Zero-shot drug repurposing with geometric deep learning and clinician centered design

Kexin Huang[1,†,*], Payal Chandak[2,*], Qianwen Wang[1], Shreyas Havaldar[3], Akhil Vaid[3,4], Jure Leskovec[5], Girish Nadkarni[4], Benjamin S. Glicksberg[3,4], Nils Gehlenborg[1], and Marinka Zitnik[1,6,7,8,‡]

[1]Department of Biomedical Informatics, Harvard Medical School, Boston, MA 02115
[2]Harvard-MIT Program in Health Sciences and Technology, Cambridge, MA 02139
[3]Hasso Plattner Institute for Digital Health, Icahn School of Medicine at Mount Sinai, NY 10029
[4]Charles Bronfman Institute for Personalized Medicine, Icahn School of Medicine at Mount Sinai, NY 10029
[5]Department of Computer Science, Stanford University, Stanford, CA 94305
[6]Broad Institute of MIT and Harvard, Cambridge, MA 02142
[7]Harvard Data Science Initiative, Cambridge, MA 02138
[8]Kempner Institute for the Study of Natural and Artificial Intelligence, Harvard University, MA 02134

[†] Present address: Department of Computer Science, Stanford University
[*] Equal contribution
[‡] Corresponding author: marinka@hms.harvard.edu

Historically, drug repurposing – identifying new therapeutic uses for approved drugs – has been attributed to serendipity. While recent advances have leveraged knowledge graphs and deep learning to identify potential therapeutic candidates, their clinical utility remains limited due to their dependence on existing knowledge about diseases. Here, we introduce TxGNN, a geometric deep learning approach designed for "zero-shot" drug repurposing, enabling therapeutic predictions even for diseases with no existing medicines. Trained on a medical knowledge graph, TxGNN utilizes a graph neural network and metric-learning module to rank therapeutic candidates as potential indications and contraindications across 17,080 diseases. When benchmarked against eight leading methods, TxGNN significantly improves prediction accuracy for indications by 49.2% and contraindications by 35.1% under stringent zero-shot evaluation. To facilitate interpretation and analysis of the model's predictions, TxGNN's Explainer module offers transparent insights into the multi-hop paths that form TxGNN's predictive rationale. Clinicians and scientists found TxGNN's explanations instrumental in contextualizing and validating its predicted therapeutic candidates during our user study. Many of TxGNN's novel predictions have shown remarkable alignment with off-label prescriptions made by clinicians within a large healthcare system, affirming their real-world utility. TxGNN provides drug repurposing predictions that are more accurate than existing methods, consistent with off-label prescription decisions made by clinicians, and can be investigated through multi-hop interpretable explanations.

Introduction

The healthcare demands of billions globally underscore the pressing need to develop therapies for many diseases that currently lack treatments. Of over 7,000 rare diseases worldwide, only 5-7% of rare diseases have FDA-approved drugs[1]. In nations with aging populations, an enormous need for therapeutics lies in the growing burden of neurological disorders such as Parkinson's and Alzheimer's diseases[2]. Further, few therapeutics exist for neglected tropical diseases that affect populations in tropical and subtropical regions[3]. Leveraging existing therapies and expanding their use by identifying new therapeutic indications via drug repurposing can alleviate the global disease burden and address unmet disease needs. Drug repurposing can lead to significantly faster translation to the clinic and lower development costs than designing a novel drug from scratch since there is ample data about safety and efficacy of existing drugs[4] (Figure 1a). For instance, over 60% of therapies approved for a neglected tropical disease, leishmaniasis, have been repurposed[5,6]. The fundamental premise behind repurposing is that drugs can have pleiotropic effects beyond the mechanism of action of their direct targets[7]. Approximately 30% of FDA-approved drugs are issued at least one post-approval new indication, and many drugs have accrued over 10 indications throughout years[8]. However, most repurposed drugs are the result of serendipity[9,10]. Noticing off-label prescriptions of clinicians, as in the cases of gabapentin and bupropion, has led to many repurposed indications[10]. For other drugs, including sildenafil, indications that become the basis for repurposing are discovered fortuitously through patient experience[8]. The connection between drug candidates and their potential new applications is not identified systematically because the underlying mechanism 'connecting' them is either very intricate and unknown or dispersed and buried in a growing sea of information[9].

Owing to technological advances, the effects of drugs can now be prospectively matched to new indications by systematically analyzing medical knowledge graphs[7,11]. The new strategies rely on identifying therapeutic candidates based on their effects on cell signalling, gene expression, and disease phenotypes[7,12–14]. Machine learning has been used to analyze high-throughput molecular interactomes to unravel genetic architecture perturbed in disease[14,15] and help design therapies to target them[16]. To provide therapeutic predictions, geometric deep learning models optimized on large medical knowledge graphs[17] can extract disease signatures and match them to therapeutic candidates based on the proximity of therapeutic mechanisms to networks perturbed in disease[17–21].

Although computational approaches have identified promising repurposing candidates for complex diseases[18,22,23], there remain two key factors that, if addressed, could significantly enhance the clinical impact of repurposing predictions made by machine learning models. (1) First, existing methods assume that diseases for which we would like to make therapeutic predictions are well-understood and likely to have existing therapies. While this is certainly the case for some diseases (*e.g.*, hypertensive disorder is indicated with 103 medications across diverse patient populations[11]), there is a long tail of diseases that do not satisfy this assumption. Of 17,080 diseases examined in our study, 92% have no indications, and only 5.4% have more than one indicated medication. Despite the passage of the Orphan Drug Act in the United States in 1983 represented a launching point for a rare disease drug development revolution for these patients, around 95% of rare diseases have no FDA-approved drugs and up to 85% of rare diseases do not have even one drug developed that would show promise in rare disease treatment, diagnosis or prevention[24]. This long tail of diseases with few or no therapies and limited molecular understanding presents the most fruitful challenge clinically. (2) Second, a repurposed indication for a therapeutic candidate can be unrelated to the indication for which the drug was initially approved. Thalidomide was originally proposed to help with morning sickness during pregnancy and then retracted for causing congenital disabilities[10]. It was repurposed in 1964 for an autoimmune complication of leprosy and again in 2006 for multiple myeloma[10]. Another example of therapeutic repositioning, sildenafil was originally studied for angina and hypertension but later repurposed for erectile dysfunction[10]. Collectively, we refer to these challenges as the zero-shot drug repurposing problem (Figure 1b).

To be clinically useful, machine learning models must make "zero-shot" predictions; that is, they need to extend therapeutic predictions to diseases whose understanding is incomplete and, further, to diseases with no approved drugs. Unfortunately, the ability of machine learning models to identify therapeutic candidates for diseases with incomplete, sparse data and zero known therapies drops drastically[18,25] (as we demonstrate across eight benchmarks in Figures 2c and 2d). Here, we introduce TxGNN, a geometric deep learning approach for zero-shot drug repurposing that can predict therapeutic use for diseases with limited or no therapies (Figure 1c). Foundation models like TxGNN are transforming deep learning: instead of training disease-specific models for every disease, TxGNN is a single pretrained model that is adapted to many diseases. TxGNN is trained on a medical knowledge graph that collates decades of biological research across 17,080 diseases, including complex and rare diseases (Figure 1d). TxGNN uses a graph neural network model to embed therapeutic candidates and diseases into a latent representation space and is optimized to reflect the geometry of TxGNN's medical knowledge graph. To make therapeutic predictions under zero-shot settings, TxGNN has a metric learning module to learn similarities between diseases with existing drugs and diseases without any drugs in order to transfer knowledge between these diseases and achieve zero-shot prediction. Once trained, TxGNN performs zero-shot inference on new diseases without additional parameters or fine-tuning.

We demonstrate TxGNN by evaluating its therapeutic predictions across stringent hold-out sets against drugs from key disease areas and against recently approved drugs. We additionally compare TxGNN's novel predictions to off-label prescriptions in a hospital system and conduct a user study with clinicians and scientists to evaluate the potential of novel predictions. First, we go beyond the classical approach of testing machine learning models on random subsets of indications by creating hold-out datasets that prevent the model from taking shortcuts[26] and ensuring that the model can transfer to challenging testing settings when the model encounters diseases with no known therapies. Across six such settings, TxGNN consistently outperforms eight state-of-the-art methods and gains up to 59.3% and 17.8% in accuracy in predicting indications and contraindications compared to the second-best approach. Next, we curate indications that received FDA approval only after TxGNN's medical knowledge graph was built. We observe that TxGNN consistently ranks newly introduced drugs highly. On average, TxGNN ranks the approved drug in the first third (30%) of all predictions and as high as the top 2-4% for specific drugs.

We develop an TxGNN Explainer module that allows introspecting TxGNN's predictions and identifies the relationships most critical for making a prediction. We collect these critical relationships to build explanatory reasoning paths for predicted indications. Clinicians and scientists can interact with explanations using our graphical user interface at http://txgnn.org. To evaluate explanatory reasoning paths and their utility for end users, we conducted a user study. In the user study, 91.6% clinicians and scientists agreed that TxGNN's explanations were valuable in sorting through TxGNN's predictions and helpful for planning the downstream evaluation of predicted indications. Finally, we examine whether TxGNN's novel predictions align with clinical decisions on off-label prescriptions in the medical records of 1,272,085 patients from a large healthcare system. For each of the 480 diseases phenotyped in the medical records, we rank drug candidates in the order predicted by TxGNN. We find that the top-1 predictions have a 107% greater average likelihood of real-world prescription than the bottom-50% predictions. Our analyses suggest that TxGNN's predictions are closely aligned with clinical practices and can offer valuable insights into potential novel uses of existing medicines.

Results

Overview of zero-shot drug repurposing in TxGNN. Zero-shot drug repurposing involves predicting therapeutic candidates for diseases that do not have any existing indications (Figure 1b). Zero-shot drug repurposing is a new problem in deep learning research that has not been considered previously. Mathematically, the model takes a drug-disease pair as input and provides the likelihood of the drug acting on the disease as output. We previously curated and validated a large-scale medical knowledge graph[11] (Figure 1d) consisting of 9,388 indications and 30,675 contraindications that form the gold-standard labels for evaluation[27]. The knowledge graph covers a vast range of 17,080 diseases where 92% have no FDA-approved drugs, including rare diseases and less-understood complex diseases. The knowledge graph also comprises 7,957 potential candidates for drug repurposing, ranging from FDA-approved drugs to experimental drugs investigated in ongoing clinical trials. TxGNN operates on the principle that effective drugs can target disease-perturbed and disease-associated networks of biomolecules, and it has two modules: (1) the TxGNN *Predictor* module enables the accurate prediction of indications and contraindications in the zero-shot setting and (2) the TxGNN *Explainer* module provides interpretable multi-hop explanations that connect the drug to the disease (Figure 1c).

TxGNN Predictor The Predictor module consists of a graph neural network (GNN) optimized on the relationships within the biomedical knowledge graph (Methods 2.2). Through large-scale pretraining, the GNN produces biologically meaningful representations for any entity in this knowledge graph. Then, using self-supervised learning, this GNN is finetuned to predict relationships between therapeutic candidates and diseases. TxGNN leverages a metric learning procedure to make zero-shot predictions. TxGNN capitalizes on the insight that diseases are intrinsically related[12, 16] by leveraging molecular mechanisms of well-annotated diseases to enhance predictions on diseases with limited annotations (Figure 2a, Figure S2). This is achieved by creating a disease signature vector for each disease based on its neighbors in the knowledge graph. The similarity between a pair of diseases is measured by the normalized dot product of their signature vectors. Since most disease pairs do not share underlying pathologies, they have low similarity scores. In contrast, a relatively high similarity score (>0.2) between diseases suggests complementary mechanisms. Description of TxGNN model and its architecture can be found in Methods 2 and Figure S1.

When querying a specific disease, TxGNN retrieves similar diseases, generates embeddings for them, and then adaptively aggregates them based on their similarity to the queried disease. The aggregated output embedding summarizes knowledge borrowed from similar diseases fused with the query disease embedding. TxGNN processes different downstream therapeutic tasks, such as indication and contraindication prediction, in a unified manner using shared drug and disease embeddings (Methods 2.3). This step can also be interpreted as a graph rewiring technique in the graph machine learning literature (Figure S3). Given a query disease, TxGNN ranks drugs based on their predicted likelihood scores, offering a prioritized list of therapeutic candidates with potential for repurposing.

TxGNN Explainer While TxGNN's Predictor provides likelihood scores for therapeutic candidates, these scores alone are insufficient for trustworthy model deployment. Both clinicians and scientists seek to understand the reasoning behind these predictions to validate the model's hypotheses and better understand the disease pathology. To this end, TxGNN Explainer delves into the knowledge graph to pinpoint and succinctly present relevant biological concepts for the drug-disease pair of interest (Figure 4a). This conceptual subgraph mirrors the analytical process clinical researchers use to examine relationships between therapeutic candidates and disease and how the drug perturbs local biological networks to produce a therapeutic effect on disease.

TxGNN employs a self-explaining approach called GraphMask[28] (Methods 2.6). This method generates a sparse yet sufficient subgraph of biological entities considered critical to each therapeutic use prediction. Then, it yields an importance score between 0 and 1 for every edge in this subgraph, with 1 indicating the edge is vital for prediction and 0 suggesting it is irrelevant. TxGNN Explainer combines the drug-disease subgraph and edge importance scores to produce multi-hop paths connecting the disease to predicted therapeutic candidates. Unlike widely recognized explainability techniques such as SHAP[29] that generate feature attribution maps, TxGNN Explainer offers granular and easy-to-understand multi-hop explanations that are, as we show in the user study, aligned with the clinician/scientist's intuition.

We developed a clinician-centered user interface to present these subgraph explanations (Figure 4b) that is openly accessible at http://txgnn.org. The interface visualizes the explanatory rationales from TxGNN to assist clinicians and scientists in reasoning about therapeutic use predictions. Amongst a range of designs, as shown in Figures S4 and S5, we focused on visual path-based reasoning because our research demonstrated that this design choice enhanced clinician comprehension and satisfaction[30].

Comparative assessment of TxGNN in zero-shot drug repurposing. We evaluated model performance in drug repurposing across various hold-out datasets. We generated a hold-out dataset by sampling diseases from the knowledge graph. These diseases were deliberately omitted during the training phase and later served as test cases to gauge the model's ability to generalize its insights to previously unseen diseases. These held-out diseases were either chosen randomly, following a standard evaluation strategy, or specifically selected to evaluate zero-shot prediction. In our study, we used both types of hold-out datasets to thoroughly evaluate methods. We compared TxGNN to eight established methods in predicting therapeutic use. They included network medicine statistical techniques, including KL and JS divergence[18], graph-theoretic network proximity approach[22], and diffusion state distance (DSD)[31], state-of-the-art graph neural network methods, including relational graph convolutional networks (RGCN)[21,32], heterogeneous graph transformer (HGT)[33], and heterogeneous attention networks (HAN)[34], and a natural language processing model, BioBERT[35]. More information regarding each baseline is in Methods 3.6.

