DRUG REPURPOSING FOR MULTIPLE COVID STRAINS USING COLLABORATIVE FILTERING

Otto Kißig[†], **Martin Taraz**[†], **Sarel Cohen**[‡], **Vanja Doskoč**[‡] **& Tobias Friedrich**[‡]
Hasso Plattner Institute, University of Potsdam, Germany
[†]firstname.lastname@student.hpi.de, [‡]firstname.lastname@hpi.de

ABSTRACT

The ongoing COVID-19 pandemic demands for a swift discovery of suitable treatments. The development of completely new compounds for such a novel disease is a challenging, time intensive process. This amplifies the relevance of *drug repurposing*, a technique where existing drugs are used to treat other diseases. A common bioinformatical approach to this is based on *knowledge graphs*, which compile relationships between drugs, diseases, genes and other biomedical entities. Then, graph neural networks (GNNs) are used for the drug repurposing task as they provide a good link prediction performance on such knowledge graphs. Building on state-of-the-art GNN research, Doshi & Chepuri (2020) construct the remarkable model DR-COVID. We re-implement their model and extend the approach to perform significantly better. We propose and evaluate several strategies for the aggregation of link predictions into drug recommendation rankings. With the help of clustering of similar target diseases we improve the model by a substantial margin, compiling a top-100 ranking of candidates including 32 currently being in COVID-19-related clinical trials. Regarding the re-implementation, we offer more flexibility in the selection of the graph neighborhood sizes fed into the model and reduce the training time significantly by making use of data parallelism.

1 INTRODUCTION

1.1 PREFACE

With COVID-19, a global pandemic with severe socio-economic implications impacting nearly every part of our daily lives is active (Nicola et al., 2020). The surprising nature and the rapid spread makes finding an effective treatment as urgent as challenging, since the disease-specific knowledge is limited in the beginning and the pandemic costs additional lives every day. Because known and approved drugs are already well-studied, they pose a good starting point for accelerated development of treatments, and an emerging tactic in fighting COVID-19 (Shah et al., 2020). DrugBank, which is a comprehensive web resource containing structured information about drugs approved by the US Food and Drug Administration as well as experimental drugs, contained more than $2\,300$ approved drugs and over $4\,500$ experimental drugs as of 2018; both numbers are strongly increasing (Wishart et al., 2018). This amount of data emphasises the need for computer aided discovery and development of treatments.

Drug repurposing with knowledge graphs, as described by Ashburn & Thor (2004), is the current state-of-the-art technique for utilizing machine learning to predict whether known drugs are a possible treatment for new diseases. Applying drug repurposing allows for a better way to maneuver through the pandemic. It can lead to better treatments for patients infected with one of the COVID-19 strains and a better understanding of the mechanism of the individual COVID-19 diseases. The concept of drug repurposing has been first described by Ashburn & Thor (2004). However, today we approach the problem using machine learning methods, focusing on deep learning approaches. The idea of predicting unknown links between entities in a knowledge graph is traditionally known as *Collaborative Filtering*, as described by Sarwar et al. (2001). In this work we build on the idea of *graph embeddings*, which map a fixed-size feature vectors to graph nodes and relations. A state-of-the-art technique for the creation of such embeddings based on deep neural networks (DNNs) is TRANSE, which was proposed by Bordes et al. (2013).

Regarding the specific application of drug repurposing relying on edge prediction in a knowledge graph of biomedical data (see Section 2), Gysi et al. (2020) present a novel classification approach to this problem by implementing and merging various different ideas and techniques into one ensemble classifier. At its core, they deploy a DNN with a encoder-decoder structure. The encoder mechanism of it, which is based on parts of the *Decagon* graph neural network by Zitnik et al. (2018), was initially proposed for the prediction of side effects of concurrent drug use.

1.2 Our Contribution

This work offers two main contributions to the deep learning and the bioinformatics community.

