Explainable Drug Repurposing in Context via Deep Reinforcement Learning

Lise Stork[1(✉)], Ilaria Tiddi[1], René Spijker[2,3], and Annette ten Teije[1]

[1] Vrije Universiteit Amsterdam, Amsterdam, The Netherlands
l.stork@vu.nl
[2] Cochrane Netherlands, Julius Center for Health Sciences and Primary Care, Utrecht, The Netherlands
[3] University Medical Center Utrecht, Utrecht University, Utrecht, The Netherlands

Abstract. Biomedical knowledge graphs encode domain knowledge as biomedical entities and relationships between them. Graph traversal algorithms can make use of these rich sources for the discovery of novel research hypotheses, e.g. the repurposing of a known drug. Traversed paths can serve to explain the underlying causal mechanisms. Most of these models, however, are trained to optimise for accuracy w.r.t. known gold standard drug-disease pairs, rather than for the explanatory mechanisms supporting such predictions. In this work, we aim to improve the retrieval of these explanatory mechanisms by improving path quality. We build on a reinforcement learning-based multi-hop reasoning approach for drug repurposing. First, we define a metric for path quality based on coherence with context entities. To calculate coherence, we learn a set of phenotype annotations with rule mining. Second, we use both the metric and the annotations to formulate a novel reward function. We assess the impact of contextual knowledge in a quantitative and qualitative evaluation, measuring: (i) the effect training with context has on the quality of reasoning paths, and (ii) the effect of using context for explainability purposes, measured in terms of plausibility, novelty, and relevancy. Results indicate that learning with contextual knowledge significantly increases path coherence, without affecting the interpretability for the domain experts.

Keywords: Explainable AI · Multi-hop Reasoning · Reinforcement Learning · Drug Repurposing

1 Introduction

Drug discovery is challenging and costly [8]: it can take very long, an average of ∼9 years for a drug to get approved by the relevant bodies [5]. Repurposing an already approved drug is a good alternative: it may reveal new interesting drug targets and pathways. However, considerable background knowledge about the biochemical properties of drugs and diseases, their relationships, and the causal mechanisms between them is required. By making knowledge about biomedical associations and processes machine-readable, automated methods can aid

C. Pesquita et al. (Eds.): ESWC 2023, LNCS 13870, pp. 3–20, 2023.
https://doi.org/10.1007/978-3-031-33455-9_1

experts in coming up with interesting new purposes for known drugs, to be further tested in clinical trials.

Modern Semantic Web technologies have shown a huge effort being directed toward the generation of structured biomedical knowledge, with the use of shared ontologies such as SNOMED-CT[1], DrugBank[2], the Cochrane Linked Data Vocabulary[3] or UMLS[4]. Researchers have aimed at linking such independent knowledge bases into federated biomedical networks with a.o. genes, pathways, biological processes, compounds and diseases [1,16,17], in support of automated drug repurposing or, more broadly, the discovery of new knowledge. The problem of drug repurposing on structured data can be formulated as a link prediction task in which known drug-disease pairs are used as gold standard data to predict novel ones [3,9,22,32].

Reinforcement learning (RL)-based multi-hop reasoning for drug repurposing has the advantage that reasoning paths can serve as explanations for newly discovered links, but the key challenge is the discovery of meaningful paths. Liu et al. [22] use logical rules, specifically *metapaths* (node types + relations between them), mined using AnyBurl [24] to guide the RL agent in discovering paths between diseases and compounds. Such metapaths can then be ranked by domain experts in order to assess the interpretability of the paths following these rules [23,35]. In most biomedical knowledge graphs, however, metapaths have many distinct instantiations. An example being the metapath *Compound–binds–Gene–associates–Disease* in Hetionet[5], of which the *Gene-associates-Disease* relation alone has an average number of ∼94 gene associations per disease[6]. The gene-disease associations catalogued by Gwas (See Footnote 6) are statistical, meaning that variations in these genes *may* contribute to the development of diseases or traits. This demonstrates that, often, only a subset of instantiated paths describe valid causal mechanisms, and not only the metapath but also the instantiated path, i.e., its entities, dictates their relevance.

In this work, we propose to guide the reinforcement agent's path traversal using auxiliary contextual knowledge about the phenotype(s) of a disease, i.e., its known symptoms. We formulate the task of drug repurposing as a *contextual* link prediction problem, in which we guide multi-hop reasoning using knowledge about a set of context entities. In RL-based multi-hop reasoning for drug repurposing [6], an agent traverses a graph of entities from disease to drug, rewarded when a terminal entity is reached. In our work, additional knowledge about the clinical phenotype of a disease is used to guide the path traversal of such agent (for an example, see Fig. 4). To the best of our knowledge, no research so far has focused on RL-based multi-hop reasoning using context entities.

We apply our methodology to Hetionet (See Footnote 5), a single integrative KG connecting biological entities such as genes and pathways. We assess the

[1] https://www.snomed.org/.
[2] https://go.drugbank.com/.
[3] https://data.cochrane.org/concepts/.
[4] https://www.nlm.nih.gov/research/umls/index.html.
[5] https://het.io/.
[6] https://www.ebi.ac.uk/gwas/.

impact of contextual knowledge in a quantitative and qualitative evaluation and aim at measuring: (i) the effect training with context has on the reasoning paths, and (ii) the effect of using context in causal explanations about phenotype-drug tuples on the plausibility, novelty and relevancy of these explanations as assessed by domain experts. Lastly, we present a real-world dataset consisting of patient populations (one or more diseases and symptoms), extracted from systematic reviews, that can be used for phenotype or population-based drug repurposing. Our contribution is twofold:

1. a RL method for multi-hop reasoning based on context, named CoCo (Coherent with Context), for the discovery of interesting clinical hypotheses;
2. a real-world dataset of small graphs representing patient populations (diseases, conditions, symptoms and comorbidities) from clinical trials.

2 Related Work

First, we discuss automated scientific discovery in general. Then, we move to AI for medicine, and finally multi-hop reasoning for drug repurposing.

