Task-Driven Knowledge Graph Filtering Improves Prioritizing Drugs for Repurposing

Florin Ratajczak (✉ florin.ratajczak@helmholtz-muenchen.de)

Siemens Healthineers

Mitchell Joblin

Siemens AG

Martin Ringsquandl

Siemens AG

Marcel Hildebrandt

Siemens AG

Research Article

Keywords: Knowledge Graphs, Knowledge Graph Embeddings, Drug Repurposing, Link Prediction

Posted Date: August 30th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-721705/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Version of Record: A version of this preprint was published at BMC Bioinformatics on March 4th, 2022.

See the published version at <https://doi.org/10.1186/s12859-022-04608-y>.

RESEARCH

Task-driven Knowledge Graph Filtering Improves Prioritizing Drugs for Repurposing

Florin Ratajczak^{1*}, Mitchell Joblin², Martin Ringsquandl² and Marcel Hildebrandt²

¹Digital Technology & Innovation, Siemens Healthineers, Erlangen, Germany

²Corporate Technology, Siemens AG, Munich, Germany

*Correspondence:

florin.ratajczak@helmholtz-muenchen.de

¹Digital Technology & Innovation, Siemens Healthineers, Erlangen, Germany

Full list of author information is available at the end of the article

Abstract

Background: Drug repurposing aims at finding new targets for already developed drugs. It becomes more relevant as the cost of discovering new drugs steadily increases. To find new potential targets for a drug, an abundance of methods and existing biomedical knowledge from different domains can be leveraged. Recently, knowledge graphs have emerged in the biomedical domain that integrate information about genes, drugs, diseases and other biological domains. Knowledge graphs can be used to predict new connections between compounds and diseases, leveraging the interconnected biomedical data around them. While real world use cases such as drug repurposing are only interested in one specific relation type, widely used knowledge graph embedding models simultaneously optimize over all relation types in the graph. This can lead the models to underfit the data that is most relevant for the desired relation type. We propose a method that leverages domain knowledge in the form of metapaths and use them to filter two biomedical knowledge graphs (Hetionet and DRKG) for the purpose of improving performance on the prediction task of drug repurposing while simultaneously increasing computational efficiency.

Results: We find that our method reduces the number of entities by 60% on Hetionet and 26% on DRKG, while leading to an improvement in prediction performance of up to 40.8% on Hetionet and 12.4% on DRKG, with an average improvement of 17.5% on Hetionet and 5.1% on DRKG. Additionally, prioritization of antiviral compounds for SARS CoV-2 improves after task-driven filtering is applied.

Conclusion: Knowledge graphs contain facts that are counter productive for specific tasks, in our case drug repurposing. We also demonstrate that these facts can be removed, resulting in an improved performance in that task and a more efficient learning process.

Keywords: Knowledge Graphs; Knowledge Graph Embeddings; Drug Repurposing; Link Prediction

1 Background

Drug repurposing (also known as drug repositioning) involves the strategy of finding new indications for already approved drugs outside of their original indications [1]. It has several advantages over the *de novo* drug development, including shortening the timespan of development as well as saving costs [2, 3]. Historically, drug repurposing has lacked systematic approaches. The identification of new indications for a drug was largely due to casual observations during trials or clinical use [2]. Recently,

¹computational approaches have emerged that are driven by an increasing availability
²of suitable data [4, 2]. These methods usually scale to a large number of different
³drugs and diseases and have varying degrees of robustness, but they are also very
⁴specific to their domain and thus limited in the amount and type of data they
⁵can leverage. This limitation has recently been addressed by the development of
⁶datasets that span several domains and are modeled in the form of knowledge
⁷graphs, specifically for drug repurposing, and computational approaches that can
⁸leverage such data types [5, 6, 7].

⁹ In general, a knowledge graph (KG) is a collection of facts which are encoded as
¹⁰triples, e.g. (*Aspirin*, *treats*, *Headache*) [8]. In this example, *Aspirin* and *Headache*
¹¹are entities which are connected by the relationship *treats*. Entities can have types
¹²that further describe them and group them into categories. In this example, *Aspirin*
¹³is of type *Compound*, whereas *Headache* is of type *Disease*. This allows integrat-
¹⁴ing data from various domains, which makes KGs a highly flexible data structure
¹⁵[8]. Entities can also be connected via different relationships, e.g. entities of the
¹⁶type *Compound* can be connected to entities of type *Gene* in various ways, with
¹⁷*upregulates* and *downregulates* describing diametrically opposed concepts [5].

¹⁸ Recently, considerable effort is directed towards developing specialized machine
¹⁹learning models for KGs [9, 10, 11, 12, 13]. While the native representation of a
²⁰large graph is high-dimensional, these methods aim at projecting the graph into a
²¹lower-dimensional latent space in a way that best preserves the graph structure.
²²In most cases, this is achieved by learning distributed vector representations for
²³all entities and relationships which embed these concepts into the latent space.
²⁴Once the feature vectors, or embeddings, are learned, the models can score triples
²⁵(*head*, *relationship*, *tail*) by using model-specific scoring functions and yield the
²⁶probability of the given triple being present in the graph or not. Since the triples
²⁷in the graph represent true facts, this probability can be interpreted as the prob-
²⁸ability of the scored triple being a true fact. While triples that are present in the
²⁹graph should score a higher probability than triples that are absent, it is also pos-
³⁰sible for triples that are absent to score a high probability. This might indicate a
³¹previously undiscovered but true triple, given that the model has learned mean-
³²ingful representations of the concepts embedded in the latent space. Finding these
³³high-probability undiscovered triples is a task known as link prediction, or KG
³⁴completion. For drug repurposing, this corresponds to high-probability but absent
³⁵triples of the type (*Compound*, *treats*, *Disease*), which can then be used to priori-
³⁶ze compound-disease pairings for expert revision and clinical trials.