1. We improve the results of Doshi & Chepuri (2020) by applying a fine tuned clustering mechanism as a post prediction step that increases the number of predicted drugs that indeed were or are in clinical trials today. In this way we improve the state-of-the-art. Additionally, we propose a well-specified method of creating top-100 predictions that yield possible treatment candidates, and clarify the process as a whole.
2. We re-implement[1] the model described by Doshi & Chepuri (2020) and improve it by offering more flexibility in the selection of the graph neighborhood sizes fed into the model. Furthermore, we reduce the training time significantly by making use of high data parallelity and introducing vectorized operations at the most performance-critical spots. Lastly, we improve its readability and usability by following the proposed idioms of the PyTorch authors (Paszke et al., 2019) as well as relying on well-tested utilities provided by scikit-learn (Pedregosa et al., 2011) rather than manual metric calculations.

2 Dataset

We rely on the Drug Repurposing Knowledge Graph (DRKG) by Ioannidis et al. (2020), which compiles data from different biomedical databases. It contains $97\,238$ entities belonging to 13 entity types and $5\,874\,261$ triplets belonging to 107 edge types. We restrict ourselves to 98 edge types between 4 entity types, namely gene, compound, anatomy and disease, which leaves us with a knowledge graph with $69\,036$ entities and $4\,885\,854$ edges. In particular, it contains drugs and substances as *compound* entities, as well as different COVID-19 variants as *disease* entities. The edge types include e.g. *compound-treats-disease* edges, which is the kind of edge our model predicts.

One part of DRKG are the precomputed TransE embeddings trained using `dgl-ke` by Zheng et al. (2020). To train our model to predict whether an edge in some *compound-treats-disease* relation exists, we have to create suitable training samples. To provide our model not only positive examples for training but also negative samples, for each positive edge we sample 30 negative edges in the dataset, which results in a ratio similar to Dr-COVID in order to ensure comparability. This process tries to account for the actual imbalance of edges and non-edges on the entire dataset. Note that, by doing so, the sets of edges included in the dataset are not complete, however, they are quite certain to be correct. Consequently, the positive edges are given a higher weight in the loss calculation, and the slightly higher number of negative edges (which are not certain to be truly negative) are given a lower weight. To prevent too much imbalance during individual loss computations on batches, we utilize a weighted random batch sampler that over-samples the positive samples yielding an expected ratio of $1 : 1.5$ of positive to negative samples in each batch.

3 Model Architecture

The architecture of our model is illustrated in Figure 1. It consists of a SIGN (Frasca et al., 2020) architecture encoder, which provides an embedding $y \in \mathbb{R}^{250}$ for each node of the graph. We apply $tanh$ to the output of each individual encoder step and forward it into our decoder. To each pair of two encodings y_u, y_v of two nodes u, v, the decoder assigns a score $s_{u,v} \in [0, 1]$. This score measures the probability for a *compound-treats-disease* edge between nodes u and v to exist. The decoder consists of two linear layers $\ell_1(u)$ and $\ell_2(v)$ that process the encodings y_u and y_v via a

[1] Our implementation of the experiments conducted and our re-implementation of the Dr-COVID model can be found here: `drive.google.com/file/d/12JJFe8wsfGrqq7IRPpWZkZDQl05TsqKQ`.

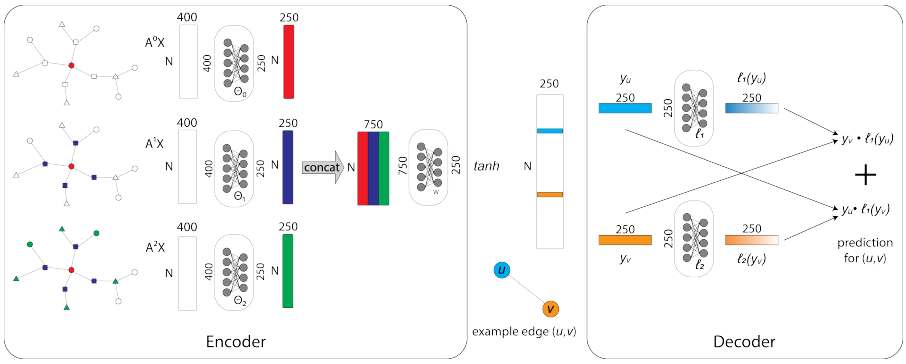

Figure 1: The architecture of our model as described in Section 3.

