IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 29, No. 4, August, 2024, pp. 1541-1548
Journal homepage: http://iieta.org/journals/isi

# Bridging Data Complexity with GATNet for Learning in Interconnected Electronic Medical Records Graphs

Swathi Mirthika G.L., Sivakumar B*

Department of Computing Technologies, School of Computing, College of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur 603203, Chengalpattu, India

Corresponding Author Email: sivakumb2@srmist.edu.in

Copyright: ©2024 The authors. This article is published by IIETA and is licensed under the CC BY 4.0 license (http://creativecommons.org/licenses/by/4.0/).

https://doi.org/10.18280/isi.290427

**Received:** 4 January 2024
**Revised:** 9 July 2024
**Accepted:** 2 August 2024
**Available online:** 21 August 2024

***Keywords:***
*electronic medical record, graph database, heterogeneous graph, knowledge graph, link prediction*

**ABSTRACT**

Heterogeneous graphs are a data format for graphs that could define complicated and diverse real-world interactions by accommodating distinct sorts of nodes and edge types. Heterogeneous graphs organize varied medical data to help patients, therapies, drugs, and healthcare practitioners make informed decisions. Medical recommendation systems use them to represent and analyze complicated connections between healthcare data items. Heterogeneous graphs can potentially be constructed and analyzed using the Graph Attention Network (GAT). The purpose of this research is to tackle the issue of implementing a complicated and extremely diverse dataset, which consists of: Using the GATNet (Graph Attention Network) method, we will show how to perform two things: (1) Construct a model with several attributes and relationships using EMR (electronic medical record), and (2) Use that model in a disease prognostic prediction challenge. The initial graph database utilizes a graphical depiction of a patient's progression, showcasing a query of a predictive network that produces analytical findings of AUROC–0.75 and AUPRC–0.17 which is 0.03% & 0.02% higher compared to the existing models.

## 1. INTRODUCTION

Applications such as social networks, recommendation systems, and knowledge graphs have all benefited from the increased use of graph-based data. The performance of programs that rely on graph-based data is adversely affected by the difficulty of detecting missing relationships between nodes. Faster access to a great amount of information has resulted from the growth of innovation and the broad availability of internet services. However, this has also led to an explosion in the amount of data available online, making it harder for people to zero down on relevant results. Several methods with less computational needs have evolved as a solution to this problem, making it simpler and quicker to get to the information we need.

As a result, research and development into recommender systems has gained strength. The ideas of link prediction and heterogeneous graph creation are relevant when discussing knowledge graphs. Knowledge graphs are organized illustrations of information that include of things (nodes) and their connections (edges). These connections can vary in kind, and entities can possess diverse qualities. Constructing a heterogeneous graph and predicting links are two essential challenges in the study of a knowledge graph.

The connection between these two notions is based on the idea that a well-designed diverse knowledge graph serves as the basis for accurate link prediction. Utilizing a complete and well-organized graph that encompasses entities and links from many domains enables the development of more precise and relevant link prediction models. The graph's heterogeneity, characterized by its varied entity and relationship types, is a valuable resource for forecasting novel connections or absent interactions among entities.

The process of constructing a heterogeneous graph is fundamental for creating intricate knowledge graphs that consist of various entities and relationships. On the other hand, link prediction utilizes these graphs to make forecasts about potential or absent connections between entities, capitalizing on the organized information stored in the graph.

In conventional electronic medical record (EMR) systems, data is structured and administered within relational database systems, whereby there exists no inherent linkage between the recorded data. In order to demonstrate, it is common practice in database design to establish relationships across various databases through the use of foreign keys. These foreign keys are often connected to a column within a table, indicating the connection between the data tables rather than the individual data pieces. In contrast, graph databases establish connections between data records in order to efficiently organize data attributes, with a particular emphasis on the interconnections between data pieces. Entities and links are employed in order to enhance space efficiency and provide expedited querying for extensive datasets in comparison to relational mapping.

A link prediction is an issue that involves predicting a connection between two nodes based on the characteristics of those nodes. This problem is connected to research topics that are relevant to the study of the long-term state of the network. Each of the nodes in the system as well as any further

connections that have been identified. Utilizing a social network's historical data allows for the investigation of its current & possible future states and the prediction of the kinds of shifts and alterations that will take place within the latter.

The purpose of link prediction is to find a set of missing or future ties between users by estimating the probability of presence (or development) for each of the non-existing network nodes. This can be done in order to complete the network [1]. If an edge does not already exist in a network, link prediction can help find it. To determine which nodes in a network, refer to the same person, entity resolution analyses the attributes of each node and the connections between them. Despite being separate tasks, link prediction and heterogeneous graph creation have similar objectives, such as describing and analyzing complicated relationships. The development of a heterogeneous graph serves as the fundamental basis for link prediction tasks inside a structured and linked system. This style facilitates the prediction of connections and interactions between diverse entities and node types.

When it comes to completing a knowledge graph, one of the most basic tasks is link prediction, which makes use of preexisting relationships to infer new ones and therefore construct a fuller knowledge graph. There is a plethora of proposed approaches for carrying out the link-prediction task, each of which makes use of a different representational strategy. In order to generate a more complete knowledge graph, link prediction is a crucial task in knowledge base completion that makes use of preexisting ties to infer new links. The link-prediction task has been approached in a variety of ways based on different representational methodologies [2-4]. On the basis of link similarity score propagation via stochastic process in networks with nodes properties, a link prediction technique is proposed. According to the similarity of the properties on the nodes connected by the link, the algorithm assigns an ability to bring to each network link. In addition to its importance in other fields, such as medication development and knowledge graph building, link prediction is an essential step in these endeavors as well.

