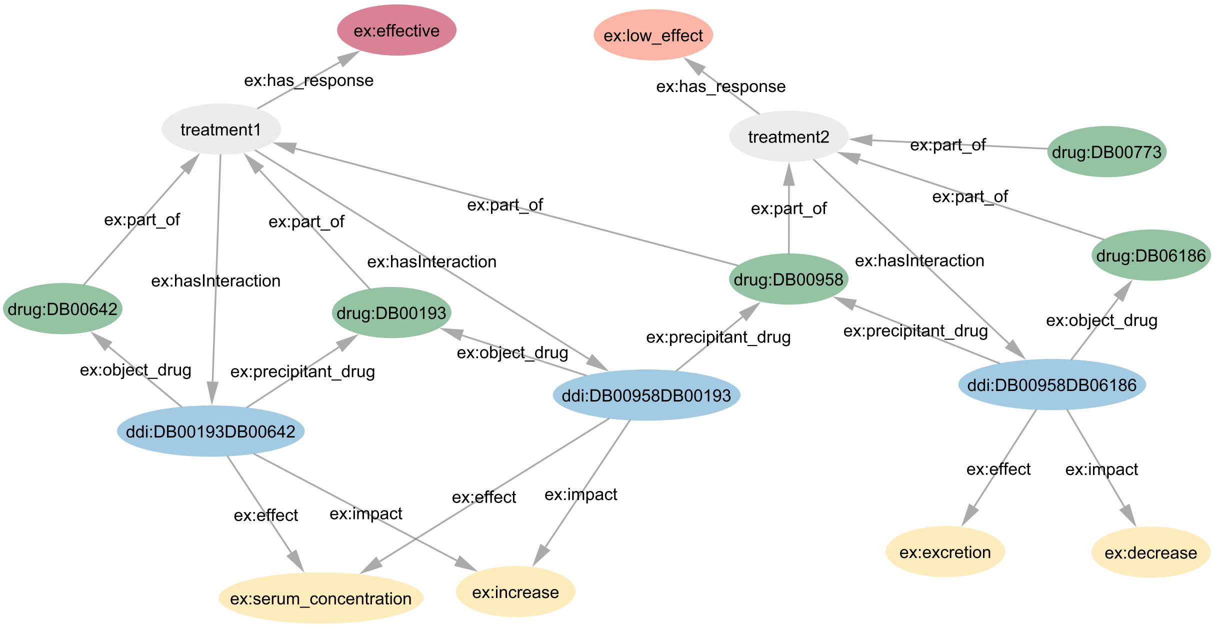


Fig. 8. **Portion of treatment knowledge graph** $\mathcal{KG}$. The *treatment1* is composed by three drugs represented by the nodes, *drug:DB00642*, *drug:DB00193*, and *drug:DB00958*. The *treatment2* also contains three drugs and shares *drug:DB00958* with *treatment1*. The blue node *ddi:DB00958DB06186* represents a DDI in the *treatment2* where the *drug:DB00958* is the precipitant, and *drug:DB06186* is the object drug. The effect of this DDI is represented by the yellow node *ex:excretion* and the impact by the node *ex:decrease*. Then, the treatment *treatment2* has a low effective response represented by the property *ex:has_response*.

4.2. Symbolic system. Deductive database

Let $\tau = \langle \mathcal{KG}, r, prediction, DS, KGE\rangle$ be the input abstract target prediction, where $\mathcal{KG}$ is the polypharmacy treatment knowledge graph, $r = has\_response$, $prediction = tail$, DS the deductive database system, and KGE the knowledge graph embedding algorithm. The IDB of the DS comprises a set of rules to deduce new DDIs in treatments. A DDI is deduced when a set of drugs are taken together and is represented as a relation in the minimal model of the deductive database DS. The extensional database corresponds to statements about interactions between drugs stated in $\mathcal{KG}$. The ground predicates included in the EDB are the following; they are extracted from the KG by executing SPARQL queries:

$rule_1(serum, increase).$
$rule_1(metabolism, decrease).$
$rule_1(absorption, increase).$
$rule_1(excretion, decrease).$

$rule_2(serum, decrease).$
$rule_2(metabolism, increase).$
$rule_2(absorption, decrease).$
$rule_2(excretion, increase).$

precipitant(DB00958DB06186, DB00958).
object(DB00958DB06186, DB06186).
effect(DB00958DB06186, excretion).
impact(DB00958DB06186, decrease).

SPARQL queries in Listing 3 and Listing 4 declaratively define the ground $rule_1$ and $rule_2$ in the EDB. Both queries are executed on top of the $\mathcal{KG}$; the CONSTRUCT query returns RDF triples in the form of subject, predicate and object. The predicate in the RDF triples represents the ground predicate in the EDB.

```
PREFIX tkg: <http://research.tib.eu/lung-cancer/vocab/>
PREFIX tkge: <http://research.tib.eu/lung-cancer/entity/>
CONSTRUCT {?E <rule1> ?I} WHERE {
        ?ddi a tkge:DDI .
        ?ddi tkg:effect ?E .
        ?ddi tkg:impact ?I .
        FILTER((?E in (tkge:serum, tkge:absorption) && ?I="increase") ||
               (?E in (tkge:metabolism, tkge:excreation) && ?I="decrease")) }
```

Listing 3. **SPARQL query to ground the extensional predicate** $rule_1(E, I)$

```
PREFIX tkg: <http://research.tib.eu/lung-cancer/vocab/>
PREFIX tkge: <http://research.tib.eu/lung-cancer/entity/>
CONSTRUCT {?E <rule2> ?I} WHERE {
        ?ddi a tkge:DDI .
        ?ddi tkg:effect ?E .
        ?ddi tkg:impact ?I .
        FILTER((?E in (tkge:serum, tkge:absorption) && ?I="decrease") ||
               (?E in (tkge:metabolism, tkge:excreation) && ?I="increase")) }
```

Listing 4. **SPARQL query to ground the extensional predicate** $rule_2(E, I)$

The facts included in the ground predicates *precipitant, object, effect*, and *impact* from the EDB are extracted using the CONSTRUCT query of Listing 5. The EDB contains thousands of facts for those predicates; therefore, only a few ground facts are presented.

```
PREFIX tkg: <http://research.tib.eu/lung-cancer/vocab/>
PREFIX tkge: <http://research.tib.eu/lung-cancer/entity/>
CONSTRUCT { ?ddi <precipitant> ?A .
            ?ddi <object> ?B .
            ?ddi <effect> ?E .
            ?ddi <impact> ?I } WHERE {
    ?ddi a tkge:DDI .
    ?ddi tkg:precipitant ?A .
    ?ddi tkg:object ?B .
    ?ddi tkg:effect ?E .
    ?ddi tkg:impact ?I }
```

Listing 5. **SPARQL query to extract the ground the extensional predicates** *precipitant(ddi,A), object(ddi,B), effect(ddi,E)*, and *impact(ddi,I)*

The above-mentioned $rule_1$ identifies the combinations of effect and impact that alter the toxicity of an object drug, while $rule_2$ determines the combinations of effect and impact that alter the effectiveness of an object drug. The predicates $rule_1$ and $rule_2$ represent the effect and impact of pharmacokinetic DDIs. The intensional database (a.k.a. *IDB*) comprises Horn rules that state when a new DDI can be deduced as a result of the combination of the treatment's drug. These rules are negation free; thus, the interpretation of the deductive database corresponds to the minimal model of the *EDB* and *IDB*. The intensional database relies on the fact that pharmacokinetic DDIs cause the concentration of one of the interacting drugs (a.k.a. object) to be altered when combined with the other drug (a.k.a. precipitant). Thus, the absorption, distribution, metabolism, or excretion rate of the object drug is affected. Whenever the object drug absorption is decreased (resp. increased), the bioavailability of the drug is also affected. Furthermore, any alteration in the metabolism or excretion of the object drug has consequences on the therapeutic efficacy and toxicity of the drug. The following Datalog rules state the effect of pharmacokinetic DDIs:

$$\begin{aligned}&\mathit{precipitant(ID,\ A),\ object(ID,\ B),\ effect(ID,\ E),\ impact(ID,\ I)}\\&\quad\Rightarrow \mathit{ddi}(A, E, I, B).\end{aligned}\tag{2}$$

$$\begin{aligned}&\mathit{ddi}(A, E, I, B)\\&\quad\Rightarrow \mathit{inferred\_ddi}(A, E, I, B).\end{aligned}\tag{3}$$

$$\begin{aligned}&\mathit{inferred\_ddi}(A, E_2, I_2, B), \mathit{ddi}(B, E, I, C), \mathit{rule}_1(E, I), \mathit{rule}_1(E_2, I_2), (A! = C)\\&\quad\Rightarrow \mathit{inferred\_ddi}(A, E, I, C).\end{aligned}\tag{4}$$

$$\begin{aligned}&\mathit{inferred\_ddi}(A, E2, I2, B), \mathit{ddi}(B, E, I, C), \mathit{rule}_2(E, I), \mathit{rule}_2(E_2, I_2), (A! = C)\\&\quad\Rightarrow \mathit{inferred\_ddi}(A, E, I, C).\end{aligned}\tag{5}$$