sigmoid function, that is, $\sigma(y_v \cdot \ell_1(y_u) + y_u \cdot \ell_2(y_v))$. The loss of the model is computed using a binary cross entropy loss with logits with weights to account for the imbalanced training dataset, as described in Section 2.

4 Output Interpretation

In this section we present different strategies of interpreting the scores that the model outputs for the application of predicting the top-k most promising compound nodes for a given set of disease nodes D. Note that this is important as there are multiple COVID-19 diseases. Let n be the total amount of compound nodes. Predicting all $n \cdot |D|$ edge combinations, our model yields a matrix of scores $S \in \mathbb{R}^{|D| \times n}$. For each of the following strategies we first perform a standardization of the scores per disease using $\hat{s}_{dc} = \frac{s_{dc} - \mu(s_{d*})}{\sigma(s_{d*})}$, with d being the index of a disease in D, c being the index of the compound, $\mu(s_{d*})$ and $\sigma(s_{d*})$ denoting the mean and standard deviation over all diseases.

As certain "mild" diseases may be affected by plenty compounds resulting in those being linked more likely, we apply the standardization to achieve a better comparability across different diseases. This allows us to identify the best suited compounds for every disease individually and compare those. However, this could also give good scores to some compounds in the case of diseases with no "good" scores in the first place, potentially yielding some less useful proposals.

An aggregation strategy takes our matrix of standardized scores $(\hat{s}_{dc})$ and derives a list of compounds from it, the top-k of which are our result. We propose the following aggregation strategies.

- **Global Score Mean:** We calculate the means of $(\hat{s}_{dc})$ along axis 0, that is, over all diseases per compound; then we sort the compounds by their respective scores and select the top-k.
- **Global Score Maximum:** We find the maxima of $(\hat{s}_{dc})$ along axis 0; then again we sort the compounds and select the top-k.
- **Union over Disease Rankings:** We calculate top-x compounds per disease with x as small as possible such that we get at least k unique compounds in the union over all diseases. We then concatenate all those top-x lists together to get a top-k compound list.
- **Cluster Score Maximum:** Grouping similar disease types that share traits can be used to enhance the accuracy of our top-k predictions. We perform such a grouping using the k-means clustering algorithms with different parameter settings for the number of clusters. For each cluster, which now represents a group of similar diseases, we use a mean reduction to calculate the score of a compound and then reduce to the maximum across these clusters. A sensible number of clusters to create can be chosen by performing a principal component analysis (PCA) (Pearson, F.R.S., 1901) on the standardized scores. For our data this indicates values of 2 to 3 (for more details see Appendix, Figure 2).
- **Union over Cluster Rankings:** We perform the top-x selection on clusters calculated with the clustering method described above. This not only allows us to use a greater x because we have fewer lists to pick from, but also to get more consistent top picks because of the internal averages that we apply inside each cluster.

5 EVALUATION

To test our compound ranking methods, we apply each to retrieve a top-100 list of proposed candidates. We then compute the number of intersections with the compounds that are part of a clinical trial related to COVID-19 according to the U.S. National Library of Medicine (World Health Organization). For this we use a compiled Kaggle dataset (Pandey, 2021) containing compound names.

We implement the model using PyTorch (Paszke et al., 2019). We train it using the AdamW optimizer (Kingma & Ba, 2015). The data is split using scikit-learn (Pedregosa et al., 2011) with 90% of the data as training data. The training is performed on Google Colab utilizing a Nvidia Tesla T4 and takes 30 minutes to prepare the graph dataset. We train our model using 5 epochs with a starting learning rate of 10^{-5} and a weight decay of 10^{-2}. Each training epoch took us 30 seconds which is a significant improvement over the 610 seconds of the implementation by Doshi & Chepuri (2020) and can be attributed to the exploitation of data parallelism in the decoder part of the model.

