Knowledge Graph Embedding
for Ecotoxicological Effect Prediction

Erik B. Myklebust[1,2(✉)], Ernesto Jimenez-Ruiz[2,3], Jiaoyan Chen[4],
Raoul Wolf[1], and Knut Erik Tollefsen[1]

[1] Norwegian Institute for Water Research, Oslo, Norway
`ebm@niva.no`
[2] Department of Informatics, University of Oslo, Oslo, Norway
[3] Alan Turing Institute, London, UK
[4] Department of Computer Science, University of Oxford, Oxford, UK

Abstract. Exploring the effects a chemical compound has on a species takes a considerable experimental effort. Appropriate methods for estimating and suggesting new effects can dramatically reduce the work needed to be done by a laboratory. In this paper we explore the suitability of using a knowledge graph embedding approach for ecotoxicological effect prediction. A knowledge graph has been constructed from publicly available data sets, including a species taxonomy and chemical classification and similarity. The publicly available effect data is integrated to the knowledge graph using ontology alignment techniques. Our experimental results show that the knowledge graph based approach improves the selected baselines.

Keywords: Knowledge graph · Semantic embedding · Ecotoxicology

1 Introduction

Extending the scope of risk assessment models is a long-term goal in ecotoxicological research. However, biological effect data is only available for a few combinations of chemical-species pairs.[1] Thus, one of the main efforts in ecotoxicological research is the design of tools and methods to extrapolate from known to unknown combinations in order to facilitate risk assessment predictions on a population basis.

The Norwegian Institute for Water Research (NIVA) is a leading Norwegian institute for fundamental and applied research on marine and freshwaters.[2] The Ecotoxicology and Risk Assessment programme at NIVA has through the last years developed a risk assessment system called RAdb.[3] This system has been applied to several case studies based on agricultural/industrial runoff into lakes

---

[1] Chemical and compound are used interchangeably.

[2] NIVA Institute: https://www.niva.no/en.

[3] NIVA Risk Assessment Database: https://www.niva.no/en/projectweb/radb.

or fjords. However, the underlying relational database structure of RAdb has its limitations when dealing with the integration of diverse data and knowledge sources. This limitation is exacerbated when these resources do not share a common vocabulary, as it is the case in our ecotoxicology risk assessment setting.

In this paper we present a preliminary study of the benefits of using Semantic Web tools to integrate different data sources and knowledge graph embedding approaches to improve the ecotoxicological effect prediction. Hence, our contribution to the NIVA institute is twofold:

(i) We have created a knowledge graph by gathering and integrating the relevant biological effect data and knowledge. Note that the format of the source data varies from tabular data, to SPARQL endpoints and ontologies. In order to discover equivalent entities we exploit internal resources, external resources (*e.g.*, Wikidata [21]) and ontology alignment (*e.g.*, LogMap [12]).
(ii) We have evaluated three knowledge graph embedding models (TransE [5], DistMult [23] and HolE [17]) together with the (baseline) prediction model currently used at NIVA. Our evaluation shows a considerable improvement with respect to the baseline and the benefits of using the knowledge graph models in terms of recall and $F_{\beta=2}$ score. Note that, in the NIVA use case, *false positives* are preferred over *false negatives* (*i.e.*, missing the hazard of a chemical over a species).

The rest of the paper is organised as follows. Section 2 provides some preliminaries to facilitate the understanding of the subsequent sections. In Sect. 3 we describe the use case where the knowledge graph and prediction models are applied. The creation of the knowledge graph is described in Sect. 4. Section 5 introduces the effect prediction models, while Sect. 6 presents the evaluation of these models. Finally, Sect. 7 elaborates on the contributions and discusses future directions of research.

2   Preliminaries

Knowledge Graphs. We follow the RDF-based notion of knowledge graphs [4] which are composed by RDF triples $\langle s, p, o \rangle$, where s represents a subject (a class or an instance), p represents a predicate (a property) and o represents an object (a class, an instance or a data value *e.g.*, text, date and number). RDF entities (*i.e.*, classes, properties and instances) are represented by an URI (Uniform Resource Identifier). A knowledge graph can be split into a TBox (terminology), often composed by RDF Schema constructors like class subsumption (*e.g.*, `ncbi:taxon/6668 rdfs:subClassOf ncbi:taxon/6657`) and property domain and range (`ecotox:affects rdfs:domain ecotox:Chemical`),[4] and an ABox (assertions), which contain relationships among instances (*e.g.*,

---

[4] The OWL 2 ontology language provides more expressive constructors. Note that the graph projection of an OWL 2 ontology can be seen as a knowledge graph (*e.g.*, [1]).

`ecotox:chemical/330541 ecotox:affects ecotox:effect/202`) and semantic type definitions (*e.g.*, `ecotox:taxon/28868 rdf:type ecotox:Taxon`). RDF-based knowledge graphs can be accessed with SPARQL queries, the standard language to query RDF graphs.

Ontology Alignment. Ontology alignment is the process of finding mappings or correspondences between a source and a target ontology or knowledge graph [10]. These mappings are typically represented as equivalences among the entities of the input resources (*e.g.*, `ncbi:taxon/13402 owl:sameAs ecotox:taxon/Carya`).

Embedding Models. Knowledge graph embedding [22] plays a key role in link prediction problems where the goal is to learn a scoring function $S : \mathcal{E} \times \mathcal{R} \times \mathcal{E} \rightarrow \mathbb{R}$. $S(s, p, o)$ is proportional to the probability that a triple $\langle s, p, o \rangle$ is encoded as true. Several models have been proposed, *e.g.*, Translating embeddings model (TransE) [5]. These models are applied to knowledge graphs to resolve missing facts in largely connected knowledge graphs, such as DBpedia [14]. Embedding models have also been successfully applied in biomedical link prediction tasks (*e.g.*, [2,3]).