³⁷ As with most machine-learning approaches, the data that is used to train the
³⁸model is key to the performance of the trained model. Both quality and quantity of
³⁹the data have to be sufficient for the model to extract information and generalize to
⁴⁰previously unseen examples. Since it is nontrivial to judge the quality of the data
⁴¹in a graph context, most approaches focus on quantity, which leads to increasingly
⁴²larger KGs. KGs such as WikiData [14], BabelNet [15] or Yago [16] already span
⁴³millions of entities and billions of facts [8]. As the availability of knowledge grows in
⁴⁴the biomedical domain, more databases are compiled into larger KGs [5, 7]. The un-
⁴⁵derlying assumption is that a larger KG with more entities and relations will likely
⁴⁶be better for most, if not all, purposes. Knowledge graph embedding (KGE) meth-
ods aim at simultaneously optimizing predictions for all relations that are present

¹in a graph. However, in real applications such as drug repurposing, it is usually
²very few or just one specific relation that is of interest. Other real-world use cases
³on KGs that focus on one task only are predicting the involvement of compounds
⁴in gene regulation [17] and compound side effects [18]. It is therefore possible that
⁵while optimizing for all relations, the KGE model underfits the data on the relation
⁶of relevance, especially when that relation is underrepresented in the graph. This
⁷implies that by tailoring the KG to a specific task the performance of the model in
⁸that task can be improved. We aim at investigating this conjecture by deliberately
⁹and programmatically removing entities and triples from two biomedical KGs and
¹⁰analyzing the impact thereof on training and performance in a link prediction task
¹¹for drug repurposing. We realize this by applying a metapath based filtering ap-
¹²proach which allows us to inject domain knowledge about higher order relationships
¹³of the prediction task into the filtering procedure and create task-specific KGs.

¹⁴ We make the following contributions:

- ¹⁵ • We show that two large biomedical KGs contain facts that are not only irrele-
¹⁶ vant, but even consistently reduce prediction performance for drug repurpos-
¹⁷ ing.
- ¹⁸ • We propose a method to filter a KG based on metapaths that retains facts
¹⁹ which are supportive to the prediction task and results in up to 60% reduction
²⁰ in the data set size.
- ²¹ • We conduct an empirical study on 5 KGE approaches and show that our
²² method leads to a consistent improvement in terms of prediction performance
²³ (up to 41%).

²⁴

²⁵1.1 Related Work

²⁶KGE models have gained popularity for drug repurposing lately. Similar embedding
²⁷methods have been employed by [19], although with a focus on repurposing drugs
²⁸for Diabetes Mellitus. In [6] the authors of the Hetionet graph database have utilized
²⁹a custom path-based metric in combination of regularized linear regression for drug
³⁰repurposing on Hetionet. [20] have compiled the GNBR, a biomedical KG that is
³¹also incorporated into DRKG which is used in this work, to predict new treatments
³²for rare diseases using KGEs. KGs have also been used in the adjacent fields of drug
³³repurposing, like predicting interactions between drugs and genes [21], diseases and
³⁴genes [5, 22] or polypharmacy side effects [23]. There have also been endeavours
³⁵for large-scale computational drug repurposing specifically for COVID-19 [24, 25],
³⁶some also involving graphs [26, 27, 28].

³⁷1.2 Preliminaries

³⁸KGs consist of an entity set $\mathcal{E}$ and a set of binary relations $\mathcal{R}$. Each entity $e \in \mathcal{E}$
³⁹is assigned a type in $\mathcal{T}$ according to a mapping $\phi : \mathcal{E} \rightarrow \mathcal{T}$. While every entity is
⁴⁰unique, multiple entities can be assigned the same type. For example, while there is
⁴¹only one entity *Aspirin* in the graph, which is of the type *Compound*, there can be
⁴²multiple entities of the type *Compound* in the graph. We can define a KG in terms
⁴³of a set of triples $\mathcal{KG} \subset \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ of the form of (h, r, t) . The neighbourhood $N(e)$
⁴⁴of an entity e is the set of entities that are adjacent to e . Every triple contained in
⁴⁵the graph considered a true fact and every triple not contained in the graph taken
⁴⁶to be unknown (i.e., the open world assumption).

2 Results

2.1 Reduction in Dataset Size

In Hetionet, from the initial 45,158 entities and 2,249,807 triples, only 19,364 and 1,516,799 remain after applying our metapath-based filtering method (see Table 1). When it comes to the entities involved in the prediction task, from the initial 1538 entities of type compound and 136 entities of type disease, only 1503 compound entities and 136 disease entities remain after filtering. In DRKG, from the initial 96,121 entities and 5,874,261 triples only 57,592 and 4,748,143 remain after filtering (see Table 1). From the initial 23,347 compounds and 4,952 diseases, 19,633 and 4,813 remain after filtering. In both datasets, the relative frequency of entities with types Compound and Disease increases after filtering (see Figure 1). This effect is more pronounced on Hetionet than on DRKG. Other Node types that increase in frequency after the modification in both datasets are Anatomy, Pathway, Pharmacologic Class and Symptom, which all are closely related to the target domain. Node types that are uniformly reduced are Biological Process and Cellular Component. These node types refer to broader biological concepts and are not as domain specific.

In summary, applying our method to both KGs results in a significant reduction in both the number of entities and triples. Interestingly, the filtering process seems to increase the relative presence of the two entities types relevant to our prediction task (i.e., compounds and diseases) and of entity types that are closely related to it.

Table 1 Effects of downsampling on the Number of entities and relations and the average degree.

Dataset	observed variable	Original	Modified	%-Change
Hetionet	Entities (k)	47	19	-60%
	Triples (M)	2.2	1.5	-32%
	Avg. Degree	24	39	+65%
DRKG	Entities (k)	96	71	-26%
	Triples (M)	5.6	5.4	-4%
	Avg. Degree	61	77	+26%

2.2 Impact on Training

Our graph filtering procedure impacts the samples drawn by the negative sampler during training (see Figure 2), which are used to calculate the loss (see Sec. 5.2.7). Both the negative and the positive samples achieve higher scores on average on the modified dataset. In addition, the entropy as a measure of distance between the two distributions increases from 0.104 on the original dataset to 0.120 on the modified dataset on Hetionet. The increase in entropy taken together with the change to a bimodal distribution indicate that the negative samples drawn after modifying the dataset come from two underlying distributions, one being easier and one being harder to distinguish from the positive samples (see Figure 2 B). In comparison, the negative samples drawn from the original dataset have a uniform distribution and receive lower average probabilities (see Figure 2 A). On DRKG, the entropy between the positive and negative samples decreases from 0.074 on the original dataset to 0.032 on the modified version of DRKG. This indicates that the distributions become more alike, meaning that the negative samples become harder to distinguish from the positive samples (see Figure 2 C & D).