Initially, we followed the standard evaluation strategy where drug-disease pairs were randomly shuffled, and a subset of these pairs was set aside as a hold-out set (testing set; Figure 2c). Under this strategy, the diseases being evaluated as hold-outs may already have had indications and contraindication relationships with drugs in the training set. Therefore, the learning objective was to identify additional therapeutic candidates for well-studied diseases. This evaluation method aligns with the approach predominantly used in literature[21]. Our experimental results in this setting concur, with 3 of 8 existing methods achieving AUPRC greater than 0.8, and HAN as the best at 0.873 AUPRC. TxGNN also had a comparable performance as established methods. In predicting indications, TxGNN achieved a 4.3% increase in AUPRC (0.913) over the strongest baseline, HAN.

As shown by above experiments, machine learning methods can help identify repurposing opportunities for diseases that already have some FDA-approved drugs[14–18,22,23]. However, Duran et al.[36] reason that many methods simply retrieve additional therapeutic candidates that are similar to existing ones across biological levels. This suggests the standard evaluation strategy is unsuitable for evaluating diseases that have no FDA-approved drugs (Figure 1b). Given this limitation, we evaluate models under zero-shot drug repurposing. We began by holding out a random set of diseases and then moved all their associated drugs to the hold-out set (Figure 2d). From a biological standpoint, the model was required to predict therapeutic candidates for diseases that lacked treatments, meaning it had to operate without any available data on drug similarities. In this scenario, TxGNN outperformed all existing methods by a large margin. TxGNN significantly improves over the next best baseline in predicting both indications (19.0% AUPRC gain) and contraindications (23.9% AUPRC gain). While established methods achieved satisfactory results in conventional drug repurposing evaluations, they often fell short on more challenging zero-shot drug repurposing scenarios. TXGNN was the only method that achieved consistent performance in both settings.

Benchmarking TXGNN for zero-shot drug repurposing across disease areas. Diseases with biological similarities often share therapeutic candidates[12]. For instance, beta-blockers are effective in treating a multitude of cardiovascular issues, including heart failure, cardiac arrest, and hypertension. Likewise, selective serotonin reuptake inhibitors (SSRIs) can address various psychiatric conditions such as major depressive disorder (MDD), anxiety disorder, and obsessive-compulsive disorder (OCD). If, during training, a model learns that an SSRI is indicated for MDD, it does not take a large leap to suggest that the same SSRI could be effective for OCD during testing[25]. This phenomenon is known as shortcut learning[37,38] and underlies many of deep learning's failures[39,40]. Shortcut decision rules tend to perform well on standard benchmarks but typically fail to transfer to challenging testing conditions[41], such as the real-world scenario of predicting therapeutic candidates for rare or neglected diseases.

To evaluate drug repurposing models in challenging conditions, we curated a stringent hold-out dataset that contained a group of biologically related diseases that we refer to as a disease area. Given the diseases in a specific disease area, all their indications and contraindications were removed from the training dataset. Further, a large fraction (95%) of the connections from medical entities to these diseases were excluded from the training dataset. For diseases in the chosen area, these conditions simulated limited molecular characterization and lack of existing treatments (Figure 3a). In this study, we considered five disease area hold-out datasets characterized in Table 1 and listed here in order of increasing disease area size. First are 'adrenal gland' diseases like Addison and ectopic crushing syndrome. Second, 'anemia' with conditions such as thalassemia and hemoglobin C disease. Third, 'mental health' disorders like anorexia nervosa and depressive disorder. Fourth, 'cardivascular' diseases, including long QT syndrome and mitral valve stenosis. Finally, 'cancer' diseases such as neurofibroma and Leydig cell tumors.

We benchmarked the performance of all methods above on these rigorous hold-out datasets in Figure 3b-f and found that TXGNN consistently improved predictive performance over existing methods. For indications, TXGNN had 59.3%, 42.3%, 36.2%, 10.2%, 0.5% relative gain in AUPRC over the next best baseline across adrenal glands, anemia, mental health, cancer, and cardiovascular disease hold-outs respectively. For contraindications, TXGNN robustly improved over the next best baseline, with relative gains ranging from 11.8% to 17.8%. For indication prediction, the natural language processing method, BioBERT, had the best performance (in 4/5 disease area hold-outs) amongst the group of established methods. For contraindication prediction, the graph-based method, RGCN, was the best baseline across 4 of 5 hold-out datasets, and BioBERT's performance gain observed for indication prediction disappeared. TxGNN was consistently the best-performing method across all five disease area hold-outs for both indication and contraindication prediction tasks. These rigorous benchmarks demonstrate that TxGNN was broadly generalizable and produced accurate predictions in zero-shot drug repurposing settings.

In 4 of 5 disease area hold-outs, TxGNN's relative gains in performance over existing methods grew as the number of diseases in the hold-out shrunk, suggesting that TxGNN could be particularly effective for small clusters of diseases. While TxGNN demonstrated significantly higher performance in 4 of 5 disease area hold-outs, its performance was on par with existing methods in the cardiovascular hold-out. This could be attributed to a lack of knowledge of related diseases in the training dataset when entire disease areas are held out. Upon visualizing the latent representations of the TxGNN Predictor, we discovered that it facilitates knowledge transfer from distant diseases to those with scarce information (Figure S6). Additional evaluation metrics are described in Figure S7. In ablation studies, we demonstrated that each component of TxGNN Predictor is indispensable to the model's predictive performance (Figures S8 and S9).

TxGNN's multi-hop explanations reflect its predictive rationale. TxGNN's Explainer extracts concise multi-hop relationships between drugs and diseases from the medical knowledge graph to provide supporting evidence for TxGNN's predictions. TxGNN's Explainer identifies explanations as maximally predictive subgraphs of the medical knowledge graph that connect the query drug with the query disease through multiple hops following relationships in the knowledge graph, such that the predictive performance of these subgraphs is comparable to the performance of the entire knowledge graph. To gauge the quality of explanations, we compared the AUPRC of TxGNN's predictions when leveraging the whole knowledge graph to the AUPRC derived from using only the explanatory subgraphs. A strong correlation would indicate that TxGNN's Explainer effectively identifies the most pivotal connections[42] and that explanations faithfully capture TxGNN's internal reasoning for making predictions[43].

Based on the most predictive relationships (i.e., edges with importance scores greater than 0.5, representing an average of 14.9% of edges from the knowledge graph), the performance of TxGNN's model exhibited a minor decline from AUPRC=0.890 (STD: 0.006) to AUPRC=0.886 (STD: 0.005). On the other hand, when considering the remaining irrelevant relationships from the knowledge graph (i.e., edges with importance scores less than 0.5, accounting for an average of 85.1% of edges) and excluding those deemed predictive by TxGNN, the model's predictive performance drastically decreased from AUPRC=0.890 (STD: 0.006) to AUPRC=0.628 (STD: 0.026). Together, these analyses indicated that TxGNN's explanatory subgraphs faithfully capture elements of the knowledge graph that TxGNN uses to make predictions. The TxGNN Explainer effectively discerned the pivotal relationships between a drug and disease, ensuring explanations aligned with its predictive rationale.

TxGNN Explainer supports human-centric evaluation of therapeutic candidates. We showcased the importance of TxGNN's multi-hop pathway explanations in facilitating human-AI collaboration by conducting a user study with clinicians and scientists. In our experiments (Figure 4c), we engaged five clinicians, five clinical researchers, and two pharmacists. These participants were shown 16 drug-disease combinations with TxGNN's predictions, where 12 predictions were accurate. For each pairing, participants indicated whether they agreed or disagreed with TxGNN's predictions using the explanations provided (Figure S10). Our results suggest that giving visual explanations improved users' performance in evaluating model predictions, such as determining if a proposed therapeutic candidate can treat a disease.

When comparing TxGNN Explainer's performance to a version without explanations, we evaluated user accuracy, exploration time, and user confidence (Figure 4d). The results revealed a significant improvement in both accuracy (+46%) and confidence (+49%) when users were given explanations. Users took more time to think, integrating their prior knowledge with TxGNN's explanations, and therefore trusted the model predictions more (confidence +49%). With TxGNN Explainer, participants discerned between accurate and inaccurate predictions more effectively with TxGNN Explainer than using TxGNN predictions alone (accuracy +46%).

Study participants reported greater satisfaction when using TxGNN Explainer compared to the baseline (Figure 4e), where 11/12 (91.6%) participants agreed or strongly agreed that the predictions and explanations made by TxGNN were valuable. In contrast, when no explanations were provided, 8/12 (75.0%) participants disagreed or strongly disagreed with relying on TxGNN's predictions. Additionally, participants expressed significantly more confidence in correct predictions made by TxGNN ($t(11) = 3.64, p < 0.01$) with TxGNN Explainer than in the baseline without explanations. Some participants stated that path-based explanations were useful for guiding downstream evaluations, such as examining biological mechanisms and possible adverse drug events of predicted therapeutic candidates. Our user study offers empirical evidence of optimized human-AI collaboration in predicting therapeutic applications with TxGNN Explainer.

Evaluation of TxGNN's novel predictions in a large electronic health record system. TxGNN's remarkable performance in previous evaluations suggests that its novel predictions—*i.e.*, therapies not yet FDA-approved for a disease but ranked highly by TxGNN —might hold significant clinical value. As these therapies have not yet been approved for treatment, there is no established gold standard against which to validate them. Recognizing the clinical practice of off-label drug prescription, we used the enrichment of disease-drug pair co-occurrence in a health system's electronic health records as a proxy measure of being a potential indication. From the Mount Sinai Health System medical records, we curated a cohort of 1,272,085 adults with at least one drug prescription and one diagnosis each (Figure 5a). This cohort was 40.1 percent male, and the average age was 48.6 years (STD: 18.6 years). The racial and sex breakdown is in Figure 5b-c. Diseases were included if at least one patient was diagnosed with it, and drugs were included if prescribed to a minimum of ten patients (Table 2 and Methods 3.8), resulting in a broad spectrum of 480 diseases and 1,290 drugs as illustrated in Figure 5d.

Across these medical records, we measured disease-drug co-occurrence enrichment as the ratio of the odds of using a specific drug for a disease to the odds of using it for other diseases. We derived 619,200 log-odds ratios (log-ORs) for each drug-disease pair. We found that FDA-approved drug-disease pairs exhibited significantly higher log-ORs than other pairs (Figure 5e). Contraindications represented a confounding factor in this analysis because adverse drug events could increase the co-occurrence between drug-disease pairs. In our study of contraindications, we found no significant enrichment in co-occurrence of drug-disease pairs, which suggested that adverse drug effects were not a major confounding factor.

For each disease in the medical records, TxGNN produced a ranked list of potential therapeutic candidates. We omitted drugs already linked to the disease, categorized the remaining novel candidates into top-1, top-5, top-5%, and bottom-50%, and calculated their respective mean log-ORs (Figure 5f). We found that the top-1 novel TxGNN prediction had, on average, a 107% higher log-OR than the mean log-OR of the bottom-50% predictions. This suggested that TxGNN's top candidate had much higher enrichment in the medical records and, thereby, a greater likelihood of being an appropriate indication. In addition, the log-OR increased as we broadened the fraction of retrieved candidates, suggesting that TxGNN's prediction scores were meaningful in capturing the likelihood of indication. Although the average log-OR stands at 1.09, the top-1 therapeutic candidate proposed by TxGNN had a log-OR of 2.26, approaching the average log-OR of 2.92 for

FDA-approved indications. This analysis highlights the potential clinical utility of TxGNN's top predictions.

We present a case study of TxGNN's predicted therapeutic candidates for Wilson's disease, a genetic disorder causing excessive copper accumulation that frequently instigates liver cirrhosis in children (Figure 3g). We found that TxGNN predicted likelihoods close to zero for most therapeutic candidates, with only a select few drugs highly likely to be indications. TxGNN ranked Deferasirox as the most promising therapy for Wilson's disease. Wilson's disease and Deferasirox had a log-OR of 5.26 in the medical records, and literature indicates that Deferasirox may effectively eliminate hepatic iron[44]. In a separate analysis, we evaluated TxGNN on ten therapies that were recently approved by the FDA (Table S1). TxGNN consistently ranked newly introduced drugs favorably and placed the approved drug within the top 5% of therapeutic candidates in two instances. These case studies highlight that TxGNN's novel predictions have the potential to align closely with clinical decision-making on drug prescription.

Discussion

Tapping into the dormant potential of existing drugs, drug repurposing has already had demonstrable success in addressing the global healthcare demands. Yet, existing deep learning models for drug repurposing are based on the assumption that diseases for repurposing predictions are well-understood and already possess existing therapies. This overlooks the vast array of disorders—92% of the 17,080 diseases we analyzed—lacking such pre-existing indications and in-depth molecular characterization. The clinical imperative thus leans heavily on addressing the needs of these lesser-known disorders, many of which fall into the category of complex, neglected, or rare diseases[45–47]. This has culminated in what we refer to as zero-shot drug repurposing.

We introduce TxGNN, a geometric deep learning model that addresses this problem head-on, specifically targeting diseases with limited molecular understanding and no treatment avenues. TxGNN achieves state-of-the-art performance in drug repurposing by leveraging a network medicine principle that focuses on disease-treatment mechanisms[17]. When asked to suggest therapeutic candidates for a disease, TxGNN identifies diseases with shared pathways, phenotypes, and pathologies, extracts relevant knowledge, and fuses it back into the disease of interest. By effectively capturing these latent relationships between diseases, TxGNN can generalize to diseases with few treatment options and perform zero-shot inference for unseen diseases. The design behind TxGNN that enables effective zero-shot drug repurposing can be adapted to a wide range of problems, such as disease-target identification and phenotype modeling.

TxGNN Predictor can propose indications and contraindications in a unified formulation across 17,080 diseases and significantly improves predictive performance compared to existing methods under the real-world constraints of zero-shot inference. Further, the therapies predicted by TxGNN correlate strongly with data from electronic health records. TxGNN's therapeutic hypotheses can be further tested at scale and in parallel using medical records. This can be done by comparing patient cohorts with the disease who were prescribed the predicted drugs to matched patient cohorts who were not prescribed those drugs.

TxGNN Explainer generates multi-hop pathways that reflect its internal reasoning for each proposed therapeutic candidate. This enables clinicians and scientists to sift through its predictions and investigate the underlying disease-treatment mechanisms. In our user study, the interactive TxGNN Explainer allowed participants to more easily engage with the model predictions and debug failure points. This highlights the importance of clinician-centered design and explainability in integrating machine learning into drug prescription and development decisions[48].