Evaluation Metrics. We use (A)ccuracy, (P)recision, (R)ecall, (F_β) score to evaluate the models. They are defined as

$$A = \frac{tp + tn}{tp + tn + fp + fn} \tag{1}$$

$$P = \frac{tp}{tp + fp} \tag{2}$$

$$R = \frac{tp}{tp + fn} \tag{3}$$

$$F_\beta = (1 + \beta^2)\frac{PR}{\beta^2 P + R} \tag{4}$$

where tp, tn, fp, and fn stand for *true positive*, *true negative*, *false positive*, and *false negative*, respectively. Essentially, accuracy is the proportion of correct classifications. Recall is a measure of how many expected positive predictions were found by our model, and precision is the proportion of predictions that were correctly classified. F_β is a combined measure of precision and recall. $\beta = 1$ gives equal weight, while $\beta < 1$ favours precision and $\beta > 1$ favours recall. Here we use $F_{\beta=1}$ (F_1 in short) and $F_{\beta=2}$.

As the above metrics all depend on a selected threshold, we also use area under the receiver operating characteristic (ROC) curve (AUC) to measure and compare the overall pattern recognition capability of the prediction models. ROC is the curve of true positive rate ($tp/(tp + fn)$, i.e., recall) and false positive rate ($fp/(fp + tn)$), with the threshold ranging from 0 to 1 using a small step. AUC is the area under this curve, its values range between 0 and 1. Larger AUC indicates higher performance.

3 NIVA Use Case: Ecotoxicology and Risk Assessment

Ecotoxicology is a multidisciplinary field that studies the ecological and toxicological effects of chemical pollutants on populations, communities and ecosystems. Risk assessment is the result of the intrinsic hazards of a substance combined with an estimate of the environmental exposure (*i.e.*, Hazard + Exposure = Risk).

The Computational Toxicology Program within NIVA's Ecotoxicology and Risk Assessment section aims at designing and developing prediction models to assess the effect of chemical mixtures over a population where traditional laboratory data cannot be easily acquired.

Figure 1 shows the risk assessment pipeline followed at NIVA. *Exposure* is data gathered from the environment, while *effects* are hypothesis that are tested in a laboratory. These two data sources are used to calculate risk, which is used to find (further) susceptible species and the mode of action (MoA) or type of impact a compound would have over those species. Results from the MoA analysis are used as new effect hypothesis.
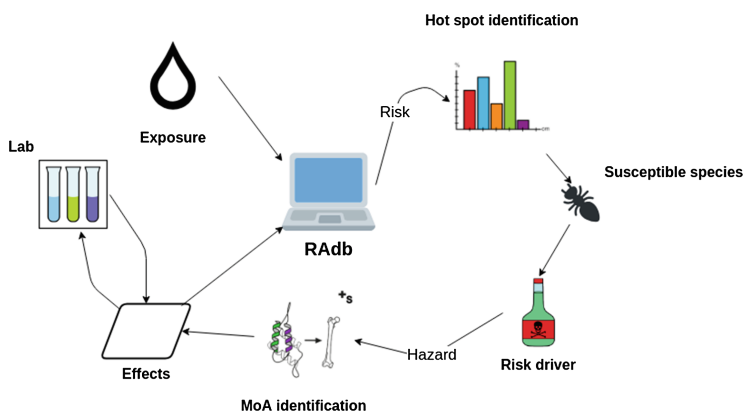

Fig. 1. NIVA risk assessment pipeline.

The effect data is gathered during experiments in a laboratory, where the population of a single species is exposed to a concentration of a toxic compound. Most commonly, the mortality rate of the population is measured at each time interval until it becomes a constant. Although the mortality at each time interval is referred to as *endpoint* in the ecotoxicology literature, we use *outcome* of the experiment to avoid confusion. Table 1 shows the typical outcomes and their proportion within the effects data. To give a good indication of the toxicity to a species, these experiments need to be repeated with increasing concentrations until the mortality reaches 100%. However, this is time consuming and is generally not done (*sola dosis facit venenum*). Hence, some compounds may appear

Table 1. The 10 most frequent outcomes in ECOTOX effect data.

Proportion	Abbreviation	Description
0.21	NR	Not reported
0.17	NOEL	No-observable-effect-level
0.16	LC50	Lethal concentration for 50% of test population
0.14	LOEL	Lowest-observable-effect-level
0.05	NOEC	No-observable-effect-concentration
0.05	EC50	Effective concentration for 50% of test population
0.04	LOEC	Lowest observable effect concentration
0.03	BCF	Bioconcentration factor
0.02	NR-LETH	Lethal to 100% of test population
0.02	LD50	Lethal dose for 50% of test population
0.11	Other	

more toxic than others due to limited experiments. Thus, when evaluating prediction models, (higher values of) recall are preferred over precision.

Risk assessment methods require large amounts of effect data to efficiently predict long term risk for the ecosystems. The data must cover a minimum of the chemicals found when analysing water samples from the ecosystem, along with covering species present in the ecosystem. This leads to a immense search space that is close to impossible to encompass in its entirety. Thus, it is essential to extrapolate from known to unknown combinations of chemical-species and suggest to the lab (ranked) effect hypothesis. The state-of-the-art within effect prediction are quantitative structure–activity relationship models (QSARs). These models have shown promising results for use in risk assessment, *e.g.*, [19]. However, QSARs have limitations with regard the coverage of compounds and species. These models use some chemical properties, but they usually only consider one or few species at a time. In this work we contribute with an alternative approach based on knowledge graph embeddings where the knowledge graph provides a global and integrated view of the domain.

