INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Prediction of Disease-Gene Associations by an Ensemble of Knowledge Graph Completion

Ruhina Karani[1]*, **Jay Mehta**[1], **Jay Mistry**[1], **Harit Koladia**[1], **Chetashri Bhadane**[1]

1 Department of Computer Engineering, Dwarkadas J. Sanghvi College of Engineering, Mumbai, Maharashtra, India

OPEN ACCESS

Received: 12-01-2024

Accepted: 12-02-2024

Published: 07-03-2024

Citation: Karani R, Mehta J, Mistry J, Koladia H, Bhadane C (2024) Prediction of Disease-Gene Associations by an Ensemble of Knowledge Graph Completion. Indian Journal of Science and Technology 17(11): 1078-1086. https://doi.org/10.17485/IJST/v17i11.83

* **Corresponding author**.

ruhina.karani@djsce.ac.in

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objective: This research aims to enhance genetic counselors' efficiency in analyzing genetic data across diverse medical settings, spanning prenatal scans, tumor testing, carrier typing, and Fluorescence In Situ Hybridization (FISH), etc. The objective is to employ graph-based techniques for identifying potential gene-disease associations and recommending personalized medical interventions. The application scope extends to areas like personalized medicine, newborn screening, and genetic probing. **Methods**: The study utilizes a novel technique within the PrimeKG genetic graph database, predicting gene-disease associations. Ensembles are constructed from six models - TransE, TransD, TransR, TransH, ComplEx, and DistMult. The hits@10 metric evaluates the model's effectiveness, measuring the accuracy of predictions within the top 10 ranked associations. The ensemble achieves a score of 0.52, indicating a significant proportion of correct predictions in the top ten associations. **Findings**: The research presents an efficient approach to identify gene-disease associations, demonstrating a high hits@10 metric accuracy (0.52). This method significantly aids medical professionals in making informed patient care decisions, with potential applications in prenatal scans, tumor testing, carrier typing, and more. The findings underscore the utility of graph-based techniques in transforming disease identification and treatment through genetic data analysis. **Novelty**: This study introduces a unique approach, leveraging ensembles from diverse models, to predict gene-disease associations within the PrimeKG genetic graph database. The hits@10 metric underscores the model's efficiency, presenting a novel and valuable contribution to the field of genetic data analysis and healthcare decision-making.

Keywords: Gene Disease Prediction; Machine Learning; Knowledge Graph Completion; Drug Discovery; Healthcare

1 Introduction

In the realm of computational medicine, the application of machine learning has gained prominence, especially with the availability of high-throughput genetic data. Machine learning algorithms excel in handling the complexity of genetic data, as demonstrated in Genome-Wide Association Studies (GWAS)[1]. Noteworthy contributions include the diagnostic potential of serum microRNAs in distinguishing pancreatic and biliary tract cancer patients[2]. Challenges in healthcare machine learning, such as bias and uncertainty, emphasize the need for a combined approach with clinical judgment[2]. Deep learning frameworks, like DeepGO, outperform traditional methods in prioritizing disease-related genes[3]. Network-based methods, leveraging genome-wide association data, emphasize the importance of diverse data sources for accurate identification of disease-associated genes[2]. Jafari et al.[4] bioinformatics analysis identified biomarkers for pancreatic cancer, underscoring the significance of exploring various genes and pathways.

Addressing the research gap in predicting gene-disease associations, the proposed system combines six cutting-edge Knowledge Graph completion techniques[5,6]. Leveraging PrimeKG comprehensive biological knowledge base enhances accuracy by integrating diverse data sources to construct functional gene networks and prioritize disease-related genes[7]. This system holds potential for improving genetic counseling and personalized medicine, offering more effective analysis and uncovering potential disease predispositions[8].

