Title

Knowledge integration and decision support for accelerated discovery of antibiotic resistance genes

Journal

Nature Communications, 13(1)

ISSN

2041-1723

Authors

Youn, Jason
Rai, Navneet
Tagkopoulos, Ilias

Publication Date

2022

DOI

10.1038/s41467-022-29993-z

Peer reviewed

nature COMMUNICATIONS

ARTICLE

https://doi.org/10.1038/s41467-022-29993-z OPEN

Knowledge integration and decision support for accelerated discovery of antibiotic resistance genes

Jason Youn[1,2,3], Navneet Rai[1,2,3] & Ilias Tagkopoulos[1,2,3]✉

We present a machine learning framework to automate knowledge discovery through knowledge graph construction, inconsistency resolution, and iterative link prediction. By incorporating knowledge from 10 publicly available sources, we construct an *Escherichia coli* antibiotic resistance knowledge graph with 651,758 triples from 23 triple types after resolving 236 sets of inconsistencies. Iteratively applying link prediction to this graph and wet-lab validation of the generated hypotheses reveal 15 antibiotic resistant *E. coli* genes, with 6 of them never associated with antibiotic resistance for any microbe. Iterative link prediction leads to a performance improvement and more findings. The probability of positive findings highly correlates with experimentally validated findings ($R^2 = 0.94$). We also identify 5 homologs in *Salmonella enterica* that are all validated to confer resistance to antibiotics. This work demonstrates how evidence-driven decisions are a step toward automating knowledge discovery with high confidence and accelerated pace, thereby substituting traditional time-consuming and expensive methods.

[1] Department of Computer Science, University of California, Davis, CA 95616, USA. [2] Genome Center, University of California, Davis, CA 95616, USA. [3] USDA/NSF AI Institute for Next Generation Food Systems (AIFS), University of California, Davis, CA 95616, USA. ✉email: itagkopoulos@ucdavis.edu

For computational methods to be effective, the integration and ingestion of biological data at scale are paramount[1–3]. To this end, various initiatives[4–6] have transitioned from relational databases that store data using tables and are often limited by their scalability[7] to graph databases that efficiently process dense interrelated datasets[8] by utilizing the Resource Description Framework (RDF) triple of subject, predicate, and object[9]. This design helps to identify patterns among data, and to utilize the information content they carry to gain insights into the mechanisms of action, associations, and testable hypotheses[2,10]. In the biomedical domain, knowledge graphs[11] with thousands to millions of RDF triples are used to organize knowledge in life sciences[12,13], including health conditions such as cancer[14] and cardiovascular disease[15]. In the case of antibiotic resistance genes (ARGs), there exist both graph databases like CARD[16] and ARDB[17] that represent ontologies, as well as traditional databases like MEGARes[18], ARGO[19], and ARG-ANNOT[20] that store ARG sequencing data. Current challenges include unreported or unresolved conflicted information between two or more sources[21,22], lack of negative findings[23,24] that is necessary to train machine learning models, focus on only one relation type[25], inability to directly integrate results across sources due to incompatible meta-data[26], all of which limits their suitability as a training set for machine learning models. Similarly, extracting training data from published literature is challenging as it is often hidden in supplementary tables and figures[27,28], may be inaccessible or incompatible[29,30], which hinders any knowledge synthesis and analysis[31,32].

Automating the integration of heterogeneous biomedical data and their organization so they are machine learning-ready for downstream analysis and knowledge discovery is important for any life science field. One such area is the discovery of ARGs and relationships. Antibiotic resistance poses a major threat to the efficacy of the antibacterial drugs, which leads to increased mortality and costs[33]. Identification of ARGs has traditionally been performed through time-consuming and expensive culture-based methods[34] and more recently through bioinformatics analysis of whole-genome sequencing samples, including BLAST-based[20,35] and deep learning-based[36,37] methods. Outside of the domain of antibiotic resistance, there have been multiple attempts to discover biological knowledge from knowledge graphs[38–42] by formulating it as a knowledge graph completion (KGC)[43] problem, where the objective is to complete (discover) the missing links (new knowledge) in the graph. Graph feature models[44,45] and latent feature models[46,47] have traditionally been used for KGC, whereas models that utilize pre-trained language models (LM)[48,49] have recently achieved state-of-the-art results.

In this study, we present a methodology (Knowledge Integration and Decision Support, or KIDS) that constructs an inconsistency-free knowledge graph that supports multiple triple types and can be used to generate hypotheses over multiple iterations (Fig. 1). We apply the KIDS framework to the area of *Escherichia coli* antibiotic resistance, which leads to a knowledge graph consisting of 651,758 triples of 23 RDF triple types in total, among which 9 triple types are negative. To resolve inconsistencies, we computationally predicted, and experimentally validated 236 sets of inconsistencies with 94.07% accuracy. We then demonstrate how the automated process allows the discovery of previously unknown ARGs. KIDS achieved an average of 0.77 AUCPR and 0.86 AUROC in predicting the ARGs over two iterations of hypothesis generation, validation, and integration with existing knowledge, with the predicted ARG probability being highly correlated with validated findings ($R^2 = 0.94$). Furthermore, our analysis led to the discovery of six ARGs that we have validated experimentally, among which five homologs in *Salmonella enterica* also showed antibiotic resistance.

Results

The landscape of *E. coli* antibiotic resistance genes and processes. We applied the KIDS framework on the biological domain of *E. coli* and constructed a multi-relational knowledge graph[50] (see Methods) that consists of 651,758 triples (Fig. 2a, b and Supplementary Data 1). Raw data to construct the knowledge graph were curated from a total of ten sources (Section 1.1.1 of Supplementary Information) that include information about antibiotic resistance, effects of antibiotics on the expression patterns, gene-regulatory relations with transcription factors, and the impact of genes on the biology of an organism at the molecular, cellular, and organism levels[51], all regarding *E. coli* genes (Fig. 2c). The resulting knowledge graph provides a comprehensive view of the positive *E. coli* antibiotic resistance with 18-fold more genes and 3-fold more antibiotics than CARD[16] (Fig. 2d, Supplementary Table 1). Among the 23 triple types of the knowledge graph, 14 positive triple types account for the 31,216 (4.8%) associations as genes are less likely to confer resistance to an antibiotic (Fig. 2e). The knowledge graph contains antibiotic exposure times at six different time points ranging from 30 min to 7 days (Supplementary Table 2). From the total of 466,752 possible gene-antibiotic pairs, 358,674 pairs (76.9%) were connected via either a positive or negative 'confers resistance to antibiotic (CRA)' predicate, with the rest being candidates for either association (Supplementary Fig. 1).

Resolved inconsistencies help discover new knowledge. We identified 236 sets of inconsistencies in our intermediate knowledge graph (Supplementary Data 2, Fig. 3a) between the findings of two sources Tamae et al.[21] and Liu et al.[52] for positive and negative counterparts of the predicate 'CRA after 18 h' despite their identical experimental setup (Supplementary Table 6). We then applied the AverageLog[53] inconsistency resolution algorithm (see Methods) to select which one of the two conflicting facts is more likely to be true by iteratively updating the source trustworthiness and belief of triple (Fig. 3b). Results show that we were able to accurately resolve these inconsistencies (94.07% accuracy, 50.0% F1-score, 33.3% precision, 3.0% baseline precision) when compared to the ground truth wet-lab validation (Fig. 3b, Supplementary Table 3), which was performed by measuring and comparing the minimum inhibitory concentrations (MICs) of the single-gene knock-out strain and the wild-type strain on the LB agar plate (see Methods, Supplementary Data 8). We then trained the hypothesis generator before and after resolving inconsistencies, to test how inconsistency resolution affects knowledge discovery. This led to two previously unidentified antibiotic-resistant relationships (*surA*, CRA, Vancomycin) and (*asmA*, CRA, Vancomycin) with significantly increased probabilities after the inconsistency resolution (0.024–0.882 and 0.005–0.213, respectively) that we validated experimentally.