Figure 1 Relative frequencies of node types in **A** Hetionet and **B** DRKG. The lighter bars denote the frequencies in the original datasets whereas the darker bars denote the frequencies in the modified dataset. The change in relative frequency from the original to the modified datasets is denoted as percentages at the end of the bars. A: Anatomy, BP: Biological Process, CC: Cellular Component, C: Compound, D: Disease, G: Gene, MF: Metabolic Function, P: Pathway, PC: Pharmacologic Class, SE: Side Effect, S: Symptom, Atc: Atc, T: Tax

The training loss is higher on the modified dataset than on the original dataset (see Figure 3). Conversely, the evaluation performance on the validation set reaches equal levels after fewer epochs and higher global optima on the modified dataset compared to the original dataset. This indicates a higher learning efficiency on the modified dataset. On the original dataset, the evaluation performance begins to saturate earlier, resulting in an earlier stop and worse global optima compared to the modified dataset. The higher training loss indicates that the optimization for the average task in the graph is becoming worse, while the higher evaluation performance indicates that the optimization for drug repurposing improves. This shows that the modification of the dataset is not a free lunch, but instead a trade-off between worse performance on the average tasks but better performance in the task of interest.

2.3 Increased Performance

The performances of almost all models have increased after metapath-based filtering (see Table 2). On Hetionet the performance change ranges from +0.6% up to +40%, whereas on DRKG it ranges from a small hit to the performance with -1.2% to an increase of 12.4%. Like with the downsampling, the effects of our method are less pronounced on DRKG than on Hetionet.

2.4 SARS-CoV2 Case Study

As shown in Figure 4 **A**, training the original dataset does not allow the model to adequately capture the relevance of favipiravir and danoprevir in its role as an antiviral compound against SARS-CoV2. The modified dataset however allows the model to prioritize all antiviral compounds over the median rank. It also prioritizes the antiviral compounds stronger than the model trained on the original dataset, especially visible in danoprevir and favipiravir. Only ritonavir is less prioritized by the model trained on the modified dataset than by the model trained on the original

Table 2 Effects of downsampling on the performance.

Dataset	Model	Original (MRR)	Modified (MRR)	%-Change
Hetionet	TransE	0.2199	0.2550	+15.9%
	DistMult	0.2270	0.2284	+0.6%
	ComplEx	0.1975	0.2782	+40.8%
	RESICAL	0.2428	0.2940	+21.1%
	ConvE	0.1312	0.1647	+25.5%
	Random	0.0343	0.0331	-3.5%
DRKG	TransE	0.0837	0.0826	-1.2%
	DistMult	0.0899	0.0931	+3.6%
	ComplEx	0.0896	0.0950	+5.9%
	RESICAL	0.0578*	0.0650*	+12.4%
	ConvE	0.0618*	0.0649*	+4.9%
	Random	0.0102	0.0086	-16.7%

*) Due to computational constraints, no hyperparameter optimization (HPO) has been performed for these models. The hyperparameter setting was chosen based on the results of the HPO on Hetionet.

dataset. Taking into account the average rank of the four compounds with all 27 SARS-CoV entites in the graph, it is clear that the modified dataset helps the model to prioritize the antiviral compounds (see Figure 4 B). All compounds receive lower mean ranks when the model is trained on the modified dataset than when it is trained on the orig dataset.

Figure 3 Training loss and evaluation performance curves for the best performing ComplEx models on the original and the modified dataset of Hetionet (A) and DRKG (B). The red crosses indicate at which epoch the model achieved the best evaluation score before being early stopped three epochs later. The dashed grey lines demonstrate the performance of the best performing model for easier comparison.

Figure 4 Predicted ranks of four antiviral Compounds and SARS-CoV2 entities compared to the ranks of all Compounds. The light green and blue bars show the ranks of predictions of the type (SARS CoV2 entity, $treats^{-1}$, Compound) for all Compounds in the graph, using the original and the modified dataset for training. The predicted ranks of the four antiviral Compounds are highlighted. The red and dark green bars show the change in the predicted rank of a given antiviral compound when the model was trained on the original dataset versus when the model was trained on the modified dataset. In A, the SARS-CoV2 entity is the SARS-CoV2 Spike entity. B shows the mean ranks of all 27 SARS-CoV2 entities in the graph with each of the four antiviral compounds (for all individual ranks see Additional File 1 — Table S3). Expected median rank is 9816, lower ranks indicate higher-than-average probabilities of treatment.

3 Discussion

Our experiments show that our method increases the average degree per entity (see Table 1), thus increasing the neighborhood richness, which produces entity embeddings that achieve higher performance on the drug repurposing task. It also increases the relative proportion of *Compound* and *Disease* entities, thus making them have a greater impact on the overall loss. On DRKG, it also removes a significant percentage of Compounds, indicating that these Compounds were not densely connected to any disease by any of the given metapaths. However, since these entities are not considered during the evaluation of the performance, their removal is

¹not responsible for the improvements reported in Table 2. If such a model would be
²used in production, however, it would be of utmost importance to remove such pos-
³sibly biased and low-confidence predictions, as is done by our task-driven metapath
⁴filtering procedure.

⁵ Entities that are very densely connected to Compounds and Diseases and are
⁶intuitively related to our prediction task, like Pharmacological Class, Side Effects
⁷and Symptoms, are not heavily downsampled and increase in relative frequency (see
⁸Figure 1). Cellular Components and Biological Processes on the other hand, where
⁹a significant proportion of the entities are not expected to play a vital role in drug-
¹⁰disease mechanisms, are uniformly reduced in relative frequency. Genes are heavily
¹¹reduced as well by almost 50% (see Additional File 1 — Table S1 and Additional
¹²File 1 — Table S2) but retain their relative frequency in the graph (see Figure 1).
¹³This indicates that, while Genes play a crucial role in the mechanistic connections
¹⁴between Compounds and Diseases, a preliminary selection of certain Genes might
¹⁵be helpful for future studies of biological networks.

¹⁶ Additionally, since the walker has a higher chance to pass through a densely
¹⁷connected entity compared to a sparsely connected entity, well established entities
¹⁸are favoured during the task-driven metapath filtering. This relies on the assumption
¹⁹that e.g. a gene that has been well researched will have more adjacent relations than
²⁰a gene has not been researched that well yet, since most of its possible connections to
²¹compounds and diseases, or its roles in metabolic functions, are not discovered yet.
²²Applying this example also to other entity types, the stochastic nature of our task-
²³driven metapath filtering method might provide a rudimentary form of retaining
²⁴established entities while filtering out less established entities.