2 Methodology

2.1 Dataset Employed

The PrimeKG dataset is a biological knowledge base consolidated by Chandak, Payal et al.[9]. Building on previous efforts in constructing knowledge graphs based on diseases, PrimeKG utilizes 20 high-quality resources to define 17,080 diseases and their 4,050,249 associations across ten important biological domains. These domains cover biological processes and pathways, anatomical and phenotypic scales, perturbations in amino acids linked to disease, as well as a comprehensive list of approved and experimental drugs and their therapeutic action. Figure 1 visualizes a subgraph of the entire knowledge base.

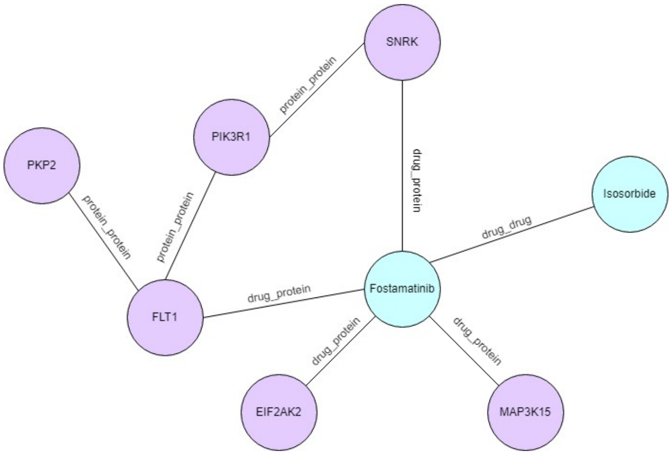

Fig 1. Subgraph from PrimeKG dataset. Purple nodes are genes and cyan nodes are drugs

2.2 Proposed Architecture

Ensemble learning is a machine-learning technique that combines the predictions of different models to make a final prediction[10]. This idea is used to combine the strengths of different models to improve the overall performance of the prediction, as different models use varied embedding spaces for encoding entities and relations. Figure 2 depicts a flow diagram of the proposed architectures, which is mainly divided into three stages namely, data preprocessing, model training, and evaluation. In the data preprocessing step, the csv files from the PrimeKG dataset are used to generate true triples. The entities and relationships are converted to integers by label encoding, this reduces the processing time and allows faster querying times while training the model. The triples generated in this step are used in model training; A training and test set is created from the true triples for evaluating the model after training.

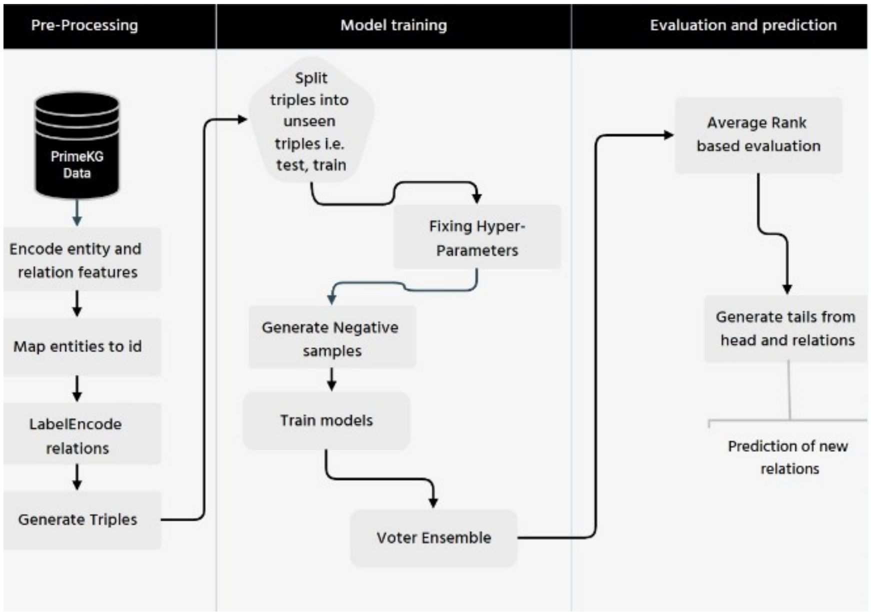
Pre-Processing
Model training
Evaluation and prediction
PrimeKG Data
Encode entity and relation features
Map entities to id
LabelEncode relations
Generate Triples
Split triples into unseen triples i.e. test, train
Fixing Hyper-Parameters
Generate Negative samples
Train models
Voter Ensemble
Average Rank based evaluation
Generate tails from head and relations
Prediction of new relations