Machine-Supported Scientific Discovery. Automated hypothesis discovery has been subject of study for a long time, since seminal works such as [2,33]. These works aimed at supporting scientists formulating testable hypotheses, either through suggesting literature or by discovering co-occurrences and correlations in data, sometimes using structured knowledge such as Medical Subject Headings (MeSH)-terms. The Knowledge Integration Toolkit (KnIT) [26] used methods such as matrix factorisation and graph diffusion to generate testable hypotheses in the biomedical domain. Methods for generated data insights were also presented in other fields, such as astronomy, geoscience or neuroscience [12,27]. These models create variants of hypotheses, which scientists can then refine and assess empirically.

One way to formulate the task of hypothesis generation, is through link prediction over knowledge graphs [3,9,22,32], or the generation of a small graph representing complex hypotheses. Attempts in this direction are the work of [15], where social sciences hypotheses are generated using a specific set of ontological classes, or [7] using generative adversarial models (GANs) for the prediction of small molecular graphs. Predictions made by such models are, however, not easily explained, which hampers trust especially in sensitive domains such as clinical medicine [19].

AI for Medicine. Through rapid technological developments and data digitisation, AI has found many applications in the pharmaceutical domain, from drug design through protein structure prediction with AlphaFold [18], to drug screening through toxicity prediction, and drug repurposing [28]. A downside of using common machine learning techniques for applications in medicine is the black-box nature of most of the prediction systems, hampering trust in such systems

in sensitive domains such as medicine. Explainable AI methods tackle this issue by providing transparent reasoning for the models in a variety of tasks, prediction included. These systems generally provide explanations either by eliciting the models' inner workings (e.g. visual cues or anchors [25]) or by using feature importance [20,30]. Supplying an AI model with structured, machine-readable background knowledge about known cause and effect relationships within the problem domain [4] can instead support both the generation of hypotheses as well as their explanation [3,9,10,22]. Rule mining [13,24] or path-search algorithms over large-scale knowledge graphs [6,34], which carry the potential to provide predictions with understandable explanations, have proven to be effective in more recent years. RL-based multi-hop reasoning for drug repurposing is an example task.

Table 1. Hetionet metaedges (predicates + types) and number of facts.

Metaedge	Count
participates(Gene, Biol.Proc.)	559504
expresses(Anatomy, Gene)	526407
regulates(Gene, Gene)	265672
includes(Gene, Gene)	147164
causes(Compound, Symptom)	138944
downregulates(Anatomy, Gene)	102240
upregulates(Anatomy, Gene)	97848
participates(Gene, Mol.Func.)	97222
participates(Gene, Pathway)	84372
participates(Gene, Cell.Comp.)	73566
covariates(Gene, Gene)	61690
downregulates(Compound, Gene)	21102
upregulates(Compound, Gene)	18756
associates(Disease, Gene)	12623
binds(Compound, Gene)	11571
upregulates(Disease, Gene)	7731
downregulates(Disease, Gene)	7623
resembles(Compound, Compound)	6486
localizes(Disease, Anatomy)	3592
presents(Disease, Symptom)	3357
includes(Pharma.Class, Compound)	1029
resembles(Disease, Disease)	543
treats(Compound, Disease)	483
palliates(Compound, Disease)	390

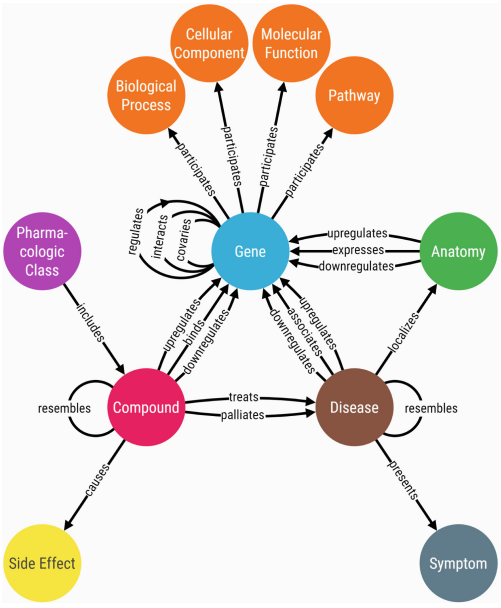

Fig. 1. Semantic schema of Hetionet.

Multi-hop Reasoning for Drug Repurposing. Link prediction has been proposed for the task of drug repurposing, in which a link between a disease and compound is predicted. Himmelstein et al. [16] for instance, obtained and integrated data from publicly available sources about biomedicine to create Hetionet (See Footnote 5) (see Fig. 1) and identified network patterns, call *metapaths*, to distinguish treatments from non-treatments, e.g., *Compound-binds-Gene-associates-Disease*. Sosa et al. [32] used the Global Network of Biomedical

Relationships (GNBR) [29] to develop a knowledge graph embedding-based drug repurposing method to predict novel treatments for diseases. They assessed the validity of these hypotheses using a variety of sources, and, similarly to Himmelstein et al. [16], discovered meaningful metapaths explaining newly discovered links.

One of the challenges to RL-based multi-hop reasoning, is that RL agents learn without the help of gold standard reasoning paths. An agent can therefore learn from nonsensical or meaningless search trajectories that incidentally lead to a correct answer [21]. Finding meaningful higher-order neighbourhoods is challenging. The injection of additional knowledge in the path traversal can guide multi-hop reasoning to learn from a more meaningful subset of trajectories. For instance, [22] used metapaths from [16] to train a RL agent to walk a graph of biomedical knowledge, receiving a reward if the path found for the *Compound-treats-Disease*, or vice-versa, matched a metapath.

In our work, we hypothesise that not only metapaths dictate the meaningfulness of a path, but also its entities. Therefore, we train a RL agent by rewarding paths of which the entities are coherent with the phenotype of a disease.

3 Multi-hop Reasoning Coherent with Context (CoCo)

We present preliminaries (Sect. 3.1), after which we discuss our main approach (Sect. 3.2).