²⁵ The increase in performance reported in Table 2 is possibly due to the negative
²⁶samples that are generated during training becoming more similar to the positive
²⁷samples. In the initial graphs, the unconstrained topology makes it possible that
²⁸certain entities are in great distance or even unreachable from a *Compound* or
²⁹*Disease* entity. This makes it possible for the negative sampling to draw negative
³⁰samples that are easily distinguishable from positive samples and that carry no
³¹information for our prediction task. On the modified graphs however, the walking
³²strategy applied during filtering makes sure that all entities are at most half of the
³³length of the longest metapath away from a *Compound* or *Disease* entity. This
³⁴makes the samples generated by the negative sampler more alike to the positive
³⁵samples that they are being compared to and increases their informative value
³⁶relative to our prediction task, thus increasing learning efficiency. In Figure 2 it can
³⁷be seen that after the filtering, the negative samples move closer to the predictions
³⁸of the positive samples. The predicted probabilities of the positive samples on the
³⁹other hand move closer to 1, resulting in more confident predictions. This scenario
⁴⁰makes the learning problem harder, which results in a higher training loss (see
⁴¹Figure 3). The more confident predictions of positive relations alongside with the
⁴²fact that in the modified dataset most of the graph is in direct proximity of entities
⁴³of the type *Compound* and *Disease* simultaneously results in a higher evaluation
⁴⁴performance on the validation and test holdout sets, as can be seen in Figure 3 and
Table 2, respectively.

⁴⁵ Since lopinavir and ritonavir are well established antiviral drugs, it is not surpris-
⁴⁶ing that they earn the lowest ranks from the four selected compounds. However, the

¹results shown in Figure 4 indicate that the model trained on the original dataset
²does not adequately capture the relevance of favipiravir and danoprevir in its role
³as antiviral compounds against SARS-CoV2. The modified dataset however allows
⁴the model to effectively prioritize these compounds. Lopinavir and ritonavir appear
⁵not to be effective against SARS-CoV2, despite being an established treatment for
⁶other viral diseases [29, 30, 31]. Favipiravir and danoprevir on the other hand show
⁷promise as treatment options for SARS-CoV2 [30, 32], which further emphasizes
⁸the relevance of the increased prioritization of these compounds by modifying the
⁹dataset. Since none of the four antiviral compounds is directly connected to any of
¹⁰the 27 SARS-CoV2 entities in the graph, this illustrates that the modified dataset
¹¹is better suited to generate new insights and prioritize drugs for repurposing than
¹²the original dataset, even when the disease does not yet have any treatment present
¹³in the graph at all.

¹⁴ However, as indicated by comparing the results of the two datasets, for the degree
¹⁵of downsampling as well as for the performance, the improvements generated by this
¹⁶downsampling method are greater in Hetionet than they are for DRKG. Compared
¹⁷to Hetionet, the relative reduction of entities in the graph as well as the relative
¹⁸increase in the average entity degree is smaller in DRKG (see Table 1). This is
¹⁹possibly due to the fact that entities of the types *Compound* and *Disease* as well
²⁰as *treats*-relations are severely underrepresented in Hetionet while they represent
²¹a large part of the graph in DRKG (see Figure 1). This indicates that the effect
²²of the metapath-based filtering is more pronounced the more the relevant entities
²³and relations are underrepresented in the graph. Over time KGs will expand into
²⁴an increasing number of domains and contain diversified sets of entity types and
²⁵relations. This will lead to individual relations and entity types being increasingly
²⁶underrepresented in comparison to the whole graphs. Since most real world use
²⁷cases like drug repurposing focus on a small set of relations, underrepresentation of
²⁸the relevant relations will be a widespread problem. Especially since in most KG
²⁹scenarios scarce data of interest is enriched with large quantities of supporting data,
³⁰the portion of the data that is most valuable for the target domain is almost always
³¹heavily underrepresented. Therefore, methods to alleviate this underrepresentation
³²problem and to identify and remove irrelevant information on a per-task basis will
³³be crucial. Finally, the modification of future KGs will be necessary not only from
³⁴a theoretical and performance point of view, but also regarding practical aspects
³⁵like training time and memory requirements. If the information gathered in KGs
³⁶expands to an increasing amount of domains, better methods to carve out task-
³⁷specific subgraphs from these emerging supergraphs will become indispensable.

³⁸

³⁹4 Conclusion

⁴⁰We present a metapath-based approach to KG filtering which allows one to include
⁴¹domain knowledge in the form of metapaths. The selection of metapaths can be
⁴²adapted to modify a KG and tailor it for a specific task, which we demonstrate by
⁴³applying it for the task of drug repurposing. We present evidence that a proportion
⁴⁴of up to 60% of a KG can be removed by this filtering method while simultaneously
⁴⁵improving the performance in the task by up to 40%. This result demonstrates that
⁴⁶the removed information is in fact counterproductive to learning embeddings for

¹the specific task. During training, the loss on the modified graph is higher, while
²simultaneously the link prediction performance for drug repurposing is improved.
³This indicates that, while the model has worse performance on the average task,
⁴it has an improved performance in the relevant task, i.e. drug repurposing. This
⁵is in part attributed to the entity type of relevance becoming more frequent, an
⁶increase in the average degree and the samples drawn during training becoming
⁷more informative to the task. Furthermore it is shown that the models trained on
⁸the modified graph reach equal performance earlier than the models trained on the
⁹original graphs, which emphasizes a higher learning efficiency, while simultaneously
¹⁰consuming less resources. Given the improved prediction performance in drug repur-
¹¹posing in general and in the specific case of antiviral compounds and SARS-CoV2
¹²it can be concluded that the task-specific modified graph is less general and more
¹³specific to the use case than the original graph.

¹⁴ Future work can be applied in several directions. While we show that KGs contain
¹⁵counterproductive information that can be reduced to improve performance in a
¹⁶certain task, this has to be done as a preliminary step before training the model.
¹⁷Allowing a model to prune a graph by removing entities or gradually reducing
¹⁸their impact during training, possibly guided by domain knowledge, would be the
¹⁹next logical step. A similar approach has recently been developed for graph neural
²⁰networks [33]. A different direction of future work is to further investigate the trade-
²¹off between different intensities of filtering and changes in performance. This would
²²be necessary to formulate the filtering as an optimization problem and to possibly
²³find global optima for a given task and scenario.

²⁴**5 Methods**

²⁵**5.1 Task-Driven Metapath Filtering**

²⁶We propose a method to filter a KG based on a set of metapaths M . The metapaths
²⁷ $m_1, m_2, \dots, m_l \in M$ can be chosen to incorporate domain knowledge in the filtering
²⁸process. They further enable us to constrain the maximum distance between every
²⁹node in the graph and an entity of the types of interest. In the example of drug
³⁰repurposing, the entity types of interest are Compounds and Diseases. Thus, we
³¹can choose the metapaths so that all the entities in the graph are within a certain
³²distance of Compounds and Disease and are necessarily connected to them via
³³entities of the types of our choice.