We compare the top-100 results of predicting compounds for `SARS-CoV2 E` of our obtained model to those predicted utilizing the weights of Doshi & Chepuri (2020). While their model's top-100 predictions include 22 compounds showing up in clinical trials, we only reach 10. We suspect the hand-made adjustments to the dataset utilizing undisclosed data sources are responsible for this discrepancy, as this is the sole missing part in our reimplementation. Consequently, we used their published weights for the evaluation of the post classification methods presented in Section 4.

The results of the the different post classification procedures can be found in Table 1. We see that our Union over Cluster Rankings with KMeans(k=3) outperforms the other approaches, yielding 32 hits. As indicated by the PCA on the standardized scores (see Figure 2 in the appendix) the choice of $k = 3$ is sensible. In contrast to that, DR-COVID's aggregation method, Union over Disease Rankings, reaches just 21 hits in our evaluation process.

Aggregation strategy	# hits
Single Disease (median)	20
Global Score Maximum	22
Global Score Mean	30
Cluster Score Maximum with KMeans(k=8)	18
Cluster Score Maximum with KMeans(k=3)	20
Union over Disease Rankings (DR-COVID, Doshi & Chepuri (2020))	21
Union over Cluster Rankings with KMeans(k=8)	24
Union over Cluster Rankings with KMeans(k=3) (our model)	**32**

Table 1: Hits of proposed candidates in actual clinical trials for different aggregation techniques.

6 CONCLUSION AND FUTURE WORK

Collaborative filtering can help the swift development of drugs in the face of a global pandemic. Rather than filtering the promising candidates by hand, one relies on graph neural networks. These build on top of comprehensive knowledge graphs connecting chemical compounds, diseases and individual genes over many different relationship types and can help with this task without actually understanding the semantic meaning of an individual relationship type: They are mainly good at detecting similar nodes in a graph. This makes them useful across many fields and in contexts beyond bioinformatics. As already shown by Ioannidis et al. (2020), predicting candidates for COVID-19 treatments using deep learning is a promising technique. We have been able to clarify the evaluation part of DR-COVID by Doshi & Chepuri (2020) and proposed an aggregation technique yielding slightly better results. Our own implementation improves both training speed as well as readability.

It remains open work to measure the effects of feeding the model with varying neighborhood information depths, which our own implementation of DR-COVID specifically enables. Furthermore, a thorough analysis of the types, stages and amount of clinical trials as well the role a drug plays for a study (e.g., main treatment, mitigation of side effects, etc.) remains to be conducted.

REFERENCES

Ted T. Ashburn and Karl B. Thor. Drug repositioning: identifying and developing new uses for existing drugs. *Nature Reviews Drug Discovery*, 2004.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Neural Information Processing Systems (NeurIPS)*, pp. 2787–2795, 2013.

Siddhant Doshi and Sundeep Prabhakar Chepuri. Dr-COVID: Graph neural networks for SARS-CoV-2 drug repurposing. *CoRR*, 2020.

Fabrizio Frasca, Emanuele Rossi, Davide Eynard, Ben Chamberlain, Michael Bronstein, and Federico Monti. SIGN: Scalable inception graph neural networks. *CoRR*, 2020.

Deisy Morselli Gysi, Ítalo Do Valle, Marinka Zitnik, Asher Ameli, Xiao Gan, Onur Varol, Helia Sanchez, Rebecca Marlene Baron, Dina Ghiassian, Joseph Loscalzo, and Albert-László Barabási. Network medicine framework for identifying drug repurposing opportunities for COVID-19. *CoRR*, 2020.