Rule (3) states the base case of the *IDB*. The predicate symbol *ddi* represents the DDIs with their effect and impact in $\mathcal{KG}$. Precipitant drug A generates effect E (e.g., absorption, excretion, metabolism, serum concentration) with impact I (e.g., increase or decrease) in object drug B. The predicate symbol *inferred_ddi* expresses a deduced DDI, where the first term is the precipitant drug, the second and third terms represent the value of the property effect and impact of the DDIs deduced, and the last term is the object drug. Rule (4) and (5) define the effects of combining drugs that interact in a polypharmacy treatment and comprises the clauses to deduce relationships encoded in $\mathcal{KG}$. The head predicate *inferred_ddi* becomes valid when the predicate symbols in the body of the rule are also valid. The DDIs deduced from the Rule (4) increase the toxicity of the object drug, and the DDIs deduced from Rule (5) alter the effectiveness of the object drug. Those deduced DDIs are aggregated to the $\mathcal{KG}$; they represent valuable insights into each treatment. Each DDI deduced, which is part of the minimal model of the *IDB* predicate *inferred_ddi(A,E,I,C)*, is inserted into the $\mathcal{KG}$ using the query shown in Listing 6. From the motivating example, we can observe that by applying the DDI deductive system to the treatment `T1` in Fig. 3(a), a new DDI is deduced in Fig. 3(b); it represents a new triple enhancing the treatment information, reducing thus, data sparsity.

```
PREFIX tkg: <http://research.tib.eu/lung-cancer/vocab/>
INSERT DATA {
        <ddi> tkg:precipitant <A> .
        <ddi> tkg:object <C> .
        <ddi> tkg:effect <E> .
        <ddi> tkg:impact <I>
        }
```

Listing 6. **SPARQL query to insert the deduced DDI from the intensional predicate** *inferred_ddi(A,E,I,C)*

4.3. Sub-symbolic system. Knowledge graph embedding model

Once the deductive system *DS* deduces new DDIs, the Knowledge Graph Embedding algorithm *KGE* is applied to learn a latent representation of the entities in a low-dimensional space. The *DS* increases the relationships in the ego networks $ego(v)$ such as $I(v) \in \{C_1, C_2\}$, $C_1 = D(\mathit{has\_response})$, and $C_2 = R(\mathit{has\_response})$. The *DS* minimizes the data sparsity issues by augmenting the description of the treatments with newly deduced DDIs. Then, *KGE* is able to improve the entities' representation in the embedding space. Thus, the scoring function $\phi(h, r, t)$ of the *KGE* is improved, and the link prediction task infers missing links that correspond to triples $\langle h, r, ?\rangle$, where $I(h) = D(r)$ and $r = \mathit{has\_response}$. Thus, h are entities of class *Treatment*, and entities in the object position t are of class *Response*. Symbolic and sub-symbolic systems are highly complementary to each other. Sub-symbolic AI

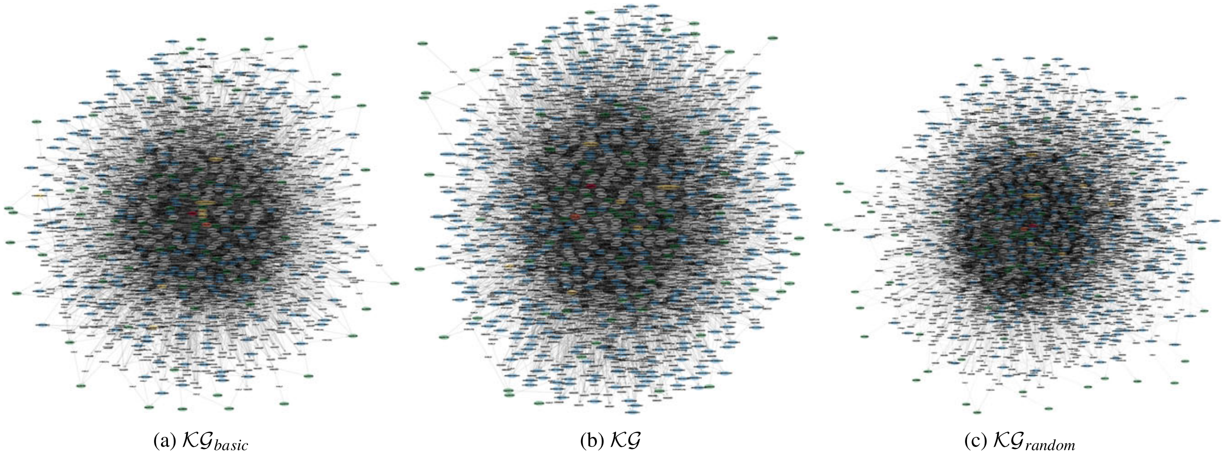

(a) $\mathcal{KG}_{basic}$ (b) $\mathcal{KG}$ (c) $\mathcal{KG}_{random}$

Fig. 9. **Benchmarks to evaluate.** Fig. 9(a) represents the $\mathcal{KG}_{basic}$, and it includes treatments from clinical records and pharmacokinetic DDI extracted from Drugbank. Figure 9(b) represents the $\mathcal{KG}$ and it includes treatments from clinical records, pharmacokinetic DDI extracted from Drugbank, and a new set of pharmacokinetic DDI deduced by the DDI Deductive System. Figure 9(c) represents the $\mathcal{KG}_{random}$ and includes treatments from clinical records, pharmacokinetic DDI extracted from Drugbank, and the same number of new links deduced in $\mathcal{KG}$ is generated randomly.

systems are able to solve complex problems that humans cannot analyze to draw conclusions or make predictions. Sub-symbolic methods are generally robust to data noise, while symbolic systems are vulnerable to data noise, which contrasts with the strength of sub-symbolic approaches.

5. Experimental study

We empirically assess the impact of the DDIs encoded in $\mathcal{KG}$ on our approach's behavior. In particular, this work explores the following research questions: **RQ1** Can the problem of predicting treatment effectiveness be effectively modeled as a problem of link prediction? **RQ2** Can the symbolic system for an abstract target prediction improve the link prediction performance of the KGE models? **RQ3** Can knowledge encoded in drug-drug interactions enhance the accuracy of the predictive task?

5.1. Experiment setup

We empirically evaluate the effectiveness of our approach to capture knowledge encoded in $\mathcal{KG}$ and predict polypharmacy treatment response.

5.1.1. Benchmarks

We conduct our evaluation over three Knowledge Graphs represented in Fig. 9. $\mathcal{KG}_{basic}$ is the Knowledge Graph which only contains for each polypharmacy treatment the DDIs and their effect extracted from Drugbank. The second Knowledge Graph, $\mathcal{KG}$, includes not only the DDIs extracted from DrugBank, but also the ones deduced by Deductive Database *DS*, i.e., it contains new deduced DDIs and their effects. Lastly, the third Knowledge Graph, $\mathcal{KG}_{random}$ is created from $\mathcal{KG}_{basic}$; it also includes the same number of links included in $\mathcal{KG}$ but these links are randomly generated, i.e., they correspond to false or true relationships. We **aim** to validate whether the links discovered by our DDI Deductive System improve the prediction of treatment responses.