3.1 Preliminaries

Domain Knowledge. As auxiliary knowledge to drive the learning of multi-hop patterns between diseases and compounds, we use knowledge of symptoms (i.e. phenotypes). We base our choice on a few assumptions from biomedicine:

Assumption 1. *A detailed understanding of how a condition's symptoms relate to underlying molecular processes can help in elucidating the molecular mechanisms underlying these conditions, useful for identifying new drug targets [31, 37].*

Assumption 2. *Shared symptoms can indicate shared genes between diseases [31]. Symptoms can thus serve as additional knowledge for representation learning of diseases and their associations with genes and other genotypes.*

Biomedical Knowledge Graphs. A biomedical knowledge graph $\mathcal{G}$ is a collection of biomedical facts $\{(e_1, r, e_2)\} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ where $\mathcal{E}$ and $\mathcal{R}$ are a set of entities (such as, genes or proteins, compounds, molecular functions, cellular components, biological processes, pathways, diseases) and relations (binds, associates, treats, downregulates, etc.), respectively. Inverse relations are indicated by an underscore: binds $\rightarrow$ _binds.

Logical Rules. Rule mining methods such as AnyBurl [24] or GPFL [13] mine logical rules from large knowledge graphs for the task of link prediction. Logical rules can be written in the form $head \leftarrow body$, in which the body can be seen as evidence for the head. We use lowercase letters to denote constants (ground entities e and relations $r \in \mathcal{G}$) and uppercase letters to denote variables. An example of a ground path rule of length n is shown below. Straight ground rules, rules without cycles in the body, can be divided into cyclic ($e_1 = e_{n+1}$), or acyclic rules ($e_1 \neq e_{n+1}$).

$$r_0(e_0, e_1) \leftarrow r_1(e_1, e_2), \ldots, r_n(e_n, e_{n+1}) \tag{1}$$

With rule mining, path rules are generalised into different rule types. Every method has a *language bias*, dictating what kind of rules can be learned. In this work, we focus on Both Anchored Rules (BAR) or instantiated rules, as they are capable of expressing relationships between pairs of entities. BAR rules are generalisations of acyclic straight rules in which an atom in the head e_i and the tail atom e_j are anchored by a constant, cfr. Eq. 2.

$$\textbf{BAR}: r_t(X, e_i) \leftarrow r_1(X, V1), r_2(V1, e_j) \tag{2}$$

We exemplify this with data from our use case, as shown in Eq. 3, which expresses that if a disease presents *Sensorineural Hearing Loss*, it was often associated with a gene that participates in the *positive regulation of Fc receptor mediated stimulatory signaling* pathway, with a confidence of α. For readability, we will call these associations *rule-based phenotype annotations*.

$$\begin{aligned} \alpha = 0.56 \quad & presents(X, e_i) \leftarrow associates(X, V1), participates(V1, e_j) \\ & e_i = \text{Sensorineural Hearing Loss} \\ e_j = \text{positive regulation of Fc receptor mediated stimulatory signaling} & \end{aligned} \tag{3}$$

The confidence score α refers to standard confidence [11], and is calculated by the number of correct predictions the rule suggests over the training set (support), divided by the number of possible groundings of the body atom of the rule.

Knowledge Graph Multi-hop Reasoning. Given a query $(e_{q1}, treats, ?)$ or $(?, treats, e_{q2})$, this approach aims to predict the missing element ?, through a k-hop reasoning path $e_1 \xrightarrow{r_1} e_2 \xrightarrow{r_2} \ldots \xrightarrow{r_k} e_{k+1}$. We extend the task to reasoning in context. Given a query $(e_{q1}, treats, ?, c_q)$, or $(?, treats, e_{q1}, c_q)$, in which c_q refers to additional knowledge about the query q, knowledge graph reasoning in context aims to predict the missing element ? through a k-hop reasoning path $p_q = e_1 \xrightarrow{r_1} e_2 \xrightarrow{r_2} \ldots e_{k+1}$, coherent with $c_q = \{e_1, \ldots, e_n\}$.

3.2 Proposed Approach

Link prediction through graph traversal as a Markov decision process has been proposed in the MINERVA algorithm [6]. Our methodology, which we call CoCo

(Coherent with Context), extends the MINERVA algorithm, with the novelty that we (i) formulate our queries as ternary relations $(e_{q1}, r_q, e_{q2}, c_q)$: a treats relation between a disease and drug, given a (set of) symptom(s), respectively, which we (ii) train and evaluate using a novel path coherence metric as reward. A schematisation of our method is presented in Fig. 2.

Path Coherence. We define Path Coherence (PC) as a score between a reasoning path p_q and context c_q for a given query q. Symptoms in the Hetionet graph do not contain any direct links to genes, pathways or other biomedical entities other than diseases. The human phenotype ontology[7], includes curated phenotype annotations, but the coverage is rather low. Therefore, we use GPFL [13] to mine these associations in the form of Both Anchored Rules (BAR), as exemplified in Eq. 4, with the added advantage of interpretability.

$$\begin{aligned} \alpha = 0.33 \quad presents(X, h_i) &\leftarrow associates(X, t_j) \\ h_i &= \text{Ataxia Telangiectasia} \\ t_j &= \text{Gene ATP7B} \end{aligned} \tag{4}$$

Path Coherence (Eq. 5) will be calculated based on these associations. In Eq. 5, $\mathbf{p}_q$ refers to the multi-hop reasoning path, $(t_j, \alpha_j) \in \mathbf{r}_{c_q}$ refers to the subset of rule-based phenotype annotations for which the head entity h_j is in the set of context entities c_q. t_j then refers to the tail entity of the rule, and α_j to the standard confidence. The metric sums up standard confidences for rule-based phenotype annotations that link a query's context entities to its path entities.

$$PC(\mathbf{p}_q, \mathbf{c}_q) = \sum_{e_i \in \mathbf{p}_q} \sum_{(t_j, \alpha_j) \in \mathbf{r}_{\mathbf{c}_q}} \alpha_{j\{e_i = t_j\}} \tag{5}$$

States. The state of the RL agent is encoded by the current location of the agent e_t, the entity e at time t, as well as the query (e_{q1}, r, e_{q2}). More formally, $\mathcal{S}_t \in \mathcal{S} = \Big(e_t, (e_{q1}, r, e_{q2})\Big)$.

Actions. The set of possible actions available to the agent at state $\mathcal{S}_t$ is encoded as $\mathcal{A}_{\mathcal{S}_t}$, and denotes all outgoing edges from the current entity e_t, as well as their tail entities. $\mathcal{A}_t$ denotes an action taken at time t. As with the MINERVA algorithm, we include self loops, allowing an entity to stay at the current node in case the agent requires fewer steps to reach e_{q2}.