Fig 2. Architecture of the Stacked Ensemble

Model training involves selecting hyperparameter values such as margin, learning rate, epochs, and batch size. During the scoring function calculations, negative triples are generated to distinguish between true and negative data. Model training is done using stochastic gradient descent, and the loss history is shown to ensure correct convergence. Trained models provide predictions to construct a voter ensemble, and normalized forecasts are utilized for ranking and evaluation. Mean rank and hits@k are metrics that quantify ensemble accuracy by comparing true predictions among the top-k predicted triples. Tails are predicted using head and relation inputs, prioritized by score. The suggested stacking method employs three models as a voting ensemble, with normalized scores to prevent overfitting. During the training process, entities and relations are normalized. The network is divided into training and test sets for individual model training and final ensemble evaluation[11].

2.3 Problem Definition

The knowledge graph completion problem is a significant challenge in machine learning and artificial intelligence, focusing on deducing missing connections within a knowledge graph, where entities are linked through triples representing relationships as shown in Figure 3.

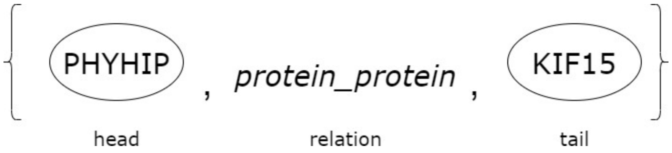

Fig 3. Example of a triple from the PrimeKG dataset

Knowledge graph completion entails describing relationships between items in a structured fashion, known as triples (h, e, t). Entities act as nodes, while relationships are represented as directed edges. Traditional techniques use embeddings, which map things and relationships to a low-dimensional vector space. The proximity of vectors reflects the probability of relationships. Structured metrics and the hinge loss function are two examples of contemporary techniques. Training data consists of real triples (D) and false triples (D') produced by manipulating entities. The margin hyperparameter (γ) determines score discrepancies. Overall, knowledge graph completion uses embeddings, structured metrics, and margin control to address inference issues in graph-based representations[12].

2.4 TransE

Several approaches are taken to learn the low-dimensional embeddings of relations, one such approach is translation-based. TransE algorithm is the most basic translation-based approach[13]. It models the relationship between nodes and relations as translation vectors in the embedding space. In the embedding space, each item and relation is represented as a low-dimensional vector. A translation vector between the embeddings of both entities, projected onto the embedding space of the relation, is used to describe the relationship between two entities and a relation. TransE's scoring function is provided by Equation (1),

$$g(h,\, e,\, t) \;=\; ||h \,+\, e \,-\, t|| \tag{1}$$

TransE algorithm learns representations for nodes and edges by optimizing a margin-based ranking function, similar to other embedding-based methods for knowledge graph completion. The objective function aims to score true triples higher than false triples by a margin of at least γ.

2.5 TransH

TransH is a translation-based knowledge graph completion algorithm[14]. By adding relation-specific hyperplanes, which more effectively represent the interactions between entities and relations, the TransH algorithm expands on the TransE method. Relation normalization is applied to the relation-specific projection matrices in the model, to ensure that the projection matrices are orthogonal and have unit norms. The normalization steps help prevent overfitting and improve the generalization of the model. In TransH, distances between things are calculated in the projected space by projecting them onto a hyperplane that is unique to the relation in which they take part. Formally, the TransH algorithm defines the score function for a triple is given by Equation (2),