Currently, the NIVA RAdb is under redevelopment, giving opportunities to include sophisticated effect prediction approaches, like the one presented in this paper, as a novel module for improving domain wide regulatory risk assessment.

4   A Knowledge Graph for Toxicological Effect Data

Risk assessment involves different data sources and laboratory experiments as shown in Fig. 1. In this section we describe the relevant datasets and their integration to create the *Toxicological Effects and Risk Assessment* (TERA) knowledge graph (see Fig. 2).
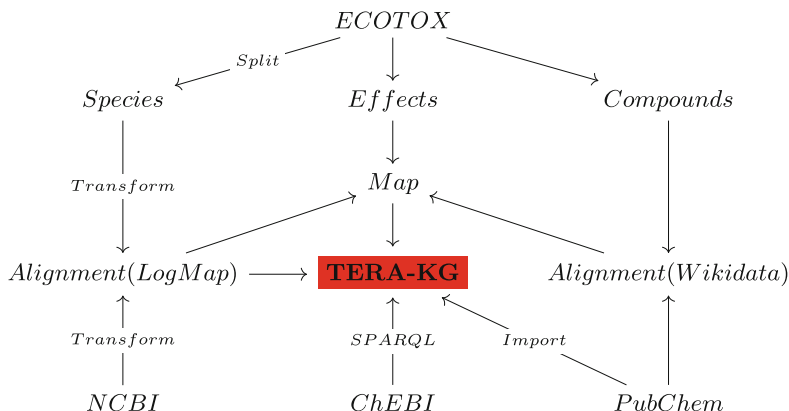
Fig. 2. Data sources in the TERA knowledge graph. Compound classification is available from PubChem. Chemical class hierarchy comes from the ChEMBL SPARQL endpoint. Compound literals are gathered from PubChem REST API and transformed into triples. ECOTOX and PubChem identifiers are aligned using the Wikidata SPARQL endpoint. ECOTOX and NCBI taxonomies are aligned using LogMap.

4.1   The ECOTOX Database

We rely on the ECOTOXicology database (ECOTOX) [9]. ECOTOX consists of $\sim$930k tests (or experiments) derived from the literature. Currently, an ECO-TOX test considers the effect of one of $\sim$12k chemicals on one of $\sim$13k species. Which implies that less than 1% of compound-species pairs have been tested. The effect is categorised in one of a plethora of predefined outcomes. For example, the LC50 outcome implies lethal concentration for 50% of the test population. Table 1 shows the most frequent outcomes in ECOTOX.

Table 2. ECOTOX database entry examples.

test_id	reference_number	test_cas	species_number
1068553	5390	877430 (2,6-Dimethylquinoline)	5156 (Danio rerio)
2037887	848	79061 (2-Propenamide)	14 (Rasbora heteromorpha)

result_id	test_id	endpoint	conc1_mean	conc1_unit
98004	1068553	LC50	400	mg/kg diet
2063723	2037887	LC10	220	mg/L

Table 2 contains an excerpt of the ECOTOX database. ECOTOX includes information about the compounds and species used in the tests. This information, however, is limited and additional (external) resources are required to complement ECOTOX.

The number of outcomes per compound and species varies substantially. For example, there are 1,881 experiments where the compound used is *sulfuric acid*,
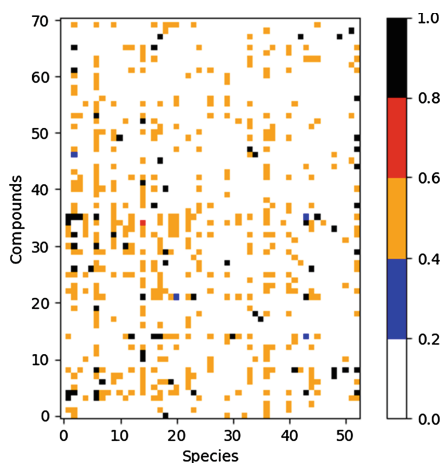
Fig. 3. ECOTOX effects data. x and y-axis represent individual species and chemicals sorted by similarity. Similarities are given by Eqs. (6) and (7) in Sect. 5.1. *i.e.*, chemicals $c_i \in C$ are indexed such that $S_{0,1} > S_{1,2} > \cdots > S_{n-1,n}$. Showing only chemicals and species that are involved in 25 or more experiments. Values relate to mortality rate of the test population, *i.e.*, LC50 corresponds to 0.5.

and 9,436 experiments where *Pimephales promelas* (fathead minnow) is the test species. The median number of experiments per chemical and species are 3 and 6, respectively. Figure 3 visualises a subset of the outcomes, here the zero values are either no effect or missing. This figure shows certain features of the data, *e.g.*, that compounds are more diversely used than species and that compound similarity is closely correlated to effects with regards to a species.

Currently, the ECOTOX database in used in risk assessment as reference data when calculating risk for a ecosystem. Essentially, comparing the reference and the observed chemical concentrations (per species). Since most compounds have multiple experiments per species, the mean and standard deviation of risk to a species can be calculated. However, if there is only one experiment for a compound-species pair we cannot calculate a standard deviation, such that the risk assessment is featureless. Therefore, estimating new effects is important to represent the natural variability of the effect data.