Environment. The environment evolves according to the transition function $\delta : \mathcal{S} \times \mathcal{A} \leftarrow \mathcal{S}$, the action chosen by the agent, by updating the current state $\mathcal{S}_t$ to the new state $\mathcal{S}_{t+1}$.

[7] https://hpo.jax.org/app/data/annotations.

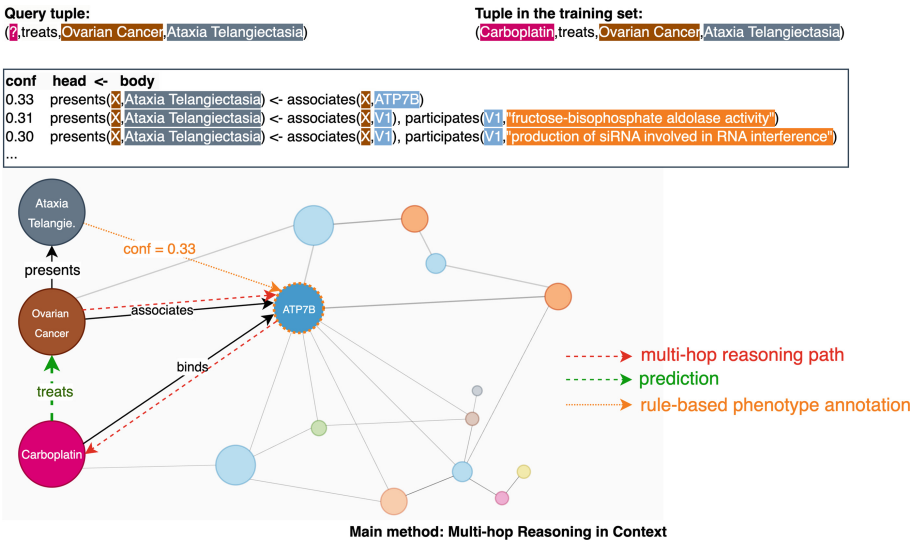

Fig. 2. A graphical representation of our methodology during training. First, the subset of rule-based phenotype annotations belonging to the query tuple's phenotype are retrieved. Second, A RL agent is trained to traverse the KG from disease to compound or vice versa. If the correct query entity is reached, a terminal reward of 1 is given. If the path is associated with the context via a rule-based annotation, an additional reward is given equal to the standard confidence of the rule (here PC = 0.33).

Policy Network. The history of our agent up to step t is denoted as $H_t = (H_{t-1}, A_{t-1})$. The policy network encodes the agent's transition history, and is parameterised by a long-term-short-term memory network (LSTM) [14], allowing for long-term dependencies between graph traversals.

$$\mathbf{h}_t = LSTM(\mathbf{h}_{t-1}, \mathbf{a}_{t-1}) \tag{6}$$

In Eq. 6, $\mathbf{a}_{t-1}$ refers to the vector space embedding of the previous action, which consists of $[\mathbf{r}_{t-1}, \mathbf{e}_t]$: the embedding of the action relation and its tail entity. If there is no previous action, $\mathbf{a}_{t-1}$ refers to the zero vector. The history-dependent action space distribution is given by Eq. 7 below, where $\mathbf{W_1}$ and $\mathbf{W_2}$ in Eq. are weight matrices learned during training. By stacking the embeddings for all the outgoing actions we obtain $\mathbf{A}_t$. A next action A_t is sampled according to Eq. 8.

$$\mathbf{d}_t = softmax(\mathbf{A}_t(\mathbf{W_2} ReLU(\mathbf{W_1}[\mathbf{h}_t; \mathbf{a}_t; \mathbf{r}_q]))) \tag{7}$$

$$A_t \sim Categorical(\mathbf{d}_t) \tag{8}$$

For each step made by the agent, we repeat Eq. (7)–(9) until the maximum path length is reached. The parameters of the LSTM network together with $\mathbf{W_1}$ and $\mathbf{W_2}$ form the parameters θ of the policy network π_θ.

Rewards. Rewards are given according to Eq. 9 at the end of k transitions, where k refers to the length of the reasoning path.

$$R(S_{k+1}) = 1_{\{t(e_{k+1})=t(e_{q2})\}} + \tau_{\{e_{k+1}=e_{q2}\}} + \rho_{\{t(e_{k+1})=t(e_{q2})\}} \sum_{e_i \in \mathbf{p}_q} \sum_{(t_j,\alpha_j)\in \mathbf{r}_{\mathbf{e}_q}} \alpha_{j\{e_i=t_j\}} \tag{9}$$

The first part of the equation reflects a type reward: it is set to 1 if the correct type $t(e_{q2})$ is reached, and 0 otherwise. The second part of the function assigns a reward of τ when the target entity is reached, 0 otherwise. The last part of the equation adds a path coherence (PC) reward, multiplied by ρ, which is only given when at least the correct type is reached $\{t(e_{k+1}) = t(e_{q2})\}$, 0 otherwise. We experimented with a stricter rule reward, only given when the agent reached e_{q2}, but hyperparameter optimisation showed this impacted accuracy.

Optimisation. As MINERVA, we employ the REINFORCE [36] algorithm to optimise the expected rewards. The agent's optimisation problem is given by Eq. 10, where $\mathcal{D}$ refers to the true underlying distribution of the $(e_{q1}, treats, e_{q2})$ triples.

$$\arg\max \mathbb{E}_{(e_{q1},treats,e_{q2})\sim\mathcal{D}} \mathbb{E}_{A_1,A_2,\ldots,A_L\sim\pi_\theta} \Big[R(S_{k+1}) | e_{q1}, e_{q2} \Big] \tag{10}$$

The second expectation is calculated over multiple rollouts (sampled trajectories) for each training example.

4 Experiments

Section 4.1 describes datasets used. Section 4.2 describes the rule-based phenotype annotations, and Sect. 4.3 describes the evaluation of our method.