KIDS accelerates knowledge discovery. The hypothesis generator module performs link prediction[43] on the incomplete knowledge graph to identify the missing links (i.e., generate hypotheses). We focused on exploring the missing CRA links between all pairwise combinations of *E. coli* genes and antibiotics (108,078 hypotheses). To this end, we applied five different variations of the hypothesis generation methods (PRA[44,54], MLP, a stacked model that combines PRA and MLP using AdaBoost[55], TransE[46], and TransD[56]; see Fig. 4b and Methods) on a reduced knowledge graph without temporal information (see Methods) that has 494,819 triples and 12 predicate types (Supplementary Table 4, Supplementary Data 1). From those methods, PRA[44,54] finds observable predicate paths between subject (source) and object (target) nodes in the graph and treats them as human-interpretable features (Supplementary

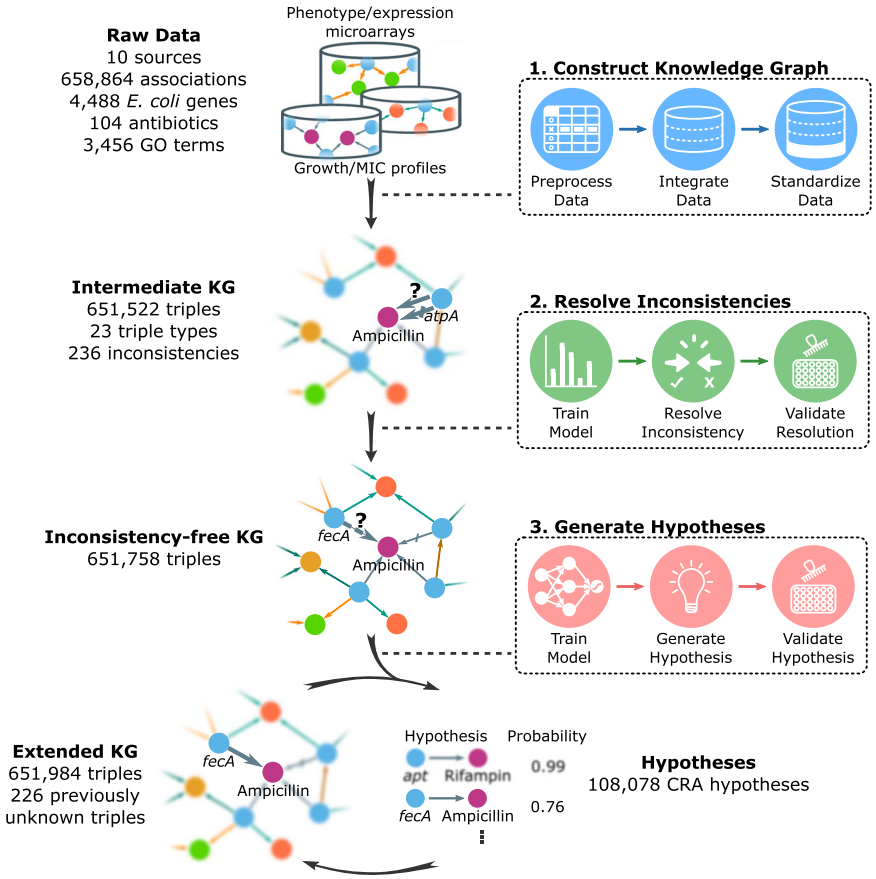


Fig. 1 Overview of the KIDS framework. First, an intermediate knowledge graph is created from ten sources by processing RDF triples that encode 23 types of associations. Second, inconsistencies are computationally resolved and experimentally validated to construct an inconsistency-free knowledge graph. Third, a hypothesis generator is trained on the knowledge graph and assigns probabilities for the missing links. Hypotheses with high probability are experimentally validated, and the results are integrated into the knowledge graph, which is used for the next iteration of hypothesis generation. GO refers to the Gene Ontology, and gray arrows denote the positive predicate 'confers resistance to antibiotic (CRA)'.

Table 9). In contrast, MLP[57] is a fully connected neural network that uses the triples represented by latent vector embeddings to predict whether any given edge is valid. We also tested translation-based graph embedding methods TransE[46] and TransD[56] (Section 1.3.5 of Supplementary Information), but we selected the stacked model as it had superior performance in testing. Evaluation of these methods, that have been optimized for F1-score, using 5-fold cross-validation shows that the stacked model had the best performance in terms of AUCPR with a 154.4% increase when compared to PRA (0.28 vs. 0.11, respectively, p value $= 2.1 \times 10^{-6}$) and a 27.7% increase when compared to MLP (0.28 vs. 0.22, respectively, p value $= 3.0 \times 10^{-3}$) (Fig. 4c, Supplementary Table 5), while the baseline was 0.02.

We used the stacked model to generate 226 CRA hypotheses of varying probability that we subsequently tested experimentally (Supplementary Data 3). Of those hypotheses, 64 (28.3%) were validated as positives (Fig. 5a). After adding those results to the knowledge graph, we ran a second iteration of KIDS, which produced another 90 hypotheses, from which 29 (28.8%) were positively validated (Fig. 5a). From these two iterations, we computationally predicted and experimentally validated, similar to the wet-lab validation performed for the inconsistency resolver (Section 1.3.12 of Supplementary Information), a total of 93 CRA hypotheses for 83 *E. coli* genes that confer resistance to one or more of 15 antibiotics (Fig. 5e, Supplementary Data 4). The KIDS-generated hypotheses are reliable as the calibrated output for each hypothesis is a highly correlated confidence score ($R^2 = 0.94$) with the validated outcome (Fig. 5a). For instance, hypotheses with probability >0.8 have a high true positive rate with 29 out of 37 tested hypotheses (78.4%) to yield an ARG, whereas hypotheses with probability ≤0.2 have a true positive rate with only 14 out of 163 tested hypotheses (8.59%) to yield an ARG. Interestingly, KIDS produced improved hypotheses in the second iteration with the addition of the newly discovered results (Fig. 5b–d). The KIDS-generated hypotheses are positively correlated with high consistency when compared to the random baseline (Kendall's tau[58] $= 0.96$ vs. 0.00, respectively, p value $< 2.2 \times 10^{-308}$; RBO[59] $= 0.56$ vs. 0.00, respectively, p value $< 2.2 \times 10^{-308}$; Section 1.3.11 of Supplementary Information).

AI-driven discovery of six antibiotic resistance genes. Extensive literature search on the 83 *E. coli* genes that are implicated in the CRA hypotheses identified 15 genes that are previously unknown ARG for *E. coli*, with 6 of them (1 from the first iteration and 5 from the second iteration) not appearing as an ARG for any bacteria. Those 6 are the following: *ftsP*, *hdfR*, *lrp*, *proV*, *qorB*, and *rbsK* (Fig. 6), which have never been reported to be involved in antibiotic resistance (Supplementary Data 4). Further investigation on the biological processes reveals they are part of a diverse repertoire of functions related to amino acid metabolism, nutrient transport, and regulation. More specifically, FtsP is a cell division protein that is required for bacterial growth during stress conditions. FtsP stabilizes or protects the divisional assembly during stress condition[60]. HdfR, which is an H-NS-dependent *flhDC* regulator, represses the expression of the flagellar master operon

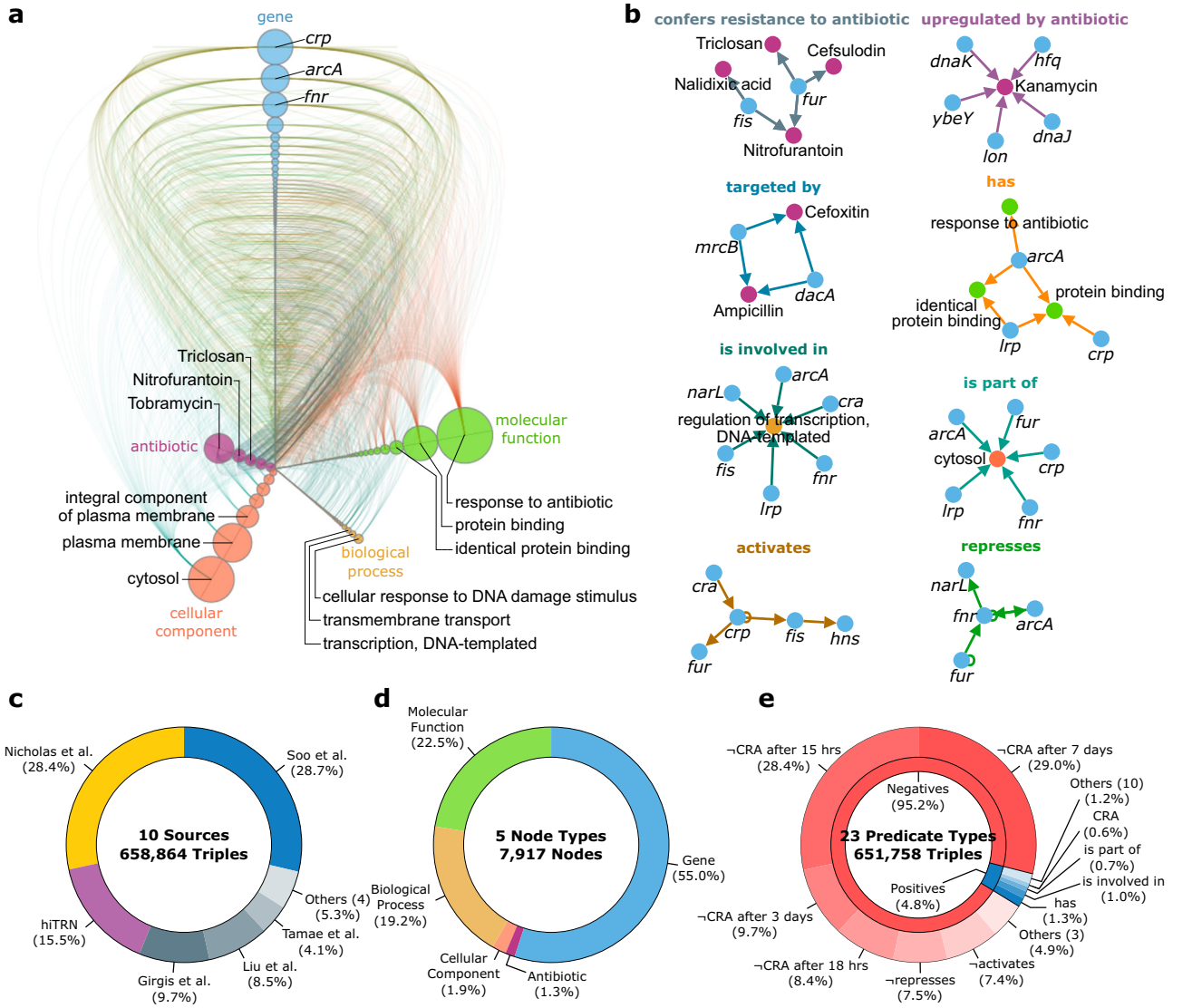


Fig. 2 The inconsistency-free *E. coli* knowledge graph. a Hive plot visualization of the knowledge graph's major components, with each axis corresponding to one of five different node types: gene, antibiotic, cellular component, biological process, and molecular function. The size of a node is its in and out degree. Only the 5% highest degree nodes from each node type and their positive connections are shown. **b** The top highest degree nodes for each of the eight positive predicates in the knowledge graph. **c**–**e** Breakdown of the knowledge graph representation in terms of data sources, node, and predicate types. CRA denotes the predicate 'confers resistance to antibiotic', whereas ¬CRA denotes 'confers no resistance to antibiotic'.