5.1.2. Knowledge graph embedding models

We utilize eleven models to compute latent representations, e.g., vectors, of entities and relations in the three KGs and then employ them to infer new facts. In particular, we utilize three main families of models:

- Tensor Decomposition models such as *HolE* and *RESCAL*.

Table 2

Scoring function and complexity of embedding models. Adapted from [31]

Embedding model	Scoring function	Complexity
HolE	$r(h \star t)$	$\mathcal{O}(\lvert E\rvert d + \lvert\mathcal{R}\rvert d)$
RESCAL	$h^T W_r t = \sum_{i=1}^{d} \sum_{i=1}^{d} w_{ij}^{(r)} h_i t_j$	$\mathcal{O}(\lvert E\rvert d + \lvert\mathcal{R}\rvert d^2)$
RotatE	$-\lVert h \circ r - t \rVert$	$\mathcal{O}(\lvert E\rvert d + \lvert\mathcal{R}\rvert d)$
QuatE	$t(h \otimes r)$	$\mathcal{O}(\lvert E\rvert d + \lvert\mathcal{R}\rvert d)$
TransE	$\lVert h + r - t \rVert$	$\mathcal{O}(\lvert E\rvert d + \lvert\mathcal{R}\rvert d)$
TransH	$\lVert h_\perp + d_r - t_\perp \rVert$	$\mathcal{O}(\lvert E\rvert d + 2\lvert\mathcal{R}\rvert d)$
TransD	$\lVert h_\perp + r - t_\perp \rVert$	$\mathcal{O}(2\lvert\mathcal{E}\rvert d + 2\lvert\mathcal{R}\rvert d)$
TransR	$\lVert h_r + r - t_r \rVert$	$\mathcal{O}(\lvert E\rvert d + \lvert\mathcal{R}\rvert d^2)$
UM	$\lVert h - t \rVert$	$\mathcal{O}(\lvert E\rvert d)$
SE	$\lVert M_{r,1} h - M_{r,2} t \rVert$	$\mathcal{O}(\lvert E\rvert d + 2\lvert\mathcal{R}\rvert d^2)$
ERMLP	$w^T g(W[h; r; t])$	$\mathcal{O}(\lvert E\rvert d + \lvert\mathcal{R}\rvert d + k(3d + 2) + 1$

- Geometric models such as *RotatE*, *QuatE*, and the Trans* family models *TransE*, *TransH*, *TransD*, *TransR*.
- Deep Learning models such as *UM*, *SE* and *ERMLP*.

The symbolic-sub-symbolic system proposed is implemented in eleven embedding models from different families [31]. Holographic embeddings (*HolE*) [23] computes circular correlation, denoted by $\star$ in Table 2, between the embeddings of head and tail entities. *RESCAL* [24] is an algorithm of relational learning based on tensor factorization, where it models entities as vectors and relations as matrices. In *RESCAL*, the relation matrices W_r contain weights $w_{i,j}$ between the i-th factor of h and j-th factor of t. *RotatE* [37] represents each relation as a rotation from the head entity to the tail entity in the complex latent space. The rotation r is applied to h by operating a Hadamard product (denoted by $\circ$ in Table 2). *QuatE* [44] operates on the quaternion space and learns hypercomplex valued embeddings (quaternion embeddings) to represent entities and relations, and $\otimes$ represents the Hamilton product. *TransE* [5] proposes a geometric interpretation of the latent space and interprets relation vectors as translations in vector space, $h + r \approx t$. *TransE* can not naturally model 1-n, n-1 and n-m relationships. Suppose a relation r with cardinality 1-n, (h, r, t_1), (h, r, t_2) then the model fits the embeddings in order to ensure $h + r \approx t_1$ and $h + r \approx t_2$, i.e. $t_1 \approx t_2$. Translation on hyperplanes (*TransH*) [41] is an extension of *TransE* that aims to overcome the limitations of *TransE*. *TransH* interprets a relation as a translating operation on a hyperplane. Furthermore, each relation r is represented by the hyperplane's norm vector (w_r) and the translation vector (d_r) on the hyperplane. The variables $h_\perp$ and $t_\perp$ denote a projection of head vector h and tail vector t to the hyperplane w_r. *TransR* [21] represents entities and relations in distinct vector spaces and learns embeddings by translation between projected entities. $h_r = h * M_r$ where M_r corresponds to a projection matrix $M_r \in \mathbb{R}^{dxk}$ that projects entities from the entity space to the relation space; further $r \in \mathbb{R}^k$. *TransD* [17] employs separate projection vectors for each entity and relation. In the score function of *TransD*, the variables $h_\perp$ and $t_\perp$ are defined as, $h_\perp = M_{rh} h$ and $t_\perp = M_{rt} t$, where M_{rh}, $M_{rt} \in \mathbb{R}^{m \times n}$ are two mapping matrices defined as follows: $M_{rh} = r_p h_p + I^{m \times n}$ and $M_{rt} = r_p t_p + I^{m \times n}$. The subscript p means the projection vectors, and $I^{m \times n}$ denotes the identity matrix of size $m \times n$. The Unstructured Model (*UM*) [4] is a simplified version of *TransE* where it does not consider differences in relations and only models entities as embeddings. This model can be beneficial in KGs containing only a single relationship type. The Structured Embedding (*SE*) [6] model defines two matrices $M_{r,1}$ and $M_{r,2}$ to project head and tail entities for each relation. SE can discern between the subject and object roles of an entity since it employs different projections for the embeddings of the head and tail entities. *ERMLP* [10] is a model based on a multi-layer perceptron and uses a single hidden layer. In the score function, the variable $W \in \mathbb{R}^{k \times 3d}$ represents the weight matrix of the hidden layer, the variable $w \in \mathbb{R}^k$ represents the weights of the output layer, and g is the activation function. In Table 2, the variable k corresponds to the number of neurons in the hidden layer.

The PyKEEN (Python KnowlEdge EmbeddiNgs) framework [2] is used to learn the embeddings. The hyperparameters utilized to train the model are epoch number 200 and training loops: stochastic local closed world assumption (sLCWA). The negative sampling techniques used are Uniform negative sampling and Bernoulli negative sampling. The embedding dimensions and the rest of the parameters are set by default. To assure statistical

Table 3

Statistics of knowledge graph. Metrics to measure size, diversity, and sparsity in knowledge graph

KG	T	E	R	RE	EE	RD	ED
$\mathcal{KG}_{basic}$	5630	1069	7	1.615	10.846	804.286	10.533
$\mathcal{KG}$	6675	1069	7	1.726	10.989	953.571	12.488
$\mathcal{KG}_{random}$	6675	1069	7	1.710	11.291	953.571	12.488

robustness, we apply 5-fold cross-validation. For evaluating the performance of embeddings methods, we measure the metrics: $Precision = \frac{TP}{TP+FP}$, $Recall = \frac{TP}{TP+FN}$, $F1\text{-}Score = \frac{2*(precision*recall)}{(precision+recall)}$.

5.1.3. Implementations

The pipeline for predicting polypharmacy treatment response has been implemented in Python 3.9. Experiments were executed using 12 CPUs Intel® Xeon(R) W-2133 at 3.60 GHz, 64 GB RAM, and 1 GPU GeForce GTX 1080 Ti/PCIe/SSE2 with 12 GB VRAM. We used the library pyDatalog[3] to develop the Deductive System and the library PyKEEN,[4] to learn the embeddings.

5.2. Metrics to characterize the benchmarks

Table 3 shows the statistics of the three KGs. We considered the metrics, Number of Triples (T), Entities (E), and Relations (R), to measure the size in KG. The metrics Relation entropy (RE) and Entity entropy (EE) are considered to measure diversity and Relational density (RD) and Entity density (ED) to measure sparsity in the KG.