4.1 Datasets

Hetionet Training, Validation and Test Set. Hetionet (See Footnote 5) consists of declarative knowledge within the biomedical domain, represented as binary relations between biomedical entities. Even though the graph contains binary relations between diseases and compounds, as well as between diseases and symptoms, it does not contain complex n-ary relations such as a relation between a disease, a set of symptoms, and a compound. Table 1 and 2 show statistics of the Hetionet graph. It's schema is shown in Fig. 1.

Table 2. Hetionet general dataset statistics.

Entities	Relations	Triples	Avg. node degree
47,031	24	2,250,197	95.8

For training and evaluation, we thus artificially construct such n-ary relations. First, we retrieve all *treats* triples as well as all *presents* triples from

Hetionet. From those triples, we construct tuples (all valid combinations) of the form *(Compound, Disease, Symptom)* and split them into training, validation and test set. As we are interested in discovering paths that give insight into the mechanics of drug treatments, we remove the edges *resembles* and *palliates*. Table 3 shows final train, test and validation-set statistics.

Real-World Dataset. Additionally, we extract a real-world dataset of populations from Cochrane's systematic reviews, which exemplifies how our approach can be used to predict drug targets or compounds based on complex phenotypes. The Cochrane Linked Data Project[8] has semantically annotated a collection of systematic reviews–a syntheses of clinical trials belonging to a specific research question–according to the PICO ontology[9], producing a small graph for each systematic review (a "PICO"). First, we extract unique patient populations and their diseases, conditions and symptoms from the PICO graphs[10]. In order to reason over the extracted subgraphs, we join them with the Hetionet graph in the following manner:

1. *Filtering PICO graph.* Using a SPARQL query, we extract populations and their disease or phenotype entities. In PICO graphs, these are all entities from the Cochrane Linked Data Vocabulary (CLDV)[11] of type `condition`. The query used to create these simplified graphs can be found online[12].
2. *Linking populations to Hetionet.* Both the CLVD and Hetionet use different codes to uniquely describe their biomedical entities. In order to join both graphs, we therefore replace all relevant nodes with UMLS identifiers using the steps enumerated below. The script used for this processing step can be found online (See Footnote 12).
   (a) `owl:sameAs` links are added between nodes in the CLDV and equivalent concepts from widely used vocabularies[13]. The CLDV maps disease and outcome entities to the following vocabulaires: MedDRA, MeSH, and SNOMED-CT.
   (b) vocabularies that are used to represent diseases and symptoms in Hetionet are DOID and UMLS, respectively. Both vocabularies, as well as those mentioned in (a), are downloaded from UMLS version 2021AA[14].
   (c) downloaded vocabularies are preprocessed by extracting all medical entities and linking them their respective UMLS identifiers using `owl:sameAs`.

[8] https://linkeddata.cochrane.org/.
[9] https://linkeddata.cochrane.org/pico--ontology.
[10] Initially, intervention nodes were extracted as well, but many of these proved too coarse grained—e.g. *viral agents*—to be useful for multi-hop reasoning.
[11] https://data.cochrane.org/concepts/.
[12] https://github.com/lisestork/coco.
[13] See the following concept for diabetes: https://data.cochrane.org/concepts/r4hp38bjj6qx, which they indicate is linked to unique codes from MedDRA: (10012594,10012601), MeSH: (D003920), and UMLS: (C0011849).
[14] https://download.nlm.nih.gov/umls/kss/2021AA/umls-2021AA-full.zip.

(d) RDFpro[15] is used to smush all preprocessed vocabularies, meaning that all identifiers are replaced by their respective UMLS identifiers.
(e) all `owl:sameAs` links are removed, such that all disease, drug and symptoms are represented by codes from the UMLS vocabulary.

3. *Appending types.* As a last step, types are appended to the UMLS codes, e.g. `Disease::C0348393` to ensure the graph remains compliant with the Hetionet ontology after smushing, given that some UMLS codes are used to represent distinct types in the Hetionet ontology (e.g., to describe a side effect as well as a disease).

After preprocessing, we end up with 357 population tuples. Statistics are shown in Table 3. An example tuple: a patient population with Haematological malignancy, as well as Cardiac death, Cytomegalovirus infection, and Herpes simplex. The dataset can be found online (See Footnote 12). Further investigation is needed to evaluate these results, as it does not yet include gold standard annotated drug targets (i.e., genes) or compounds.

Table 3. Hetionet training, validation and test set statistics: dataset split, total number of triples n, total number of *(Drug, Disease, Symptom)* tuples n_+, number of diseases n_d, number of compounds n_c and number of symptoms n_s.

dataset	n	n_+	n_d	n_c	n_s
Hetionet train	950	27,632	71	272	372
Hetionet validation	236	–	41	97	–
Hetionet test	300	9,200	56	119	363
Real-world dataset	357	–	60	–	276

4.2 Rule-Based Phenotype Annotations

To create a set of rule-based phenotype annotations, we applied GPFL [13] to the full Hetionet graph to mine rules for the relation *Disease-presents-Symptom*, as GPFL specialises in learning BAR rules (see Sect. 3.1). Recall that the set of rule-based phenotype annotations $\mathbf{r}$ is needed in the reward function used to train the policy network of the graph traversal algorithm (see Sect. 3.2), and is used for evaluation. We preprocessed the rules before use, by taking only high quality rules (which [13] defines as rules with a confidence score > 0.1, and a head coverage of > 0.01), and only those for which the head constant was a symptom (not a disease). Subsequently, we turned these rules into a rule set $\mathbf{r}$ (cfr. Eq. 11).

$$\mathbf{r} = \{(h_0, t_0, \alpha_0), ...(h_n, t_n, \alpha_n)\} \tag{11}$$

The final ruleset consists of 4,981,259 rules with 1,746 unique head atoms and 33,449 unique tail atoms, and can be found online (See Footnote 12). On average, each head atom has 2,853 rules (min: 1, max: 48,961).

[15] https://rdfpro.fbk.eu/.

4.3 Experimental Results

We evaluate our model on Hetionet in a quantitative and qualitative evaluation. We aim at measuring: (i) the effect training with contextual knowledge has on the quality of the reasoning paths, (ii) the effect training with contextual knowledge has on accuracy for the drug repurposing task, and (iii) the effect of using context in causal explanations on plausibility, novelty and relevancy as assessed by domain experts.