4.2   Dataset Integration into the TERA Knowledge Graph

Figure 2 shows the different datasets and their transformation that contribute in the creation of the TERA knowledge graph. For example Triples *(vii)–(ix)* in Table 3 have been created from the ECOTOX effect data.

Each compound in the ECOTOX effect data has a identifier called CAS Registry Number assigned by the Chemical Abstracts Service. The CAS numbers are proprietary, however, Wikidata [21] (indirectly) encodes mappings between CAS numbers and open identifiers like *InChIKey*, a 27 character hash of the

International Chemical Identifier (InChI) that encodes the chemical information in a unique manner. Hence, other datasets, such as PubChem [20], can be used to gather chemical features and classification of compounds. PubChem is already available as a knowledge graph and can be imported directly. However, the PubChem hierarchy only contains permutations of compounds. To create a full taxonomy for the chemical data, we use the ChEMBL SPARQL endpoint to extract the classification (provided by the ChEBI ontology [6]) for the relevant PubChem compounds. For example Triples *(v)* and *(vi)* in Table 3 come from the integration with PubChem and ChEMBL.

Table 3. Example triples from the TERA knowledge graph

#	subject	predicate	object
(i)	ecotox:group/Worms	owl:disjointWith	ecotox:group/Fish
(ii)	ncbi:division/2	owl:disjointWith	ncbi:division/4
(iii)	ecotox:taxon/34010	rdfs:subClassOf	ecotox:taxon/hirta
(iv)	ncbi:taxon/687295	rdfs:subClassOf	ncbi:taxon/513583
(v)	compound:CID10198308	rdf:type	obo:CHEBI_134899
(vi)	compound:CID10198308	pubchem:formula	''$C_7H_6O_6S$''
(vii)	ecotox:chemical/115866	ecotox:affects	ecotox:effect/001
(viii)	ecotox:effect/001	ecotox:species	ecotox:taxon/26812
(ix)	ecotox:effect/001	ecotox:endpoint	LC50
(x)	ecotox:taxon/33155	owl:sameAs	ncbi:taxon/311871

Aligning ECOTOX and NCBI. The species lineage in ECOTOX is not complete and therefore this (missing) information has been complemented with the NCBI taxonomy [16], a curated classification of all of the organisms in the public sequence databases (around 10% of the species on Earth). The tabular data provided for the ECOTOX species and the NCBI taxonomies has been transformed into subsumptions and disjointness triples (see first four triples in Table 3). Leaf nodes are treated as instance entities.

Since there does not exist a complete and public alignment between ECOTOX species and the NCBI Taxonomy, we have used the LogMap [11,12] ontology alignment systems to index and align the ECOTOX and NCBI vocabularies. ECOTOX currently only provides a subset of the mappings via its web search interface. We have gathered a total of 929 ground truth mappings for validation purposes. The lexical indexation provided by LogMap left us with 5,472 possible NCBI entities to map to ECOTOX (we focus only on instances, *i.e.*, leaf nodes). LogMap identified 4,681 (instance) mappings to ECOTOX ($\sim$40% of its entities) covering all 929 mappings from the (incomplete) ground truth, thus, an estimated recall of 100%. The mappings computed by LogMap have been included to the TERA knowledge graph as additional equivalence triples (see Triple *(x)* in Table 3 as example).

5 Effect Prediction Models

In this section we introduce the selected machine learning models to solve the effect prediction problem shown in Fig. 4. We use the known effects, denoted as *Affects* and *Not affects* in the figure, to predict whether or not new proposed chemical-species pairs are *true* (Affects) or *false* (Not affects).[5]

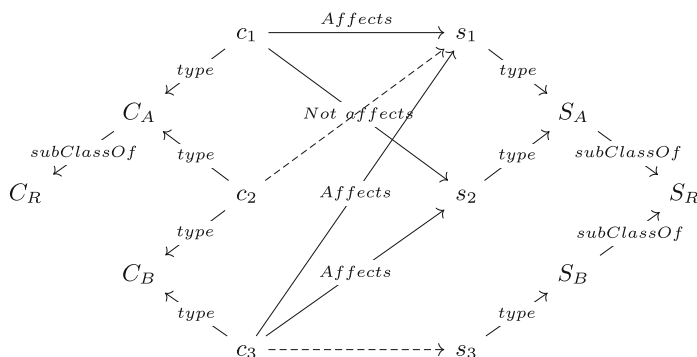
Fig. 4. The effect prediction problem. Lowercase s_j and c_i are instances of species and compounds, while uppercase denote classes in the hierarchy. Solid lines are observations and dashed lines are to be predicted. *i.e.*, does c_2 affect s_1?

Effect Data Sampling. A balance between positive and negative effect data samples is desired, therefore, we choose outcomes in categories (refer to Table 1): NOEL, LCp, LDp, NR-LETH, and NR-ZERO (p ranges from 0 to 100). We are only concerned about the mortality rate in experiments, consequently, we treat LC* and LD* identically. In addition, NR-LETH is treated as LC100. For simplicity, we treat the effects as binary entities. Hence, the outcome for a compound-species pair c, s is defined as

$$f(c, s) = \begin{cases} 1 & \text{if } (c, s) \in \text{LCp } \cup \text{ LDp } \cup \text{ NR-LETH} \\ 0 & \text{if } (c, s) \in \text{NOEL } \cup \text{ NR-ZERO.} \end{cases} \tag{5}$$

For example, according to Fig. 4, $f(c_1, s_1) = 1$ (*i.e.*, c_1 affects s_1) and $f(c_1, s_2) = 0$ (*i.e.*, c_1 does not affects s_1), while $f(c_2, s_1)$ is unknown and thus a prediction is required for this chemical-species pair.