$$g\ (h,\, e,\, t) = \ \|eh \,-\, et\|\, p \tag{2}$$

Here, *eh* and *et* are the projected embeddings of the head entity and tail entity onto the hyperplanes unique to the relation e. p is a hyperparameter that controls the norm used to compute the distance between *eh* and *et*. The projection of entities onto relation-specific hyperplanes is achieved through the following formula in Equation (3),

$$ex \;=\; x - (x * e) * e \tag{3}$$

The formula projects x, the embedding of an entity onto the hyperplane perpendicular to e, by subtracting the component of x along e. This way, each entity is represented by a vector that lies on a hyperplane that is specific to the relation it participates in. To train the TransH model, a margin-based ranking loss function is used, which aims to distinguish true triples from false ones.

2.6 TransR

TransR is a knowledge graph completion algorithm. The TransR algorithm extends the TransE algorithm, which represents entities and relations in the same embedding space[14,15], by introducing a separate embedding space for relations. In TransR, entities are projected into a relation-specific space before being compared with other entities through a learned transformation matrix. The TransR algorithm defines the scoring function for a given head, relation, and tail is given by Equation (4),

$$f\ (h,\, e,\, t) \;=\; ||Me(eh) \,+\, r \,-\, Me(et)||2 \tag{4}$$

Here *Me* is a relation-specific transformation matrix that maps an entity embedding from the entity space to the relation space of *e*. *eh* and *et* are the encodings of the head and tail entity projected into the individual relation space of *e*. To train the TransR model, a margin-based ranking loss function is used. The regularization term encourages the embeddings to have low rank, which means that they can be decomposed into a few latent factors. The regularization term in TransR is defined in Equation (5) as follows,

$$R \;=\; \lambda||M||2 \tag{5}$$

λ is a hyperparameter that controls the strength of the regularization, M is a projection matrix that maps the embeddings of the entities from the entity space to the relation space.

2.7 DistMult

DistMult is a method for knowledge graph completion that was proposed in[16]. Specifically, the bilinear model computes the score of a triple (*h*,*e*,*t*) as the dot product of the embeddings of *h* and *e*, multiplied by the embedding of *t*. The DistMult algorithm addresses these issues by using a diagonal matrix to represent the relation embeddings instead of a full matrix. This reduces the number of parameters to be learned and improves generalization. The DistMult algorithm also uses a simpler score function that takes the Hadamard product of the embeddings of the subject and relation, multiplied by the embeddings of the object. The score function is given by Equation (6),

$$g(h,\ e,\ t)\ =<\ h\odot e,t\ > \tag{6}$$

Here, $\odot$ denotes the element-wise product of two vectors. The DistMult algorithm builds encoding for entities and relations by maximizing a margin-based ranking goal, similar to previous embedding-based algorithms for knowledge graph completion. The objective function aims to score true triples higher than false triples by a margin.

2.8 ComplEx

The ComplEx algorithm is a method for knowledge graph completion. To learn the embeddings of entities and relations, the ComplEx algorithm maximizes a margin-based ranking objective, which aims to distinguish true triples from false triples[17]. The score function *f(h, e, t)* is given in Equation (7) as the inner product of the embeddings of *h* and *e*, conjugated with the embedding of *t*.

$$g(h,\ e,\ t)\ =\ Re(<\ h,\ et*>) \tag{7}$$

Here, $Re()$ denotes the real part of a complex number denotes complex conjugation. The ComplEx algorithm learns the embeddings of entities and relations *b*. The ComplEx algorithm uses the same score function and embeddings learned during training for both training and evaluation.