The metrics RE and EE measure the distribution of relationships and entities in the KG, respectively. Higher values of RE mean that all possible relations are equally probable, and lower values mean one or more relations have a high probability. The values of the metric RE mean that all possible relations in $\mathcal{KG}$ are more equally probable than all possible relations in $\mathcal{KG}_{basic}$ and $\mathcal{KG}_{random}$. The three KGs have a higher EE value than RE as they use a small set of manually defined relations but contain many entities. The metrics RD and ED measure the sparsity of entities and relationships in the KG, respectively. We measure sparsity as information density, where RD means average triples per relation and ED is the average triples per entity. $\mathcal{KG}_{basic}$ has the lower average triples per relation and entity, while $\mathcal{KG}$ and $\mathcal{KG}_{random}$ have the higher average triples per entity. The metrics evaluated in Table 3 are defined in the paper [25], implemented by us and available at the GitHub repository.[5]

5.3. Impact of capturing symbolic knowledge

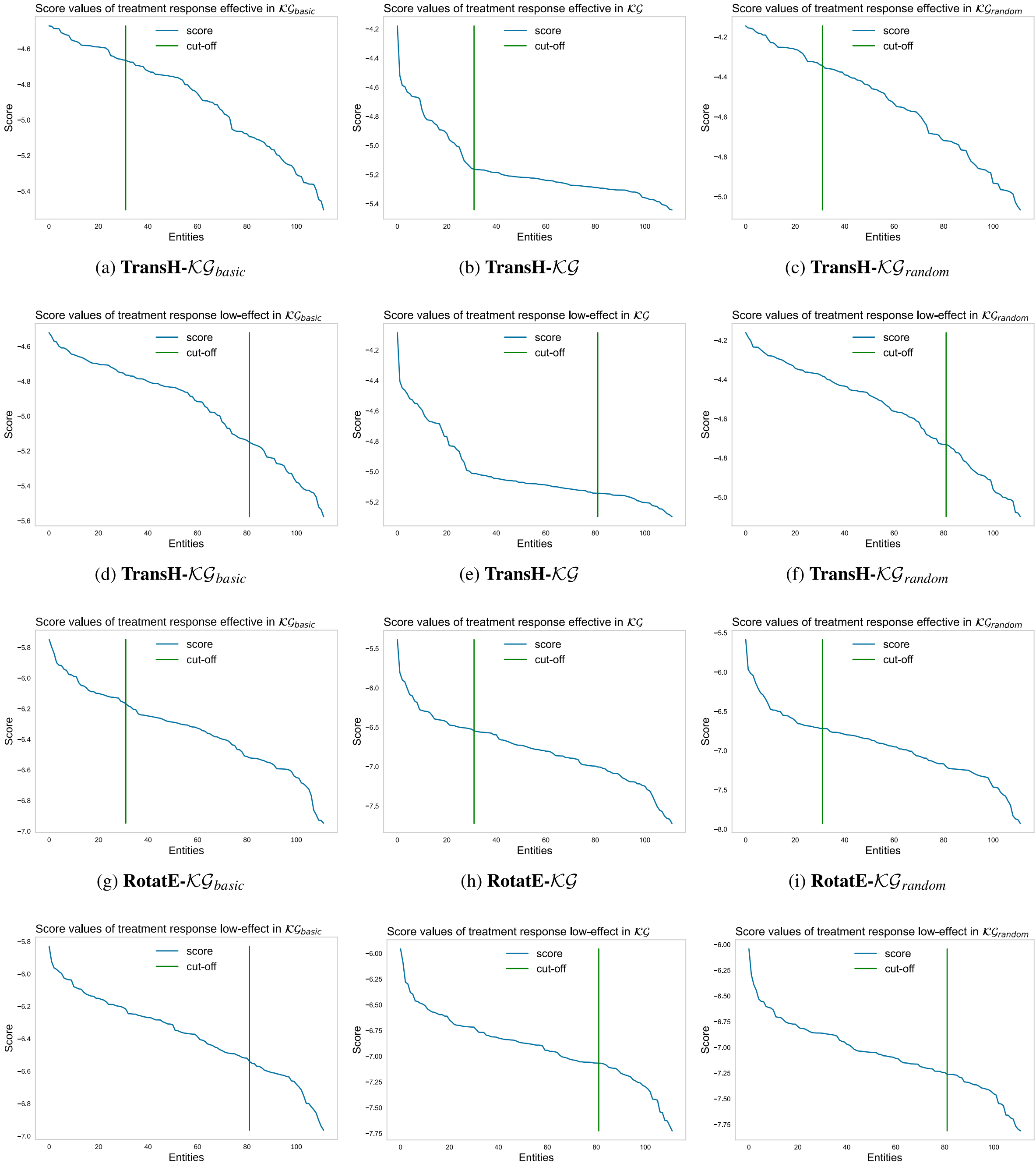
Figure 10 shows the behavior of the scoring function for the entities predicted by *TransH* and *RotatE* embedding models. For the purpose of brevity, we only show the score value results for two embedding models. The evaluation material is available .[6] We can notice how *DS* for the prediction property $r = has\_response$ is impacting the KGE models. Figure 10(a) to 10(c) and Fig. 10(g) to 10(i) show the score values of the entities predicted on the link prediction task given the predicate *ex:has_response* and object *effective* by the *TransH* and *RotatE* models, respectively. Figure 10(d) to 10(f) and Fig. 10(j) to 10(l) report on the score values of the entities predicted given the predicate *ex:has_response* and object *low-effect* by the *TransH* and *RotatE* models, respectively. We can observe that the models have different behaviors for each KG. The vertical line in each plot represents the cut-off in a specific percentile. The percentile used for each KG was based on the percentage of links to the entity *effective* and *low-effect* in the KG, e.g., the percentile for the effective treatments is 27, because the amount of links to treatment response (*effective* and *low-effect*) is 548 and 149 are effective treatments ($100 * 149/548$). The portion of entities predicted, delimited by the vertical line, is evaluated in terms of precision, recall, and f1-score.

[3] https://sites.google.com/site/pydatalog/home

[4] https://pykeen.readthedocs.io/en/stable/index.html

[5] https://github.com/SDM-TIB/Statistics_KnowledgeGraph

[6] https://github.com/arivasm/Neuro-Symbolic_Treatment-Response.git



(a) **TransH-$\mathcal{KG}_{basic}$** (b) **TransH-$\mathcal{KG}$** (c) **TransH-$\mathcal{KG}_{random}$**

(d) **TransH-$\mathcal{KG}_{basic}$** (e) **TransH-$\mathcal{KG}$** (f) **TransH-$\mathcal{KG}_{random}$**

(g) **RotatE-$\mathcal{KG}_{basic}$** (h) **RotatE-$\mathcal{KG}$** (i) **RotatE-$\mathcal{KG}_{random}$**

(j) **RotatE-$\mathcal{KG}_{basic}$** (k) **RotatE-$\mathcal{KG}$** (l) **RotatE-$\mathcal{KG}_{random}$**

Fig. 10. **Score value of the predicted entities.** The green line represents the cut-off at the percentile 27 for *effective* treatments and 73 for *low-effect* treatments for the three KGs.

5.4. Evaluating the performance of our integrated symbolic-sub-symbolic system

The selected portions of entities predicted are measured precision, recall, and f1-score on average because of cross-validation. Figure 11 and Fig. 12 show the evaluation of the Link Prediction task through Uniform negative

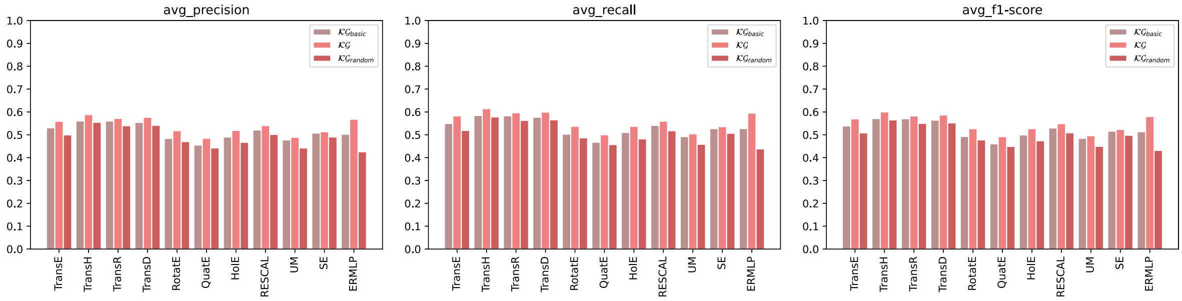


Fig. 11. Evaluation of the link prediction task in terms of precision, recall, and f-measure. Utilizing uniform negative sampling.