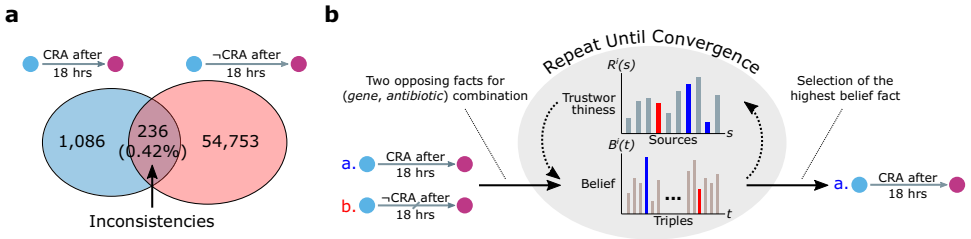
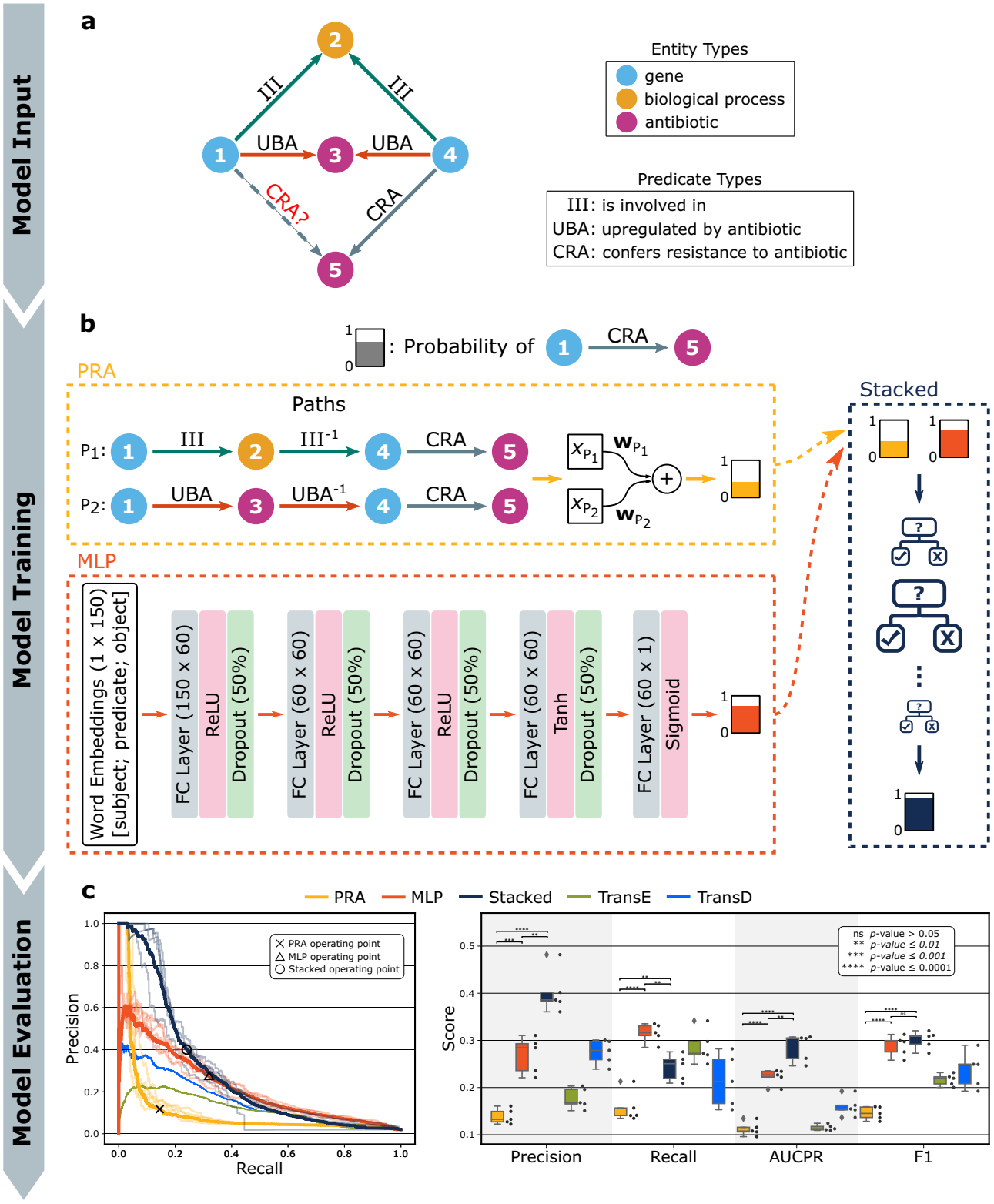


Fig. 3 Inconsistency resolution. a Venn diagram showing the inconsistencies detected in the intermediate knowledge graph, where the inconsistency is defined as two or more sources supporting a conflicting fact. **b** The inconsistency resolution algorithm is iteratively trained using the intermediate knowledge graph. Once the training converges, it is used to select the triple with the higher belief among the inconsistencies. See Methods for more details. The blue and purple nodes represent gene and antibiotic, respectively. CRA denotes the predicate 'confers resistance to antibiotic', whereas ¬CRA denotes 'confers no resistance to antibiotic'.

flhDC[61] and induces the expression of the *gltBD* operon, which is involved in acid resistance[62,63]. Lrp encodes a leucine-responsive regulatory protein, which regulates at least 10% of the genes in *E. coli*, including regulation of major porins OmpC and OmpF that determine the permeability of the cell membrane[64,65]. ProV is predicted to be a component of an osmoresponsive ABC



transport system and involved in osmosensing[66]. QorB is a NAD(P)H:quinone oxidoreductase, which catalyzes the reduction of quinone. *E. coli* strain overexpressing *qorB* shows defects in growth and a significant decrease in several enzymes involved in carbon metabolism[67]. Interestingly, oxidoreductases have been reported to involve antibiotic resistance[68]. RbsK is a sugar kinase that, in addition to phosphorylation of ribose, facilitates stress-induced mutagenesis in *E. coli*[69]. Mutations in sugar kinase genes such as *waaP* of *S. enterica* lead to increase susceptibility to antibiotic polymyxin[70,71].

We did not identify any statistically significant homologs (*E* value < 0.05) of these six genes among the 4577 ARGs from CARD[72], while the best hit was for *OXA-541* of *Pseudomonas putida* for *lrp* (91.7% sequence similarity, *E* value = 0.12) (Section 1.3.13 of Supplementary Information, Supplementary Data 10). The prevalence of these six genes across the human digestive

Fig. 4 Hypothesis generator architecture, training, and evaluation. a Illustration of the training and evaluation of the hypothesis generator (HG). The task of the HG is to associate a probability to a putative link for the 'confers resistance to antibiotic' or CRA between two nodes (nodes 1 and 5 here). **b** Three HG architectures, PRA, MLP, and Stacked, an ensemble method of a majority voting schema of the other two, were constructed and evaluated. Additional translation-based models like TransE and TransD were also tested although not illustrated here (see Methods). **c** Precision-recall, AUCPR, and F1-score for the five methods ($n = 5$, 5-fold cross-validation). Black circles denote raw data points. The box represents the interquartile range, the middle line represents the median, the whisker line extends from minimum to maximum values, and the diamond represents outliers. For PRA vs. MLP, all scores were statistically significant (precision p value $= 1.1 \times 10^{-4}$, recall p value $= 1.4 \times 10^{-5}$; AUCPR p value $= 2.9 \times 10^{-6}$; F1-score p value $= 1.6 \times 10^{-6}$). For PRA vs. Stacked, all scores were also statistically significant (precision p value $= 2.2 \times 10^{-6}$, recall p value $= 2.0 \times 10^{-3}$; AUCPR p value $= 2.1 \times 10^{-6}$; F1-score p value $= 3.9 \times 10^{-7}$). Finally, for MLP vs. Stacked, all scores were significant (precision p value $= 1.1 \times 10^{-3}$, recall p value $= 1.5 \times 10^{-3}$; AUCPR p value $= 3.0 \times 10^{-3}$) except for F1-score (p value $= 0.37$). Note that all methods have been optimized for the F1-score, and the p values were calculated using the two-sided t-test.

microbiome ranges from 0.67 to 8.79% (Section 1.3.14 of Supplementary Information, Supplementary Table 12). Finally, to investigate the antibiotic resistivity of genes homologous to the six previously unknown ARGs in other bacterial genera, we identified five homologs *ftsP*, *lrp*, *proV*, *rbsK*, and *yifA* (*hdfR* in *E. coli*) in *S. enterica* with >78% similarity in nucleotide sequences, while the homolog of *qorB* was not identified (Section 1.3.15 and 1.3.16 of Supplementary Information). Wet-lab validation revealed that knocking out these five genes in *S. enterica* also increased susceptibility to antibiotics (Supplementary Data 9).