Quantitative Evaluation. We apply our method CoCo to Hetionet, as discussed in Sect. 3.2. After hyperparameter optimisation, we ran CoCo as well as MINERVA ten times using the following hyperparameters: learning rate: 0.004, $\rho = 2$, $\tau = 3$, 2 LSTM layers, and a hidden layer size of 128.

Results. To evaluate path quality, we calculate Path Accuracy (PA):

$$\text{PA} = \frac{n \text{ correct preds with } PC > 0}{n \text{ correct preds}} \tag{12}$$

We average PA over ten runs per model (CoCo and MINERVA), see Table 4. Moreover, to show that learning with context does not sacrifice hits@k scores, we compare CoCo to MINERVA on the basis of hits@1, hits@5 and Mean Reciprocal Rank (MRR), averaged over ten runs per model. Lastly, to demonstrate the difference in predictions and explanations between CoCo and MINERVA, we show statistics for the hits@1 predictions (Table 5): entities of correct predictions, as well as statistics of rewarded tail entities t_i. We trained both models for path length (PL) $\in \{2, 4\}$, as training with a PL $= 3$ yielded significantly worse results. We hypothesise this is due to Hetionet's semantics: the number of path patterns between disease and compound are reduced to a less meaningful subset.

Table 4. APA(%), Hits@k(%) and MRR (%) scores (± standard deviation) for CoCo and the model without context (MINERVA), averaged over 10 runs, for path length (PL) $\in \{2, 4\}$.

Model	PL	APA	Hits@1	Hits@5	MRR
CoCo	2	**65.61** (±23.19)	**9.64** (±2.4)	14.11 (±2.25)	11.60 (±2.10)
MINERVA	2	**59.63** (±14.94)	**9.14** (±2.98)	14.64 (±3.74)	11.79 (±3.10)
CoCo	4	**57.45**(±6.98)	**8.68** (±1.44)	**18.15** (±1.87)	**13.44** (±1.47)
MINERVA	4	45.87 (±4.25)	**8.58** (±2.15)	**18.44** (±1.60)	**13.37** (±1.72)

We observe from Table 4 and Table 5 that for CoCo, APA as well as the number of distinct rewarded tail entities t_i, has increased significantly for paths of length 4 (paired t-test, $p < 0.5$). This can indicate that patterns have been discovered between diseases and genes based on disease phenotypes. From Table 4,

we additionally observe that hits@k results have not decreased nor increased significantly.

Figure 3 shows rewarded rule tail types $t(t_i)$ during training. We can see that rewarded entities are mostly of type *Gene* and *Anatomy*, and that these numbers are increasing during training, indicating that a relationship between disease-drug tuples and these entities is learned. Other entities are not rewarded often during training. By observing Fig. 1, we hypothesise that this is likely due to path length, as entities of type *Molecular Function*, *Biological Process*, *Cellular Component* and *Pathway* are more hops away from diseases and compounds.

Qualitative Evaluation. For our qualitative evaluation, we evaluate the multi-hop reasoning paths on the basis of plausibility, novelty and relevancy to the domain, through an online survey. For both models (CoCo and MINERVA), we extracted the path of the highest-ranked prediction for each example from the test set. From this subset, we randomly sampled paths from the five most used metapaths, making sure that half of those had a PC of > 0.

Table 5. Hits@1 statistics: path length (PL), number of rule hits n_r (number of unique rewarded rule tail atoms n_t), number of unique diseases n_d, number of unique compounds n_c, and number of unique drug-disease triples n_t.

Model	PL	n_r (n_t)	n_d	n_c	n_t
CoCo	2	143 (19)	26	46	49
MINERVA	2	109 (23)	27	61	63
CoCo	4	**191** (**56**)	30	64	73
MINERVA	4	115 (41)	27	65	70

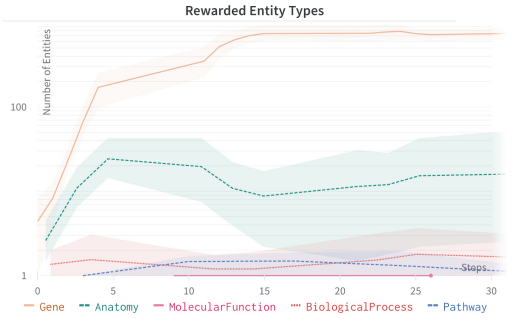

Fig. 3. Rewarded tail atom types $t(t_i)$ during training, averaged over ten runs. Note that the y-axis has a log scale.

Plausibility. For plausibility, we asked four annotators to rate each explanation with one of three categories: *implausible, partially plausible, plausible*. Amongst these annotators, the Krippendorff's inter annotator agreement was α_k 0.17, indicating a slight agreement ($0.01 < \alpha_k < 0.20$). The average plausibility score was 0.70, with a statistically insignificant difference between scores for paths generated by the model trained with and without context—0.73 and 0.66, respectively. Path length did influence interpretability scores significantly, with a score of 0.82 and 0.57 for path length 2 and 4 respectively, indicating that background knowledge about path semantics (for instance using metapaths in the reward function [22]) during training is needed to improve interpretability of longer paths.

Unexpectedly, using context in the explanation, as shown in Fig. 4, had a negative impact on the plausiblity score, 0.58 versus 0.86 for with and without context, respectively. We believe the result to be due to the associations (such as *Ovarian cancer* → *Ataxia Telangiectasia*) being unclear or difficult to interpret by the annotators due to the semantics of the *Disease-presents-Symptom* relation. Annotators indicated issues with semantics, i.e., *"Ataxia Telangiectasia is not a symptom of Ovarian cancer"*. Some annotators indicated a clearer description of the exact association would be more insightful *"association lacks further explanation of mechnism or relation"*. Framing such explanations differently, or in more detail, might therefore improve plausibility.

Novelty. To measure novelty, we asked annotators to indicate whether the fact was new to them or not, *True* or *False*. Amongst four annotators, the Krippendorff's α_k was -0.20, indicating a less than chance agreement ($\alpha_k < 0$). Given the varying expertise among the annotators (pharmacovigilance, geriatrics and multimorbidity, evidence synthesis, immunology, and molecular biology, drug therapies and safe use), we argue that the result is to be expected. No other result for novelty is calculated, as all explanations that were found plausible were also known to at least one of the annotators.