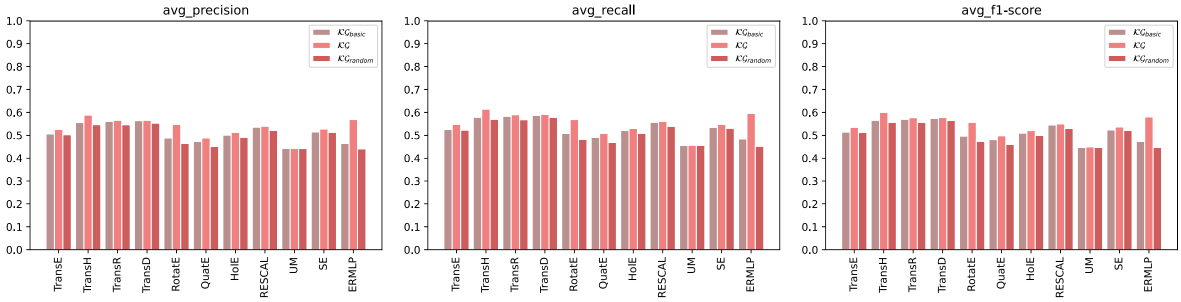


Fig. 12. Evaluation of the link prediction task in terms of precision, recall, and f-measure. Utilizing Bernoulli negative sampling.

sampling and Bernoulli negative sampling, respectively. Uniform sampling randomly chooses the candidate entity based on a uniform probability between all possible entities. Bernoulli sampling corrupts the head with probability p and the tail with $1 - p$, where p is an average number of unique tail entities per unique head entities given a relation r. The relation with cardinally *1-n* has a higher probability of corrupting the head, and relations *n-1* have a higher probability of corrupting the tail. Figure 11 and 12 show the results of the three benchmarks. Each plot depicts the results of a metric for each embedding model and KG. The best performing embedding model in the three metrics is *TransH*. The KGE models have all better performance in $\mathcal{KG}$ regarding the three metrics evaluated in both negative sampling techniques. In addition, the worst performance is observed in $\mathcal{KG}_{random}$. The *DS* minimizes the data sparsity issue with meaningful relationships and enhances the predictive performance of KGE models. These results suggest that the deduced DDIs by the Deductive System *DS* are meaningful to the treatment responses. More importantly, they put the crucial role of the deduced relations into perspective.

5.5. *Discussion*

The techniques proposed in this paper rely on known relations between entities to predict novel links in the KG. During the experimental study, we observed that these techniques could improve the prediction of treatment effectiveness. Figure 13 shows a box plot of cosine similarity. Considering the KGE model with better performance *TransH*, we computed the cosine similarity between the embedding entities of type treatment. Five treatments with a low-effect response are selected, and *TransH* in $\mathcal{KG}_{basic}$ misclassifies them, but *TransH* in $\mathcal{KG}$ predicts them correctly. Next, all the treatments with a low-effect response are selected. Thus, the cosine similarity is computed between the selected treatment and the list of treatments with the same response. The first box represents the result of *TransH* in $\mathcal{KG}_{basic}$; this box contains the similarity values between the treatment *treatment*355 y the list of treatment with the same response that *treatment*355. The second box depicts the result of *TransH* in $\mathcal{KG}$ for the same treatment in the first box. We can observe that the five treatments are more similar to the list of treatments in $\mathcal{KG}$ than in $\mathcal{KG}_{basic}$. The first quartile, median, and third quartile values in the boxplot are higher in $\mathcal{KG}$ than in $\mathcal{KG}_{basic}$. Therefore, these outcomes put in evidence the quality of the deduced links in $\mathcal{KG}$ and their impact on the accuracy of the KGE models in the resolution of the task of predicting treatment effectiveness.

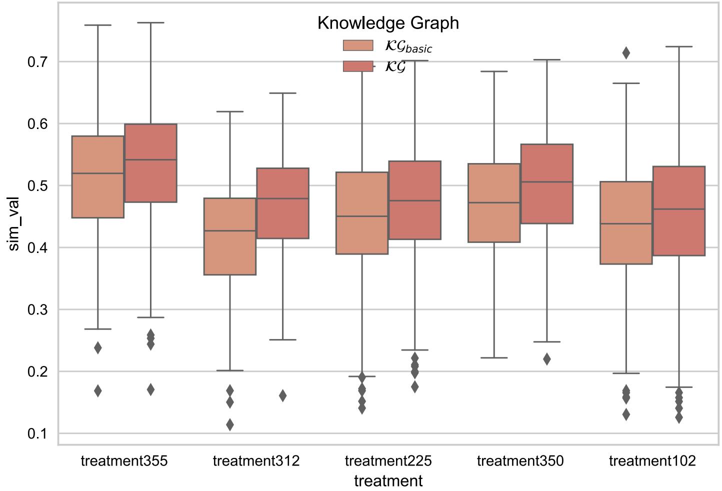


Fig. 13. **Boxplot of cosine similarity**. The boxplot illustrates the distribution of cosine similarity values between treatments in x-axe with a list of treatments. We observe the five treatments in the x-axe are more similar to the treatments in $\mathcal{KG}$ than in $\mathcal{KG}_{basic}$.

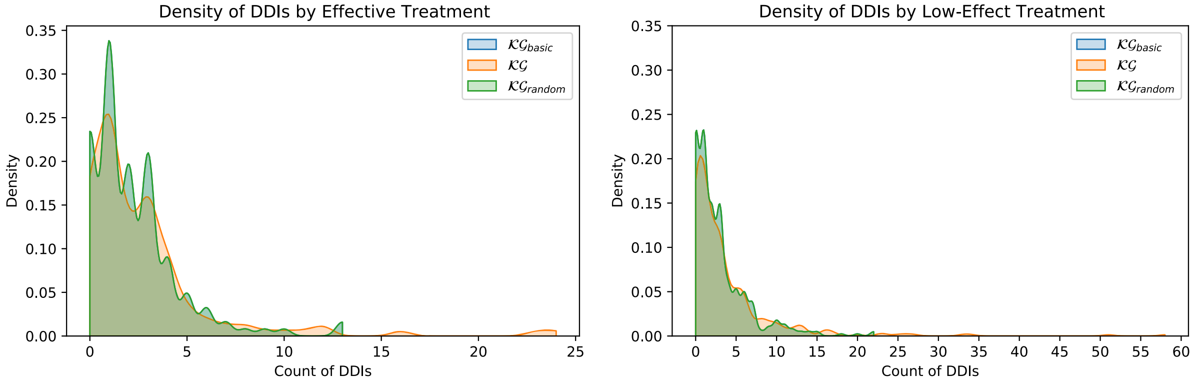


Fig. 14. **The distribution of DDIs by treatment for each KG**. Figure 14(a) shows the density of treatments by DDIs for the treatment response *effective* in $\mathcal{KG}_{basic}$, $\mathcal{KG}$, and $\mathcal{KG}_{random}$. Figure 14(b) shows the density of treatments by DDIs for the treatment response *low-effect*.

Figure 14 shows the distribution of DDIs by treatment in $\mathcal{KG}_{basic}$, $\mathcal{KG}$, and $\mathcal{KG}_{random}$. The x-axis represents the count of DDIs in treatment, and the y-axis represents the density of treatments in the KG with a specific x value. We utilized the Kernel Density Estimation (KDE) function to compute the probability density of the count of DDIs in each KG. We can observe for both treatment response *effective* and *low-effect* that $\mathcal{KG}$ have less density for treatments with five or fewer DDIs than the other two KGs and more density for treatments with more than five DDIs than the rest of the KGs. Furthermore, most treatments with *effective* response contain less than five DDIs while treatments with *low-effect* response contain more than five DDIs. These outcomes put into evidence the crucial role that implicit DDIs have on a treatment's response and the need to deduce them using symbolic systems.

Analysis of deduced DDI by Treatment classes: Fig. 15 exhibits the distribution of DDIs by treatment response in both $\mathcal{KG}_{basic}$ and $\mathcal{KG}$. The DDI Deductive System deduces new DDIs in 23.1% of treatments with *low-effect* responses, while only 10.7% of treatments with *effective* responses deduce new DDIs. This analysis indicates that the DDI Deductive System deduces more than twice the number of DDIs in *low-effect* response treatments than in *effective* response treatments.