Discussion

In this work, we presented the KIDS framework as an automated method for knowledge organization and discovery. We demonstrated the power of the KIDS platform in the context of *E. coli* antibiotic resistance, a research area with a need for such a method, as the emergence of antibiotic resistance renders existing antibacterial drugs less efficient and thus necessitates a constant race to discover new ways to fight microbial infections[73]. Out of the 6 ARGs discovered in this work, we found that 5 homologs in *S. enterica* also conferred resistance to an antibiotic, indicating that the knowledge gained in this study can be easily translated to closely related bacteria. Current computational tools identify potential ARGs by genomic and metagenomic sequence analysis, which has limited performance when the reference database does not include similar ARG sequences. Similarly, just looking at homology is not sufficient for discovering ARGs. Among the 129 genes from the lowest probability range [0.0, 0.2] that we have validated to have no antibiotic resistance, we found 9 homologous ARGs in CARD that have significant E value (<0.05) with >68.6% sequence similarity (Supplementary Data 10). KIDS removes the dependency to reference sequences as its power stems from guilt-by-association and pattern discovery within the knowledge graph. Although manual literature curation and experimental validation were tedious and time-consuming, we found that the KIDS framework generates actionable hypotheses that lead to automated knowledge discovery with high confidence and efficiency.

On the summary statistics, the improvement from resolved inconsistencies was small, most likely because only 7 out of the 236 inconsistencies (3.0%) we experimentally resolved and further validated in the wet-lab were positive triples (Supplementary Data 2), and therefore reinstating them back to the knowledge graph where 1606 positive CRA triples exist (Supplementary Data 1) did not affect the knowledge graph much (1606–1613, a 0.44% increase). However, we found two previously unknown antibiotic-resistant relationships (*surA*, CRA, Vancomycin) and (*asmA*, CRA, Vancomycin) only after reinstating the resolved inconsistencies into the knowledge graph, something that demonstrates the importance of inconsistency resolution and coherence in our knowledge. For the lack of negative findings, our knowledge graph is the first to include both the positive findings (14 triple types, 31,216 triples) and the negative findings (9 triple types, 620,542 triples) to the best of our knowledge. Although the majority of the hypothesis generation models we tested did not use these negatives and instead generated them either through closed-world assumption or corruption through random sampling, our best model (stacked) did utilize these negatives. We believe there is still a potential to take advantage of these negative findings in other machine learning models. To address the focus on only one relation type, our knowledge graph contains 23 relation types (Supplementary Table 2) as opposed to a single relation type from other sources (Section 1.1.1 of Supplementary Information). Finally, regarding the inability to directly integrate results across sources due to incompatible meta-data, this is still a problem for this and any other framework, as it is related to data incompatibility during their generation and reporting, something that we as a community need to collaboratively work on by adhering to standards like FAIR[74].

Although translation-based graph embedding models have shown state-of-the-art performance in some benchmark datasets[75,76], they performed worse than models like MLP and Stacked for our *E. coli* knowledge graph (Supplementary Table 5). This may be due to the known limitations of these methods where they are unable to handle knowledge graphs with complex and diverse entities and relations (e.g., one-to-many, many-to-one, many-to-many)[77] or do not utilize semantic information[78]. For example, in our knowledge graph, many genes are known to confer resistance to a specific antibiotic (many-to-one). Therefore, these genes will be close to each other in the embedding space, making it difficult to differentiate them from each other. This leaves room for performance improvement of the hypothesis generation methods by utilizing the current state-of-the-art link prediction methods[48,49] which take advantage of pre-trained LM like BERT[28] and RoBERTa[79] and approach the problem as a natural language processing task. Unlike graph embedding approaches[46,80–83], LM-based methods generalize well to unseen nodes or edges in graph[49]. However, the application of such methods on the biological domain remains a challenge as LM models are usually not trained on biological data, except BioBERT[84], in which case further fine-tuning of the LM model to the specific domain (*E. coli* ARG here) is desired. For the scope of this work, we used a stacked (MLP and PRA) hypothesis generation method, inspired by the Knowledge Vault[57] project.

There are a few areas of improvement. First, knowledge inference rules (see Methods) were generated upon visual inspection of the 23 triple types of the knowledge graph. There are automatic knowledge graph construction methods[85,86] that can potentially do this automatically, but we leave it for future work as their precision is not at the human level nor has been tested in the biomedical domain. Second, although our knowledge graph contains temporal information, we discarded them when training the hypothesis generator. Allowing temporal features[87–89], we could expand our research to generate time-specific hypotheses, using techniques such as sequence-to-sequence learning methods[90,91]. Third, the major bottleneck of

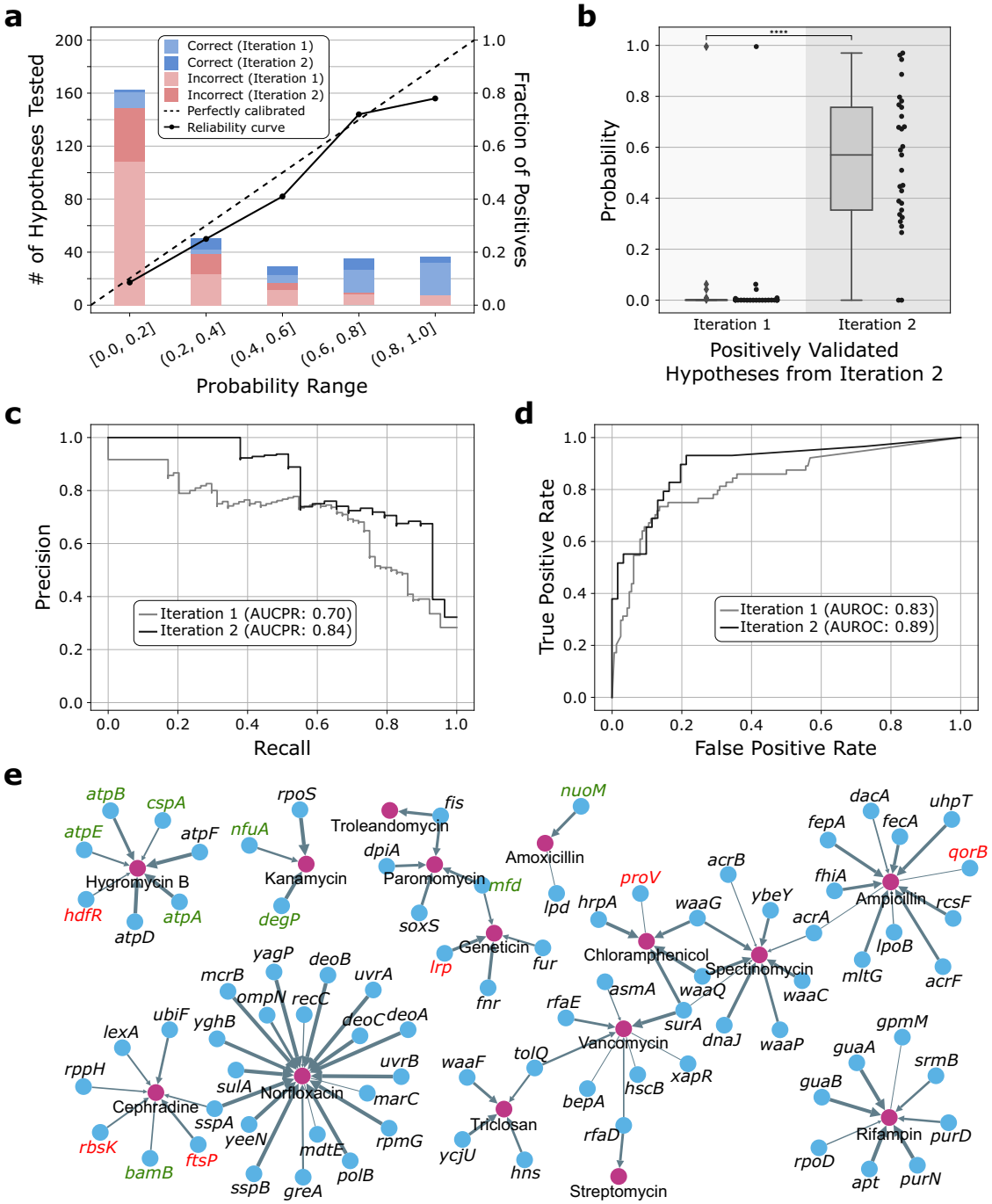


Fig. 5 Accelerated missing link discovery through iterative learning. a A high correlation between the probability assignment by the hypothesis generator and forward experimental validation (226 and 90 validated hypotheses from the first and second iteration, respectively; $R^2 = 0.94$). **b** The probability distribution of the positively validated hypotheses from the second iteration (i.e., dark blue bar in **a**) compared to the probability of the same hypotheses from the first iteration ($n = 29$ positively validated second iteration hypotheses). Updating the knowledge graph with the validated hypotheses in the first iteration (i.e., light blue and red bars in **a**) and re-training of the hypothesis generator led to the 14-fold probability increase (0.55 vs. 0.04, respectively, p value $= 1.1 \times 10^{-11}$), which in turn enabled the discovery that would not have been possible with only one iteration of hypothesis generation. The box represents the interquartile range, the middle line represents the median, the whisker line extends from minimum to maximum values, and the diamond represents outliers. The p value was calculated using the two-sided t-test. **c**, **d** The precision-recall (PR) and receiver operating characteristic (ROC) curves of the generated hypotheses compared against our wet-lab validation results. The AUCPR and AUROC of the second iteration hypotheses increased by 19.4% and 7.3%, respectively, when compared to the first iteration hypotheses. **e** We predicted and validated 64 CRA hypotheses from iteration 1 and 29 CRA hypotheses from iteration 2 for a total of 83 *E. coli* genes (blue node) that confer resistance (gray arrow) to one or more of 15 antibiotics (purple node). Genes with green and red labels indicate previously unknown genes that are not associated with antibiotic resistance in *E. coli* (9 genes) or any microbe (6 genes), respectively. The edge thickness is proportional to the KIDS predicted probability.