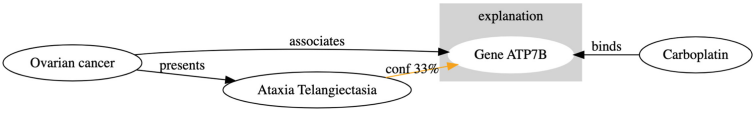

Fig. 4. A 2-hop reasoning path between Ovarian cancer and Carboplatin, associated through an instantiated logical rule with Ataxia Telangiectasia, a symptom of Ovarian cancer (as encoded in Hetionet).

Relevancy. For relevancy, we asked annotators to indicate, on a five point Likert Scale (*Strongly disagree-Strongly agree*), whether a fact was relevant to the domain of biomedicine. To three of the four annotators, the causal explanations were generally found relevant to the domain, given that in only 4 of the 24 cases one of the annotators chose *Neutral* or *Disagree/Strongly disagree*. The four cases that were rated as less relevant to the domain proved to be semantically incorrect. For one case, the symptom appeared to be the cause of the disease rather than a symptom, as noted by two annotators: *"fetal hypoxia is a cause of epilepsy, rather than a symptom"*. Moreover, in two cases, the metapath appeared not relevant, such as: *Disease-associates-Gene-associates-Disease-associates-Gene-_binds-Compound*, for which annotators indicated they did not understand the reasoning. One annotator was less optimistic about path relevancy, specifically for paths of length two. It was indicated that, even though these were plausible, they were deemed less relevant as they would be easier to devise by humans. Longer paths, on the other hand, are deemed less plausible, but potentially more relevant, as it is more challenging for humans to think up longer chains of reasoning.

Table 6. Predictions for Cochrane's populations for $PC = 0$ and $PC > 0$, with their systematic review codes.

Prediction	α	Review
Haematological malignancy & Herpes simplex → Raloxifene	0	CD012601
Haematological malignancy & Herpes simplex → Dactinomycin	0.11	CD012601
ADHD & Hyperkinesis → Haloperidol	0.15	CD005042
ADHD & Hyperkinesis → Thioridazine	0	CD005042
Hypertension & Hypervigilance → Ciclopirox	0.70	CD004351
Hypertension & Hypervigilance → L-Aspartic Acid	0	CD004351
Gestational diabetes & Hypertension → L-Glutamine	1	CD005542
Gestational diabetes & Hypertension→ Dinoprostone	1.15	CD005542

Lastly, we ran CoCo on our real-world dataset of populations from Cochrane systematic reviews. The full dataset including predictions, as exemplified in Table 6, can be found online (See Footnote 12). Examples from Table 6 show that predictions can be ranked based on path coherence (PC), where predictions with a $PC < 0$ do not take into account a disease's phenotype, and those with a $PC > 0$ are related to the disease's phenotype. Given that a detailed understanding of how a condition's symptoms relate to underlying molecular processes can help in elucidating the molecular mechanisms underlying these conditions, such predictions can be useful for identifying interesting drug targets. However, without gold standard annotations of these populations with drug targets and compounds, it is challenging to quantify the exact improvement.

5 Conclusion

Based on two assumptions (Assumption 1 and 2), we employed a novel approach for drug repurposing in context. We based our work on a RL-based multi-hop reasoning approach for drug repurposing. First, we defined a metric for path quality based on path coherence with a set of context entities (symptoms). Second, we annotated the context entities with biomedical entities using logical rule mining. Third, we used the measure for path coherence as a reward during training. We evaluated: (i) the effect of training with these logical rules on the reasoning paths, (ii) whether including these associations increased the interpretability of the paths when presented to domain experts. Moreover, we presented a real-world dataset of populations for multi-hop reasoning.

First, we discovered that after training with context, reasoning paths extracted for predictions on the test set changed significantly. They were found to be more coherent with their context without sacrificing prediction accuracy. Second, we found that an increased path length was found more interesting, given that longer reasoning chains would be more challenging to discover by humans.

However, longer reasoning paths were also found less plausible or semantically incorrect. The addition of metapaths (as done in [22]) to our methodology would resolve this issue.

A limitation of our method is that it only learns from the similarity of diseases based on their phenotype, whereas it does not take into account differences between phenotypes of related diseases nor restrictions for drug treatments due to co-occurring diseases. Moreover, even though we prove that paths discovered during testing appear more coherent with context, it is challenging to quantify the exact improvement without a larger gold standard dataset of drug-symptom-disease tuples. This will be looked at in future work.

Acknowledgements. The authors would like to thankfully acknowledge Lotty Hooft and Alexandre Renaux for their valuable opinions and discussions related to this work. This work is supported by the EU Horizon 2020 research programme MUHAI, grant no. 951846.

References

1. Agrawal, M., Zitnik, M., Leskovec, J.: Large-scale analysis of disease pathways in the human interactome. In: Pacific Symposium on Biocomputing, no. 212669, pp. 111–122 (2018). https://doi.org/10.1142/9789813235533_0011
2. Bahler, D., Stone, B., Wellington, C., Bristol, D.W.: Symbolic, neural, and Bayesian machine learning models for predicting carcinogenicity of chemical compounds. J. Chem. Inf. Comput. Sci. **40**(4), 906–914 (2000). https://doi.org/10.1021/ci990116i
3. Bakal, G., Talari, P., Kakani, E.V., Kavuluru, R.: Exploiting semantic patterns over biomedical knowledge graphs for predicting treatment and causative relations. J. Biomed. Inform. **82**(January), 189–199 (2018). https://doi.org/10.1016/j.jbi.2018.05.003
4. Blomqvist, E., Alirezaie, M., Santini, M.: Towards causal knowledge graphs-position paper. In: KDH@ ECAI (2020)
5. Brown, D.G., Wobst, H.J.: A decade of FDA-approved drugs (2010–2019): trends and future directions. J. Med. Chem. **64**(5), 2312–2338 (2021). https://doi.org/10.1021/acs.jmedchem.0c01516
6. Das, R., et al.: Go for a walk and arrive at the answer: reasoning over paths in knowledge bases using reinforcement learning. In: 6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings (2018)
7. De Cao, N., Kipf, T.: MolGAN: an implicit generative model for small molecular graphs. arXiv preprint arXiv:1805.11973 (2018)
8. DiMasi, J.A., Grabowski, H.G., Hansen, R.W.: Innovation in the pharmaceutical industry: new estimates of R&D costs. J. Health Econ. **47**, 20–33 (2016). https://doi.org/10.1016/j.jhealeco.2016.01.012
9. Drancé, M., Boudin, M., Mougin, F., Diallo, G.: Neuro-symbolic XAI for computational drug repurposing. In: KEOD, vol. 2, pp. 220–225 (2021)
10. Fu, G., Ding, Y., Seal, A., Chen, B., Sun, Y., Bolton, E.: Predicting drug target interactions using meta-path-based semantic network analysis. BMC Bioinform. **17**(1), 1–10 (2016)