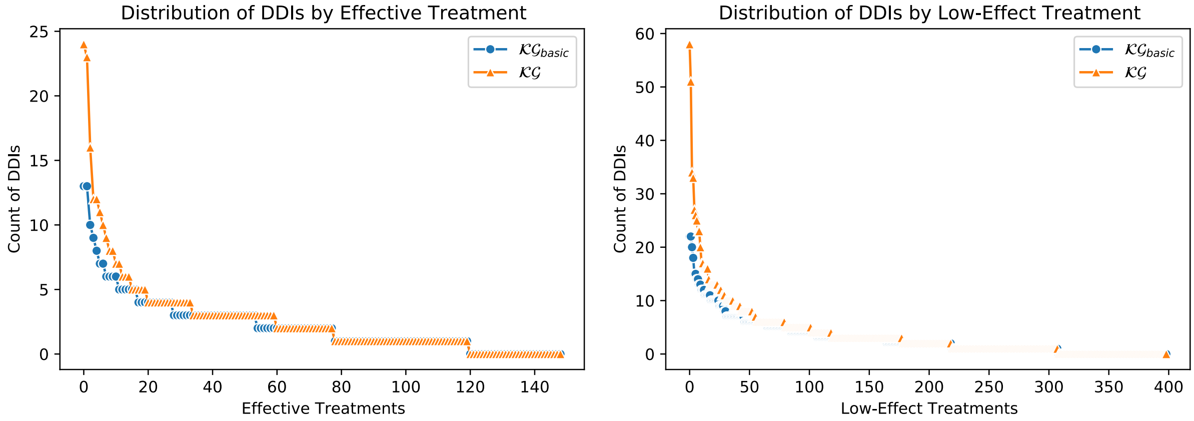


Fig. 15. Distribution of DDIs by treatment response.

6. Related work

6.1. Neuro-symbolic artificial intelligence

Neuro-Symbolic Artificial Intelligence is a highly active area that has been studied for decades [8]. Neuro-symbolic AI focuses on integrating symbolic and sub-symbolic systems. Several approaches employ translation algorithms from a symbolic representation to a sub-symbolic representation and vice versa [3]. The aim is to provide a neuro-symbolic implementation of logic, a logical characterization of a neuro-system, or a hybrid learning system that contributes features of symbolic and sub-symbolic systems [8,38]. Real applications are possible in areas with social relevance and high economic impacts, such as bioinformatics, robotics, fraud prevention, and the semantic web [3]. Methods utilized in neuro-symbolic integration in some of the aforementioned applications include translation algorithms between logic and networks. Also, the community has focused on studying the systems empirically through case studies and real-world applications. An example of a neuro-symbolic system in the field of bioinformatics is the Connectionist Inductive Learning and Logic Programming (CILP) [9]. In the field of vision-based tasks, such as semantic image labeling, high-performance systems have been produced. Karpathy et al. [19] propose an approach introduced for the recognition and labeling tasks for the content of different regions of the images; it combines Convolutional Neural Networks over the image regions together with bidirectional Recurrent Neural Networks over sentences. Once this mapping of images and sentences in the embedding space has been established, a structured objective is introduced that aligns the two modalities through multimodal embedding. The emerging system performs better than classical approaches, where tasks involving semantic descriptions are associated with databases that contain background knowledge, and computer image processing approaches are based on rule-based techniques.

Despite the progress of Neuro-Symbolic Artificial Intelligence, the scope and applicability of symbol processing are limited. Furthermore, these systems do not examine polynomial overload when integrating both paradigms. Our work leverages the symbolic system, independent of the application domain, and improves the predictive precision of KGE models. Moreover, in our approach, the deductive database is addressed to an abstract target prediction which renders the computational complexity polynomial-time. Thus, we show the positive impact on the overall performance of a predictive model implemented using KGEs considering a deductive system.

6.2. Knowledge graph embedding in biomedical field

Knowledge graphs are becoming increasingly important in the biomedical field. Discovering new and reliable facts from existing knowledge using KGE is a cutting-edge method. KG allows a variety of additional information to be added to aid reasoning and obtain better predictions.

Zhu et al. [46] develop a process for constructing and reasoning multimodal Specific Disease Knowledge Graphs (SDKG). SDKG is based on five cancers and six non-cancer diseases. The principal purpose is to discover reliable knowledge and provide a pre-trained universal model in that specific disease field. The model is built in three parts: structure embedding (S) with TransE, TransD, and ConvKB, category embedding (C), and description embedding (D) with BioBERT to convert description annotations into vectors. The best results are obtained when description embedding is combined with structure embedding, specifically with the ConvKB embedding model. Karim et al. [18] propose a new machine-learning approach for predicting DDIs based on multiple data sources. They integrated drug-related information such as diseases, pathways, proteins, enzymes, and chemical structures from different sources into a KG. Then different embedding techniques are used to create a dense vector representation for each entity in the KG. These representations are introduced in traditional machine learning classifiers and a neural network architecture based on a convolutional LSTM (Conv-LSTM), which was modified to predict DDIs. The results show that the combination of KGE and Conv-LSTM performs state-of-the-art results.

The above-mentioned research aims to discover reliable knowledge based on knowledge graphs using KGE models. However, they are limited by the data sparsity issue of the KGE models and the lack of symbolic reasoning. We overcome this limitation by integrating a Neuro-Symbolic AI system, enabling expressive reasoning and robust learning to improve the predictive performance of KGE models.

6.3. Polypharmacy side effect prediction and drug-drug interactions prediction

In recent years, there has been a growing interest in Pharmacovigilance. Extensive research has been conducted to predict potential DDI. One approach to predicting potential DDI is based on similarity [12,36,40,43], with the core idea of predicting the existence of a DDI by comparing candidate drug pairs with known interacting drug pairs. These approaches define a wide variety of drug similarity measures for comparison. The known DDIs that are very similar to a candidate pair provide evidence for the presence of a DDI between the candidate pair drugs. Sridhar et al. [36] propose a probabilistic approach for inferring unknown DDIs from a network of multiple drug-based similarities and known DDIs. They used the probabilistic programming framework Probabilistic Soft Logic. This symbolic approach predicts three types of interactions [36], CYP-related interactions (CRDs), where both drugs are metabolized by the same CYP enzyme, NCRDs, where no CYP is shared between the drugs and general DDI from Drugbank. Furthermore, they considered seven drug-drug similarities. Thus, they found five novel DDIs validated by external sources. A framework to predict DDIs is presented in [12]; they exploit information from multiple linked data sources to create various drug similarity measures. Then, they build a large-scale and distributed linear regression learning model to predict DDIs. They evaluate their model to predict the existence of drug interactions, considering the DDIs as symmetric. A neural network-based method for drug-drug interaction prediction is proposed in [30]. They use various drug data sources in order to compute multiple drug similarities. They computed drug similarity based on drug substructure, target, side effect, off-label side effect, pathway, transporter, and indication data. The proposed method first performs similarity selection and then integrates the selected similarities with a nonlinear similarity fusion method to obtain high-level features. Thus, they represent each drug by a feature vector and are used as input to the neural network to predict DDIs.

Other approaches focus on predicting DDIs and their effects [20,22,32,47]. Beyond knowing that a pair of drugs interact, it is essential to know the effect of DDI in polypharmacy treatments. In [20], propose a novel deep learning model to predict DDIs and their effects. They use additional features based on structural similarity profiles (SSP), Gene Ontology term similarity profiles (GSP), and target gene similarity profiles (TSP) to increase the classification accuracy. The proposed model uses an auto-encoder to reduce the dimension of the resulting vector from the combination of SSP, TSP, and GSP. The benchmark used has 1597 drugs and 188’258 DDIs with 106 different types. The model works as a multi-label classification model where the deep feed-forward network has an output layer of size 106, representing the number of DDI types. The results show that the model obtains equal or better results in 101 out of 106 DDI types than baseline methods. Also, they demonstrate how adding the features GSP and TSP increases the accuracy of DDIs prediction. Marinka Zitnik et al. [47] present Decagon, an approach for predicting the side effects of drug pairs. The approach develops a new convolutional graph neural network for link prediction. They construct a multi-modal graph of protein-protein interactions, drug-protein target interactions, and the DDI side effects. The graph encoder model produces embeddings for each node in the graph. They proposed a new model that assigns separate processing channels for each relation type and returns an embedding for each node in the graph. Then, the Decagon decoder for polypharmacy side effects relation types takes pairs of embeddings and produces a score. Thus, Decagon can predict the side effect of a pair of drugs.