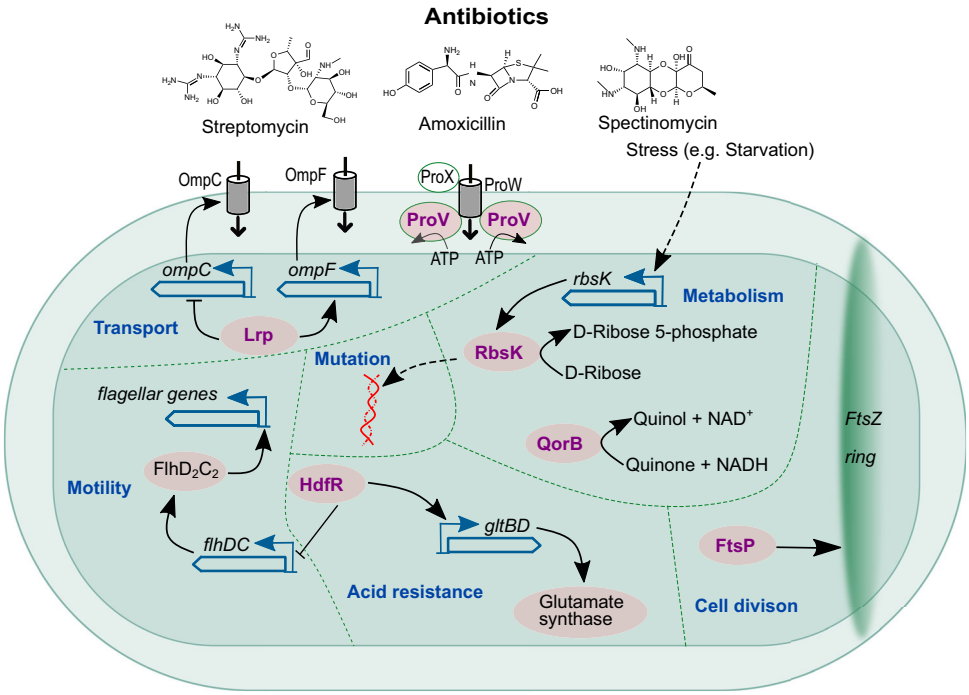


Fig. 6 Mode of action of 6 previously unknown genes discovered to be involved in antibiotic resistance. The proteins of these genes are shown in purple. Solid arrows indicate upregulation while blocking bars indicate downregulation. Dotted arrows indicate indirect regulation.

the KIDS framework is its dependency on expert-guided manual curation of data in RDF triple format. An automated data curator would be a boon to adding information from existing literaure[86,92]. In addition, we expect better initialization schemes, such as those based on pre-trained word embeddings trained using scientific literature instead of random initialization, to further improve performance[93–95]. Concomitantly, we would like to apply KIDS to other bacteria and replicate the success that we observe in *E. coli*. Finally, evaluating the impact of data size on learning performance (Supplementary Fig. 1) can help to determine how well this method can generalize to other microbes with limited training data.

Taken together with other advances in optimal experimental design[96,97], interpretable machine learning[98,99], and automated research and development approaches[100], the proposed framework paves the way for a systematic, optimized, and reproducible way to elucidate complex biological systems in shorter timescales, with less manual labor, and unprecedented fidelity.

Methods

Knowledge graph constructor. The knowledge graph construction process is shown in Supplementary Fig. 2 with detailed examples.

Data integration. We merge distinctive sets of knowledge from ten different sources (Section 1.1.1 of Supplementary Information) in a unified format using binary relationships known as an RDF triple of the form (subject, predicate, object), where subject and object are the nodes (biological entities) in the graph, and the predicate is the edge (relation) between them.

Synonym resolution. For entity types gene and antibiotic in the integrated knowledge graph, a name mapping table is applied to resolve the synonyms as multiple representations may exist for a single entity. For gene name mapping, Accession IDs to external databases and synonym lists of all *E. coli* genes downloaded from EcoCyc[101] are mapped to the original gene symbol. For all antibiotics, we map all synonyms listed in ChemIDplus[102] to its MeSH name (defined as MeSH heading in ChemIDplus). This name mapping table is in Supplementary Data 5.

Knowledge inference. As a data augmentation step, we added 15 sets of rules that we manually created to bridge existing gaps in the knowledge representation. As an example, a new triple (*sucD*, has, response to antibiotic) can be inferred from an existing triple (*sucD*, CRA, Cephradine). The full list of rules created is listed in Supplementary Data 6.

Inconsistency resolver. The inconsistency resolution process is shown in Supplementary Fig. 2 with detailed examples.

Inconsistency detection. To detect inconsistencies in the knowledge graph, we manually defined nine sets of rules (Supplementary Data 7) upon close inspection of the knowledge graph. In this work, we treat a set of triples that share the same subject and object entities connected by conflicting predicates as an inconsistency. For example, triples (*atpA*, CRA after 18 h, Ampicillin) and (*atpA*, confers no resistance to antibiotic after 18 h, Ampicillin) are considered one set of inconsistency.

Inconsistency resolution. Let t be a triple and s be a source, then $t \in T$ and $s \in S$, where T and S is the group of all triples and all sources, respectively. If we let T_s be all the triples of source s, then $T = \bigcup_{s \in S} T_s$. Each triple $t \in T$ belongs to a mutual exclusion set $M_t \subseteq T$, a set of triples that are mutually exclusive with one another. In an inconsistency-free setting, a triple t belongs to one unique M_t. In other words, $|M_t| = 1$ means there exist no conflicts in M_t. Assuming there exists one true triple $\bar{t}$ in each mutual exclusion set M, the goal of the inconsistency correction methods is to predict $\bar{t}$ for all M with $1<|M|$. Prediction of $\bar{t}$ is done by measuring belief of triple t, $B(t)$ (i.e., the level of confidence that triple t is true), among all t in M and by assigning t with the highest belief $\text{argmax}_{t \in M} B(t)$. Although the specific way to measure $B(t)$ varies across methods, it is commonly estimated based on the source trustworthiness $R(S_t)$ (i.e., level of trust assigned to the source), where $S_t = \{s : s \in S, t \in T_s\}$ is the set of sources with t. We compute the trustworthiness $R(s)$ and belief $B(t)$ iteratively until convergence. We used the AverageLog[53] among others (Section 1.2.1 of Supplementary Information), and the equations to update $R^i(s)$ and $B^i(t)$ for each iteration i are as follows [Eq. 1–2]:

$$R^i(s) = \log|T_s| \frac{\sum_{t \in T_s} B^{i-1}(t)}{|T_s|}, \quad (1)$$

$$B^i(t) = \sum_{s \in S_t} R^i(s), \quad (2)$$

where $R^i(s)$ and $B^i(t)$ are normalized to prevent a numerical explosion by dividing with $\max_{s \in S} R^i(s)$ and $\max_{t \in T} B^i(t)$, respectively. $B^0(t)$ is set to 0.5 for all $t \in T$. Performance evaluation of alternative inconsistency resolution methods can be found in Supplementary Figs. 6–11 and Supplementary Tables 7, 8.

Hypothesis generator

Preprocessing. There was not enough training data to train the hypothesis generator if we were to treat each predicate of varying temporal information distinctly. Therefore, we ultimately decided to modify the knowledge graph by removing the temporal information from the 14 predicates (e.g., 'CRA after 15 h' to CRA). This process reduced the size of the knowledge graph by 24.1% from 651,758 triples to 494,819 triples (Supplementary Data 1) and the number of predicates from 23 to 12 (Supplementary Table 4).