³⁴Formally, a metapath m is a defined sequence of entity types $t \in T$. Given a
³⁵metapath $m = (t_1, t_2, \dots, t_{n-1}, t_n)$ of length n , a random metapath walker can start
³⁶at any entity e_1 of the type t_1 . In the next step, the walker samples all entities
³⁷of the type t_2 in the neighbourhood $N(e_1)$ of e_1 and selects one at random. Now
³⁸the walker samples all entities of the type t_3 in the neighbourhood $N(e_2)$ of e_2
³⁹and selects one at random again. This procedure is repeated until the walker either
⁴⁰reaches an entity e_n of type t_n or arrives at an entity e_i that has no entity e_{i+1} of
⁴¹the required type t_{i+1} in his neighbourhood $N(e_i)$. In both cases, the walker stops
⁴²and starts a new, unrelated walk at an entity of type t_1 . After a the walker started
⁴³a predefined number of starts at each entity of type t_1 , the procedure is repeated
⁴⁴and the metapath is traversed in the opposite direction, starting at entity type t_n .

⁴⁵This procedure yields a set of complete walks $\mathcal{W}_{complete}$, where the walker reached
⁴⁶an entity e_n of type t_n , and a set of prematurely stopped walks $\mathcal{W}_{stopped}$, where

Table 3 Hetionet and DRKG Entity Types. All figures are before any preprocessing has taken place.
[6, 7]

Entity Type	Hetionet	DRKG
Anatomy	402	400
Atc	0	4,048
Biological Process	11,381	11,381
Cellular Component	1,391	1,391
Compound	1,552	24,313
Disease	137	5,103
Gene	20,945	39,220
Molecular Function	2,884	2,884
Pathway	1,822	1,822
Pharmacologic Class	345	345
Side Effect	5,734	5,701
Symptom	438	415
Tax	0	215
Total	47,031	97,238

the walker stopped at an entity e_i that has no required next node in its neighborhood. To avoid small disconnected graphs, two walks in $\mathcal{W}_{complete}$ are concatenated if the last entity in one walk is also the first entity in another walk. This concatenation procedure is continued until a sufficient length of the concatenated walk is reached, which is then placed in $\mathcal{W}_{concatenated}$. Now, a subset of entities $\mathcal{E}' = \{e | e \in W \text{ for some walk } W \in \mathcal{W}_{concatenated}\}$ is taken, clearly $\mathcal{E}' \subset \mathcal{E}$. In other words, only entities e that appear at least once in the concatenated walks $\mathcal{W}_{concatenated}$ are assigned to $\mathcal{E}'$. This subset $\mathcal{E}'$ is further called the modified version of the original set of entities $\mathcal{E}$. Thus, a new task specific graph $\mathcal{KG}' \subset \mathcal{E}' \times \mathcal{R}' \times \mathcal{E}'$ is created which is a modification of the original graph $\mathcal{KG} \subset \mathcal{E} \times \mathcal{R} \times \mathcal{E}$.

5.2 Experimental Setup

To assess the effectiveness of our method, we have produced modified versions of two datasets by applying walk-based filtering as described in section 5.1. The effect on both of the datasets has been evaluated by training five different KGE models on the original datasets and on the task specific modified datasets.

5.2.1 Datasets

Hetionet is a biological KG constructed from the integration of 29 publicly available databases [5, 6]. From these databases, a heterogeneous graph has been constructed following the schema displayed in Figure 5 A. In total it consists of 47,031 entities of 11 different types (see Table 3), which are connected by 2,250,197 relations of 24 types (see Figure 3). DRKG [7] is a superset of several KGs from the biomedical domain and is the largest compilation of biomedical knowledge in graph form to date. It integrates Hetionet, GNBR, DrugBank, String, IntAct, DGIdb and individual publications into one comprehensive biological KG with 97,238 entities of 13 types (see Table 3) and 5,874,261 relations of 107 types (see Figure 5 B).

5.2.2 Dataset Preprocessing

Removal of ambiguous relations To reduce ambiguity in the results, all relations between the entities that are used during evaluation should be of the desired type. In our case, the entity types are *Compound* and *Disease*, and the desired relation between these entity types is *treats*. All relations between entities of the type *Compound* and *Disease* that are not *treats* are removed. In Hetionet, this is only

removes the *palliates* relation. In DRKG, the removed relations are *Pa* (alleviates, reduces), *Sa* (side effect/adverse event), *J* (role in disease pathogenesis), *Mp* (biomarkers of disease progression) from GNBR in addition to the *palliates* relations from Hetionet. The remaining relations between compounds and diseases are *T* (treatment/therapy), *C* (inhibits cell growth), *Pr* (prevents, suppresses) from GNBR, *treats* from Drugbank and *treats* from Hetionet. Although *C* and *Pr* from GNBR are not worded as treatments, their character is more curative than palliative and are therefore included into the analysis. During this step, 21,447 triples are removed from DRKG and 390 triples are removed from Hetionet.

Unification of related relations for prediction In the case of DRKG, there are several relations between entities of the type *Compound* and *Disease* that are described as *treats* or similar concepts. To allow the information learned from one of these relations to be transferred to the others, all remaining non-ambiguous relations between entities of the type *Compound* and *Disease* are replaced by a unifying *treats* relation. Since there is an overlap of the various *treats*-like relations, unifying them removes another 1,793 triples from the graph. After preprocessing, DRKG contains 96,121 entities, 4952 of which are diseases and 23347 are compounds.

Introduction of inverse relations Since most KGE models represent relations in a directed fashion, inverse relations have been introduced into the graphs. This means that for every relation r there is an inverse relation r^{-1} that is connected to the same entities, but in the opposite direction. For example, in addition to the triple $(Compound, treats, Disease)$, there now also exist triples in the opposite direction $(Disease, treats^{-1}, Compound)$.

5.2.3 Metapath Selection

The metapaths selected for use in our method are based on those used by [6], but omitting the relations and only considering sequences of entity types (see Additional File 2 — Table S4). For DRKG, three additional metapaths have been added in the

¹direct neighbourhood of the *Compound* entity type to also include the additional
²entity type *Atc* (see Additional File 2 — Table S5). The metapaths have been
³traversed in both directions, once starting from the compounds and once from the
⁴diseases. The number of starts for every compound and every disease has been set
⁵to 1000 per metapath, the length of the concatenated walks is set to 100 entities
⁶and the number of concatenated walks gathered per metapath is set to 5000.