2.9 ConvE

ConvE is a method for knowledge graph completion. The ConvE algorithm combines knowledge graph embedding with convolutional neural networks (CNNs) to capture local patterns and interactions among entities and relations[18,19]. The ConvE algorithm models the relationship between entities and relations using Equation (8)

$$g\ (h,\ e,\ t)\ =\ b([ge(h);\ gr(r)]\ *\ W) \tag{8}$$

Here, *ge* and *gr* are functions that map entities and relations to 2D image-like matrices. [*ge*(h); *gr*(r)] is the con- catenation of the 2D matrices of h and r along the feature dimension. W is the weight matrix of the CNN layer. *b*(.) is a non-linear activation function. The score function $g(h,\ e,\ t)$ should be small if the triple $(h,\ r,\ t)$ is true, and large if the triple is false. To learn the embeddings of entities and relations, the ConvE algorithm minimizes a margin-based ranking loss function, which aims to distinguish true triples from false triples.

3 Results and Discussions

In the ensemble model, each individual model was trained on a common training set using a pipeline. The training process for each model consisted of 100 epochs. However, due to limited computing resources, further experiments with extended training were limited. During training, a batch size of 1280 was chosen to strike a suitable balance between memory usage, training time, and model performance. Additionally, a checkpoint mechanism was implemented to create backups after each epoch. This ensured that in the event of a failure, training could be resumed from the last saved checkpoint, eliminating the need to start the training process from scratch. To optimize the training process, the Adam optimizer was utilized. Adam is a popular choice for optimizing neural network models. The optimizer was applied to each model using their specific parameters, allowing for efficient training and parameter updates. For model evaluation, Rank Based Evaluations were performed to procure metrics for this study. After training, loss was plotted for the 100 epochs as shown in Figure 4.

The individual and ensemble models are evaluated on the basis of rank-based metrics including Hits@3, Hits@5, Hits@10, and Mean Rank.

The KB embedding model seeks to anticipate an entity given a relation and another entity in the KB completion or link prediction task. To be more precise, it infers a tail entity *t* given a head entity and a relation (*h*,*e*) or a head entity and a relation

(h,e) given a tail entity t. Using the rating task below, we first evaluated how well the knowledge graph embedding models performed: To construct the distorted triples for all test triples (h, e, t), the head entity h was taken out and supplanted by all other entities save the head entity; (2) scores for the distorted triples and the test triple are computed using a scoring function; and (3) score values are sorted in descending order, and (4) the test triple's accurate head entity was recorded in order to determine its rank. The algorithms are evaluated on their ability to predict missing relationships between entities in a knowledge graph. Given a test set of triples with one or more missing relationships, the models compute the scores of all possible relationships between the entities in each triple.

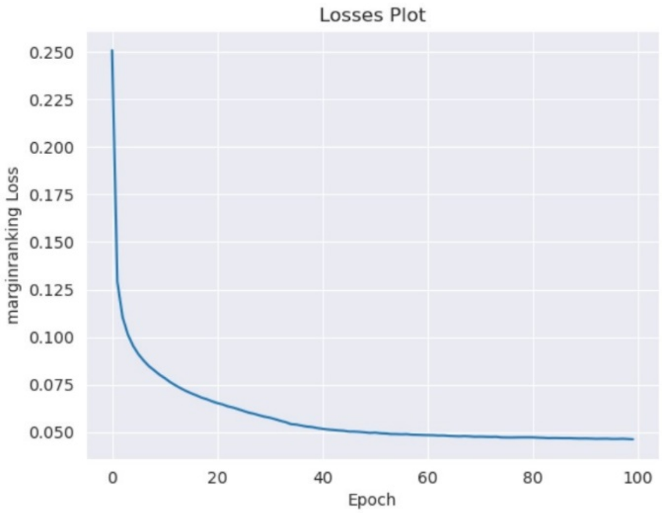

Fig 4. Loss curve plotted after training

Figure 5 displays the realistic arithmetic mean ranks computed for various models computed for head, this evaluation metric is calculated as the arithmetic mean over all individual ranks. Its value ranges from 0 to infinity, where a lower value indicates better performance. This figure clearly indicates that ComplEx model has better adjusted average mean rank, followed by ConvE. One advantage this metric holds over hits@k is its sensitivity to changes in overall model performance, capturing not only specific results under a certain cutoff but also providing an insight into the average performance.