Knowledge Graphs. We rely on the TERA knowledge graph (see excerpts in Table 3 and Fig. 4) to feed the knowledge graph embedding algorithms. For simplicity we discard the ECOTOX species entities that have not a correspondence to NCBI. Note that we currently do not consider literals.

---

[5] The models are implemented with Keras [7]. Data and codes available from: https://github.com/Erik-BM/NIVAUC.

5.1 Baseline Model (M_1)

This (baseline) prediction model is based on the current prediction method used at NIVA. The basic idea of this method is to find the nearest-neighbour from the observed samples. In this context, the nearest neighbours are defined by hierarchy distance for species and similarity for compounds. Therefor, we first define a adjacency matrix for the taxonomy and a similarity matrix for compounds.

$$A_{i,j} = \frac{1}{|P(s_i, r)| + |P(s_j, r)| - 2|P(s_i, r) \cap P(s_j, r)| + 1} \tag{6}$$

where r is the taxonomy root, $P(x, r)$ is the classes in the path from x to r, and $|\cdot|$ denotes the cardinality. One basic approach to calculate the chemical similarity is using the Jaccard index of the binary fingerprints of the compounds [18]. Hence, the similarity matrix is defined as

$$S_{i,j} = J(c_i, c_j) = \frac{|(F_i)_2 \cap (F_j)_2|}{|(F_i)_2 \cup (F_j)_2|} \tag{7}$$

We define a matrix $E \in \mathbb{R}^{|C| \times |T|}$, where C and T denote the set of compounds and species respectively. E contains all the observed effects (training set):

$$E_{i,j} = \begin{cases} 1 & \text{if } (c_i, \text{ affects, } s_j) \\ 0 & \text{else} \end{cases} \tag{8}$$

We can then make the prediction with A, S, and E, as shown in Algorithm 1. The algorithm terminates when t_{max} neighbours are visited or $p > 0$.

5.2 Multilayer Perceptron (M_2)

Our second prediction model is a Multilayer perceptron (MLP) network with n hidden layers. The model can be expressed as:

$$\boldsymbol{y}^0 = [\boldsymbol{e}_c, \boldsymbol{e}_s] \tag{9}$$

$$\boldsymbol{y}^t = ReLu(\boldsymbol{y}^{t-1} W_t + \boldsymbol{b}_t) \tag{10}$$

$$\hat{y} = \sigma(\boldsymbol{y}^n W_n + \boldsymbol{b}_n) \tag{11}$$

where $t = 1, 2, ..., n$. $[\cdot, \cdot]$ denotes vector concatenation. $ReLu$ is the rectifier function and σ is the logistic sigmoid function. W_t are the weight matrices and b_t are the biasses for each layer. $\boldsymbol{e}_c, \boldsymbol{e}_s \in \mathbb{R}^k$ are the embedded vectors of c and s. For example $\boldsymbol{e}_c$ is defined as

$$\boldsymbol{e}_c = \boldsymbol{\delta}_c W_C \tag{12}$$

where $\boldsymbol{\delta}_c$ is the one-hot encoded vector for entity c, $W_C \in \mathbb{R}^{|C| \times k}$ is an embedding transformation matrix to learn.

Input: E, A, S, c_i, s_j
Output: p, effect prediction for c_i, s_j
$i', j' \leftarrow i, j$;
$t_1 \leftarrow t_{max}$;
$p \leftarrow E_{i,j}$;   `// 0 if no overlap between train and test`
 while $t_1 > 0$ **do**
    $i' \leftarrow \arg\max_{k \neq i} S_{i,k}$;   `// find index of most similar compound`
    $A' \leftarrow A$; $t_2 \leftarrow t_{max}$;   `// copy A and reset counter`
    reset j;   `// reset to j in input` s_j
    while $t_2 > 0$ **do**
        $j' \leftarrow \arg\max_{k \neq j} A'_{j,k}$;   `// index of closest specie`
        $p \leftarrow \max(p, E_{i',j'})$;   `// update prediction`
        $A'_{j,j'} \leftarrow 0$;   `// set seen indices to zero`
        $t_2 \leftarrow t_2 - 1$;
        if $p > 0$ **then return** p;
        $i, j \leftarrow i', j'$;   `// update`

    end
    $S'_{i,i'} \leftarrow 0$;   `// set seen indices to zero`
    $t_1 \leftarrow t_1 - 1$;
end
return p;

Algorithm 1: Baseline prediction model algorithm (M_1).

A dropout layer is stacked after each hidden layer to prevent the network from overfitting. The model is optimised using `ADAGRAD` [8] with the following log loss function:

$$L(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\frac{1}{N} \sum_{i=1}^{N} [y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)] \tag{13}$$

5.3   Knowledge Graph (KG) Embedding and MLP ($M_2^\star$)

We have extended the MLP model (M_2) by feeding it with the TERA KG-based embeddings of c (*i.e.*, the chemical) and s (*i.e.*, the species), which encode the information of the taxonomy and compound hierarchies, among other semantic relationships. Note that the TERA knowledge graph also includes similarity triples about compounds. These triples represent pairs of compounds c_i and c_j where their similarity $S_{i,j}$ (as in Eq. 7) is above a threshold ϕ.