⁷

⁸5.2.4 KGE Methods

⁹We trained five different KGE Models: TransE [9], RESCAL [10], DistMult [11],
¹⁰ComplEx [13] and ConvE [12]. Translational methods, such as the seminal work
¹¹TransE [9], embed both entities and relations into the same vector space. Subse-
¹²quently, translational methods consider additive functions to model the actions of
¹³relations on entities. Factorization-based approaches such as RESCAL [10], Dist-
¹⁴Mult [11], or ComplEx [13] define multiplicative scoring functions on the entity
¹⁵embeddings. Specifically, each relations $r \in \mathcal{R}$ induces a parametrized bi-linear
¹⁶form $(\cdot, \cdot)_r : R^d \times R^d \rightarrow R$. Next to methods that combine graph neural networks
¹⁷and KGE methods, the method ConvE [12] is commonly employed for KBC. ConvE
¹⁸[12] learns 2D-convolutional networks over the embedding space to obtain expres-
¹⁹sive feature maps that serve as input to a bilinear product that scores candidate
²⁰triples.

²¹

²²5.2.5 Performance Measure

²³The link prediction performance is evaluated only on triples of the type $(Compound, treats, Disease)$
²⁴and $(Disease, treats^{-1}, Compound)$. Negative samples are generated by permuting
²⁵the tail entities and restricting the permuted tail entities to be of the same type as
²⁶the original tail entity (i.e. for $(Compound, treats, Disease)$, we permute the tail
²⁷entity only with other Diseases). The metric chosen for evaluation is the filtered
²⁸Mean Reciprocal Rank (MRR):

²⁹

$$³⁰ \quad MRR = \frac{1}{|Q|} \sum_{i=1}^Q \frac{1}{rank_i} \quad (1)$$

³¹

³²where Q is the number of queries and $rank_i$ is the rank of the i th cor-
³³rect result according to the scoring function $(h, t)_r$. In our scenario, we have
³⁴one query for every true triple of the types $(Compound, treats, Disease)$ and
³⁵ $(Disease, treats^{-1}, Compound)$. We compare this true triple to false triples with
³⁶the same head entity and relationship, but permuted tail entities. The permuted
³⁷tail entities are all entities in the graph with the same type as the true tail entity
³⁸which are not connected to the head entity by the given relation. Then, predicted
³⁹probabilities are sorted in descending order and the reciprocal of the rank of the
⁴⁰one true triple. The mean of the reciprocals is the MRR. An MRR of 1 means that
⁴¹for every query, the correct result is always on rank 1. It is called filtered because
⁴²each true triple is compared individually against all the false triples (i.e. observed
⁴³triples are excluded from the ranking). Thus, multiple true triples for a single head
⁴⁴entity don't influence each other's ranks.

⁴⁵ Additionally, we only consider entities of the types *Compound* and *Disease* for
⁴⁶evaluation that occur in both the full and the filtered graph, regarding both the

¹true and the permuted triples. To evaluate the difficulty of the prediction task we
²include the performance of a random classifier. To obtain the random classifier
³results, the scores of a hyperparameter-optimized model for all compound-disease
⁴pairs are taken. Then, the ground truth adjacency matrix between Compounds
⁵and Diseases is randomized and the performance of the scores on the randomized
⁶ground truth is evaluated. This procedure is repeated 100 times and the mean of
⁷the performances is taken.

⁸

⁹5.2.6 Validation

¹⁰Prior to our metapath-based filtering, two holdout sets have been taken for both
¹¹of the datasets. The first holdout set is used for the selection of optimal hyper-
¹²parameters (validation), whereas the second holdout set is used to assess the final
¹³performance per model, after all parameters and hyperparameters are fixed (test).
¹⁴On Hetionet, these holdout sets each contain 12.5% of the *treats* relations and their
¹⁵inverses each, whereas on DRKG they each contain 10% of the *treats* relations and
¹⁶their inverses. The remainder of the *treats*-relations and all other triples in the
¹⁷graph are used for training. If an entity is removed by the metapath-based filtering
¹⁸procedure, then all triples containing this entity are removed from all holdout sets,
¹⁹resulting in identical holdout sets for the original and the modified datasets.

²⁰

²¹5.2.7 Post-Training Analyses

²²To better understand the impact that the filtering procedure has on the training
²³and on the final embeddings, comparable models are chosen after training is finished
²⁴for further inspection. The models have to produce good results on both Hetionet
²⁵and DRKG in both the original and modified setting to be chosen. Additionally, the
²⁶hyperparameter optimization (HPO) has to be completed and the best performing
²⁷model after HPO is chosen for the post-training analyses.

²⁸

²⁹*Effect on Negative Sampling* During training, negative samples are drawn from the
³⁰graph by corrupting existing triples, i.e. positive samples. Although the truth value
³¹of these negative samples is unknown, they are being treated as false during training
³²(local closed world assumption). The training loss is determined based on comparing
³³the predicted probabilities of the positive (existing) samples to the sampled negative
³⁴(not existing) samples. Thus, the samples drawn during negative sampling can have
³⁵a non-negligible effect on the training. The effect of our metapath-based filtering
³⁶procedure on the samples drawn during negative sampling is examined by drawing
³⁷200,000 positive and 200,000 negative samples in batches of 2,000 each. The batches
³⁸are scored using a trained model. The predictions are scaled to be in the interval
³⁹ $[0, 1]$ where 0 indicates a low probability of the triple being true and 1 indicates a
⁴⁰high probability of the triple being true. The resulting distributions are compared
⁴¹visually and by entropy.

⁴²

⁴³*Case-Study SARS CoV2* COVID-19 is a current global health challenge and the
⁴⁴search for optimal treatment options is still ongoing. It is caused by the severe acute
⁴⁵respiratory syndrome coronavirus-2 (SARS CoV2). To better understand how the
⁴⁶change in model performance after filtering can translate into insights, the final

models will be compared in how they rank four compounds that have been in the discussion as potential treatments of the disease and all 27 SARS CoV2-related disease entities in DRKG. The chosen compounds are danoprevir, favipiravir, lopinavir and ritonavir, which are all currently investigated as antiviral drugs in the treatment of COVID-19 [34, 35, 32]. The antiviral compound remdesivir is not included in the analysis because it is not present in DRKG. Since the mechanistic target of the antiviral compounds, the RNA-dependent RNA polymerase (RdRp) [36], is not present in DRKG, we focus on the SARS-CoV2 Spike protein entity's rank and the mean rank over all 27 SARS-CoV2 entities. None of the SARS-CoV2 entities have an adjacent *treats* relation, thus the dataset reflects that they have no known treatment. Each SARS-CoV2 entity is ranked against each of the 19633 remaining compound entities in triples of the form (*SARSCoV2entity*, *treats*⁻¹, *Compound*). The rank of the antiviral compounds compared to all the other compounds in the graph is noted. A low rank reflects a high priority, i.e. a high probability of a *treats*⁻¹ relation between the disease and compound entities.