All the approaches mentioned above are limited to predicting DDIs and their effects between pairs of drugs. However, in our view, the interactions and their effects need to be considered as a whole and not in pairs in polypharmacy treatments. Our symbolic system resorts to a set of rules that state the implicit definition of new DDIs generated as a result of the combination of multiple drugs in treatment. Since cancer treatment schemes are usually composed of more than one drug, and patients may have several co-existing diseases requiring additional medications, it is of significant relevance to the deduction of DDIs holistically in a given treatment.

7. Conclusions and future work

This paper addresses the problem of Neuro-Symbolic AI integration, enabling expressive reasoning and robust learning to discover relationships over knowledge graphs. We have presented an approach that integrates symbolic-sub-symbolic systems to enhance the predictive performance of abstract target prediction in KGE models. The symbolic system is implemented by a deductive database defined for an abstract target prediction over a KG. The proposed solution builds the ego networks of the head and tail of the abstract target prediction to deduce new relationships in the ego network; it is able to enhance the ego networks of the abstract target prediction and effectively predict treatment effectiveness. Further, the sub-symbolic system implemented by a KGE model enhances the predictive performance of the abstract target prediction and completes the KG. The performance of the proposed approach is assessed in a knowledge graph for lung cancer to discover treatment effectiveness. Predicting treatment effectiveness is effectively modeled as a link prediction problem, and exploiting DDI Deductive System improves existing embedding models by performing the treatment prediction task. Results of a 5-fold cross-validation process demonstrate that our approach, integrating neuro-symbolic systems, improves the eleven KGE models evaluated. The presented approach using the symbolic system's reasoning can enhance the ego networks of the abstract target prediction and effectively predict treatment effectiveness. Thus, our work broadens the repertoire of Neuro-Symbolic AI systems for discovering relationships over a KG. As for future work, we envision having a more fine-grained description of the DDIs and a descriptive profile of the patients and improving the model.

Acknowledgements

Ariam Rivas is supported by the German Academic Exchange Service (DAAD). The authors thank the BIOMEDAS program for training. This work has been partially supported by the EU H2020 RIA funded projects CLARIFY with grant agreement No 875160, EraMed P4-LUCAT No 53000015, and Opertus Mundi GA 870228, as well as the Federal Ministry for Economic Affairs and Energy (BMWi) project SPEAKER (FKZ 01MK20011A). Furthermore, Maria-Esther Vidal is partially supported by Leibniz Association in the program "Leibniz Best Minds: Programme for Women Professors", project TrustKG-Transforming Data in Trustable Insights with grant P99/2020.

References

[1] F. Aisopos, S. Jozashoori, E. Niazmand, D. Purohit, A. Rivas, A. Sakor, E. Iglesias, D. Vogiatzis, E. Menasalvas, A.R. Gonzalez, G. Vigueras, D. Gomez-Bravo, M. Torrente, R. Lopez, M.P. Pulla, A. Dalianis, A. Triantafillou, G. Paliouras and M.-E. Vidal, Knowledge graphs for enhancing transparency in health data ecosystems, in: *Semantic Web*, 2023, https://www.semantic-web-journal.net/content/knowledge-graphs-enhancing-transparency-health-data-ecosystems-0.

[2] M. Ali, H. Jabeen, C.T. Hoyt and J. Lehmann, The KEEN universe: An ecosystem for knowledge graph embeddings with a focus on reproducibility and transferability, 2020, in press. ISBN 978-3-030-30796-7.

[3] T.R. Besold, A. d'Avila Garcez, S. Bader, H. Bowman, P. Domingos, P. Hitzler, K.-U. Kühnberger, L.C. Lamb, P.M.V. Lima, L. de Penning, G. Pinkas, H. Poon and G. Zaverucha, *Chapter 1. Neural-Symbolic Learning and Reasoning: A Survey and Interpretation1, Neuro-Symbolic Artificial Intelligence: The State of the Art*, 2021. doi:10.3233/faia210348.

[4] A. Bordes, X. Glorot, J. Weston and Y. Bengio, A semantic matching energy function for learning with multi-relational data, *Machine Learning* (2014), 1–30, https://hal.archives-ouvertes.fr/hal-00835282. doi:10.1007/s10994-013-5363-6.

[5] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston and O. Yakhnenko, Translating embeddings for modeling multi-relational data, in: *Advances in Neural Information Processing Systems*, C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani and K.Q. Weinberger, eds, Vol. 26, Curran Associates, Inc., 2013, https://proceedings.neurips.cc/paper/2013/file/1cecc7a77928ca8133fa24680a88d2f9-Paper.pdf.

[6] A. Bordes, J. Weston, R. Collobert and Y. Bengio, Learning structured embeddings of knowledge bases, in: *25th Conference on Artificial Intelligence (AAAI)*, San Francisco, United States, 2011, pp. 301–306, https://hal.archives-ouvertes.fr/hal-00752498.

[7] S. Ceri, G. Gottlob and L. Tanca, What you always wanted to know about datalog (and never dared to ask), *IEEE Transactions on Knowledge and Data Engineering* **1**(1) (1989), 146–166. doi:10.1109/69.43410.

[8] A. d'Avila Garcez and L.C. Lamb, *Neurosymbolic AI: The 3rd Wave*, 2020, arXiv:2012.05876.

[9] A.S. d'Avila Garcez, K. Broda and D.M. Gabbay, Neural-symbolic learning systems – foundations and applications, in: *Perspectives in Neural Computing*, 2002.

[10] X. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun and W. Zhang, Knowledge vault: A web-scale approach to probabilistic knowledge fusion, in: *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'14*, Association for Computing Machinery, New York, NY, USA, 2014, pp. 601–610. ISBN 9781450329569. doi:10.1145/2623330.2623623.

[11] H.K.G. Fernlund, A.J. Gonzalez, M. Georgiopoulos and R.F. DeMara, Learning tactical human behavior through observation of human performance, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* **36**(1) (2006), 128–140. doi:10.1109/TSMCB.2005.855568.

[12] A. Fokoue, M. Sadoghi, O. Hassanzadeh and P. Zhang, Predicting drug-drug interactions through large-scale similarity-based link prediction, in: *The Semantic Web. Latest Advances and New Domains*, Springer International Publishing, 2016. ISBN 978-3-319-34129-3.

[13] C. Gutierrez and J.F. Sequeda, Knowledge graphs, *Commun. ACM* **64**(3) (2021), 96–104. doi:10.1145/3418294.

[14] A. Heuvelink, Cognitive models for training simulations, PhD thesis, Vrije Universiteit Amsterdam, 2009, https://research.vu.nl/en/publications/cognitive-models-for-training-simulations.

[15] P. Hitzler, A. Eberhart, M. Ebrahimi, M.K. Sarker and L. Zhou, Neuro-symbolic approaches in artificial intelligence, *National Science Review* **9**(6) (2022), nwac035. doi:10.1093/nsr/nwac035.

[16] A. Hogan, E. Blomqvist, M. Cochez, C. D'amato, G.D. Melo, C. Gutierrez, S. Kirrane, J.E.L. Gayo, R. Navigli, S. Neumaier, A.-C.N. Ngomo, A. Polleres, S.M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab and A. Zimmermann, *Knowledge Graphs, ACM Comput. Surv.* **54**(4) (2021), 37. doi:10.1145/3447772.

[17] G. Ji, S. He, L. Xu, K. Liu and J. Zhao, Knowledge graph embedding via dynamic mapping matrix, in: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Association for Computational Linguistics, Beijing, China, 2015, pp. 687–696, https://aclanthology.org/P15-1067. doi:10.3115/v1/P15-1067.

[18] M.R. Karim, M. Cochez, J.B. Jares, M. Uddin, O.D. Beyan and S. Decker, Drug-drug interaction prediction based on knowledge graph embeddings and convolutional-LSTM network, in: *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics, BCB 2019, Niagara Falls*, NY, USA, September 7–10, 2019, X.M. Shi, M. Buck, J. Ma and P. Veltri, eds, ACM, 2019, pp. 113–123. doi:10.1145/3307339.3342161.