While TxGNN offers remarkable promise in addressing the zero-shot drug repurposing problem, it is not without limitations. Its efficacy is contingent on the quality of the knowledge graph, which could not only have incomplete knowledge of certain disease areas but also be faced with literature bias from varied data sources. A promising avenue for future research involves applying uncertainty quantification over graphs to determine the reliability of model predictions. Additionally, our user study engaged a small sample size (N=12) of clinicians and scientists. While the results were statistically significant, a larger study would incorporate a greater diversity of user perspectives. Despite our feasibility analysis of TxGNN's predictions using medical records, unforeseeable confounding factors might bias the enrichment scores measured. To mitigate those factors, we use a multitude of comprehensive evaluation settings, including rigorous hold-out datasets, a user study, and a systematic medical record analysis. While our goal has been to introduce a computational method for predicting therapeutic candidates across diverse zero-shot scenarios, detailed analysis of specific repurposing candidates warrants future investigation.

Our zero-shot drug repurposing model, TxGNN, predicts therapeutic candidates even for diseases with no FDA-approved drugs and with minimal molecular knowledge. The Explainer module enhances transparency of TxGNN's predictions, fostering trust and aiding downstream evaluations. TxGNN streamlines the drug repurposing process, especially where the need for disease-specific datasets hinders drug development. This is pivotal for conditions with scarce data, like rare diseases and emerging pathogens. In the quest for cost-effective therapeutic innovations, models like TxGNN highlight the computational potential in revealing novel therapeutic avenues.

Data availability. TxGNN's website is at https://zitniklab.hms.harvard.edu/projects/TxGNN. The knowledge graph dataset is available at Harvard Dataverse under a persistent identifier https://doi.org/10.7910/DVN/IXA7BM. All clinical and electronic medical record data were deidentified, and the Institutional Review Board at Mount Sinai, New York City, U.S., approved the study.

Code availability. Python implementation of the methodology developed and used in the study is available via the project website at https://zitniklab.hms.harvard.edu/projects/TxGNN. The code to reproduce results, documentation, and usage examples are at https://github.com/mims-harvard/TxGNN. To facilitate the usage of the algorithm, we developed a TxGNN Explainer, a web-based app available at http://txgnn.org to access TxGNN's predictions.

Acknowledgements. K.H., P.C., and M.Z. gratefully acknowledge the support of NIH R01-HD108794, US DoD FA8702-15-D-0001, awards from Harvard Data Science Initiative, Amazon Faculty Research, Google Research Scholar Program, Bayer Early Excellence in Science, AstraZeneca Research, Roche Alliance with Distinguished Scientists, and Kempner Institute for the Study of Natural and Artificial Intelligence at Harvard University. P.C. was supported, in part, by the Harvard Summer Institute in Biomedical Informatics. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funders.

Authors contribution. P.C. retrieved, processed, and analyzed the knowledge graph. K.H. and P.C. developed and implemented new machine learning methods, benchmarked machine learning models, and analyzed model behavior, all together with M.Z. Q.W. and N.G. implemented the clinician-centered visual explorer of model predictions and performed a user study to evaluate its usability. S.H., A.V., G.N. and B.S.G. performed a validation study examining new predictions of therapeutic use through the electronic health record system. K.H., P.C, Q.W., S.H., A.V., J.L., G.N, B.S.G., N.G., and M.Z. contributed new analytic tools and wrote the manuscript. All authors discussed the results and contributed to the final manuscript. M.Z. designed the study.

Competing interests. The authors declare no competing interests.

Inclusion and ethics statement in global research. We have complied with all relevant ethical regulations. Our research team represents a diverse group of collaborators. Roles and responsibilities were clearly defined and agreed upon among collaborators before the start of the research. All researchers were included in the study design, study implementation, data ownership, intellectual property, and authorship of publications. Our research did not face severe restrictions or prohibitions in the setting of the local researchers, and no specific exceptions were granted for this study in agreement with local stakeholders. Animal welfare regulations, environmental protection and risk-related regulations, transfer of biological materials, cultural artifacts, or associated traditional knowledge out of the country do not apply to our research. Our research does not result in stigmatization, incrimination, discrimination, or personal risk to participants. Appropriate provisions were taken to ensure the safety and well-being of all participants involved. Our team was committed to promoting equitable access to resources and benefits resulting from the research.

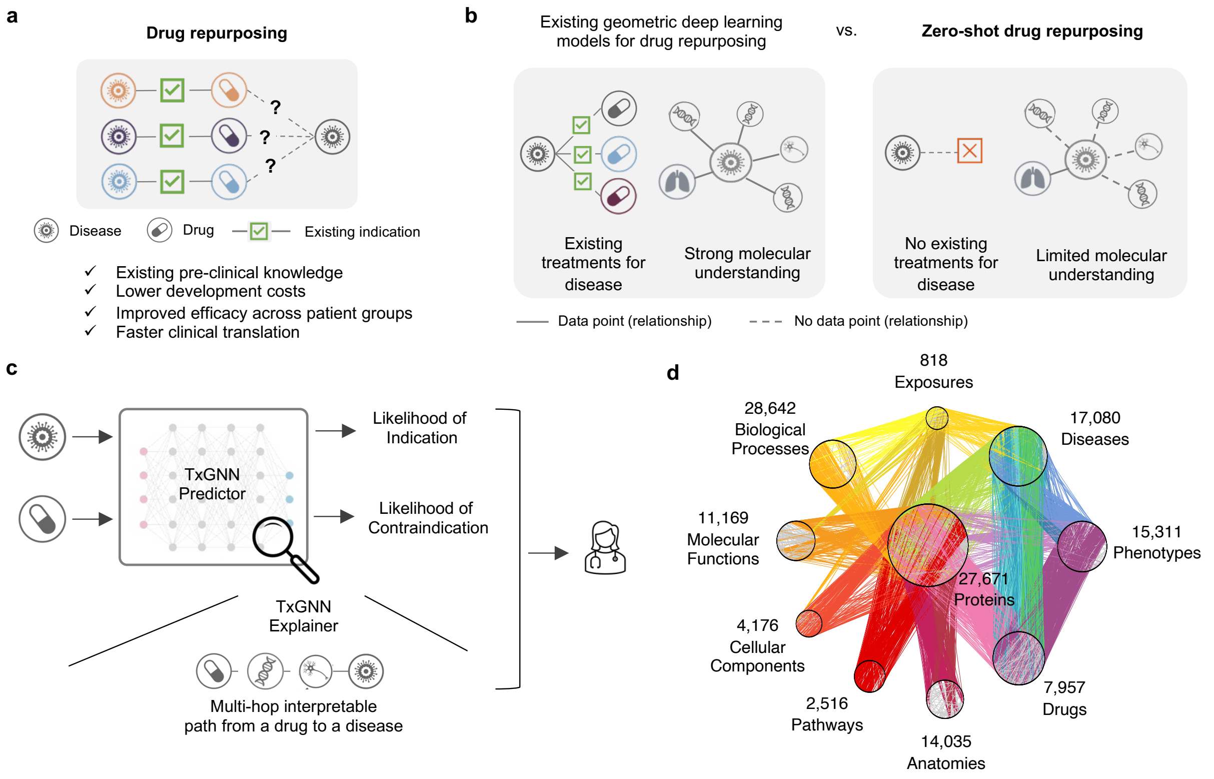
a
Drug repurposing
?
?
?
Disease
Drug
Existing indication
✓ Existing pre-clinical knowledge
✓ Lower development costs
✓ Improved efficacy across patient groups
✓ Faster clinical translation
b
Existing geometric deep learning models for drug repurposing
vs.
Zero-shot drug repurposing
Existing treatments for disease
Strong molecular understanding
No existing treatments for disease
Limited molecular understanding
Data point (relationship)
No data point (relationship)
c
TxGNN Predictor
Likelihood of Indication
Likelihood of Contraindication
TxGNN Explainer
Multi-hop interpretable path from a drug to a disease
d
818 Exposures
28,642 Biological Processes
17,080 Diseases
11,169 Molecular Functions
15,311 Phenotypes
27,671 Proteins
4,176 Cellular Components
7,957 Drugs
2,516 Pathways
14,035 Anatomies

Figure 1: TxGNN is a geometric deep learning approach for drug repurposing across challenging diseases with no known treatments and limited molecular understanding. a. Drug repurposing involves exploring new therapeutic applications for existing drugs to treat different diseases. By capitalizing on abundant pre-existing safety and efficacy data, it can dramatically cut down the cost and time to deliver life-saving therapeutics. **b.** Although AI-based drug repurposing has shown promise, its success has been primarily evaluated on diseases with approved treatments and well-understood molecular mechanisms. However, many diseases of critical pharmaceutical interest lack any available treatments (i.e., zero-shot) and exhibit unclear disease mechanisms. These inherent constraints pose challenges to existing AI methods. In this work, we tackle this problem head-on by formulating it as a zero-shot drug repurposing challenge. **c.** TxGNN presents a novel AI framework that generates actionable predictions for zero-shot drug repurposing. TxGNN geometric deep learning model incorporates a vast and comprehensive biological knowledge graph to accurately predict the likelihood of indication or contraindication for any given disease-drug pair. Additionally, TxGNN generates explainable multi-hop paths, facilitating a scientist-friendly understanding of how the prediction is grounded in biological mechanisms in the KG. The combined power of rich predictions and path-based explanations empowers practitioners to prioritize the most promising drug repurposing candidates. **d.** To support our drug repurposing efforts, we develop a large-scale therapeutics-driven knowledge graph that integrates 17 primary data sources. This knowledge graph paints a comprehensive landscape of biological mechanisms across 17,080 diseases and 7,957 repurposable drugs, compiling scientific knowledge for zero-shot drug repurposing endeavors.

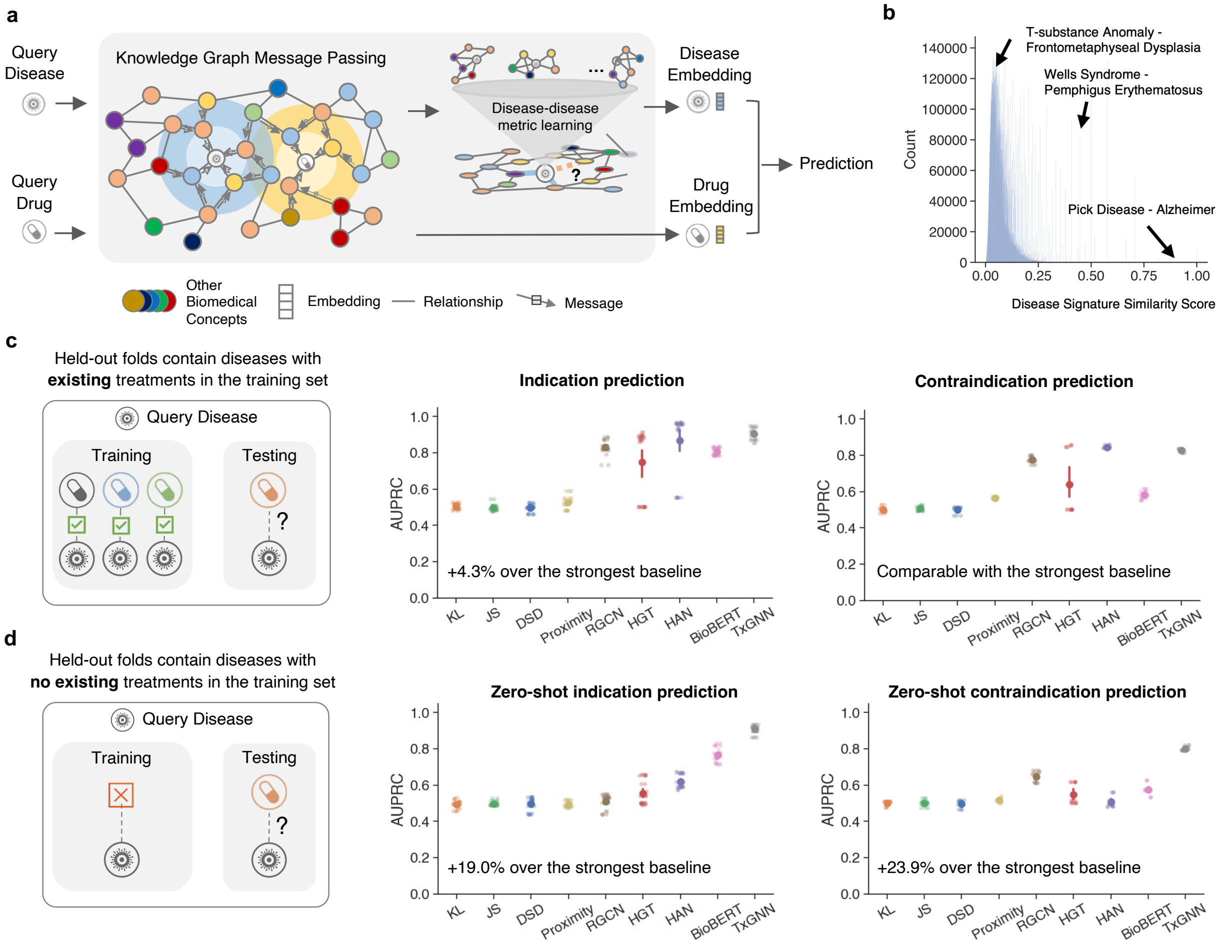

Figure 2: TxGNN predicts indications and contraindications for diseases of no known treatments with high precision. a. TxGNN is a deep learning model that learns to reason over large-scale knowledge graph on predicting the relationship between drug and disease. In zero-shot repurposing, there is limited indication and mechanism information available for the query disease. Our key insight revolves around the interconnectedness of biological systems. We recognize that diseases, despite their distinctiveness, can exhibit partial similarities and share multiple underlying mechanisms. Based on this motivation, we have developed a specialized module known as disease pooling, which harnesses the power of network medicine principles. This module identifies mechanistically similar diseases and employs them to enhance the information available for the query disease. The disease pooling module has demonstrated significant improvements in the prioritization of repurposing candidates within zero-shot settings. **b.** The TxGNN disease similarity score provides a nuanced and meaningful measure of the relationship between diseases. For instance, disease pairs with low similarity scores, such as T-substance anomaly and frontometaphyseal dysplasia (score: 0.084), indicate a lack of shared mechanisms. Conversely, significant similarity is observed when two diseases receive relatively high scores (>0.2). For instance, Wells syndrome and pemphigus erythematosus exhibit a similarity score of 0.433. Both diseases are skin disorders caused by autoimmune dysregulation, although they differ in phenotypic manifestations, with Wells syndrome characterized by redness and swelling and pemphigus erythematosus characterized by blisters. Moreover, certain disease pairs display exceptionally high similarity scores, such as Pick's disease and Alzheimer's (similarity: 0.909), due to their shared neurological causes. This metric empowers TxGNN to discover similar diseases that can inform and enrich the understanding of query diseases lacking treatment and mechanistic information. **c.** The conventional AI-based repurposing evaluates indication predictions on diseases where the model may have seen other approved drugs during training. In this scenario, we show that TxGNN achieves good performance along with existing methods. **d.** To provide a more realistic evaluation, we introduce a novel setup for assessing zero-shot repurposing, where the model is evaluated on diseases that have no approved drugs available during training. In this challenging setting, we observe a significant degradation in performance for baseline methods. In contrast, TxGNN consistently exhibits robust performance, surpassing the best baseline by up to 19% for indications and 23.9% for contraindications. These results highlight the advanced reasoning capabilities of TxGNN when confronted with query diseases lacking treatment options. The evaluation utilizes the area under the precision-recall curve (AUPRC) and is conducted with five random data splits. The mean performance is highlighted, while the standard error is represented by error bars.