Declarations

Funding

Funding has been provided by Siemens Healthineers, Erlangen, Germany and Siemens AG, Munich, Germany. The funding bodies did not play any roles in the design of the study and collection, analysis, and interpretation of data and in writing the manuscript.

Availability of data and materials

The data that is used in this work is available for download at the original author's terms.

For Hetionet, see <https://github.com/hetio/hetionet>

For DRKG, see <https://github.com/gnn4dr/DRKG>

Competing interests

The authors declare that they have no competing interests.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Authors' contributions

FR developed the method, conducted the experiments and analyses and prepared the manuscript. MJ, MR and MH provided guidance during the development and contributions to the manuscript. All authors read and approved the final manuscript.

Acknowledgements

Not applicable.

Author details

¹Digital Technology & Innovation, Siemens Healthineers, Erlangen, Germany. ²Corporate Technology, Siemens AG, Munich, Germany.

References

- Ashburn, T.T., Thor, K.B.: Drug repositioning: identifying and developing new uses for existing drugs. *Nat Rev Drug Discov* **3**(8), 673–683 (2004)
- Pushpakom, S., Iorio, F., Eyers, P.A., Escott, K.J., Hopper, S., Wells, A., Doig, A., Guilliams, T., Latimer, J., McNamee, C., Norris, A., Sanseau, P., Cavalla, D., Pirmohamed, M.: Drug repurposing: progress, challenges and recommendations. *Nat Rev Drug Discov* **18**(1), 41–58 (2019)
- Nosengo, N.: Can you teach old drugs new tricks? *Nature* **534**(7607), 314–316 (2016)
- Hurle, M.R., Yang, L., Xie, Q., Rajpal, D.K., Sanseau, P., Agarwal, P.: Computational drug repositioning: from data to therapeutics. *Clin Pharmacol Ther* **93**(4), 335–341 (2013)
- Himmelstein, D.S., Baranzini, S.E.: Heterogeneous Network Edge Prediction: A Data Integration Approach to Prioritize Disease-Associated Genes. *PLoS Comput Biol* **11**(7), 1004259 (2015). doi:10.1371/journal.pcbi.1004259

6. Himmelstein, D.S., Lizée, A., Hessler, C., Brueggeman, L., Chen, S.L., Hadley, D., Green, A., Khankhanian, P., Baranzini, S.E.: Systematic integration of biomedical knowledge prioritizes drugs for repurposing. *eLife* **6**, 26726 (2017). doi:10.7554/eLife.26726
7. Ioannidis, V.N., Song, X., Manchanda, S., Li, M., Pan, X., Zheng, D., Ning, X., Zeng, X., Karypis, G.: DRKG - drug repurposing knowledge graph for covid-19 (2020)
8. Paulheim, H.: Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic web* **8**(3), 489–508 (2017)
9. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: *NeurIPS*, vol. 26 (2013).
https://proceedings.neurips.cc/paper/2013/file/1cecc7a77928ca8133fa24680a88d2f9-Paper.pdf
10. Nickel, M., Tresp, V., Kriegel, H.-P.: A three-way model for collective learning on multi-relational data. In: *ICML* (2011)
11. Yang, B., Yih, W., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases. In: Bengio, Y., LeCun, Y. (eds.) *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (2015).
http://arxiv.org/abs/1412.6575
12. Dettmers, T., Minervini, P., Stenetorp, P., Riedel, S.: Convolutional 2d knowledge graph embeddings. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32 (2018)
13. Trouillon, T., Dance, C.R., Gaussier, E., Welbl, J., Riedel, S., Bouchard, G.: Knowledge graph completion via complex tensor factorization. *J. Mach. Learn. Res.* **18**(1), 4735–4772 (2017)
14. Vrandečić, D., Krötzsch, M.: Wikidata: A free collaborative knowledgebase. *Commun. ACM* **57**(10), 78–85 (2014). doi:10.1145/2629489
15. Navigli, R., Ponzetto, S.P.: Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artif. Intell.* **193**, 217–250 (2012).
doi:10.1016/j.artint.2012.07.001
16. Suchanek, F.M., Kasneci, G., Weikum, G.: Yago: A core of semantic knowledge. In: *Proceedings of the 16th International Conference on World Wide Web. WWW '07*, pp. 697–706. ACM, New York, NY, USA (2007).
doi:10.1145/1242572.1242667. https://doi.org/10.1145/1242572.1242667
17. Himmelstein, D.S., Baranzini, S.E.: Heterogeneous Network Edge Prediction: A Data Integration Approach to Prioritize Disease-Associated Genes. *PLoS Comput Biol* **11**(7), 1004259 (2015)
18. Nováček, V., Mohamed, S.K.: Predicting Polypharmacy Side-effects Using Knowledge Graph Embeddings. *AMIA Jt Summits Transl Sci Proc* **2020**, 449–458 (2020)
19. Zhu, Y., Che, C., Jin, B., Zhang, N., Su, C., Wang, F.: Knowledge-driven drug repurposing using a comprehensive drug knowledge graph. *J Health Inform* **26**(4), 2737–2750 (2020).
doi:10.1177/1460458220937101
20. 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. *Pac Symp Biocomput* **25**, 463–474 (2020)
21. Zhu, S., Bing, J., Min, X., Lin, C., Zeng, X.: Prediction of Drug–Gene Interaction by Using Metapath2vec. *Front Genet* **9**, 248 (2018). doi:10.3389/fgene.2018.00248
22. Zeng, X., Ding, N., Rodríguez-Patón, A., Zou, Q.: Probability-based collaborative filtering model for predicting gene–disease associations. *BMC Medical Genom* **10**(S5), 76 (2017). doi:10.1186/s12920-017-0313-y
23. Zitnik, M., Agrawal, M., Leskovec, J.: Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics* **34**(13), 457–466 (2018). doi:10.1093/bioinformatics/bty294
24. Riva, L., Yuan, S., Yin, X., Martin-Sancho, L., Matsunaga, N., Pache, L., Burgstaller-Muehlbacher, S., De Jesus, P.D., Teriete, P., Hull, M.V., Chang, M.W., Chan, J.F., Cao, J., Poon, V.K., Herbert, K.M., Cheng, K., Nguyen, T.H., Rubanov, A., Pu, Y., Nguyen, C., Choi, A., Rathnasinghe, R., Schotsaert, M., Miorin, L., DeJozse, M., Zwaka, T.P., Sit, K.Y., Martinez-Sobrido, L., Liu, W.C., White, K.M., Chapman, M.E., Lendy, E.K., Glynn, R.J., Albrecht, R., Rupp, E., Mesecar, A.D., Johnson, J.R., Benner, C., Sun, R., Schultz, P.G., Su, A.I., García-Sastre, A., Chatterjee, A.K., Yuen, K.Y., Chanda, S.K.: Discovery of SARS-CoV-2 antiviral drugs through large-scale compound repurposing. *Nature* **586**(7827), 113–119 (2020)
25. Mohamed, K., Yazdanpanah, N., Saghaideh, A., Rezaei, N.: Computational drug discovery and repurposing for the treatment of COVID-19: A systematic review. *Bioorg Chem* **106**, 104490 (2021)
26. Hsieh, K.L., Wang, Y., Chen, L., Zhao, Z., Savitz, S., Jiang, X., Tang, J., Kim, Y.: Drug Repurposing for COVID-19 using Graph Neural Network with Genetic, Mechanistic, and Epidemiological Validation. *Res Sq* (2020)
27. Zhang, R., Hristovski, D., Schutte, D., Kastrin, A., Fiszman, M., Kilicoglu, H.: Drug repurposing for covid-19 via knowledge graph completion. *J Biomed Inform* **115**, 103696 (2021). doi:10.1016/j.jbi.2021.103696
28. Peter Richardson, C.T.D.S.O.O.A.P.M.R.E.S.J.S. Ivan Griffin: Baricitinib as potential treatment for 2019-ncov acute respiratory disease. *Lancet Correspondence* **395**, 30 (2020)
29. Horby, P.W., Mafham, M., Bell, J.L., Linsell, L., Staplin, N., Emberson, J., Palfreeman, A., Raw, J., Elmahi, E., Prudon, B., Green, C., Carley, S., Chadwick, D., Davies, M., Wise, M.P., Baillie, J.K., Chappell, L.C., Faust, S.N., Jaki, T., Jefferey, K., Lim, W.S., Montgomery, A., Rowan, K., Juszczak, E., Haynes, R., Landray, M.J.: Lopinavir-ritonavir in patients admitted to hospital with COVID-19 (RECOVERY): a randomised, controlled, open-label, platform trial. *Lancet* (2020)
30. Taher, M., Tik, N., Susanti, D.: Drugs intervention study in COVID-19 management. *Drug Metab Pers Ther* (2021)
31. Dong, Y., Shamsuddin, A., Campbell, H., Theodoratou, E.: Current COVID-19 treatments: Rapid review of the literature. *J Glob Health* **11**, 10003 (2021)
32. Chen, H., Zhang, Z., Wang, L., Huang, Z., Gong, F., Li, X., Chen, Y., Wu, J.J.: First clinical study using HCV protease inhibitor danoprevir to treat COVID-19 patients. *Medicine (Baltimore)* **99**(48), 23357 (2020)
33. Zheng, C., Zong, B., Cheng, W., Song, D., Ni, J., Yu, W., Chen, H., Wang, W.: Robust graph representation