The embeddings are learned by applying the scoring function from one of DistMult [23], HolE [17], and TransE [5]. TransE was selected as it provides a very intuitive model. DistMult was included as it has shown state-of-the-art performance (*e.g.*, [13]), while HolE was considered as it also encodes directional relations. The score function for DistMult is defined as

$$S_D(s, p, o) = \sigma(\boldsymbol{e}_s^T W_p \boldsymbol{e}_o), \ W_p = diag(\boldsymbol{e}_p) \tag{14}$$

HolE uses a circular correlation score function, defined by

$$S_H(s, p, o) = \sigma(e_r^T[e_s \star e_o]), \; e_s \star e_o = \mathcal{F}^{-1}[\overline{\mathcal{F}(e_s)} \odot \mathcal{F}(e_o)] \qquad (15)$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ are the Fourier transform and its inverse, $\overline{x}$ is the element-wise complex conjugate, $\odot$ denotes the Hadamard product. The final method is TransE, which has the score function

$$S_T(s, p, o) = ||e_s + e_p - e_o|| \qquad (16)$$

where $||x||$ is the norm of x. e_s, e_p and e_o are the vector representation for the subject, predicate and object of a triple, respectively.

DistMult and HolE optimises for a score of 1 for positive samples and 0 for negative samples. Moreover, TransE scores positive samples as 0 and with no upper bound for negative samples. We modify the TransE score function to $S_T' = \tanh(1/S_T)$, such that $\lim_{S_T \to 0} S_T' = 1$ and $\lim_{S_T \to \infty} S_T' = 0$, to avoid modifying the labels.

The embeddings are used in the same network as the M_2 model. We train the embeddings and the classifier simultaneously using log loss and `ADAGRAD`. Training simultaneously will optimise the embeddings with regards to both the knowledge graph triples and the classifier loss.

6   Effect Prediction Evaluation

Sampling. We split the effect data 50%/50% for train/test. To prevent test set leakage, those training inputs that appear in the test set are removed, resulting in a 70%/30% split. $M_2^\star$ can be trained with the entirety of the knowledge graph, which is ignored under effect prediction. The negative knowledge graph samples are generated by randomly re-sampling subject and object of a true sample, while maintaining the distribution of predicates. We generate four negative samples per positive sample.

M_1 Model Settings. We tested the performance of M_1 with 6 choices of nearest neighbour (5, 10, 20, 30, 40, 50). In addition to Algorithm 1, we tested an alternative technique for iterating over the data. However, Algorithm 1 yielded better results. The most balanced results were found when using 30 neighbours. When using more than 30 neighbours recall increases, but accuracy and precision suffer from a considerable decrease since the use of more neighbours increases the false positive rate.

M_2/$M_2^\star$ Model Settings. The embedding dimension used in M_2 and $M_2^\star$ was based on a search among sizes 16, 64, 128 and 256. We found no difference between these parameters for M_2, therefor, 16 is chosen to aid faster training. $M_2^\star$ used a larger amount of entities and needs a larger embedding space to capture the features of the data. The performance plateaued at 128, hence, this was chosen. The models (M_2, $M_2^\star$) were trained until the loss stops improving for 5 iterations. For $M_2^\star$ we used different loss weights for the embeddings and

Table 4. Performance of the prediction models. $M_2^\star$ (S_T'), $M_2^\star$ (S_D) and $M_2^\star$ (S_H) stand for the MLP prediction models using TransE, DistMult, and HolE embedding models, respectively. *Above line*: ensemble averages of 10 clean tests. *Below line*: 10 fold cross validation on training set with standard deviation.

|	M_1 ($t_{max} = 30$)	M_2	$M_2^\star$ (S_T')	$M_2^\star$ (S_D)	$M_2^\star$ (S_H)
Accuracy	0.58	0.82	**0.83**	**0.83**	**0.83**
Precision	0.47	**0.76**	0.75	**0.76**	0.73
Recall	0.80	0.78	0.84	0.82	**0.87**
F_1 score	0.59	0.77	**0.79**	**0.79**	**0.79**
$F_{\beta=2}$ score	0.70	0.79	0.82	0.81	**0.84**
AUC	−	0.90	**0.91**	**0.91**	**0.91**
Accuracy	0.56 ± 0.01	$\mathbf{0.81 \pm 0.02}$	$\mathbf{0.81 \pm 0.02}$	$\mathbf{0.81 \pm 0.01}$	$\mathbf{0.81 \pm 0.02}$
Precision	0.55 ± 0.01	0.79 ± 0.04	$\mathbf{0.80 \pm 0.04}$	0.78 ± 0.03	0.79 ± 0.03
Recall	0.76 ± 0.03	0.84 ± 0.08	0.83 ± 0.08	$\mathbf{0.87 \pm 0.05}$	0.86 ± 0.02
F_1 score	0.65 ± 0.01	0.81 ± 0.03	0.81 ± 0.03	$\mathbf{0.82 \pm 0.01}$	0.82 ± 0.01
$F_{\beta=2}$ score	0.72 ± 0.02	0.83 ± 0.06	0.82 ± 0.06	$\mathbf{0.85 \pm 0.03}$	0.84 ± 0.01
AUC	−	$\mathbf{0.89 \pm 0.01}$	0.88 ± 0.01	$\mathbf{0.89 \pm 0.01}$	$\mathbf{0.89 \pm 0.02}$

the effect predictor. These weights were chosen such that the embeddings and effects are learned at similar rates. DistMult and HolE used 0.5 and 1.0 as loss weights for embeddings and effects models, respectively, while TransE used equal weights. We used a dropout rate of 0.2 and a similarity threshold of 0.5. Note that in $M_2^\star$ we simultaneously train the embedding models and the effect predictor. We perform *(i)* 10 fold cross validation on the training set, and *(ii)* a clean test on the unseen test set. This test consist of a ensemble of 10 models trained on the training set, each with a new set of random negative knowledge graph samples. We used an ensemble to limit the impact the random negative samples has on the results.