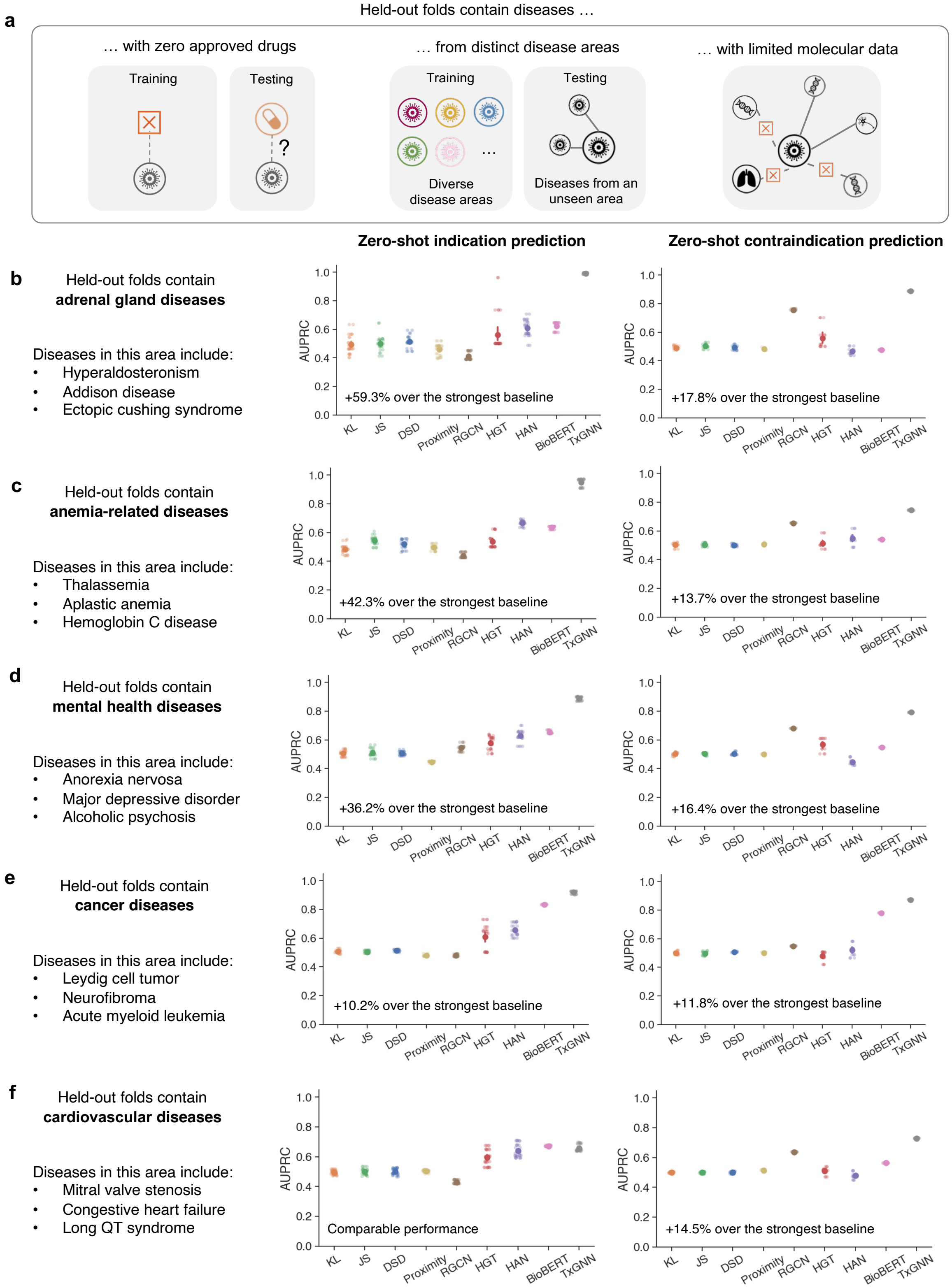

Figure 3: TxGNN accurately predicts therapeutics indications and contraindications across challenging disease areas with limited mechanism understanding. a. Zero-shot drug repurposing addresses diseases without any existing treatments and with a dearth of prior biomedical knowledge. We construct a set of 'disease area' splits to simulate these conditions. The diseases in the holdout set have (1) no approved drugs in training, (2) limited overlap with training disease set because we exclude similar diseases, and (3) lack molecular data because we deliberately remove their biological neighbors from the training set. These data splits constitute a challenging but realistic evaluation scenarios that mimic zero-shot drug repurposing settings. **b-f.** Holdout folds evaluate diseases related to adrenal glands, anemia, mental health, cancer, and cardiovascular diseases. TxGNN shows up to 59.3% improvement over the next best baseline in ranking therapeutic candidates, measured by area under the precision-recall curve.

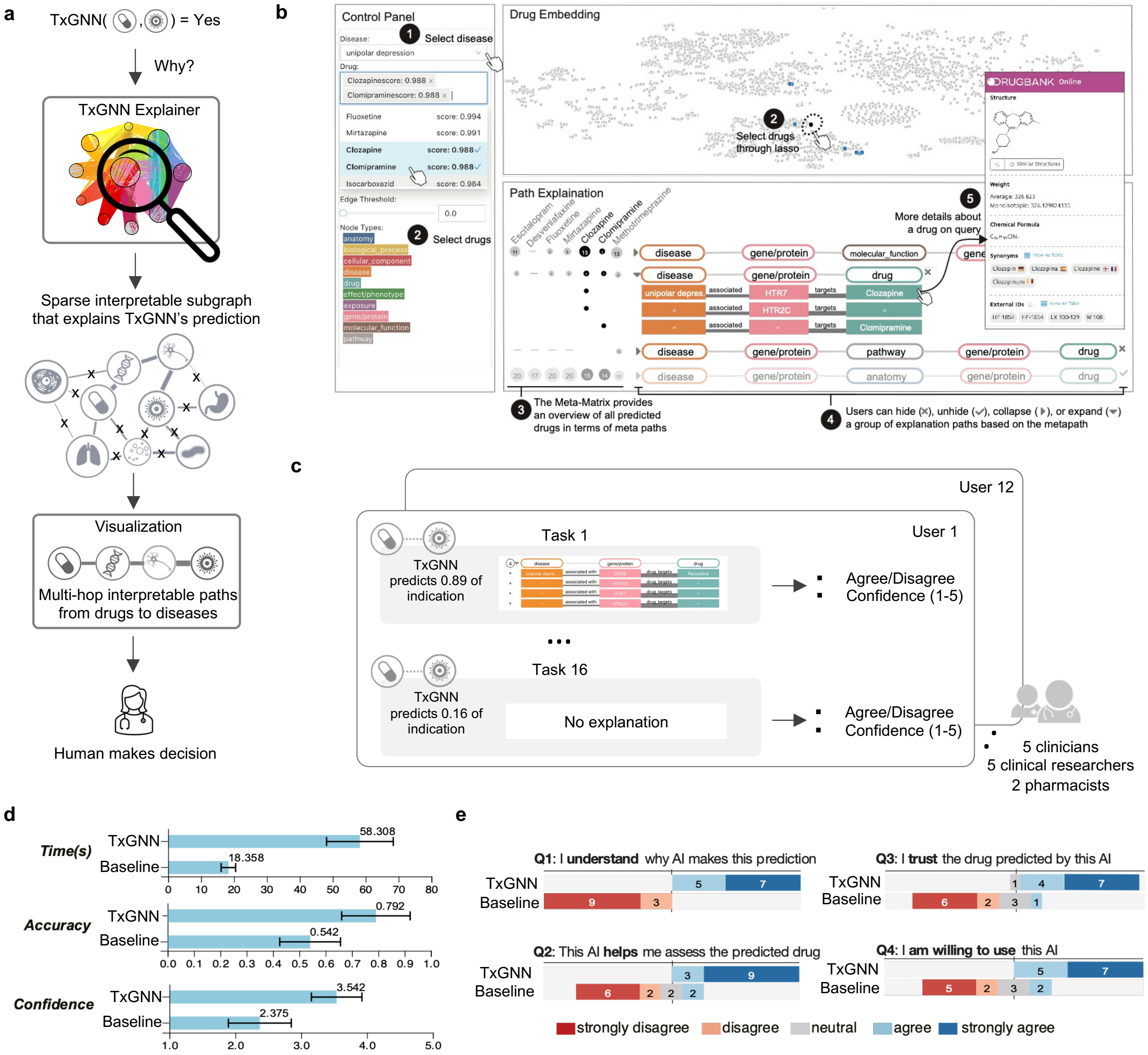
a
TxGNN(,) = Yes
Why?
TxGNN Explainer
Sparse interpretable subgraph that explains TxGNN's prediction
Visualization
Multi-hop interpretable paths from drugs to diseases
Human makes decision
b
Control Panel
1 Select disease
Disease: unipolar depression
Drug: Clozapinescore: 0.988 Clomipraminescore: 0.988
Fluoxetine score: 0.994
Mirtazapine score: 0.991
Clozapine score: 0.988
Clomipramine score: 0.988
Isocarboxazid score: 0.984
Edge Threshold: 0.0
Node Types: anatomy biological_process cellular_component disease drug effect/phenotype exposure gene/protein molecular_function pathway
2 Select drugs
Drug Embedding
2 Select drugs through lasso
DRUGBANK Online
5 More details about a drug on query
Path Explaination
Escitalopram Desvenlafaxine Fluoxetine Mirtazapine Clozapine Clomipramine Methotrimeprazine
disease gene/protein molecular_function gene
disease gene/protein drug
unipolar depres associated HTR7 targets Clozapine
associated HTR2C targets
associated targets Clomipramine
disease gene/protein pathway gene/protein drug
disease gene/protein anatomy gene/protein drug
3 The Meta-Matrix provides an overview of all predicted drugs in terms of meta paths
4 Users can hide (×), unhide (✓), collapse (▸), or expand (▾) a group of explanation paths based on the metapath
c
User 12
User 1
Task 1
TxGNN predicts 0.89 of indication
Agree/Disagree
Confidence (1-5)
Task 16
TxGNN predicts 0.16 of indication
No explanation
Agree/Disagree
Confidence (1-5)
5 clinicians
5 clinical researchers
2 pharmacists
d
Time(s) TxGNN 58.308 Baseline 18.358
0 10 20 30 40 50 60 70 80
Accuracy TxGNN 0.792 Baseline 0.542
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
Confidence TxGNN 3.542 Baseline 2.375
1.0 2.0 3.0 4.0 5.0
e
Q1: I understand why AI makes this prediction
TxGNN 5 7
Baseline 9 3
Q3: I trust the drug predicted by this AI
TxGNN 1 4 7
Baseline 6 2 3 1
Q2: This AI helps me assess the predicted drug
TxGNN 3 9
Baseline 6 2 2 2
Q4: I am willing to use this AI
TxGNN 5 7
Baseline 5 2 3 2
strongly disagree disagree neutral agree strongly agree

Figure 4: Development, visualization, and evaluation of explanations provided by TxGNN. a. Since prediction scores alone are often insufficient for trustworthy deployment of machine learning models, we develop TxGNN Explainer to facilitate adoption by clinicians and scientists. TxGNN Explainer uses state-of-the-art graph explainability techniques to identify a sparse interpretable subgraph that underlies the model's predictions. For each therapeutic candidate, TxGNN Explainer generates a multi-hop pathway composed of various biomedical entities that connects the query disease to the proposed therapeutic candidate. We develop a visualization module that transforms the identified subgraph into these multi-hop paths in a manner that aligns with the cognitive processes of clinicians and scientists. **b.** We design a web-based graphical user interface to support clinicians and scientists in exploring and analyzing the predictions and explanations generated by TxGNN. The 'Control Panel' allows users to select the disease of interest and view the top-ranked TxGNN predictions for the query disease. The 'edge threshold' module enables users to modify the sparsity of the explanation and thereby control the density of the multi-hop paths displayed. The 'Drug Embedding' panel allows users to compare the position of a selected drug relative to the entire repurposing candidate library. The 'Path Explanation' panel displays the biological relations that have been identified as crucial for TxGNN's predictions regarding therapeutic use. **c.** To evaluate the usefulness of TxGNN explanations, we conducted a user study involving 5 clinicians, 5 clinical researchers, and 2 pharmacists. These participants were shown 16 drug-disease combinations with TxGNN's predictions, where 12 predictions were accurate. For each pairing, participants indicated whether they agreed or disagreed with TxGNN's predictions using the explanations provided. **d.** We compared the performance of TxGNN Explainer with a no-explanation baseline in terms of user answer accuracy, exploration time, and user confidence. The results revealed a significant improvement in accuracy (+46%) and confidence (+49%) when explanations were provided, indicating that TxGNN Explainer contributed to the generation of trustworthy predictions. Error bars represent 95% confidence intervals. **e.** At the conclusion of the user study, participants were asked qualitative usability questions. Clinicians and scientists agreed that the explanations provided by TxGNN were helpful in assessing the predicted drug-disease relationships and instilled greater trust in the TxGNN's predictions.