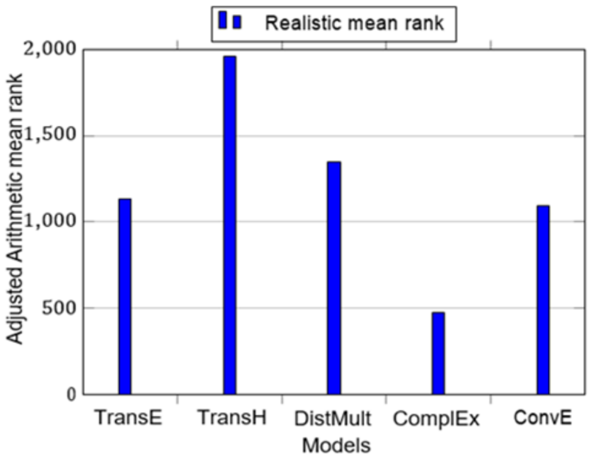

Fig 5. Arithmetic Mean Rank for Head

Table 1 shows the various hits@k values for individual models. Out of the individual models, hits@k values for ComplEx model are better than other models as it encompasses a variety of relationship types and generalizes this well due to its selection of embedding and scoring method, whereas the other models seem to be struggling generalizing the variety of relations.

Table 1. Comparison of Hits values for different models

Model	Hits@3	Hits@5	Hits@10
TransE	4.22	6.22	9.68
TransR	5.59	7.37	9.92
TransH	2.47	3.25	4.56
DistMult	12.40	14.90	18.50
ComplEx	25.98	32.64	41.82
ConvE	12.46	18.28	20.99

Figure 6 shows the adjusted arithmetic mean ranks for various models, computed for head, tail, and both. This evaluation metric is calculated as the ratio of the average rank to the expected mean rank. Its value ranges from 0 to 2, where a lower value indicates better performance. This figure indicates that for the singular models, the ComplEx model has a better-adjusted average mean rank, followed by TransE. One advantage this metric holds over arithmetic mean rank is its independence from the size of training and evaluation triples, as a small 10-triple evaluation with an average rank of 8 is bad, but an average rank of 8 is good for a 1000-triple evaluation.

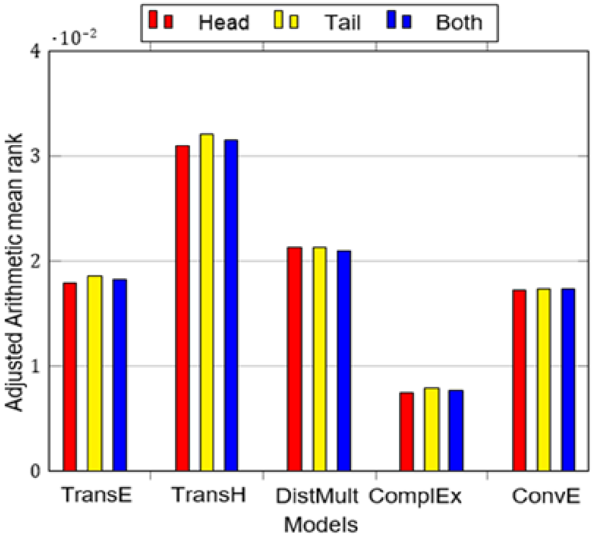

Fig 6. Adjusted Arithmetic Mean Rank

Table 2 shows the various hits@k values for the ensemble models. The hits@k values for ensembles are clearly superior to their lone counterparts. The models together perform better due to better generalization in different models, which when combined with a voting ensemble ensures that popular outcomes are scored higher than obscure ones.