Path ranking algorithm (PRA)[44,54]. The set of paired entities from the training set, linked by the CRA predicate, is first used to identify the paths used to train the model. This is done by initiating a random walk at a bounded step size starting at the subject entity. If the random walk ends up at the object entity, this path is considered successful. To reduce the size of the feature space, a path will only be considered if it links at least one object entity. Additionally, the object entity found by a path must be supported by at least a fraction, α, of the training samples. Finally, L1-regularization is used during logistic regression to reduce the feature space even more. Each path retained for the model is treated as a path feature. The value of each feature is the prior probability of reaching the object entity from the subject entity for the given sample. These path probabilities are computed recursively by assuming that every step of the path, an outgoing link to an entity, is chosen uniformly at random. After training a regularized logistic regression model to identify the parameters to these features, the final score to predict the existence of an edge in the graph is as follows [Eq. 3]:

$$score(s,o) = \sum_{P \in P_l} h_{s,P}(o) * \omega_P, \tag{3}$$

where s and o are the subject and object entities, P is one of the paths chosen by the model, P_l, $h_{s,P}(o)$ is the path probability, and ω_P is the weights determined using logistic regression. We set L1-regularization to 0.008, L2-regularization to 0.0001, and the fraction, α, to 0.01 based on a hyperparameter search performed on 5-fold cross-validation. More details on computing these probabilities can be found in their original work.

Multilayer Perceptron (MLP). The MLP, a fully connected feed-forward artificial neural network, outputs a probability of whether a given triple is true. Each entity and predicate of the knowledge graph is converted to a dense numerical vector of length 50, which is created by taking the average of the constituent word embeddings[103]. These word embeddings are randomly initialized and treated as learnable parameters for the model. A dense numerical vector of length 150, which is created by concatenating the subject, predicate, and object embeddings, is fed as an input to the network. The network contains four hidden layers, each with 60 nodes. We used ReLU[104] activation functions until the third hidden layer, followed by a Tanh activation function for the last hidden layer. Finally, the output layer uses the sigmoid activation function to produce a score between 0 and 1. We used dropout[105] after all but the last hidden layer to reduce overfitting. The model was trained to leverage the margin-based ranking loss[43] [Eq. 4]:

$$l(\omega) = \sum_{i=1}^{N} \sum_{c=1}^{C} \max\left(0, \gamma - g\left(T^i\right) + g\left(T_c^i\right)\right) + \lambda ||\omega||_2^2, \tag{4}$$

where N is the number of training edges, C is the corruption size, function $g()$ represents a complete forward pass of the network or scoring function on a given edge T, ω is the weights of the model, λ is the L2-regularization parameter, and γ is the margin that the correct edge must score higher than the corrupted edge. Based on a hyperparameter search performed on 5-fold cross-validation, we used Adam[106] optimization with a learning rate of 0.001, λ was set to 0.001, the dropout rate was set to 0.5, C was set to 100, and the margin used for training was set to 0.20.

Stacked. We stacked the two models PRA and MLP using AdaBoosted[107] decision stumps, in line with[57]. The training inputs to the model were three features: the scores produced by the PRA and the MLP and one binary value for the PRA to indicate whether the PRA was able to predict that certain sample. Note that the PRA cannot predict if no paths were found. We performed random search hyperparameter optimization over the validation set for each fold and found optimal parameters of 680 estimators and a learning rate of 1.65. Since our dataset is unbalanced, we also used SMOTE[108] sampling to synthetically create positive samples for a balanced set of positive and negative samples.

Wet-lab validation. To validate whether a gene confers resistance or not, wild-type Keio strain BW25113 and its derivative single-gene knockout (KO) strains were used[109]. MIC values of the following antibiotics were measured: Amoxicillin (Sigma, Cat# A8523), Ampicillin (Roche Diagnostics, Cat# 10835269001), Apramycin (Alfa Aesar, Cat# AAJ6661603), Cephradine (Alfa Aesar, Cat# AAJ664960), Chloramphenicol (Calbiochem, Cat# 220551), Geneticin (Teknova, Cat# 50841719), Hygromycin B (Calbiochem, Cat# 400050100MG), Kanamycin (Acros Organics, Cat# AC611290050), Levofloxacin (Chem-Impex, Cat# 50508743), Norfloxacin (Sigma, Cat# SIAL-N9890), Novobiocin (Calbiochem, Cat# 491207), Oxycarboxine (Sigma, Cat# 36185), Paromomycin (Chem-Impex, Cat# 501602750), Rifampin (Alfa Aesar, Cat# AAJ6083603), Sisomicin (TCI, Cat# I1049250MG), Spectinomycin (RPI, Cat# 50213656), Streptomycin (Across Organics, Cat# AC455340050), Sulfanilamide (Alfa Aesar, Cat# AAA1300122), Triclosan (Cayman Chemical Company, Cat# 501599771), Troleandomycin (Enzo Life Sciences, Cat# BML-EI249-0010), and Vancomycin (VWR Life Science, Cat# 97062-554). Since KO strains had a kanamycin resistance gene, the kanamycin resistance gene was removed from the required KO strains[110] to measure the resistance in kanamycin. Antibiotics and strains were preserved at −80 °C until used.

1 µL of required preserved strain was inoculated in 200 µL LB broth and grown overnight in an incubator shaker (BioTek Synergy HTX) at 37 °C. ~3 µL of grown culture was transferred, using a replicator, to LB agar plates containing different amounts of antibiotics, and plates were incubated overnight (~18 h) at 37 °C in an incubator. The next day, the absence and presence of colonies were monitored. The minimum concentration of antibiotic, at which no colonies were observed, was defined as MIC (Supplementary Data 8). In the case of metronidazole, colonies were observed at all concentrations. Metronidazole is a pro-drug and inactive, but in anaerobic conditions, this is converted to an active form by the bacteria[111,112]. The active form is toxic which leads to the killing of bacteria. As our experimental conditions were aerobic, metronidazole was converted to an active form, and we observed colonies at all concentrations. Subsequently, we removed metronidazole from our study.

Reporting summary. Further information on research design is available in the Nature Research Reporting Summary linked to this article.

Data availability

The *E. coli* antibiotic resistance knowledge graph is available at https://github.com/IBPA/KIDS.

Code availability

All code and instructions on how to reproduce the results can be found at https://github.com/IBPA/KIDS.

Received: 14 April 2021; Accepted: 4 March 2022;
Published online: 29 April 2022

References

1. Barone, L., Williams, J. & Micklos, D. Unmet needs for analyzing biological big data: a survey of 704 NSF principal investigators. *PLoS Comput. Biol.* **13**, e1005755 (2017).
2. Li, Y. & Chen, L. Big biological data: challenges and opportunities. *Genomics. Proteom. Bioinforma.* **12**, 187 (2014).
3. Kim, M. & Tagkopoulos, I. Data integration and predictive modeling methods for multi-omics datasets. *Mol. Omi.* **14**, 8–25 (2018).
4. Kumar Kaliyar, R. (2015) Graph databases: a survey. In Proc. International Conference on Computing, Communication and Automation, 785–790 (IEEE, Greater Noida, India, 2015).
5. da Silva, Waldeyr, M. C., Polyane Wercelens, Maria Emilia, M. T. Walter, Maristela, Holanda & Marcelo, Brigido. Graph databases in molecular biology. In Proc. Brazilian Symposium on Bioinformatics, 50–57 (2018).
6. Fabregat, A. et al. Reactome graph database: efficient access to complex pathway data. *PLoS Comput. Biol.* **14**, e1005968 (2018).
7. Hammes, D., Medero, H. & Mitchell, H. Comparison of NoSQL and SQL databases in the cloud. In Proc. Southern Association for Information Systems (SAIS), 21–22 (Macon, GA, 2014).
8. Rodriguez, M. A. & Neubauer, P. Constructions from dots and lines. *Bull. Am. Soc. Inf. Sci. Technol.* **36**, 35–41 (2010).
9. Cyganiak, R. et al. RDF 1.1 concepts and abstract syntax, W3C recommendation. *World Wide Web Consortium Cambridge, MA, USA* **25**, 1–22 (2015).
10. Silvescu, A., Caragea, D. & Atramentov, A. Graph Databases. Artificial Intelligence Research Laboratory Department of Computer Science, Iowa State University. (Citeseer, 2012) [online] http://people.cs.ksu.edu/~dcaragea/papers/report.pdf.
11. Ehrlinger, L. & Wöß, W. *Towards a Definition of Knowledge Graphs*. *researchgate.net* https://www.researchgate.net/publication/323316736 (2016).
12. Ernst, P., Siu, A. & Weikum, G. Knowlife: a versatile approach for constructing a large knowledge graph for biomedical sciences. *BMC Bioinforma.* **16**, 157 (2015).
13. Dumontier, M. et al. Bio2RDF release 3: a larger connected network of linked data for the life sciences. In Proc. International Semantic Web Conference (Posters & Demos), volume 1272 of CEUR Workshop Proceedings, pp. 401–404. CEUR-WS.org (Association for Computing Machinery, 2014).
14. Hasan, S. M. S. et al. Knowledge graph-enabled cancer data analytics. *IEEE J. Biomed. Heal. Inform.* **24**, 1952–1967 (2020).