11. Galárraga, L., Teflioudi, C., Hose, K., Suchanek, F.M.: Fast rule mining in ontological knowledge bases with AMIE+. VLDB J. **24**(6), 707–730 (2015). https://doi.org/10.1007/s00778-015-0394-1
12. Garijo, D., et al.: Towards automated hypothesis testing in neuroscience. In: Gadepally, V., et al. (eds.) DMAH/Poly -2019. LNCS, vol. 11721, pp. 249–257. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-33752-0_18
13. Gu, Y., Guan, Y., Missier, P.: Towards learning instantiated logical rules from knowledge graphs. arXiv preprint arXiv:2003.06071 (2020)
14. Guo, L., Sun, Z., Hu, W.: Learning to exploit long-term relational dependencies in knowledge graphs. In: International Conference on Machine Learning, pp. 2505–2514. PMLR (2019)
15. de Haan, R., Tiddi, I., Beek, W.: Discovering research hypotheses in social science using knowledge graph embeddings. In: Verborgh, R., et al. (eds.) ESWC 2021. LNCS, vol. 12731, pp. 477–494. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-77385-4_28
16. Himmelstein, D.S., et al.: Systematic integration of biomedical knowledge prioritizes drugs for repurposing. Elife **6**, e26726 (2017)
17. Jaundoo, R., Craddock, T.J.: DRUGPATH: the drug gene pathway meta-database. Int. J. Mol. Sci. **21**(9), 3171 (2020). https://doi.org/10.3390/ijms21093171
18. Jumper, J., et al.: Highly accurate protein structure prediction with AlphaFold. Nature **596**(7873), 583–589 (2021)
19. Kundu, S.: AI in medicine must be explainable. Nat. Med. **27**(8), 1328 (2021). https://doi.org/10.1038/s41591-021-01461-z
20. Lakkaraju, H., Kamar, E., Caruana, R., Leskovec, J.: Interpretable & explorable approximations of black box models. arXiv preprint arXiv:1707.01154 (2017)
21. Lin, X.V., Socher, R., Xiong, C.: Multi-hop knowledge graph reasoning with reward shaping. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP 2018) (2018)
22. Liu, Y., Hildebrandt, M., Joblin, M., Ringsquandl, M., Raissouni, R., Tresp, V.: Neural multi-hop reasoning with logical rules on biomedical knowledge graphs. In: Verborgh, R., et al. (eds.) ESWC 2021. LNCS, vol. 12731, pp. 375–391. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-77385-4_22
23. Lv, X., et al.: Is multi-hop reasoning really explainable? Towards benchmarking reasoning interpretability. arXiv preprint arXiv:2104.06751 (2021)
24. Meilicke, C., Chekol, M.W., Ruffinelli, D., Stuckenschmidt, H.: Anytime bottom-up rule learning for knowledge graph completion. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI). IJCAI/AAAI Press (2019)
25. Montavon, G., Samek, W., Müller, K.R.: Methods for interpreting and understanding deep neural networks. Digit. Sig. Process. **73**, 1–15 (2018)
26. Nagarajan, M., et al.: Predicting future scientific discoveries based on a networked analysis of the past literature. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 2019–2028 (2015)
27. Pankratius, V., et al.: Computer-aided discovery: toward scientific insight generation with machine support why scientists need machine support for discovery search. IEEE Intell. Syst. **31**(4), 3–10 (2016). https://doi.org/10.1109/MIS.2016.60
28. Paul, D., Sanap, G., Shenoy, S., Kalyane, D., Kalia, K., Tekade, R.K.: Artificial intelligence in drug discovery and development. Drug Discov. Today **26**(1), 80 (2021)

29. Percha, B., Altman, R.B.: A global network of biomedical relationships derived from text. Bioinformatics **34**(15), 2614–2624 (2018)
30. Ribeiro, M.T., Singh, S., Guestrin, C.: "Why should i trust you?" Explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1135–1144 (2016)
31. Saik, O.V., et al.: Novel candidate genes important for asthma and hypertension comorbidity revealed from associative gene networks. BMC Med. Genomics **11**(1), 61–76 (2018)
32. Sosa, D.N., Derry, A., Guo, M., Wei, E., Brinton, C., Altman, R.B.: A literature-based knowledge graph embedding method for identifying drug repurposing opportunities in rare diseases. In: Pacific Symposium on Biocomputing 2020, pp. 463–474. World Scientific (2020)
33. Swanson, D.R., Smalheiser, N.R.: An interactive system for finding complementary literatures: a stimulus to scientific discovery. Artif. Intell. **91**(2), 183–203 (1997). https://doi.org/10.1016/S0004-3702(97)00008-8
34. Tiddi, I., D'Aquin, M., Motta, E.: Walking linked data: a graph traversal approach to explain clusters. In: CEUR Workshop Proceedings (2014)
35. Wilcke, W.X., de Boer, V., de Kleijn, M.T., van Harmelen, F.A., Scholten, H.J.: User-centric pattern mining on knowledge graphs: an archaeological case study. J. Web Semant. **59**, 100486 (2019). https://doi.org/10.1016/j.websem.2018.12.004
36. Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. Mach. Learn. **8**(3), 229–256 (1992)
37. Zhou, X., Menche, J., Barabási, A.L., Sharma, A.: Human symptoms-disease network. Nat. Commun. **5**(1), 1–10 (2014)