Table 2. Comparison of Hits values for different ensemble models

Model	Hits@3	Hits@5	Hits@10
ComplEx-TransR-DistMult	21.28	25.91	30.72
DistMult-TransH-ConvE	10.53	15.22	18.66
TransE-TransR-ConvE	12.75	15.49	18.02
TransE-TransH-DistMult	14.32	16.73	19.75
ConvE-DistMult-ComplEx	**31.74**	**33.56**	**42.97**
ComplEx-TransE-ConvE	27.33	33.95	50.16
DistMult-ComplEx-TransE	14.40	16.39	23.84

From the final results, we can sense a correlation between better generalizing singular models forming better-performing ensemble models. The ensemble approach outperforms individual models and other baseline methods for link prediction in the PrimeKG dataset. The results suggest that model ensemble can be an effective technique for improving the prediction of new disease-gene associations. The comparison with other models cannot be done due to the unavailability of existing research on PrimeKG dataset.

4 Conclusion

In this study, we proposed a novel ensemble stacking of a variety of knowledge graph completion models to discover disease-gene associations. The study used the PrimeKG dataset, which includes a myriad of biological entities and represents the relations between them. This paper mainly focuses on the disease-gene relations, and evaluating the models trained on the entire dataset based on their performance on predicting new disease-gene links. A number of experiments are performed on model stacking of different permutations of individual knowledge graph completion models. The models are assessed on rank-based evaluation metrics, Mean Rank, Hits@10, Hits@5, Hits@3.

In light of the findings, we can deduce that the disease-gene interactions inferred by the ensemble technique may have valuable implications for future research. For instance, this might facilitate the study of as-yet-unidentified pathogenic pathways in human illnesses and speed up the development of new disease treatments. The same algorithm can be used to find drug-disease indications and contraindications, and disease-exposure associations, which are crucial fields of active research to understand diseases and treat them effectively.

The proposed model provides a naive approach to incorporating the learned representations of various embedding functions, which is essential to form low-level embeddings of different types of relations. However, it does not form a single embedding function to represent relations. This is especially essential when a visualization of the relations is required which cannot be provided with this approach. The visualizations of the relations aids in further understanding of the model's learning capacity, similarity in relations, and distance between relations. The model can be further augmented by using some embedding space combining functions that can capture the significance of each embedding space and have specific parameters that can control the influence of each embedding in the final embedding function.