15. Sheng, M. et al. CLMed: a cross-lingual knowledge graph framework for cardiovascular diseases. In: Ni, W., Wang, X., Song, W., Li, Y. (eds) Web Information Systems and Applications. WISA 2019. Lecture Notes in Computer Science, vol 11817. (Springer, Cham, 2019). https://doi.org/10.1007/978-3-030-30952-7_51.
16. Jia, B. et al. CARD 2017: expansion and model-centric curation of the comprehensive antibiotic resistance database. Nucleic Acids Res. 45. Oxford University Press. https://doi.org/10.1093/nar/gkw1004 (2016).
17. Liu, B. & Pop, M. ARDB—antibiotic resistance genes database. *Nucleic Acids Res.* **37**, D443–D447 (2009).
18. Lakin, S. M. et al. MEGARes: an antimicrobial resistance database for high throughput sequencing. *Nucleic Acids Res.* **45**, D574–D580 (2016).
19. Scaria, J., Chandramouli, U. & Verma, S. K. Antibiotic Resistance Genes Online (ARGO): a database on vancomycin and $β$lactam resistance genes. *Bioinformation* **1**, 5 (2005).
20. Gupta, S. K. et al. ARG-ANNOT, a new bioinformatic tool to discover antibiotic resistance genes in bacterial genomes. *Antimicrob. Agents Chemother.* **58**, 212–220 (2014).
21. Tamae, C. et al. Determination of antibiotic hypersensitivity among 4000 single-gene-knockout mutants of Escherichia coli. *J. Bacteriol.* **190**, 5981–5988 (2008).
22. Palmieri, V. et al. The graphene oxide contradictory effects against human pathogens. *Nanotechnology* **28**, 152001 (2017).
23. Nichols, R. J. et al. Phenotypic landscape of a bacterial cell. *Cell* **144**, 143–156 (2011).
24. Zhou, L., Lei, X.-H., Bochner, B. R. & Wanner, B. L. Phenotype microarray analysis of Escherichia coli K-12 mutants with deletions of all two-component systems. *J. Bacteriol.* **185**, 4956–4972 (2003).
25. Shaw, K. J. et al. Comparison of the changes in global gene expression of Escherichia coli induced by four bactericidal agents. *J. Mol. Microbiol. Biotechnol.* **5**, 105–122 (2003).
26. Louie, B., Mork, P., Martin-Sanchez, F., Halevy, A. & Tarczy-Hornoch, P. Data integration and genomic medicine. *J. Biomed. Inform.* **40**, 5–16 (2007).
27. Tshitoyan, V. et al. Unsupervised word embeddings capture latent knowledge from materials science literature. *Nature* **571**, 95–98 (2019).
28. Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv Prepr. arXiv1810.04805* (2018).
29. Begley, C. G. & Ioannidis, J. P. A. Reproducibility in science: improving the standard for basic and preclinical research. *Circ. Res.* **116**, 116–126 (2015).
30. McNutt, M. Journals unite for reproducibility. *Sci. (80-.)* **346**, 679 (2014).
31. Anderson, N. R. et al. Issues in biomedical research data management and analysis: needs and barriers. *J. Am. Med. Inform. Assoc.* **14**, 478–488 (2007).
32. Skjærven, L., Yao, X.-Q., Scarabelli, G. & Grant, B. J. Integrating protein structural dynamics and evolutionary analysis with Bio3D. *BMC Bioinforma.* **15**, 399 (2014).
33. Organization, W. H. *Antimicrobial Resistance: Global Report on Surveillance.* (WHO Press, 2014).
34. Burnham, C.-A. D., Leeds, J., Nordmann, P., O'Grady, J. & Patel, J. Diagnosing antimicrobial resistance. *Nat. Rev. Microbiol.* **15**, 697 (2017).
35. Zankari, E. et al. Identification of acquired antimicrobial resistance genes. *J. Antimicrob. Chemother.* **67**, 2640–2644 (2012).
36. Arango-Argoty, G. et al. DeepARG: a deep learning approach for predicting antibiotic resistance genes from metagenomic data. *Microbiome* **6**, 1–15 (2018).
37. Moradigaravand, D. et al. Prediction of antibiotic resistance in Escherichia coli from large-scale pan-genome data. *PLoS Comput. Biol.* **14**, e1006258 (2018).
38. Sang, S. et al. SemaTyP: a knowledge graph based literature mining method for drug discovery. *BMC Bioinforma.* **19**, 1–11 (2018).
39. Segler, M. & Waller, M. P. Chemical discovery as a knowledge graph completion problem. AITP 2017 (2017).
40. Hassani-Pak, K. & Rawlings, C. Knowledge discovery in biological databases for revealing candidate genes linked to complex phenotypes. *J. Integr. Bioinform.* **14**, 20160002 (2017).
41. Santos, A. et al. Clinical knowledge graph integrates proteomics data into clinical decision-making. *bioRxiv* (2020).
42. Jha, A., Khan, Y., Sahay, R. & d'Aquin, M. Metastatic Site Prediction in Breast Cancer using Omics Knowledge Graph and Pattern Mining with Kirchhoff's Law Traversal. *bioRxiv* (2020).
43. Nickel, M., Murphy, K., Tresp, V. & Gabrilovich, E. A review of relational machine learning for knowledge graphs. *Proc. IEEE* **104**, 11–33 (2016).
44. Lao, N. & Cohen, W. W. Relational retrieval using a combination of path-constrained random walks. *Mach. Learn.* **81**, 53–67 (2010).
45. Quinlan, J. R. Learning logical definitions from relations. *Mach. Learn.* **5**, 239–266 (1990).
46. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J. & Yakhnenko, O. Translating embeddings for modeling multi-relational data. *Adv. Neural Inform. Process. Syst.* **26**, 2787–2795 (2013).
47. Wang, Z., Zhang, J. Feng, J. & Chen, Z. Knowledge graph embedding by translating on hyperplanes. In Proc. Twenty-Eighth AAAI Conference on Artificial Intelligence. 1112–1119 (Quebec City, QC, Canada, 27–31 July 2014).
48. Yao, L., Mao, C. & Luo, Y. KG-BERT: BERT for knowledge graph completion. *arXiv Prepr. arXiv1909.03193* (2019).
49. Wang, B. et al. Structure-augmented text representation learning for efficient knowledge graph completion. In Proceedings of the Web Conference 2021. 1737–1748 (2021).
50. Rodriguez, M. & Neubauer, P. A path algebra for multi-relational graphs. In Proc. IEEE 27th International Conference on Data Engineering Workshops. 128–131. https://doi.org/10.1109/ICDEW.2011.5767613 (2011).
51. Consortium, G. O. The gene ontology resource: 20 years and still GOing strong. *Nucleic Acids Res.* **47**, D330–D338 (2019).
52. Liu, A. et al. Antibiotic sensitivity profiles determined with an Escherichia coli gene knockout collection: generating an antibiotic bar code. *Antimicrob. Agents Chemother.* **54**, 1393–1403 (2010).
53. Pasternack, J. & Roth, D. Knowing what to believe (when you already know something). In Proc. 23rd International Conference on Computational Linguistics. 877–885 (2010).
54. Lao, N., Mitchell, T. & Cohen, W. W. Random walk inference and learning in a large scale knowledge base. In Proc Conference on Empirical Methods in Natural Language Processing 529–539 (2011).
55. Freund, Y., Schapire, R. & Abe, N. A short introduction to boosting. *J.-Jpn. Soc. Artif. Intell.* **14**, 1612 (1999).
56. Ji, G., He, S., Xu, L., Liu, K. & Zhao, J. Knowledge graph embedding via dynamic mapping matrix. In Proc. 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers) 687–696 (2015).
57. Dong, X. et al. Knowledge vault: a web-scale approach to probabilistic knowledge fusion. In Proc. 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 601–610 (2014).
58. Kendall, M. G. A new measure of rank correlation. *Biometrika* **30**, 81–93 (1938).
59. Webber, W., Moffat, A. & Zobel, J. A similarity measure for indefinite rankings. *ACM Trans. Inf. Syst.* **28**, 1–38 (2010).
60. Samaluru, H., SaiSree, L. & Reddy, M. Role of SufI (FtsP) in cell division of Escherichia coli: evidence for its involvement in stabilizing the assembly of the divisome. *J. Bacteriol.* **189**, 8044–8052 (2007).
61. Ko, M. & Park, C. H-NS-dependent regulation of flagellar synthesis is mediated by a LysR family protein. *J. Bacteriol.* **182**, 4670–4672 (2000).
62. Krin, E., Danchin, A. & Soutourina, O. Decrypting the H-NS-dependent regulatory cascade of acid stress resistance in Escherichia coli. *BMC Microbiol.* **10**, 1–9 (2010).
63. Djoko, K. Y. et al. Interplay between tolerance mechanisms to copper and acid stress in Escherichia coli. *Proc. Natl Acad. Sci.* **114**, 6818–6823 (2017).
64. Tani, T. H., Khodursky, A., Blumenthal, R. M., Brown, P. O. & Matthews, R. G. Adaptation to famine: a family of stationary-phase genes revealed by microarray analysis. *Proc. Natl Acad. Sci.* **99**, 13471–13476 (2002).
65. Ferrario, M. et al. The leucine-responsive regulatory protein of Escherichia coli negatively regulates transcription of ompC and micF and positively regulates translation of ompF. *J. Bacteriol.* **177**, 103–113 (1995).
66. Gul, N. & Poolman, B. Functional reconstitution and osmoregulatory properties of the ProU ABC transporter from Escherichia coli. *Mol. Membr. Biol.* **30**, 138–148 (2013).
67. Kim, I.-K. et al. Crystal structure of a new type of NADPH-dependent quinone oxidoreductase (QOR2) from Escherichia coli. *J. Mol. Biol.* **379**, 372–384 (2008).
68. Piek, S. et al. The role of oxidoreductases in determining the function of the neisserial lipid A phosphoethanolamine transferase required for resistance to polymyxin. *PLoS ONE* **9**, e106513 (2014).
69. Al Mamun, A. A. M. et al. Identity and function of a large gene network underlying mutagenic repair of DNA breaks. *Sci. (80-.)* **338**, 1344–1348 (2012).
70. Zhao, X. & Lam, J. S. WaaP of Pseudomonas aeruginosa is a novel eukaryotic type protein-tyrosine kinase as well as a sugar kinase essential for the biosynthesis of core lipopolysaccharide. *J. Biol. Chem.* **277**, 4722–4730 (2002).
71. Yethon, J. A. et al. Salmonella enterica Serovar TyphimuriumwaaP Mutants Show Increased Susceptibility to Polymyxin and Loss of Virulence In Vivo. *Infect. Immun.* **68**, 4485–4491 (2000).
72. Alcock, B. P. et al. CARD 2020: antibiotic resistome surveillance with the comprehensive antibiotic resistance database. *Nucleic Acids Res.* **48**, D517–D525 (2020).
73. Merchel Piovesan Pereira, B., Wang, X. & Tagkopoulos, I. Biocide-Induced Emergence of Antibiotic Resistance in Escherichia coli. *Front. Microbiol.* **12**, 335 (2021).
74. Wilkinson, M. D. et al. The FAIR Guiding Principles for scientific data management and stewardship. *Sci. Data* **3**, 1–9 (2016).