[19] A. Karpathy and L. Fei-Fei, Deep visual-semantic alignments for generating image descriptions, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **39**(4) (2017), 664–676. doi:10.1109/TPAMI.2016.2598339.

[20] G. Lee, C. Park and J. Ahn, Novel deep learning model for more accurate prediction of drug-drug interaction effects, *BMC Bioinformatics* **20** (2019). doi:10.1186/s12859-019-3013-0.

[21] Y. Lin, Z. Liu, M. Sun, Y. Liu and X. Zhu, Learning entity and relation embeddings for knowledge graph completion, in: *Proceedings of the AAAI Conference on Artificial Intelligence 29(1)*, 2015, https://ojs.aaai.org/index.php/AAAI/article/view/9491.

[22] R. Masumshah, R. Aghdam and C. Eslahchi, A neural network-based method for polypharmacy side effects prediction, *BMC Bioinformatics* **22**(385) (2021). doi:10.1186/s12859-021-04298-y.

[23] M. Nickel, L. Rosasco and T. Poggio, Holographic embeddings of knowledge graphs, 2015, arXiv:1510.04935. doi:10.48550/ARXIV.1510.04935.

[24] M. Nickel, V. Tresp and H.-P. Kriegel, A three-way model for collective learning on multi-relational data, in: *Proceedings of the 28th International Conference on International Conference on Machine Learning, ICML'11*, Omnipress, Madison, WI, USA, 2011, pp. 809–816. ISBN 9781450306195.

[25] J. Pujara, E. Augustine and L. Getoor, Sparsity and noise: Where knowledge graph embeddings fall short, in: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, EMNLP 2017, Copenhagen, Denmark, September 9–11, 2017, M. Palmer, R. Hwa and S. Riedel, eds, Association for Computational Linguistics, 2017, pp. 1751–1756. doi:10.18653/v1/d17-1184.

[26] R. Ramakrishnan and J.D. Ullman, A survey of deductive database systems, *The Journal of Logic Programming* **23**(2) (1995), 125–149, https://www.sciencedirect.com/science/article/pii/0743106694000399. doi:10.1016/0743-1066(94)00039-9.

[27] A. Rivas, I. Grangel-González, D. Collarana, J. Lehmann and M.-E. Vidal, Unveiling relations in the industry 4.0 standards landscape based on knowledge graph embeddings, in: *Database and Expert Systems Applications*, 2020. doi:10.1007/978-3-030-59051-2_12.

[28] A. Rivas, I. Grangel-Gonzalez, D. Collarana, J. Lehmann and M.-E. Vidal, Discover relations in the industry 4.0 standards via unsupervised learning on knowledge graph embeddings, *Journal of Data Intelligence* **2**(3) (2021), 336–347. doi:10.26421/JDI2.3-2.

[29] A. Rivas and M.-E. Vidal, Capturing knowledge about drug-drug interactions to enhance treatment effectiveness, in: *Proceedings of the 11th on Knowledge Capture Conference, K-CAP'21*, Association for Computing Machinery, New York, NY, USA, 2021, pp. 33–40. ISBN 9781450384575. doi:10.1145/3460210.3493560.

[30] N. Rohani and C. Eslahchi, Drug-drug interaction predicting by neural network using integrated similarity, *Scientific Reports* **9** (2019). doi:10.1038/s41598-019-50121-3.

[31] 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). doi:10.1145/3424672.

[32] J.Y. Ryu, H.U. Kim and S.Y. Lee, Deep learning improves prediction of drug-drug and drug-food interactions, *Proceedings of the National Academy of Sciences* **115**(18) (2018), E4304–E4311. doi:10.1073/pnas.1803294115.

[33] A. Sakor, S. Jozashoori, E. Niazmand, A. Rivas, K. Bougiatiotis, F. Aisopos, E. Iglesias, P.D. Rohde, T. Padiya, A. Krithara, G. Paliouras and M. Vidal, Knowledge4Covid-19: A semantic-based approach for constructing a Covid-19 related knowledge graph from various sources and analyzing treatments' toxicities, *J. Web Semant.* **75** (2023), 100760. doi:10.1016/j.websem.2022.100760.

[34] A. Sakor, K. Singh, A. Patel and M. Vidal, Falcon 2.0: An entity and relation linking tool over Wikidata, in: *CIKM'20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event*, Ireland, October 19–23, 2020, M. d'Aquin, S. Dietze, C. Hauff, E. Curry and P. Cudré-Mauroux, eds, ACM, 2020, pp. 3141–3148. doi:10.1145/3340531.3412777.

[35] M.K. Sarker, L. Zhou, A. Eberhart and P. Hitzler, *Neuro-Symbolic Artificial Intelligence: Current Trends*, 2021, arXiv:2105.05330. doi:10.48550/ARXIV.2105.05330.

[36] D. Sridhar, S. Fakhraei and L. Getoor, A probabilistic approach for collective similarity-based drug-drug interaction prediction, *Bioinformatics* **32**(20) (2016), 3175–3182. doi:10.1093/bioinformatics/btw342.

[37] Z. Sun, Z.-H. Deng, J.-Y. Nie and J. Tang, RotatE: Knowledge graph embedding by relational rotation in complex space, 2019, arXiv:1902.10197. doi:10.48550/ARXIV.1902.10197.

[38] Z. Susskind, B. Arden, L.K. John, P. Stockton and E.B. John, Neuro-symbolic AI: An emerging class of AI workloads and their characterization, 2021, CoRR, arXiv:2109.06133.

[39] M. Vidal, K.M. Endris, S. Jazashoori, A. Sakor and A. Rivas, Transforming heterogeneous data into knowledge for personalized treatments – a use case, *Datenbank-Spektrum* **19**(2) (2019), 95–106. doi:10.1007/s13222-019-00312-z.

[40] S. Vilar, E. Uriarte, L. Santana, T. Lorberbaum, G. Hripcsak, C. Friedman and N.P. Tatonetti, Similarity-based modeling in large-scale prediction of drug-drug interactions, *Nature Protocols* **9** (2014). doi:10.1038/nprot.2014.151.

[41] Z. Wang, J. Zhang, J. Feng and Z. Chen, Knowledge graph embedding by translating on hyperplanes, in: *Proceedings of the AAAI Conference on Artificial Intelligence 28(1)*, 2014, https://ojs.aaai.org/index.php/AAAI/article/view/8870.

[42] D.S. Wishart, C. Knox, A.C. Guo, S. Shrivastava, M. Hassanali, P. Stothard, Z. Chang and J. Woolsey, DrugBank: A comprehensive resource for in silico drug discovery and exploration, *Nucleic Acids Research* **34** (2006), 668–672. doi:10.1093/nar/gkj067.

[43] P. Zhang, F. Wang, J. Hu and R. Sorrentino, Label propagation prediction of drug-drug interactions based on clinical side effects, *Scientific Reports* **5** (2015). doi:10.1038/srep12339.

[44] S. Zhang, Y. Tay, L. Yao and Q. Liu, Quaternion knowledge graph embeddings, in: *NeurIPS*, 2019, pp. 2731–2741, http://papers.nips.cc/paper/8541-quaternion-knowledge-graph-embeddings.

[45] Q. Zhao, J. Li, L. Zhao and Z. Zhu, Knowledge guided feature aggregation for the prediction of chronic obstructive pulmonary disease with chinese EMRs, *IEEE/ACM Transactions on Computational Biology and Bioinformatics* (2022), 1–10. doi:10.1109/TCBB.2022.3198798.

[46] C. Zhu, Z. Yang, X. Xia, N. Li, F. Zhong and L. Liu, Multimodal reasoning based on knowledge graph embedding for specific diseases, *Bioinformatics* **38**(8) (2022), 2235–2245. doi:10.1093/bioinformatics/btac085.

[47] M. Zitnik, M. Agrawal and J. Leskovec, Modeling polypharmacy side effects with graph convolutional networks, *Bioinformatics* **34**(13) (2018), i457–i466. doi:10.1093/bioinformatics/bty294.