References

1) Roman-Naranjo P, Parra-Perez AM, Lopez-Escamez JA. A systematic review on machine learning approaches in the diagnosis and prognosis of rare genetic diseases. *Journal of Biomedical Informatics*. 2023;143:1–8. Available from: https://doi.org/10.1016/j.jbi.2023.104429.
2) Kim K, Yoo D, Lee HS, Lee KJ, Park SB, Kim C, et al. Identification of potential biomarkers for diagnosis of pancreatic and biliary tract cancers by sequencing of serum microRNAs. *BMC Medical Genomics*. 2019;12(1):1–11. Available from: https://doi.org/10.1186/s12920-019-0521-8.
3) Poirion OB, Jing Z, Chaudhary K, Huang S, Garmire LX. DeepProg: an ensemble of deep-learning and machine-learning models for prognosis prediction using multi-omics data. *Genome Medicine*. 2021;13(1):1–15. Available from: https://doi.org/10.1186/s13073-021-00930-x.
4) Jafari S, Ravan M, Karimi-Sani I, Aria H, Hasan-Abad AM, Banasaz B, et al. Screening and identification of potential biomarkers for pancreatic cancer: An integrated bioinformatics analysis. *Pathology - Research and Practice*. 2023;249:154726. Available from: https://doi.org/10.1016/j.prp.2023.154726.
5) Chandak P. PrimeKG (Version 2). 2022. Available from: https://doi.org/10.7910/DVN/IXA7BM.
6) Wang Z, Gu Y, Zheng S, Yang L, Li J. MGREL: A multi-graph representation learning-based ensemble learning method for gene-disease association prediction. *Computers in Biology and Medicine*. 2023;155:106642. Available from: https://doi.org/10.1016/j.compbiomed.2023.106642.
7) Lei X, Zhang Y. Predicting disease-genes based on network information loss and protein complexes in heterogeneous network. *Information Sciences*. 2019;479:386–400. Available from: https://doi.org/10.1016/j.ins.2018.12.008.
8) Brusa M, Dickenson D. Personalized medicine and genetic newborn screening. In: Barilan YM, Brusa M, Ciechanover A, editors. Can precision medicine be personal; Can personalized medicine be precise? Oxford University Press. 2022;p. 107–C8.P56. Available from: https://doi.org/10.1093/oso/9780198863465.003.0008.
9) Chandak P, Huang K, Zitnik M. Building a knowledge graph to enable precision medicine. *Scientific Data*. 2023;10(1):1–16. Available from: https://doi.org/10.1038/s41597-023-01960-3.
10) Chen CHH, Tanaka K, Kotera M, Funatsu K. Comparison and improvement of the predictability and interpretability with ensemble learning models in QSPR applications. *Journal of Cheminformatics*. 2020;12(1):1–16. Available from: https://doi.org/10.1186/s13321-020-0417-9.
11) Dogan A, Birant D. A Weighted Majority Voting Ensemble Approach for Classification. In: 2019 4th International Conference on Computer Science and Engineering (UBMK). IEEE. 2019. Available from: https://doi.org/10.1109/UBMK.2019.8907028.
12) Ratajczak F, Joblin M, Hildebrandt M, Ringsquandl M, Falter-Braun P, Heinig M. Speos: an ensemble graph representation learning framework to predict core gene candidates for complex diseases. *Nature Communications*. 2023;14(1):1–18. Available from: https://doi.org/10.1038/s41467-023-42975-z.
13) Qian W, Fu C, Zhu Y, Cai D, He X. Translating Embeddings for Knowledge Graph Completion with Relation Attention Mechanism. In: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence. International Joint Conferences on Artificial Intelligence Organization. 2018;p. 4286–4292. Available from: https://www.ijcai.org/proceedings/2018/0596.pdf.
14) Asmara SM, Sahabudin NA, Ismail NSN, Sabri IAA. A Review of Knowledge Graph Embedding Methods of TransE, TransH and TransR for Missing Links. In: 2023 IEEE 8th International Conference On Software Engineering and Computer Systems (ICSECS). IEEE. 2023;p. 470–475. Available from: https://doi.org/10.1109/ICSECS58457.2023.10256354.

15) Wu D, Zhao J, Li M, Shi M. A Knowledge Representation Method for Multiple Pattern Embeddings Based on Entity-Relation Mapping Matrix. In: 2022 International Joint Conference on Neural Networks (IJCNN). IEEE. 2022;p. 1–8. Available from: https://doi.org/10.1109/IJCNN55064.2022.9892443.

16) Shen T, Zhang F, Cheng J. A comprehensive overview of knowledge graph completion. *Knowledge-Based Systems*. 2022;255. Available from: https://doi.org/10.1016/j.knosys.2022.109597.

17) Mohamed SK, Muñoz E, Novacek V. On Training Knowledge Graph Embedding Models. *Information*. 2021;12(4):1–19. Available from: https://doi.org/10.3390/info12040147.

18) Liu Y, Tian J, Liu X, Tao T, Ren Z, Wang X, et al. Research on a Knowledge Graph Embedding Method Based on Improved Convolutional Neural Networks for Hydraulic Engineering. *Electronics*. 2023;12(14):1–16. Available from: https://doi.org/10.3390/electronics12143099.

19) Gao Z, Pan Y, Ding P, Xu R. A knowledge graph-based disease-gene prediction system using multi-relational graph convolution networks. *AMIA Annual Symposium Proceedings*. 2023;2022:468–476. Available from: https://pubmed.ncbi.nlm.nih.gov/37128437/.