75. Toutanova, K. & Chen, D. Observed versus latent features for knowledge base and text inference. In Proc. 3rd Workshop on Continuous Vector Space Models and Their Compositionality 57–66 (2015).
76. Dettmers, T., Minervini, P., Stenetorp, P., Riedel, S. Convolutional 2D knowledge graph embeddings. in Zilberstein, Shlomo and McIlraith, Sheila and Weinberger, Kilian, (eds.) Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI-18). 1811–1818 (AAI Publications: Palo Alto, CA, USA, 2018).
77. Feng, J. et al. Knowledge graph embedding by flexible translation. In Proc. Fifteenth International Conference on the Principles of Knowledge Representation and Reasoning, 2016.
78. Wang, M., Qiu, L. & Wang, X. A Survey on Knowledge Graph Embeddings for Link Prediction. *Symmetry (Basel)* **13**, 485 (2021).
79. Liu, Y. et al. Roberta: a robustly optimized bert pretraining approach. *arXiv Prepr. arXiv1907.11692* (2019).
80. Sun, Z., Deng, Z.-H., Nie, J.-Y. & Tang, J. Rotate: knowledge graph embedding by relational rotation in complex space. *arXiv Prepr. arXiv1902.10197* (2019).
81. Yang, B., Yih, W., He, X., Gao, J. & Deng, L. Embedding entities and relations for learning and inference in knowledge bases. *arXiv Prepr. arXiv1412.6575* (2014).
82. Wang, Q., Mao, Z., Wang, B. & Guo, L. Knowledge graph embedding: a survey of approaches and applications. *IEEE Trans. Knowl. Data Eng.* **29**, 2724–2743 (2017).
83. Ji, S., Pan, S., Cambria, E., Marttinen, P. & Philip, S. Y. A survey on knowledge graphs: representation, acquisition, and applications. *IEEE Trans. Neural Networks Learn. Syst.* **33**, 494–514 (2021).
84. Lee, J. et al. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* **36**, 1234–1240 (2020).
85. Wu, X. et al. Automatic knowledge graph construction: a report on the 2019 ICDM/ICBK contest. In Proc. IEEE International Conference on Data Mining (ICDM). 1540–1545 (IEEE, 2019).
86. Bosselut, A. et al. Comet: commonsense transformers for automatic knowledge graph construction. *arXiv Prepr. arXiv1906.05317* (2019).
87. Yeh, P., Tschumi, A. I. & Kishony, R. Functional classification of drugs by properties of their pairwise interactions. *Nat. Genet.* **38**, 489 (2006).
88. Suzuki, S., Horinouchi, T. & Furusawa, C. Prediction of antibiotic resistance by gene expression profiles. *Nat. Commun.* **5**, 5792 (2014).
89. Weiss, S. J., Mansell, T. J., Mortazavi, P., Knight, R. & Gill, R. T. Parallel mapping of antibiotic resistance alleles in Escherichia coli. *PLoS ONE* **11**, e0146916 (2016).
90. Cho, K. et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv Prepr. arXiv1406.1078* (2014).
91. Hochreiter, S. & Schmidhuber, J. Long short-term memory. *Neural Comput* **9**, 1735–1780 (1997).
92. Sybrandt, J., Tyagin, I., Shtutman, M. & Safro, I. AGATHA: automatic graph mining and transformer based hypothesis generation approach. In Proc. 29th ACM International Conference on Information & Knowledge Management. 2757–2764 (2020).
93. Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S. & Dean, J. Distributed representations of words and phrases and their compositionality. *Adv. Neural Inform. Process. Syst.* **26**, 3111–3119 (2013).
94. Pennington, J., Socher, R. & Manning, C. D. Glove: global vectors for word representation. In Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP) 1532–1543 (2014).
95. Joulin, A., Grave, E., Bojanowski, P. & Mikolov, T. Bag of tricks for efficient text classification. *arXiv Prepr. arXiv1607.01759* (2016).
96. Wang, X., Rai, N., Pereira, B. M. P., Eetemadi, A. & Tagkopoulos, I. Accelerated knowledge discovery from omics data by optimal experimental design. *Nat. Commun.* **11**, 1–9 (2020).
97. Raccuglia, P. et al. Machine-learning-assisted materials discovery using failed experiments. *Nature* **533**, 73–76 (2016).
98. Ribeiro, M. T., Singh, S. & Guestrin, C. "Why should i trust you?" Explaining the predictions of any classifier. Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. p. 1135–1144 (2016).
99. Holzinger, A., Biemann, C., Pattichis, C. S. & Kell, D. B. What do we need to build explainable AI systems for the medical domain? *arXiv Prepr. arXiv1712.09923* (2017).
100. Huynh, L., Tsoukalas, A., Köppe, M. & Tagkopoulos, I. SBROME: a scalable optimization and module matching framework for automated biosystems design. *ACS Synth. Biol.* **2**, 263–273 (2013).
101. Keseler, I. M. et al. The EcoCyc database: reflecting new knowledge about Escherichia coli K-12. *Nucleic Acids Res.* **45**, D543–D550 (2016).
102. Tomasulo, P. ChemIDplus-super source for chemical and drug information. *Med. Ref. Serv. Q.* **21**, 53–59 (2002).
103. Socher, R., Chen, D., Manning, C. D. & Ng, A. Reasoning with neural tensor networks for knowledge base completion. *Adv. Neural Inform. Process. Syst.* **26**, 926–934 (2013).
104. Glorot, X., Bordes, A. & Bengio, Y. Deep sparse rectifier neural networks. In Proc. Fourteenth International Conference on Artificial Intelligence and Statistics. 315–323 (2011).
105. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**, 1929–1958 (2014).
106. Kingma, D. P. & Ba, J. Adam: a method for stochastic optimization. *arXiv Prepr. arXiv1412.6980* (2014).
107. Freund, Y. & Schapire, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* **55**, 119–139 (1997).
108. Chawla, N. V., Bowyer, K. W., Hall, L. O. & Kegelmeyer, W. P. SMOTE: synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **16**, 321–357 (2002).
109. Baba, T. et al. Construction of Escherichia coli K-12 in-frame, single-gene knockout mutants: the Keio collection. *Mol. Syst. Biol.* **2**, 8–2006 (2006).
110. Datsenko, K. A. & Wanner, B. L. One-step inactivation of chromosomal genes in Escherichia coli K-12 using PCR products. *Proc. Natl Acad. Sci.* **97**, 6640–6645 (2000).
111. Dingsdag, S. A. & Hunter, N. Metronidazole: an update on metabolism, structure-cytotoxicity and resistance mechanisms. *J. Antimicrob. Chemother.* **73**, 265–279 (2018).
112. Löfmark, S., Edlund, C. & Nord, C. E. Metronidazole is still the drug of choice for treatment of anaerobic infections. *Clin. Infect. Dis.* **50**, S16–S23 (2010).

Acknowledgements

We would like to thank the members of the Tagkopoulos lab and the reviewers for their suggestions, Nick Joodi and Minseung Kim for their help in the initial discussions, and Ameen Eetemadi for his comments on creating the figures. This work was supported by the USDA-NIFA AI Institute for Next Generation Food Systems (AIFS), USDA-NIFA award number 2020-67021-32855, and the NIEHS grant P42ES004699 to I.T. All code and instructions on how to reproduce the results can be found in https://github.com/IBPA/KIDS.

Author contributions

J.Y. performed all computational analysis, and N.R. performed all wet-lab experiments. J.Y. and N.R. created the figures. J.Y., N.R., and I.T. contributed to the critical analysis and wrote the paper. I.T. conceived and supervised all aspects of the project.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41467-022-29993-z.

Correspondence and requests for materials should be addressed to Ilias Tagkopoulos.

Peer review information *Nature Communications* thanks Mostafa Ellabaan, Jure Leskovec and the other, anonymous, reviewer(s) for their contribution to the peer review of this work. Peer reviewer reports are available.

Reprints and permission information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.