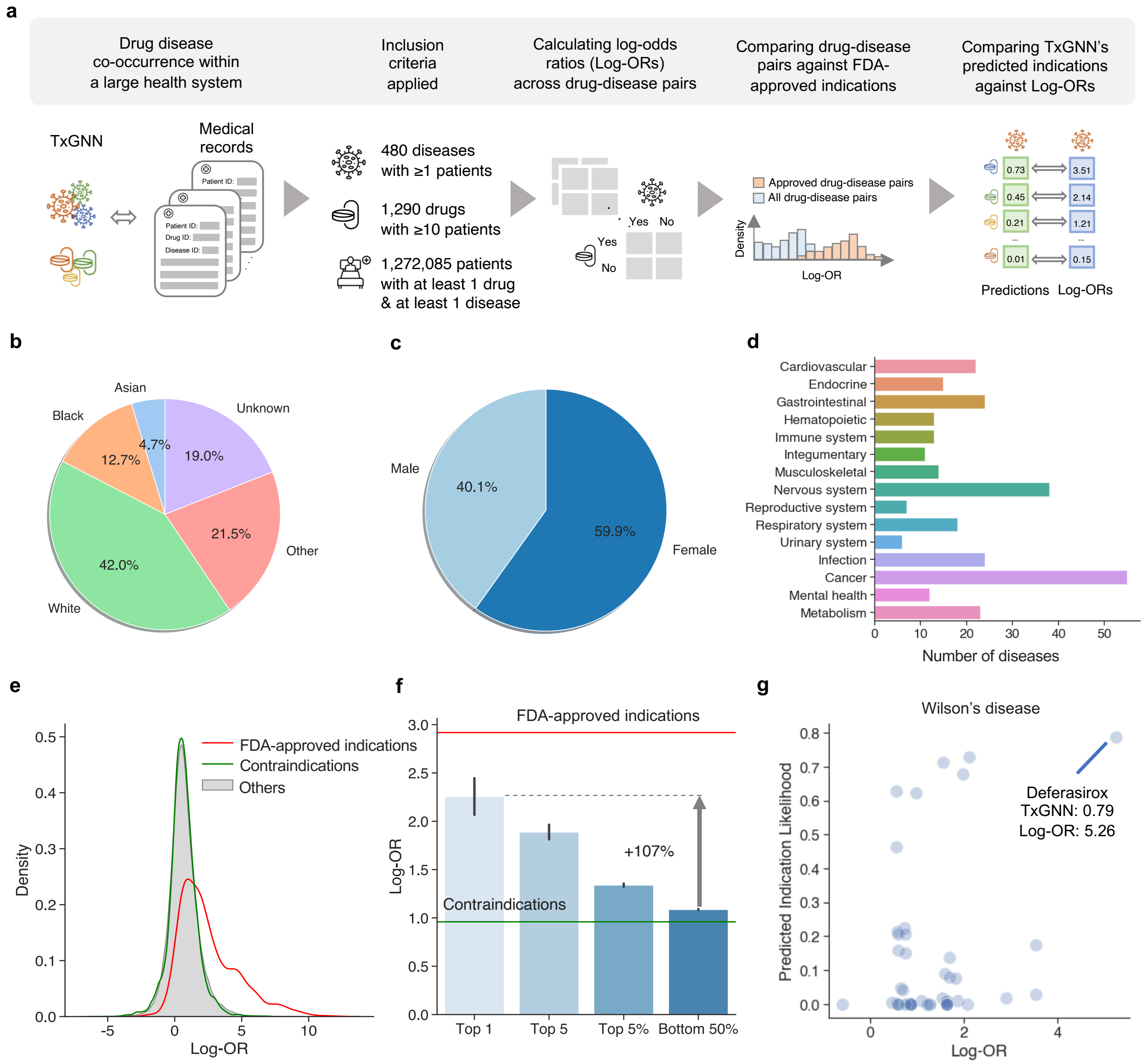

Figure 5: Evaluating TxGNN's novel predictions in a large healthcare system. a. We illustrate the steps taken to evaluate TxGNN's novel indications predictions in a Mount Sinai's electronic health record (EHR) system. First, we matched the drugs and diseases in the TxGNN knowledge graph to the EHR database, resulting in a curated cohort of 1.27 million patients spanning 480 diseases and 1,290 drugs. Next, we calculated the log-odds ratio (log-OR) for each drug-disease pair, which served as an indicator of the usage of a particular drug for a specific disease. We then validated the log-OR metric as a proxy for clinical usage by comparing drug-disease pairs against FDA-approved indications. Finally, we evaluated TxGNN's novel predictions to determine if their Log-ORs exhibited enrichment within the medical records. **b.** The racial diversity within the patient cohort. **c.** The sex distribution of the patient cohort. **d.** The medical records encompassed a diverse range of diseases spanning major disease areas, ensuring comprehensive coverage and representation. **e.** In validating log-ORs as a proxy metric for clinical prescription, we observed that while the majority of drug-disease pairs exhibited low log-OR values, there was a significant enrichment of log-OR values for FDA-approved indications. Additionally, we noted that contraindications displayed similar log-OR values to the general non-indicated drug-disease pairs, minimizing potential confounders such as adverse drug effects. **f.** We evaluated Log-ORs for the novel indications proposed by TxGNN. The y-axis represents the Log-OR of the disease-drug pairs, serving as a proxy for clinical usage. For each disease, we ranked TxGNN's predictions and extracted the average Log-OR values for the top 1, top 5, top 5%, and bottom 50% of novel drug candidates. The red horizontal line represents the average Log-OR for FDA-approved indications, while the green horizontal line represents the average Log-OR for contraindications. We observed a remarkable enrichment in the clinical usage of TxGNN's novel predictions. **g.** We provide an case study of TxGNN's predicted scores plotted against the Log-OR for Wilson's disease. Each point on the plot represents a therapeutic candidate. The top 1 most probable candidate suggested by TxGNN is highlighted, indicating its associated TxGNN score and Log-OR.

Online Methods

The Methods are structured as follows: 1) curation of knowledge graph dataset (Section 1), 2) description of machine learning approach (Section 2), and 3) outline of the experimental setup, benchmarking and evaluation (Section 3).

1 Overview of training dataset

The knowledge graph is heterogeneous, with 10 types of nodes and 29 types of undirected edges. It contains 123,527 nodes and 8,063,026 edges. Tables S2 and S3 show a breakdown of nodes by node type and edges by edge type, respectively. The knowledge graph and all auxiliary data files are available via Harvard Dataverse at https://doi.org/10.7910/DVN/IXA7BM.

1.1 Primary data resources

The knowledge graph (KG) is compiled from many primary knowledge bases that cover 10 types of biomedical entities and provide broad coverage of human disease, already-available drugs, and novel drugs in development. We briefly overview biological information retrieved from the knowledge bases, with details provided in Chandak *et al.*[11]: **Bgee**[49] contains gene expression patterns, and contributes to anatomy-protein associations where gene expression was present or absent to the KG. The **Comparative Toxicogenomics Database**[50] contributes relationships between environmental exposures and proteins, diseases, other exposures, biological processes, molecular functions, and cellular components to the dataset. **DisGeNET**[51] is an expert-curated resource about the relationships between genes and human disease and provides associations of genes with diseases and phenotypes in the KG. **DrugBank**[52] is a resource that contains pharmaceutical knowledge and supplies drug-drug and drug-protein interactions to the dataset. **Drug Central**[27] curates information about 26,698 indication edges, 8,642 contraindication edges, and 1,917 off-label use edges to the KG. **Entrez Gene**[53] is a resource maintained by the NCBI that contains associations of genes with biological processes, molecular functions, and cellular components. The **Gene Ontology**[54] network describes hierarchical associations between biological processes, molecular functions, and cellular components in the KG. The **Human Phenotype Ontology**[55] provides information on disease-phenotype, protein-phenotype, and phenotype-phenotype edges in the KG. Since the **Mondo Disease Ontology**[56] harmonizes diseases from a wide range of ontologies, including OMIM, SNOMED CT, ICD, and MedDRA, it was our preferred ontology for defining diseases and also provided hierarchical disease relations. **Protein-protein interactions** are composed of experimentally-verified interactions between proteins gathered from various resources[12,57–63]. **Reactome**[64] is an open-source, curated database for pathways that provides pathway-pathway and protein-pathway edges to the KG. The **Side Effect Resource** (SIDER)[65] contains data about adverse drug reactions and contributes drug phenotype associations to the KG. **UBERON**[66] helps include human anatomy information in the KG.

1.2 Harmonizing knowledge graph from primary data resources

To construct the knowledge graph, we harmonized ontologies for each node type, ensuring consistency by standardizing primary data sources and rectifying overlaps as described in Chandak *et al.*[11] Primary data were mapped into standardized ontologies, with drugs and diseases respectively encoded using DrugBank and Mondo Disease Ontology. For enhanced visualization in TxGNN Explainer and clarity in user studies, we introduced a 'display_relation' field as a more descriptive version of the 'relation' field, (e.g. 'disease_phenotype_negative' became 'phenotype absent').

We merged the harmonized datasets into a heterogeneous knowledge graph and extracted its largest connected component using the approach outlined in Chandak *et al.*[11]. Since the knowledge graph is designed for therapeutic use prediction, we wanted to ensure that disease nodes in the graph were meaningful representations of diseases. To this end, we adopted an approach previously validated[11] by collapsing disease nodes with nearly identical names into a single disease node. Intial disease groups were identified using automated string matching across disease names. These disease groupings were tightened using ClinicalBERT[67] embedding similarities between disease names with an empirically chosen cutoff of similarity ≥ 0.98. Finally, we manually approved the suggested disease matches and assigned names to the new groups. After grouping, 22,205 diseases in the Mondo Disease Ontology were collapsed into 17,080 grouped diseases.

2 Geometric deep learning approach

Notation. We are given a heterogeneous knowledge graph (KG) $G = (\mathcal{V}, \mathcal{E}, \mathcal{T}_R)$ with nodes in the vertex set $v_i \in \mathcal{V}$, edges $e_{i,j} = (v_i, r, v_j)$ in the edge set $\mathcal{E}$, where $r \in \mathcal{T}_R$ indicates the relation type, v_i is called the head/source node and v_j is called the tail/target node. Each node also belongs to a node type set $\mathcal{T}_V$. Each node also has an initial embedding, which we denote as $\mathbf{h}_i^{(0)}$.

Problem definition. Given a disease i and drug j, we want to predict the likelihood of the drug being (1) indicated for the disease or (2) contraindicated for the disease. The goal is to inject factual knowledge from the KG into AI application to imitate important skills possessed by human experts, i.e., reasoning and understanding when forming hypotheses and making predictions about disease treatments.

2.1 Overview of TxGNN approach

TxGNN is a deep learning approach for mechanistic predictions in drug discovery based on molecular networks perturbed in disease and targeted by therapeutics. TxGNN is composed of four modules: (1) a heterogeneous graph neural network-based encoder to obtain biologically meaningful network representation for each biomedical entity; (2) a disease similarity-based metric learning decoder to leverage auxiliary information to enrich the representation of diseases that lack molecular characterization; (3) an all-relation stochastic pre-training followed by a drug-disease centric full-graph fine-tuning strategy; (4) a graph explainability module to retain a sparse set of edges that are crucial for prediction as a post-training step. Next, we expand each module in detail.

2.2 Heterogeneous graph neural network encoder

Given a knowledge graph, we aim to learn a numerical vector (i.e., network embedding) for each node such that it captures biomedical knowledge encapsulated in the neighboring relational structures. This is achieved by transforming initial node embeddings through several layers of local graph-based non-linear function transformations to generate embeddings[32,68]. These functions are optimized iteratively, given a loss function to gradually minimize the error of making poor therapeutic use predictions. Upon convergence, optimized functions generate an optimal set of node embeddings.

Step 1: Initialization. We denote the input node embedding $\mathbf{X}_i$ for each node i, which is initialized using Xavier uniform initialization[69]. For every layer l of message-passing, there are the following three stages:

Step 2: Propagating relation-specific neural messages. For every relation type, first calculates a transformation of node embedding from the previous layer $\mathbf{h}^{(l-1)}$, where the first layer $\mathbf{h}^{(0)} = \mathbf{X}$. This is achieved via applying a relation-specific weight matrix $\mathbf{W}_{r,M}^{(l)}$ on the previous layer embedding:

$$\mathbf{m}_{r,i}^{(l)} = \mathbf{W}_{r,M}^{(l)} \mathbf{h}_i^{(l-1)} \tag{1}$$

Step 3: Aggregating local network neighborhoods. For each node v_i, we aggregate on the incoming messages $\{\mathbf{m}_{r,j}^{(l)} | j \in \mathcal{N}_r(i)\}$ from neighboring nodes of each relation r denoted as $\mathcal{N}_r(i)$ by taking the average of these messages:

$$\widetilde{\mathbf{m}^{(l)}}_{r,i} = \frac{1}{|\mathcal{N}_r(i)|} \sum_{j \in \mathcal{N}_r(i)} \mathbf{m}_{r,j}^{(l)} \tag{2}$$

Step 4: Updating network embeddings. We then combine the node embedding from the last layer and the aggregated messages from all relations to obtain the new node embedding:

$$\mathbf{h}_i^{(l)} = \mathbf{h}_i^{(l-1)} + \sum_{r \in \mathcal{T}_R} \widetilde{\mathbf{m}^{(l)}}_{r,i} \tag{3}$$

After L layers of propagation, we arrive at our encoded node embeddings $\mathbf{h}_i$ for each node i.

2.3 Predicting drug-disease relationships

Given the disease embedding and the drug embedding, we can predict the interaction between a disease-drug pair. As we have three relation types to predict for each disease-drug pair, we use a trainable weight vector $\mathbf{w}_r$ for each relation type. We then use DistMult[70] to calculate the interaction likelihood for that relation. Formally, for disease i, drug j, and relation r, we calculate the predicted likelihood p:

$$p_{i,j,r} = \frac{1}{1 + \exp(-\mathrm{sum}(\mathbf{h}_i \cdot \mathbf{w}_r \cdot \mathbf{h}_j))}. \tag{4}$$

2.4 Similarity Disease Search to Enrich Molecularly Uncharacterized Disease Embedding

Diseases receive various degrees of research, given their prevalence, complexity, and so on. For example, we know very little about the molecular underpinnings of many rare diseases[71,72]. Nevertheless, these diseases usually present the most promising therapeutic opportunities[73]. Due to the lack of understanding of these diseases, machine learning predictions have become more important than ever. However, the limited research on these diseases is reflected by the scarcity of relevant nodes and edges around these diseases in our biological knowledge graph. Because of this sparsity, the graph embedding tends to be lower quality. For example, if a disease has zero connections in the KG (i.e., no existing knowledge), then the disease embedding will be the random initialization. Empirically, we see that prevailing GNN approaches have drastically lower predictive performance on our disease-centric splits to simulate this realistic property of diseases compared to random splits (Figure 1g).

We hypothesize that the obtained network embedding for these diseases is not meaningful due to this limited prior in the KG. Thus, a model must subsidize and augment the network embedding for these molecularly uncharacterized diseases. Our key insight is that human physiology is a connected system where diseases are similar across dimensions (e.g., lung cancer is similar to brain cancer in the dimension of cancer diseases, while lung cancer is similar to asthma in the dimension of lung diseases). Therefore, if we could borrow useful information from a set of similar diseases that are relatively well-characterized in the KG through the model, we could augment the embedding of the candidate disease and improve the prediction.

To do that, we propose a three-step procedure: (1) a disease signature vector that captures the intricate disease similarities; (2) an aggregation mechanism that integrates the different similar diseases into a robust auxiliary embedding that can subsidize original disease embedding; (3) a gating mechanism to control the effect between the original disease embedding and the auxiliary disease embedding since many well-characterized diseases have sufficient embeddings and do not need subsidies. We discuss each of the three steps in detail below.