Evaluation. Figures 5a and b and Table 4 show the results of the conducted evaluation for the five effect prediction models. Figures 5a and b visualise the impact on accuracy and recall with different thresholds on the M_2–$M_2^\star$ prediction scores, while Table 4 presents the relevant evaluation metrics with a threshold of 0.5 for M_2–$M_2^\star$ and 30 neighbours for M_1. The results can be summarised as follows:

(i) M_1 is only slightly better than random choice, as the prior binary output distribution is 0.59 and 0.41. Thus it would not be appropriate for predicting effects. The false positive rate is also high, hence, M_1 would not be practical to use as a recommendation system.

(ii) M_2 is considerably better than M_1 and has balance between precision and recall. We suspect that this balance is due to random choice when the model has not previously seen a chemical or species. *i.e.*, a prediction close to the decision boundary when an input is unseen will maintain the false
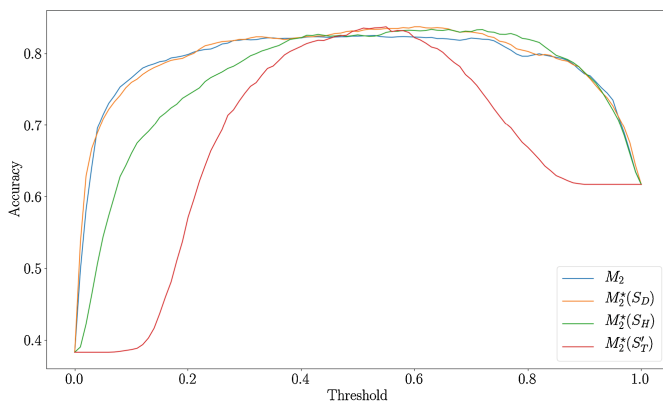
(a) Accuracy for the M_2 and $M_2^\star$ prediction models.
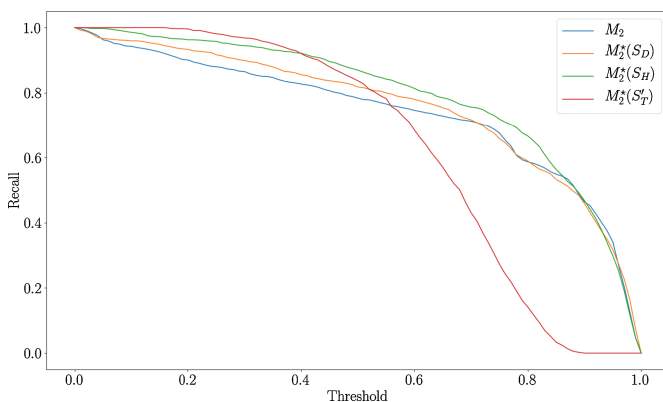
(b) Recall for the M_2 and $M_2^\star$ prediction models.

Fig. 5. Accuracy and Recall for the M_2 and $M_2^\star$ models with various thresholds.

negative/positive proportion, hence good for accuracy, not necessary for giving (interesting) recommendations to the laboratory.

(iii) Introducing the background knowledge to M_2, in the form of KG embeddings gives higher recall, without loosing accuracy. In contrast to M_2, $M_2^\star$ is more uncertain when unseen combinations are presented to the model (*in dubio pro reo*). Therefore, $M_2^\star$ is better suited to giving recommendations for cases where there is limited information about the chemical and the species in the effect data.

(iv) The best results in terms of recall, when using a threshold of 0.5 (see Table 4), are obtained by $M_2^\star$ with the embeddings provided by HolE (9 points higher than the M_2).

(v) As shown in Figs. 5a and b, lowering the decision threshold (0.30) would yield a higher recall (0.90) for the DistMult-based model, while maintaining the accuracy. TransE and HolE-based models have higher recall (0.97 and 0.94) at decision threshold 0.30, however, this comes at a cost of reduction in accuracy (0.74 and 0.79).

(vi) The highest overall $F_{\beta=2}$ score is 0.87, and is shared by all $M_2^\star$ models, albeit, at different decision boundaries, 0.34, 0.14 and 0.31 for models with TransE, DistMult, and HolE embeddings, respectively.

7   Discussion and Future Work

We have created a knowledge graph called TERA that aims at covering the knowledge and data relevant to the ecotoxicological domain. We have also implemented a proof-of-concept prototype for ecotoxicological effect prediction based on knowledge graph embeddings. The obtained results are encouraging, showing the positive impact of using knowledge graph embedding models and the benefits of having an integrated view of the different knowledge and data sources.

Knowledge Graph. The TERA knowledge graph is by itself an important contribution to NIVA. TERA integrates different knowledge and data sources and aims at providing an unified view of the information relevant to the ecotoxicology and risk assessment domain. At the same time the adoption of a RDF-based knowledge graph enables the use of *(i)* an extensive range of Semantic Web infrastructure that is currently available (*e.g.*, reasoning engines, ontology alignment systems, SPARQL query engines), and *(ii)* state of the art knowledge graph embedding strategies.

Prediction Models. The obtained predictions are promising and show the validity of the selected models in our setting and the benefits of using the TERA knowledge graph. As mentioned before, we favour recall with respect to precision. One the one hand, false positives are not necessarily harmful, while overlooking the hazard of a chemical may have important consequences. On the other hand, due to the limited experiments in terms of concentration (*i.e.*, effect data may not be complete), some chemicals may look less toxic than others while they may still be hazardous.