- 1 learning via neural sparsification. In: Proceedings of the 37th International Conference on Machine Learning
2 (ICML), pp. 11458–11468 (2020)
34. Liu, W., Zhou, P., Chen, K., Ye, Z., Liu, F., Li, X., He, N., Wu, Z., Zhang, Q., Gong, X., Tang, Q., Du, X.,
3 Ying, Y., Xu, X., Zhang, Y., Liu, J., Li, Y., Shen, N., Couban, R.J., Ibrahim, Q.I., Guyatt, G., Zhai, S.: Efficacy
4 and safety of antiviral treatment for COVID-19 from evidence in studies of SARS-CoV-2 and other acute viral
5 infections: a systematic review and meta-analysis. *CMAJ* **192**(27), 734–744 (2020)
35. Zhang, Z., Wang, S., Tu, X., Peng, X., Huang, Y., Wang, L., Ju, W., Rao, J., Li, X., Zhu, D., Sun, H., Chen,
6 H.: A comparative study on the time to achieve negative nucleic acid testing and hospital stays between
7 danoprevir and lopinavir/ritonavir in the treatment of patients with COVID-19. *J Med Virol* **92**(11), 2631–2636
8 (2020)
36. Abdelnabi, R., Morais, A.T.S.d., Leyssen, P., Imbert, I., Beaucourt, S., Blanc, H., Froeyen, M., Vignuzzi, M.,
9 Canard, B., Neyts, J., Delang, L.: Understanding the mechanism of the broad-spectrum antiviral activity of
10 favipiravir (t-705): Key role of the f1 motif of the viral polymerase. *J Virol* **91**(12) (2017).
doi:10.1128/JVI.00487-17. <https://jvi.asm.org/content/91/12/e00487-17.full.pdf>

Additional Files

- 11 Additional file 1 — Table S1: Effects of Downsampling per Entity Type in Hetionet.
- 12 Additional file 1 — Table S2: Table S2: Effects of Downsampling per Entity Type in DRKG.
- 13 Additional file 1 — Table S3: Effects of Downsampling on the ranking of antiviral compounds against SARS
14 CoV2-entities.
- 15 Shown is the rank of a given SARS-CoV2 entity with a given antiviral compound when the model is trained on the
16 full dataset versus when it is trained on the filtered dataset. The SARS-COV2 entities are ranked against all
17 remaining 19,633 compound entities by connecting them with a *treats* relation. Rank 1 means the highest
18 probability that the two entities are connected by a *treats* relation, rank 19,633 means the lowest probability, rank
19 9,816 corresponds to the median probability.
- 20 Additional file 2 — Table S4: Metapath Selection
21 Metapaths that have been selected for random metapath walking. Denoted are the entity types traversed.C:
22 Compound, D: Disease, G: Gene, A: Anatomy, PC: Pharmacologic Class, S: Symptom,BP: Biological Process, MF:
23 Metabolic Function, PW: Pathway, SE: Side Effect
- 24 Additional file 2 — Table S5: Additional Metapaths for DRKG.
- 25 Metapaths selected for DRKG in addition to the metapaths denoted in Table S4. C: Compound, Atc: Atc, D:
26 Disease, G: Gene

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [AdditionalFile1.pdf](#)
- [AdditionalFile2.pdf](#)