Network-based Disease Signature Profiling. The overall goal for this module is to obtain a signature vector $\mathbf{p}_i$ for every disease i. There are numerous ways to calculate the similarity between two diseases. As disease representations generated by the graph neural network alone are not sufficient to characterize the candidate disease, they ideally should not be directly used to calculate similarity. Instead, we resort to graph theoretical techniques that are rooted in the field of network science[16]. We consider the following three types of signature functions:

- **Protein signatures (PS):** The mechanism of actions for small molecule drugs is to act upon protein targets in the disease pathway[74]. Thus, the ideal disease signature should preserve similarity in the protein target space. If two diseases have similar proteins in their corresponding disease pathways, they are more likely to have a similar treatment mechanism[12,75]. This key observation motivates the protein signature[76]. We have a bit vector for each disease where each bit corresponds to a specific protein. A bit is flipped to one if the bit corresponds to a protein in the disease pathway. Formally, for disease i, the protein signature is defined as:

$$\mathbf{p}_i^{\text{PS}} = [\, \text{p}_1 \; \cdots \; \text{p}_{|\mathcal{V}_{\text{P}}|} \,], \tag{5}$$

where

$$\mathrm{p}_j = \begin{cases} 1 & \text{if } j \in \mathcal{N}_i^{\mathrm{P}} \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

and $\mathcal{N}_i^{\mathrm{P}}$ is the set of proteins that lie in the 1-hop neighborhood of disease i and $|\mathcal{V}_{\mathrm{P}}|$ the number of total available proteins. To calculate similarity between two diseases i, j, we use dot product:

$$\mathrm{sim}^{\mathrm{PS}}(i, j) = \mathbf{p}_i^{\mathrm{PS}} \cdot \mathbf{p}_j^{\mathrm{PS}} = |\mathcal{N}_i^{\mathrm{P}} \cap \mathcal{N}_j^{\mathrm{P}}|. \tag{7}$$

The similarity directly measures the number of intersecting proteins in the disease pathway of i, j. If the similarity is high, we know these two diseases have a larger number of intersecting diseases, which increases the probability of similar treatment mechanisms.

- **All-node-types signatures (AT):** Human knowledge about disease pathways are vastly incomplete. Thus, some diseases may not have complete protein pathways in the knowledge graph, which leads to biased protein signatures. Additional biological knowledge about diseases could potentially benefit. In the knowledge graph, other node types connect to diseases, including effect/phenotype, exposure, and disease. Since the local neighborhood can define some characteristics of diseases, we can extend the principle of protein signature, such that if two diseases share the same nodes in these additional node types, they have similar biological underpinnings. We call these all-node-types signatures. Formally, for disease i, the protein signature is defined as:

$$\mathbf{p}_i^{\mathrm{AT}} = [\ \mathrm{p}_1 \cdots \mathrm{p}_{|\mathcal{V}_{\mathrm{P}}}\ \mathrm{ep}_1 \cdots \mathrm{ep}_{|\mathcal{V}_{\mathrm{EP}}|}\ \mathrm{ex}_1 \cdots \mathrm{ex}_{|\mathcal{V}_{\mathrm{EX}}|}\ \mathrm{ep}_1 \cdots \mathrm{ep}_{|\mathcal{V}_{\mathrm{EP}}|}\ \mathrm{d}_1 \cdots \mathrm{d}_{|\mathcal{V}_{\mathrm{D}}|}\ ], \tag{8}$$

where

$$\mathrm{p}_j = \begin{cases} 1 & \text{if } j \in \mathcal{N}_i^{\mathrm{P}} \\ 0 & \text{otherwise} \end{cases}, \mathrm{ep}_j = \begin{cases} 1 & \text{if } j \in \mathcal{N}_i^{\mathrm{EP}} \\ 0 & \text{otherwise} \end{cases}, \mathrm{ex}_j = \begin{cases} 1 & \text{if } j \in \mathcal{N}_i^{\mathrm{EX}} \\ 0 & \text{otherwise} \end{cases} \mathrm{d}_j = \begin{cases} 1 & \text{if } j \in \mathcal{N}_i^{\mathrm{D}} \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

and $\mathcal{N}_i^{\mathrm{EP}}, \mathcal{N}_i^{\mathrm{EX}}, \mathcal{N}_i^{\mathrm{D}}$ is the set of effect/phenotype, exposure, diseases nodes lie in the 1-hop neighborhood of disease i and $|\mathcal{V}_{\mathrm{EP}}|, |\mathcal{V}_{\mathrm{EX}}|, |\mathcal{V}_{\mathrm{D}}|$ the number of total available effect/phenotype, exposure, diseases respectively. We also adopt the dot product as the similarity measure, which means the similarity is the sum of all shared nodes across the four node types:

$$\mathrm{sim}^{\mathrm{AT}}(i,j) = \mathbf{p}_i^{\mathrm{AT}} \cdot \mathbf{p}_j^{\mathrm{AT}} = |\mathcal{N}_i^{\mathrm{P}} \cap \mathcal{N}_j^{\mathrm{P}}| + |\mathcal{N}_i^{\mathrm{EP}} \cap \mathcal{N}_j^{\mathrm{EP}}| + |\mathcal{N}_i^{\mathrm{EX}} \cap \mathcal{N}_j^{\mathrm{EX}}| + |\mathcal{N}_i^{\mathrm{D}} \cap \mathcal{N}_j^{\mathrm{D}}|. \quad (10)$$

- **Diffusion signatures (DS):** The above two signatures rely on the first-hop neighbor of the diseases, while higher-hop neighbors may contain useful molecular characterization. Diffusion signature simulates many random walks, where each random walk is a path of length h starting from the disease i: $w = v_i \xrightarrow{e_{i,1}} v_1 \cdots v_{h-1} \xrightarrow{e_{h-1,h}} v_h$[77]. The set of visited nodes in the k-th random walk from disease node i is denoted as $\mathcal{W}_i^k$. $\cap_k \mathcal{W}_i^k$ represents the total set of visited nodes across all walks, and we can calculate the normalized visitation probability for visited node j as:

$$f_j = \frac{\sum_k \sum \mathbb{1}_{\mathcal{W}_i^k = j}}{\sum_k |\mathcal{W}_i^k|} \quad (11)$$

  These nodes correspond to a multi-hop snapshot of molecular interactions centering around the diseases, and the visitation probability corresponds to the influence level. Given this probability score, we can obtain the diffusion signature for disease node i:

$$\mathbf{p}_i^{\mathrm{DS}} = [\; f_1 \;\cdots\; f_{|\mathcal{V}_{\mathrm{P}}|} \;]. \quad (12)$$

  For diffusion signature, we still use the dot product:

$$\begin{aligned} \mathrm{sim}^{\mathrm{DS}}(i,j) = \mathbf{p}_i^{\mathrm{DS}} \cdot \mathbf{p}_j^{\mathrm{DS}} &= \sum_u^{|\mathcal{V}_{\mathrm{P}}|} \frac{\left(\sum_k \sum \mathbb{1}_{\mathcal{W}_i^k = u}\right) \cdot \left(\sum_k \sum \mathbb{1}_{\mathcal{W}_j^k = u}\right)}{(\sum_k |\mathcal{W}_i^k|)^2} \\ &\sim \sum_u^{|\mathcal{V}_{\mathrm{P}}|} \left(\sum_k \sum \mathbb{1}_{\mathcal{W}_i^k = u}\right) \cdot \left(\sum_k \sum \mathbb{1}_{\mathcal{W}_j^k = u}\right). \end{aligned} \quad (13)$$

  Note the denominator $(\sum_k |\mathcal{W}_i^k|)^2 = (|k| * h)^2$ is a constant. Intuitively, the similarity between diseases i and j is higher when two diseases visit more shared nodes at a higher frequency.

Given the selected signature for diseases and calculated similarities among the diseases, for a query disease, we can then obtain k most similar diseases for a query disease i:

$$\mathcal{D}_{\mathrm{sim},i} = \mathrm{argmax}_{j \in \mathcal{V}_{\mathrm{D}}}^{k} \mathrm{sim}(i,j) \quad (14)$$

Disease-disease metric learning. Given the set of similar diseases, we aim to obtain an embedding that fuses different similarity dimensions into a single embedding sufficient to enhance the query disease that might be sparsely annotated. We use a weighted scheme, where the similarity score weights each disease as follows:

$$\mathbf{h}_i^{sim} = \sum_{j \in \mathcal{D}_{\text{sim}}} \frac{\text{sim}(i,j)}{\sum_{k \in \mathcal{D}_{\text{sim}}} \text{sim}(i,k)} * \mathbf{h}_j. \tag{15}$$

Embedding gating. The final step is to update the original disease embedding $\mathbf{h}_i$ with the disease-disease metric learning embedding $\mathbf{h}_i^{sim}$ through a gating mechanism. The gating mechanism consists of a scalar $c \in [0,1]$ that balances between these two types of embeddings. Note that this requires special treatment because for a disease well-characterized in the knowledge graph, we do not need the disease-disease metric learning embedding, and it potentially can even bias the final embedding. The disease-disease metric learning embedding is most useful for uncharacterized diseases since the original disease embedding is insufficient to characterize molecular mechanisms. Note that the learnable attention mechanism to select whether or not to attend original/augmented embedding does not work well because the training examples are usually the most characterized, which makes the attention weight assign high importance to the original embeddings and leaves the subsidy embedding unused. Instead, we propose a heuristic algorithm that assigns weight based on the node degree for the drug-disease relation type that is under calculation: $|\mathcal{N}_i^r|$. The higher the degree, the more well-characterized the disease is, and the less weight should be assigned to the disease-disease metric learning embedding and vice-versa. Also, this scalar should have a very high value when the node degree is minimal (0 or 1) and decreases quickly when the node degree increases. To approximate this effect, we use an inflated exponential distribution density function with $\lambda = 0.7$:

$$c_i = 0.7 * \exp(-0.7 * |\mathcal{N}_i^r|) + 0.2 \tag{16}$$

We observe the result is not sensitive to λ (Supplementary Figure 6). Finally, we use parameter search and find optimal $\lambda = 0.7$. Then, we can finally obtain an augmented disease embedding:

$$\widehat{\mathbf{h}}_i = c_i * \mathbf{h}_i^{sim} + (1 - c_i) * \mathbf{h}_i \tag{17}$$

We then use this augmented disease embedding to feed into the DistMult decoder[70] described in

Section 2.3.

2.5 Training TxGNN deep graph models

Objective function. The training objective is to accurately predict whether or not a relation holds given two entities in the knowledge graph. This can be formulated as a binary classification task for each relation. The positive samples consist of all pairs (i, j) with diverse relation types $r \in \mathcal{T}_R$. We denote this as $\mathcal{D}_+$ and the label $y_{i,r,j} = 1$. Similarly, for each pair, we generate negative counterparts through sampling described in Section 3, denoted as $\mathcal{D}_-$. For each pair i, j and its relation type r, the model predicts the likelihood $p_{i,j,r}$ and the training loss is calculated via binary cross entropy loss:

$$\mathcal{L} = \sum_{(i,r,j) \in \mathcal{D}_+ \cup \mathcal{D}_-} y_{i,r,j} * \log\left(p_{i,r,j}\right) + \left(1 - y_{i,r,j}\right) * \log\left(1 - p_{i,r,j}\right) \tag{18}$$

Previous work has focused on knowledge graph completion, leading them to optimize over the entire set of relations in the knowledge graph[78]. However, since we are only interested in drug-disease relations, training on all relation types could move the model capacity toward capturing knowledge we are not interested in. Conversely, since complicated biological mechanisms drive drug-disease relations, the vast array of biomedical relations in the knowledge graph presents a unique information source that holistically describes biological systems. Thus, the challenge is to ultimately do well on a small set of relations while also transferring knowledge positively from the larger relation set.

To solve this challenge, TxGNN uses a pre-training strategy. During pre-training, TxGNN is trained to predict relations across the entire set of relation types in the KG using stochastic mini-batching. This process allows TxGNN to distill biomedical knowledge into enriched node embeddings. Next, during fine-tuning, TxGNN zooms in and trains only on the drug-disease relations to obtain more granular drug-disease-specific embeddings that optimize for the best therapeutic outcomes.

Pre-training. TxGNN is first pre-trained on millions of biomedical entity pairs across the entire set of relations. As there are millions of edges, full-graph training is computationally infeasible. Thus, we use stochastic mini-batching to train only on a set of pairs in each training step. Each epoch goes through all pairs of data in the training knowledge graph. During pre-training, degree-adjusted disease augmentation is turned off since it is unavailable for other node types. All relations are treated equally. The weights of the trained encoder model are then used to initialize the encoder model weights during fine-tuning. Note that the weight in the decoder DistMult $\mathbf{w}_r$ is reinitialized before fine-tuning to discourage the effect of negative transfer.

Fine-tuning. After pre-training, we have an initialization that captures general biological knowledge. Next, we focus on optimizing drug-disease relation prediction. To do that, we only use the samples of all drug-disease pairs (i, j) with relation types $r \in \{$indication, contraindication, rev_indication, rev_contraindication$\}$. The rest of the relations are discarded in the training objective but are included in the knowledge graph for messaging the passing of drug and disease nodes. During fine-tuning, the degree-adjusted inter-disease embedding is turned on.

The complete TxGNN model is pre-trained and fine-tuned in an end-to-end manner. The best-performing model on the validation set is then used for performance evaluation on the test set and downstream machine-learning analyses.

2.6 Explaining model predictions

Distilling model predictions into mechanisms of molecular networks perturbed in disease and targeted by therapeutics. A machine learning model can provide accurate disease treatment predictions. However, for domain scientists' adoption, prediction alone is not sufficient. Thus, a model is expected to generate why it outputs this prediction in a form familiar to domain experts' decision-making. In the case of treatment prediction, an ideal form of explanation is to simulate how drug developers approach drug-disease relation — that is, to understand how a drug perturbs the local biological system such that it creates a therapeutic effect on the disease pathway. As TxGNN leverages the large-scale biological knowledge graph, we can probe into the local neighborhood around a query drug-disease node and pinpoint the exact mechanism contributing to the prediction. However, as a biological network is complex, making meaningful explanations requires a model to prune most uninformative edges and extract a sparse version of the local neighborhood. This can be formulated as a graph explainability problem where we try to identify a sparse set of edges where the model can make a faithful prediction using these edges[42]. To achieve it, we develop a post-training graph explainability module, adapted from GraphMask approach[28], that can drop spurious edges from the dataset and retain a sparse set of edges that contribute most towards the prediction. Next, we describe the mathematical formulation of GraphMask as used by TxGNN.