Value for NIVA. The conducted work falls into one of the main research lines of NIVA's Computational Toxicology Program (NCTP) to enhance the generation of hypothesis to be tested in the laboratory [15]. Furthermore, the data integration efforts and the construction of the TERA knowledge graph also goes in line with the vision of NIVA's section for Environmental Data Science. The availability and accessibility of the best knowledge and data will enable optimal decision making.

Novelty. Knowledge graph embedding models have been applied in general purpose link discovery and knowledge graph completion tasks [22]. They have also attracted the attention in the biomedical domain to find, for example, candidate genes for a disease, protein-protein interactions or drug-target interactions

(*e.g.*, [2,3]). However, we are not aware of the application of knowledge graph embedding models in the context of toxicological effect prediction.

Future Work. The main goal in the mid-term future is to integrate the TERA knowledge graph and the machine learning based prediction models within NIVA's risk assessment pipeline. In the near future, we intend to improve the current ecotoxicological effect prediction prototype and evaluate the suitability of more sophisticated models like Graph Convolutional Networks. The TERA knowledge graph will also be extended with additional information about species (*e.g.*, interactions) and compounds (*e.g.*, target proteins) which is expected to enhance the computed embeddings and the effect predictions.

Resources. The datasets, evaluation results, documentation and source codes are available from the following GitHub repository: https://github.com/Erik-BM/NIVAUC.

References

1. Agibetov, A., et al.: Supporting shared hypothesis testing in the biomedical domain. J. Biomed. Semant. **9**(1), 9:1–9:22 (2018)
2. Agibetov, A., Samwald, M.: Global and local evaluation of link prediction tasks with neural embeddings. In: 4th Workshop on Semantic Deep Learning (ISWC Workshop), pp. 89–102 (2018)
3. Alshahrani, M., Khan, M.A., Maddouri, O., Kinjo, A.R., Queralt-Rosinach, N., Hoehndorf, R.: Neuro-symbolic representation learning on biological knowledge graphs. Bioinformatics **33**(17), 2723–2730 (2017)
4. Arnaout, H., Elbassuoni, S.: Effective searching of RDF knowledge graphs. Web Semant.: Sci. Serv. Agents World Wide Web **48** (2018)
5. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Advances in Neural Information Processing Systems 26, pp. 2787–2795. Curran Associates, Inc. (2013)
6. ChEBI-ontology: The European bioinformatics institute (2019). https://www.ebi.ac.uk/chebi/
7. Chollet, F., et al.: Keras (2015). https://github.com/fchollet/keras
8. Duchi, J., Hazan, E., Singer, Y.: Adaptive subgradient methods for online learning and stochastic optimization. J. Mach. Learn. Res. **12**, 2121–2159 (2011)
9. US EPA: Ecotoxicology knowledgebase (ECOTOX) (2019). https://cfpub.epa.gov/ecotox/
10. Euzenat, J., Shvaiko, P.: Ontology Matching, 2nd edn. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38721-0
11. Jiménez-Ruiz, E., Cuenca Grau, B.: LogMap: logic-based and scalable ontology matching. In: Aroyo, L., et al. (eds.) ISWC 2011. LNCS, vol. 7031, pp. 273–288. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25073-6_18

12. Jimenez-Ruiz, E., Cuenca Grau, B., Zhou, Y., Horrocks, I.: Large-scale interactive ontology matching: algorithms and implementation. In: The 20th European Conference on Artificial Intelligence (ECAI), pp. 444–449. IOS Press (2012)
13. Kadlec, R., Bajgar, O., Kleindienst, J.: Knowledge base completion: baselines strike back. CoRR abs/1705.10744 (2017). http://arxiv.org/abs/1705.10744
14. Lehmann, J., et al.: DBpedia - a large-scale, multilingual knowledge base extracted from Wikipedia. Semant. Web **6**(2), 167–195 (2015)
15. Myklebust, E.B., Jimenez-Ruiz, E., Rudjord, Z.C., Wolf, R., Tollefsen, K.E.: Integrating semantic technologies in environmental risk assessment: a vision. In: 29th Annual Meeting of the Society of Environmental Toxicology and Chemistry (SETAC) (2019)
16. NCBI-Taxonomy: The national center for biotechnology information (2019). https://www.ncbi.nlm.nih.gov/taxonomy
17. Nickel, M., Rosasco, L., Poggio, T.A.: Holographic embeddings of knowledge graphs. CoRR abs/1510.04935 (2015). http://arxiv.org/abs/1510.04935
18. Nikolova, N., Jaworska, J.: Approaches to measure chemical similarity – a review. QSAR Comb. Sci. **22**(9–10), 1006–1026 (2003)
19. Pradeep, P., Povinelli, R.J., White, S., Merrill, S.J.: An ensemble model of QSAR tools for regulatory risk assessment. J. Cheminform. **8**, 48 (2016)
20. PubChem: National Institutes of Health (NIH) (2019). https://pubchem.ncbi.nlm.nih.gov/
21. Vrandecic, D., Krötzsch, M.: Wikidata: a free collaborative knowledgebase. Commun. ACM **57**(10), 78–85 (2014)
22. Wang, Q., Mao, Z., Wang, B., Guo, L.: Knowledge graph embedding: a survey of approaches and applications. IEEE Trans. Knowl. Data Eng. **29**(12), 2724–2743 (2017)
23. Yang, B., Yih, W., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases. CoRR abs/1412.6575 (2015)