Vassilis N. Ioannidis, Xiang Song, Saurav Manchanda, Mufei Li, Xiaoqin Pan, Da Zheng, Xia Ning, Xiangxiang Zeng, and George Karypis. DRKG - drug repurposing knowledge graph for covid-19. `https://github.com/gnn4dr/DRKG/`, 2020.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

Maria Nicola, Zaid Alsafi, Catrin Sohrabi, Ahmed Kerwan, Ahmed Al-Jabir, Christos Iosifidis, Maliha Agha, and Riaz Agha. The socio-economic implications of the coronavirus pandemic (COVID-19): A review. *International Journal of Surgery*, 2020.

Parul Pandey. Covid19 clinical trials dataset. `https://www.kaggle.com/parulpandey/covid19-clinical-trials-dataset`, 2021. Retrieved February 19th, 2021.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. Curran Associates, Inc., 2019.

Karl Pearson, F.R.S. LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 1901.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 2011.

Badrul Munir Sarwar, George Karypis, Joseph A. Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *International Conference on World Wide Web*, 2001.

Bhumi Shah, Palmi Modi, and Sneha R. Sagar. In silico studies on therapeutic agents for COVID-19: Drug repurposing approach. *Life Sciences*, 2020.

David S. Wishart, Yannick D. Feunang, An Chi Guo, Elvis J. Lo, Ana Marcu, Jason R. Grant, Tanvir Sajed, Daniel Johnson, Carin Li, Zinat Sayeeda, Nazanin Assempour, Ithayavani Iynkkaran, Yifeng Liu, Adam Maciejewski, Nicola Gale, Alex Wilson, Lucy Chin, Ryan Cummings, Diana Le, Allison Pon, Craig Knox, and Michael Wilson. Drugbank 5.0: a major update to the drugbank database for 2018. *Nucleic Acids Res.*, 2018.

World Health Organization. International clinical trials registry platform (ictrp). `https://www.who.int/clinical-trials-registry-platform`. Online; accessed February 19th, 2021.

Da Zheng, Xiang Song, Chao Ma, Zeyuan Tan, Zihao Ye, Jin Dong, Hao Xiong, Zheng Zhang, and George Karypis. DGL-KE: training knowledge graph embeddings at scale. In *SIGIR Conference on Research and Development in Information Retrieval*, 2020.

Marinka Zitnik, Monica Agrawal, and Jure Leskovec. Modeling polypharmacy side effects with graph convolutional networks. *Bioinform.*, 2018.

A APPENDIX

In the appendix, one can find additional figures which, we hope, help understanding our submission.

To choose a sensible number of clusters to create, we perform a principal component analysis (PCA) (Pearson, F.R.S., 1901) on the standardized scores. This allows us embedding data points from $\hat{s}_{dc}$ into the 2d plane as shown in the pair plot in Figure 2. This visually suggests that 3 clusters among the diseases exist, which can be picked up by a 3-Means clustering.

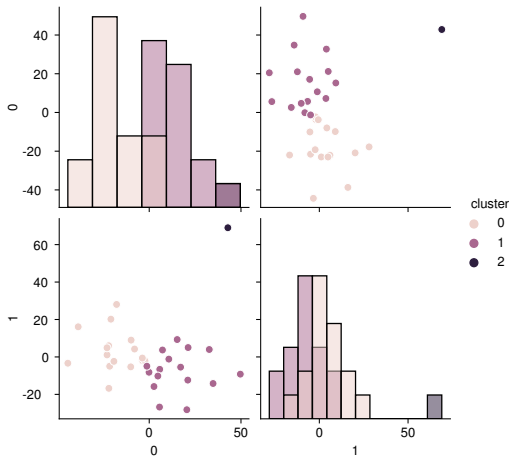

Figure 2: Pair plot showing the population of COVID-19 diseases in our dataset by their standardized scores reduced from n dimensions (the number of compounds) to two with the help of a PCA. The colors show the affiliation to a cluster derived from the application of a 3-Means clustering.