Local explanation subgraphs through pruning superfluous biomedical relations. Given a trained disease treatment prediction model, for each target node j and one of the neighbor source node i with edge $e_{i,j}$ at layer l, we have intermediate messages $\mathbf{m}_{r,i}^{(l)}$, $\mathbf{m}_{r,j}^{(l)}$ given a relation r. Given these two embeddings, we concatenate them and feed them into a relation-wise single-layer neural network parameterized by $\mathbf{W}_{g,r}^{(l)}$ to predict the likelihood of masking the message from source node i when we compute the target node j embedding, followed by a gate consisting of a sigmoid layer to squeeze the likelihood into 0 to 1 and an indicator function to decide whether or not to drop the edge:

$$z_{i,j,r}^{(l)} = \mathbf{1}_{[\mathbb{R}>0.5]}\left(\text{sigmoid}\left(\mathbf{W}_{g,r}^{(l)}\left(\mathbf{m}_{r,i}^{(l)}\|\mathbf{m}_{r,i}^{(l)}\right)\right)\right), \tag{19}$$

such that $z_{i,j,r}^{(l)} \in 0, 1$. In practice, we add a location bias of 3 to the sigmoid function at initialization. This ensures that for initialized inputs, the biased sigmoid outputs are close to 1, meaning that the gates are open at initialization, and the model can adaptively close the gates to mask edges in the subgraph. This step is crucial as random initialization starts by dropping random edges. The gap between the original and updated predictions is big, so the model minimizes the gap instead of balancing the two objectives. Next, instead of simply removing the message when the gate outputs 0, the message is replaced with a learnable baseline vector $\mathbf{b}_r^{(l)}$ for each relation r and layer l. Thus, the updated message from source node i to target node j becomes:

$$\hat{\mathbf{m}}_{i,r}^{(l)} = z_{i,j,r}^{(l)} \cdot \mathbf{m}_{i,r}^{(l)} + (1 - z_{i,j,r}^{(l)}) \cdot \mathbf{b}_r^{(l)} \tag{20}$$

Then, we can proceed with the standard message aggregation and update steps to compute the updated node embedding (Section 2.2), feed to inter-disease augmentation (Section 2.4), and generate the updated predictions $\hat{p}$ between a drug and a disease (Section 2.3). The GraphMask gate weights are optimized with two objectives. The first objective is faithfulness, where the updated predictions after masking are encouraged to be the same as the original prediction outcome. The second objective is to promote the model to mask as much as possible. These two objectives present a trade-off since larger amounts of masking would lead to a more significant gap between updated/original predictions. This can be formulated as constrained optimization using Lagrange relaxation, where we strive to maximize the Lagrange multiplier λ for constraint while minimizing the main objective. Formally, we use the loss function below:

$$\max_{\lambda} \min_{\mathbf{W}_g} \sum_{k=1}^{L} \sum_{(i,j,r)\in\mathcal{D}_+\cup\mathcal{D}_-} \mathbf{1}_{[\mathbb{R}\neq 0]} z_{i,j,r}^{(k)} + \lambda \left(\|\hat{p}_{i,j,r} - p_{i,j,r}\|_2^2 - \beta \right), \tag{21}$$

where β is the margin between the updated and original prediction. After training, we can remove edges (i, j, r) that have $z_{i,j,r}^{(k)} = 0$ and use the retained edges as the explanations. We can also use the value calculated before the indicator function to measure the level of contributions to the prediction and can be used as adjustments of more granular differences.

Necessary adaptations of GraphMask approach for biomedical knowledge graphs. We modify GraphMask[28] in the following manner to generate meaningful local explanation subgraphs of the knowledge graph. (1) Instead of a complex gate that outputs scores close to 0/1, we adopt a smooth sigmoid gate where predictions are uniform across 0 to 1. This is important because we find hard concrete map edges to 1 as long as they affect the model prediction. However, this still keeps many edges that preclude us from making acceptable medical explanations. The sigmoid gate instead allows us to distinguish the intensity of contributions and provides a flexible framework. By setting a threshold, we remove large amounts of positive edges and only retain ones crucial for the model prediction. (2) Second, while GraphMask has a single learnable weight for every edge in the dataset, we adopt a separate weight for each relation. Since embeddings across relations are different, the model assigns uniformly high scores for all edges of a given relation type despite edges varying in relevance. Using relation-specific weights allows the model to capture the importance scores of individual edges.

3 Experimental setup and implementation details

Next, we outline the experimental setup, including information on performance evaluation and dataset splits. We also provide details on the practical implementation of TxGNN deep graph models.

3.1 Creating dataset splits for rigorous performance evaluation

Our dataset presents well-studied information and includes the vast majority of existing treatments. As a result, it is easy to predict treatments for diseases with various pre-existing treatments. However, for zero-shot prediction of therapeutic use, we need to make good predictions on conditions with few or no current treatments available. The classical random split of edges of the knowledge graph into training and testing sets would not simulate this application. In the random split, for diseases with many known indications, the model would view some of these drug-disease edges in training and thus easily predict therapeutic use based on drug similarity. However, this would prevent the model from assimilating meaningful biological knowledge. Therefore, we consider the following dataset splits into training and test sets:

- **Disease area splits:** Many diseases of therapeutic interest have no existing treatments and lack significant biological knowledge. To evaluate whether TxGNN would be robust to predicting drug-disease relationships for such diseases, we develop data splits that simulate well-studied diseases as molecularly uncharacterized diseases. We cannot directly test on molecularly uncharacterized diseases, such as rare diseases, because the treatments are too few to build a confident machine learning model. We select five disease groups: cell proliferation, mental health, cardiovascular diseases, anemia, and adrenal gland diseases, and then extract groups for these diseases from the Disease Ontology hierarchy such that group includes the disease and all its children. Since these well-studied diseases have many drug-disease relationships, we can easily evaluate the model's performance during the simulation.

  For each disease, we create a separate data split as follows. First, all the drug-disease edges connected to the diseases in the group are moved to the test set. As a result, TxGNN has no information about existing indications and contraindications use edges for the chosen disease group during training. This simulates the lack of existing treatments encountered with molecularly uncharacterized diseases. Next, we remove a significant fraction (5% of the knowledge graph size) of the local 1-hop subgraph neighborhood for the disease group. Again, this simulates the limited biological understanding of molecularly uncharacterized diseases. Dataset statistics of each disease area split is provided in Table 1.

- **Systematic dataset splits:** The deployed machine learning model should excel at predicting diseases without known treatments. Predicting new treatments for diseases that already have treatments is easier than predicting diseases without treatments. This is because information about existing treatments can directly illuminate the molecular mechanism, and drug similarity can help infer new treatments. Thus, to robustly test our model, we design this split to systematically study prediction on novel unseen diseases. To do that, we first randomly split the entire set of diseases. Then, we take all drug-disease relations associated with the testing set of diseases to the test set such that there are no known treatments during training and the testing set consists of novel diseases. The testing set has around 100 different diseases in each randomly seeded run.

- **Disease-centric dataset splits:** We adopt a disease-centric evaluation to simulate realistic usage of drug candidate prioritization. First, for each disease in the test set, we pair it with all other drugs in the KG, except the drug-disease relations in the training set. Then, we make predictions for all pairs and rank based on the likelihood of interaction. We then retrieve the top K drugs and compute the recall (*i.e.*, how much drug and disease in the testing set are in the top K). Finally, we build a baseline of random screening where we randomly sample top K drugs from the drug set and compute the recall.

3.2 Modeling molecular and clinical relationships

In graphs, each edge typically has a direction and points from the source to the target node. However, in our biological knowledge graph, edges are bidirectional. For example, a drug A indicated for disease B is represented in TxGNN by a tuple $(A, \text{indication}, B)$. Similarly, disease B can be treated by drug A, corresponding to a tuple $(B, \text{rev\_indication}, A)$. For homogeneous relation type (*e.g.*, protein-protein interactions) where the head and tail node belongs to the same node types, there is no additional reverse relation type as the reverse edges are collapsed into the original relation type. Thus, we add these reverse relation types to the knowledge graph, following standard practice. For the sake of notation, when the reverse relation has a different relation type from the original type r, we denote the reverse relation type as r^c.

3.3 Negative sampling for training TxGNN models

As we only have positive data, negative data are constructed via sampling. The sampling from the unobserved simulates the realistic constraint where most possible drugs do not interact with the disease. For each relation type, we fix the source nodes and permute the target nodes through either random sampling from the set of nodes associated with this relation type's target nodes or a weighted sampling based on the degree of the target nodes. As we conduct reverse relation type construction, the source node type would also be shuffled and included in the negative samples when we do sampling for the reverse relation type. This concept of negative sampling based on shuffling target nodes is crucial. For example, suppose we want to study drugs A that can treat disease B, then we narrow down to the relation $(B, \text{rev\_indication}, A)$ instead of the $(A, \text{indication}, B)$.

3.4 Hyperparameter tuning

We conduct hyperparameter tuning using Hyperband on validation set micro AUROC using complex disease split following two stages. The first is to optimize the parameters for pre-training and fix fine-tuning parameters, where we conduct a sweep of grid search with a learning rate of $\{1e-4, 5e-4, 1e-3\}$, batch size of $\{1024, 2048\}$, and epoch size of $\{1, 2, 3\}$. Next, we fix the pre-training parameters and do a grid search for fine-tuning parameters with the hidden size of $\{64, 128, 256, 512\}$, input size of $\{64, 128, 256, 512\}$, output size of $\{64, 128, 256, 512\}$, number of inter-disease prototypes of $\{3, 5, 10, 20, 50\}$ and learning rate of $\{1e-4, 5e-4, 1e-3\}$. We obtain a final set of hyperparameters with a pre-training learning rate of $1e-3$, batch size of 1024, epoch size of 2, the fine-tuning learning rate of $5e-4$, hidden size of 512, input size of 512, output size of 512, number of prototypes 3.

3.5 Implementation details

The TxGNN is implemented using DGL[79] and PyTorch[80] Python deep learning frameworks. We use Pandas[81], Numpy[82] for data processing and computing; scikit-learn[83] for evaluation metrics; seaborn[84], matplotlib[85], UMAP[86] for visualization; Weights and Bias (https://www.wandb.ai) for training monitoring and hyperparameter tuning. We train the model with one NVIDIA Tesla V100 GPU in a server. TxGNN Explorer is implemented in JavaScript using React.js[87], D3.js[88], and Ant Design[89]. The graph data is managed using Neo4j database[90]. TxGNN Explorer communicates with TxGNN through a Python web server built with Flask[91].

3.6 Implementations about existing methods

We follow the original author's implementations for the baselines. Particularly, for network medicine statistics including KL, JS, proximity, and DSD, we used the codebase that recently benchmarked these scores for COVID-19 targets in https://github.com/Barabasi-Lab/COVID-19/tree/main. For HAN and HGT, we used the implementations in the Pytorch Geometric library, notably, the HANConv, and HGTConv layers. For BioBERT, we used the huggingface repository https://huggingface.co/dmis-lab/biobert-v1.1 to download the raw model weights and then applied an MLP decoder to make predictions. For all baselines, we use the exact same data splits as in TxGNN for a fair comparison.

3.7 Usability study of TxGNN with medical experts

We designed and developed TxGNN Explorer following a user-centric design study process[30], which compared three visual presentations of GNN explanations from users' perspectives and motivated the implementation of path-based explanations based on user feedback. We evaluated the usability of TxGNN Explorer by comparing it with a non-explanation baseline that shows drug predictions and corresponding confidence scores. Twelve medical experts (7 males, 5 females, avg. age=34.25) were recruited for the usability study through personal contacts, Slack channels, and email lists in collaborating institutions. We have obtained informed consent from all participants. We conducted the evaluation on Zoom due to COVID-19-related restrictions. Each participant logged in to the user study system (Supplementary Figure S5) using their computers and shared their screens with the interviewer. The order of predictions and the order of two conditions (TxGNN Explorer or baseline) were randomized and counterbalanced across participants. For each drug assessment task, the participants were asked to 1) decide whether this drug prediction is correct (*i.e.*, the drug can potentially be used to treat the disease) and 2) give a confidence score for their decision using a 5-point Likert scale (1=not confident at all, 5=completely confident). The study system automatically recorded the completion time for assessing each prediction. After assessing all predictions, participants provided subjective ratings for the two conditions in terms of *Trust*, *Helpfulness*, *Understandability*, and *Willingness to use* via a 5-point Likert scale (1=strongly disagree, 5=strongly agree).

3.8 Evaluations within a large healthcare system

We leveraged patient data from the Mount Sinai Health System's electronic health records (EHR) in New York City, U.S., to assess patterns from predictions in clinical practice. All clinical data were deidentified, and the Mount Sinai Institutional Review Board approved the study. The cohort consisted of over 10 million patients and was filtered for patients over 18 years of age with at least one drug and at least one diagnosis on record, leaving 1,272,085 patients. This cohort was 40.1 percent male, and the average age was 48.6 years (SD: 18.6 years). Table 2 shows the dataset's racial breakdown.

All disease and medication data were captured using the Observational Medical Outcomes Partnership (OMOP)[92,93] standard data model. We produce predictions for the 1,363 diseases with indications by training the full knowledge graph with only 5% of randomly selected drug-disease pairs as a validation set for early stopping. This experiment does not evaluate zero-shot performance for all 17,080 diseases since the model has more confidence in conditions with known indications. Disease names in the TxGNN prediction dataset were matched to SNOMED or ICD-10 codes and finally mapped to OMOP concepts in the Mount Sinai data system. We included only diseases with at least one patient diagnosis in the dataset, leaving 480 conditions. Medication names in the TxGNN prediction were matched to DrugBank ID, which was then mapped to RxNorm IDs and OMOP concepts. We included only medications with at least one patient order in the dataset, leaving 1,290 medications. Next, we included drug-disease pairs for which at least one patient was listed with both the drug and the disease, leaving 1,236 drugs and 470 diseases. For each drug-disease pair, we created a contingency table. Using the SciPy[94] library's Fisher exact function, we computed 2-sided odds ratios and p-values for each pair. Finally, we used the statsmodels[95] Python library's multi-test function to apply a two-sided Bonferonni correction on the previously generated p-values. Finally, we noted statistically significant drug-disease pairs as those with $p < 0.005$.

Disease area	Number of diseases	Number of indications	Number of Contraindications
Adrenal gland	7	41	374
Anemia	19	88	752
Cardiovascular diseases	113	453	4,242
Diseases of cell proliferation	213	1022	1079
Mental health diseases	60	355	1,567

Table 1: Statistics on disease-area-based dataset splits used to evaluate the zero-shot prediction of therapeutic use. Given all diseases in a given disease area, all indications and contraindications were removed from the dataset used to train machine learning models. Additionally, a large fraction (95%) of the connections between biomedical entities to these diseases were removed from the therapeutics-centered knowledge graph. Disease-area splits were curated to evaluate model performance on diseases with limited molecular understanding and no existing treatments.

Racial group	Count	Percent (%)
Asian	60,041	4.7
Black	162,102	12.7
White	534,305	42.0
Unknown	241,998	19.0
Other	273,639	21.5
Total number of patients	1,272,085	100.0

Table 2: Demographics of the electronic health record dataset at Mount Sinai Health System in New York City used to validate TxGNN's hypotheses on therapeutic use prediction.

References

1. Food, U. & Administration, D. Rare Disease Day 2021. https://www.fda.gov/news-events/fda-voices/rare-disease-day-2021-fda-shows-sustained-support-rare-disease-product-development-during-public (2023). [Online; accessed 19-September-2023].

2. Feigin, V. L. *et al.* Burden of neurological disorders across the us from 1990-2017: a global burden of disease study. *JAMA neurology* **78**, 165–176 (2021).

3. Vetter, N. Editor's choice. *British Medical Bulletin* **93**, 1–5 (2010).

4. Pushpakom, S. *et al.* Drug repurposing: progress, challenges and recommendations. *Nature Reviews Drug Discovery* **18**, 41–58 (2019).

5. Braga, S. S. Multi-target drugs active against leishmaniasis: A paradigm of drug repurposing. *European Journal of Medicinal Chemistry* **183**, 111660 (2019).

6. Dorlo, T. P., Balasegaram, M., Beijnen, J. H. & de Vries, P. J. Miltefosine: a review of its pharmacology and therapeutic efficacy in the treatment of leishmaniasis. *Journal of Antimicrobial Chemotherapy* **67**, 2576–2597 (2012).

7. Abdelsayed, M., Kort, E. J., Jovinge, S. & Mercola, M. Repurposing drugs to treat cardiovascular disease in the era of precision medicine. *Nature Reviews Cardiology* **19**, 751–764 (2022).

8. Sahragardjoonegani, B., Beall, R. F., Kesselheim, A. S. & Hollis, A. Repurposing existing drugs for new uses: a cohort study of the frequency of FDA-granted new indication exclusivities since 1997. *Journal of Pharmaceutical Policy and Practice* **14** (2021).

9. Sardana, D. *et al.* Drug repositioning for orphan diseases. *Briefings in Bioinformatics* **12**, 346–356 (2011).

10. Jourdan, J.-P., Bureau, R., Rochais, C. & Dallemagne, P. Drug repositioning: a brief overview. *Journal of Pharmacy and Pharmacology* **72**, 1145–1151 (2020).

11. Chandak, P., Huang, K. & Zitnik, M. Building a knowledge graph to enable precision medicine. *Scientific Data* **10**, 67 (2023).

12. Menche, J. *et al.* Uncovering disease-disease relationships through the incomplete interactome. *Science* **347** (2015).

13. Zitnik, M., Feldman, M. W., Leskovec, J. *et al.* Evolution of resilience in protein interactomes across the tree of life. *Proceedings of the National Academy of Sciences* **116**, 4426–4433 (2019).

14. Ruiz, C., Zitnik, M. & Leskovec, J. Identification of disease treatment mechanisms through the multiscale interactome. *Nature Communications* **12**, 1–15 (2021).

15. Goh, K.-I. *et al.* The human disease network. *Proceedings of the National Academy of Sciences* **104**, 8685–8690 (2007).

16. Barabási, A.-L., Gulbahce, N. & Loscalzo, J. Network medicine: a network-based approach to human disease. *Nature Reviews Genetics* **12**, 56–68 (2011).

17. Li, M. M., Huang, K. & Zitnik, M. Graph representation learning in biomedicine and healthcare. *Nature Biomedical Engineering* 1–17 (2022).

18. Gysi, D. M. *et al.* Network medicine framework for identifying drug-repurposing opportunities for covid-19. *Proceedings of the National Academy of Sciences* **118** (2021).

19. Cao, M. *et al.* Going the distance for protein function prediction: A new distance metric for protein interaction networks. *PLoS ONE* **8**, e76339 (2013).

20. Cheng, F. *et al.* Network-based approach to prediction and population-based validation of in silico drug repurposing. *Nature Communications* **9** (2018).

21. Zitnik, M., Agrawal, M. & Leskovec, J. Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics* **34**, i457–i466 (2018).

22. Guney, E., Menche, J., Vidal, M. & Barábasi, A.-L. Network-based in silico drug efficacy screening. *Nature Communications* **7**, 1–13 (2016).

23. Cheng, F., Kovács, I. A. & Barabási, A.-L. Network-based prediction of drug combinations. *Nature Communications* **10**, 1–11 (2019).

24. Fermaglich, L. J. & Miller, K. L. A comprehensive study of the rare diseases and conditions targeted by orphan drug designations and approvals over the forty years of the orphan drug act. *Orphanet Journal of Rare Diseases* **18**, 1–8 (2023).

25. Guney, E. Reproducible drug repurposing: When similarity does not suffice. In *Pacific Symposium on Biocomputing 2017*, 132–143 (World Scientific, 2017).

26. Brown, A. *et al.* Detecting shortcut learning for fair medical ai using shortcut testing. *Nature Communications* **14**, 4314 (2023).

27. Avram, S. *et al.* DrugCentral 2021 supports drug discovery and repositioning. *Nucleic Acids Research* **49**, D1160–D1169 (2021).

28. Schlichtkrull, M. S., De Cao, N. & Titov, I. Interpreting graph neural networks for NLP with differentiable edge masking. *International Conference on Learning Representations* (2021).

29. Lundberg, S. M. & Lee, S.-I. A unified approach to interpreting model predictions. *NeurIPS* **30** (2017).

30. Wang, Q., Huang, K., Chandak, P., Zitnik, M. & Gehlenborg, N. Extending the nested model for user-centric xai: A design study on gnn-based drug repurposing. *IEEE Transactions on Visualization and Computer Graphics* **29**, 1266–1276 (2023).

31. Cao, M. *et al.* Going the distance for protein function prediction: a new distance metric for protein interaction networks. *PloS one* **8**, e76339 (2013).

32. Schlichtkrull, M. *et al.* Modeling relational data with graph convolutional networks. In *ESWC*, 593–607 (Springer, 2018).

33. Hu, Z., Dong, Y., Wang, K. & Sun, Y. Heterogeneous graph transformer (2020).

34. Wang, X. *et al.* Heterogeneous graph attention network (2019).

35. Lee, J. *et al.* BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* btz682 (2019).

36. Duran-Frigola, M. *et al.* Extending the small-molecule similarity principle to all levels of biology with the chemical checker. *Nature Biotechnology* **38**, 1087–1096 (2020).

37. Bickel, S., Brückner, M. & Scheffer, T. Discriminative learning under covariate shift. *Journal of Machine Learning Research* **10** (2009).

38. Schölkopf, B. *et al.* On causal and anticausal learning. *ICML* 1255–1262 (2012).

39. Niven, T. & Kao, H.-Y. Probing neural network comprehension of natural language arguments. *Proc. 57th Annual Meeting of the Association of Computational Linguistics* 4658–4664 (2019).

40. Zech, J. R. *et al.* Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: a cross-sectional study. *PLoS medicine* **15**, e1002683 (2018).

41. Geirhos, R. *et al.* Shortcut learning in deep neural networks. *Nature Machine Intelligence* **2**, 665–673 (2020).

42. Agarwal, C., Queen, O., Lakkaraju, H. & Zitnik, M. Evaluating explainability for graph neural networks. *Scientific Data* **10** (2023).

43. Agarwal, C., Zitnik, M. & Lakkaraju, H. Probing GNN explainers: A rigorous theoretical and empirical analysis of gnn explanation methods. In *International Conference on Artificial Intelligence and Statistics*, 8969–8996 (2022).

44. Seetharaman, J. & Sarma, M. S. Chelation therapy in liver diseases of childhood: Current status and response. *World Journal of Hepatology* **13**, 1552 (2021).

45. Alsentzer, E. *et al.* Deep learning for diagnosing patients with rare genetic diseases. *medRxiv* 2022–12 (2022).

46. O'Connell, D. Neglected diseases. *Nature* **449**, 157–157 (2007).

47. Tambuyzer, E. *et al.* Therapies for rare diseases: therapeutic modalities, progress and challenges ahead. *Nature Reviews Drug Discovery* **19**, 93–111 (2020).

48. Zhang, A., Xing, L., Zou, J. & Wu, J. C. Shifting machine learning for healthcare from development to deployment and from models to data. *Nature Biomedical Engineering* 1–16 (2022).

49. Bastian, F. B. *et al.* The Bgee suite: integrated curated expression atlas and comparative transcriptomics in animals. *Nucleic Acids Research* **49**, D831–D847 (2021).

50. Davis, A. P. *et al.* Comparative Toxicogenomics Database (CTD): update 2021. *Nucleic Acids Research* **49**, D1138–D1143 (2021).

51. Piñero, J. *et al.* The DisGeNET knowledge platform for disease genomics: 2019 update. *Nucleic Acids Research* gkz1021 (2019).

52. Wishart, D. S. *et al.* DrugBank 5.0: a major update to the DrugBank database for 2018. *Nucleic Acids Research* **46**, D1074–D1082 (2018).

53. Maglott, D., Ostell, J., Pruitt, K. D. & Tatusova, T. Entrez Gene: gene-centered information at NCBI. *Nucleic Acids Research* **39**, D52–D57 (2011).

54. The Gene Ontology Consortium *et al.* The Gene Ontology resource: enriching a GOld mine. *Nucleic Acids Research* **49**, D325–D334 (2021).

55. Köhler, S. *et al.* The Human Phenotype Ontology in 2017. *Nucleic Acids Research* **45**, D865–D876 (2017).

56. Shefchek, K. A. *et al.* The Monarch Initiative in 2019: an integrative data and analytic platform connecting phenotypes to genotypes across species. *Nucleic Acids Research* **48**, D704–D715 (2020).

57. Matys, V. *et al.* TRANSFAC ® : transcriptional regulation, from patterns to profiles. *Nucleic Acids Research* **31**, 374–378 (2003).

58. Ceol, A. *et al.* MINT, the molecular interaction database: 2009 update. *Nucleic Acids Research* **38**, D532–D539 (2010).

59. Aranda, B. *et al.* The IntAct molecular interaction database in 2010. *Nucleic Acids Research* **38**, D525–D531 (2010).

60. Giurgiu, M. *et al.* Corum: the comprehensive resource of mammalian protein complexes—2019. *Nucleic acids research* **47**, D559–D563 (2019).

61. Oughtred, R. *et al.* The BioGRID database: A comprehensive biomedical resource of curated protein, genetic, and chemical interactions. *Protein Science* **30**, 187–200 (2021).

62. Szklarczyk, D. *et al.* The string database in 2021: customizable protein–protein networks, and functional characterization of user-uploaded gene/measurement sets. *Nucleic Acids Research* **49**, D605–D612 (2021).

63. Luck, K. *et al.* A reference map of the human binary protein interactome. *Nature* **580**, 402–408 (2020).

64. Jassal, B. *et al.* The reactome pathway knowledgebase. *Nucleic Acids Research* gkz1031 (2019).

65. Kuhn, M., Letunic, I., Jensen, L. J. & Bork, P. The SIDER database of drugs and side effects. *Nucleic Acids Research* **44**, D1075–D1079 (2016).

66. Mungall, C. J., Torniai, C., Gkoutos, G. V., Lewis, S. E. & Haendel, M. A. Uberon, an integrative multi-species anatomy ontology. *Genome Biology* **13**, R5 (2012).

67. Alsentzer, E. *et al.* Publicly Available Clinical BERT Embeddings 7.

68. Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O. & Dahl, G. E. Neural message passing for quantum chemistry. In *ICML*, 1263–1272 (PMLR, 2017).

69. Glorot, X. & Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*, 249–256 (2010).

70. Yang, B., Yih, W.-t., He, X., Gao, J. & Deng, L. Embedding entities and relations for learning and inference in knowledge bases. *ICLR* (2015).

71. Griggs, R. C. *et al.* Clinical research for rare disease: opportunities, challenges, and solutions. *Molecular Genetics and Metabolism* **96**, 20–26 (2009).

72. Boycott, K. M., Vanstone, M. R., Bulman, D. E. & MacKenzie, A. E. Rare-disease genetics in the era of next-generation sequencing: discovery to translation. *Nature Reviews Genetics* **14**, 681–691 (2013).

73. Thomas, S. & Caplan, A. The orphan drug act revisited. *Jama* **321**, 833–834 (2019).

74. Yıldırım, M. A., Goh, K.-I., Cusick, M. E., Barabási, A.-L. & Vidal, M. Drug—target network. *Nature Biotechnology* **25**, 1119–1126 (2007).

75. Agrawal, M., Zitnik, M. & Leskovec, J. Large-scale analysis of disease pathways in the human interactome. In *Proceedings of the Pacific Symposium on Biocomputing*, 111–122 (2018).

76. Kovács, I. A. *et al.* Network-based prediction of protein interactions. *Nature Communications* **10**, 1–8 (2019).

77. Raj, A., Kuceyeski, A. & Weiner, M. A network diffusion model of disease progression in dementia. *Neuron* **73**, 1204–1215 (2012).

78. Lin, Y., Liu, Z., Sun, M., Liu, Y. & Zhu, X. Learning entity and relation embeddings for knowledge graph completion. In *AAAI* (2015).

79. Wang, M. *et al.* Deep graph library: Towards efficient and scalable deep learning on graphs. (2019).

80. Paszke, A. *et al.* Pytorch: An imperative style, high-performance deep learning library. *NeurIPS* **32**, 8026–8037 (2019).

81. McKinney, W. *et al.* pandas: a foundational python library for data analysis and statistics. *Python for High Performance and Scientific Computing* **14**, 1–9 (2011).

82. Harris, C. R. *et al.* Array programming with numpy. *Nature* **585**, 357–362 (2020).

83. Pedregosa, F. *et al.* Scikit-learn: Machine learning in python. *Journal of Machine Learning Research* **12**, 2825–2830 (2011).

84. Waskom, M. L. Seaborn: statistical data visualization. *Journal of Open Source Software* **6**, 3021 (2021).

85. Hunter, J. D. Matplotlib: A 2d graphics environment. *Computing in science & engineering* **9**, 90–95 (2007).

86. McInnes, L., Healy, J., Saul, N. & Grossberger, L. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software* **3**, 861 (2018).

87. Inc, F. React.js. https://github.com/facebook/react.

88. Bostock, M., Ogievetsky, V. & Heer, J. D^3 data-driven documents. *IEEE Transactions on Visualization and Computer Graphics* **17**, 2301–2309 (2011).

89. Team, A. D. Ant design. https://github.com/ant-design/ant-design/.

90. Neo4j, I. Neo4j graph data platform. https://neo4j.com. Accessed: 2020-10-01.

91. Grinberg, M. *Flask web development: developing web applications with python* (" O'Reilly Media, Inc.", 2018).

92. Stang, P. E. *et al.* Advancing the science for active surveillance: rationale and design for the Observational Medical Outcomes Partnership. *Annals of Internal Medicine* **153**, 600–606 (2010).

93. Klann, J. G., Joss, M. A., Embree, K. & Murphy, S. N. Data model harmonization for the All Of Us Research Program: Transforming i2b2 data into the OMOP common data model. *PloS ONE* **14**, e0212463 (2019).

94. Virtanen, P. *et al.* SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods* **17**, 261–272 (2020).

95. Seabold, S. & Perktold, J. Statsmodels: Econometric and statistical modeling with python. In *Proceedings of the 9th Python in Science Conference*, vol. 57 (2010).