To model EMR as a heterogeneous bipartite network with attributes on nodes and edges, a new method is proposed. The latent relationships among the population may be thoroughly explored and analyzed using an efficient visualization, when combined with a focused-on patient's graph technique. For the purpose of illustrating how effective our disease-predicting model is, we devised several applicable cascade link prediction tasks that were based on the GATNet algorithm. This approach demonstrates that the performance acquired from EMR supports a sufficient significance to anticipate the result of an event that occurs within the patients and advocates for overall healthcare.

## 2. RELATED WORK

The entities and relations were comparatively diminutive, and augmenting the attributes of entities and relations is seen an imperative subsequent action. The assessment of the knowledge graph was rather uncomplicated and did not include a comparison examination of other graphs produced by the different methodologies. In addition, the utilization of knowledge graphs was initial, and there is room for enhancing the extent and profundity of knowledge graph applications in the field of recommendation [5]. The accuracy did not improve when the HGM embedding vector was concatenated with the diagnosis feature vectors, compared to when the raw lab test and diagnostic feature vectors were concatenated. This discovery suggests that the raw lab test feature vector contains distinct information that may be effectively utilized by CNN. Simultaneously, this discovery suggests that the patient vector incorporated in the HGM model may lose certain information from the original lab test feature when it is projected into a lower-dimensional latent space. To enhance the accuracy of mortality prediction, we want to conserve the information from various data points by combining all feature vectors [3, 4]. Unlike previous link prediction methods, WLNM does not make assumptions about a specific link generation mechanism, such as common neighbours. Instead, it learns this mechanism directly from the graph [6-11]. Generating a medical knowledge graph or graph representations has been the subject of extensive research recently [12-14].

A bipartite network database was created using electronic health records of patients with heart failure. The network analysis was conducted to examine the connections between patients and healthcare providers, and network statistics were calculated to show these interactions [15]. However, this study is flawed in its excessive time expenditure throughout the inquiry process. Additionally, the use of a detailed semantic knowledge network constructed from electronic medical records (EMR) for uncommon diseases highlighted the significance of partially automated schema creation in order to establish more detailed semantic connections [16]. Though this study demonstrated that the prediction job achieves superior performance when the entity types are specified, the assessment dataset had a limited range of relationship types due to the use of non-automatic labor-intensive methods. In contrast, we developed an automated procedure for creating graphs that complements the demanding work. This process also addresses the problem of labelling various types of links and resolving memory storage concerns [17].

In addition, the use of medical datasets to leverage valuable resources is becoming increasingly popular in personalized healthcare and predicting medicine applications, particularly in the context of graph neural networks. An instance of the heterogeneous similarity graph neural network was employed to examine health data based on temporal structural characteristics. This was achieved by creating numerous subgraphs and using them as input for prediction [18]. However, the techniques illustrated in this research are not well-suited to the characteristics of the EMR in network integrating. The EMR dataset possesses a distinctive structure whereby several types of datasets (such as medicine, laboratory, physical, visits, etc.) are linked together to depict the medical condition of patients. These datasets are also related through an anonymous key [19]. However, the graphs stated earlier utilize genomes, proteomics, molecular biology, or movie review datasets that do not necessarily rely on the connectivity between the nodes. As a result, a network is created where characteristics do not require any specific linkage.

In this instance, we provide methods for integrating diverse medical entities and interactions in order to forecast a patient's outcome using a graph that is built only from the EMR information. Our primary contributions are.

(i) This paper presents a novel method for developing a heterogeneous bipartite graph model using electronic medical records (EMR) that includes attributes on both nodes and edges. 7By employing a powerful visualization technique, in

combination with a graph method that prioritizes the needs and preferences of patients, it becomes possible to thoroughly examine and analyze the hidden connections within the population.

(ii) We utilized the GATNet algorithm to create practical downstream link prediction tasks, showcasing the effectiveness of our disease forecasting model [20]. This framework demonstrates that the improvements in performance achieved through the use of EMR (electronic medical records) provide a significant level of accuracy in predicting patient outcomes and strongly supports the advancement of healthcare as a whole.

In addition, we provided the approach for creating an EMR-integrated graph database. The GATNet method was then used to implement the EMR-integrated graph model in network learning [21, 22]. In this research, we demonstrated the model's efficacy by illustrating the graph database architecture and displaying query results. By forecasting the occurrence of sickness depending on the efficiency of our implementation, this study sheds light on the choices made by doctors [23].

## 3. METHODS

Figure 1 provides a concise representation of the study methodology. The datasets are associated with the International Classification of illnesses, 10th Revision (ICD-10) code of the topic and imported using comma-separated value files. The files were initially processed using Python and subsequently utilized for additional analysis with Neo4j and the Stellar graph framework. The collection consists of records for roughly 50,000 patients. There are more than 200,000 instances of medical contacts, which encompass hospital admissions, outpatient visits, and emergency department visits. The dataset comprises a range of features classified into patient demographics, clinical data, drugs, diagnoses, and procedures. The dataset utilised in this work is an extensive and intricate repository of computerised medical records, offering a thorough perspective on patients' medical backgrounds. Advanced modelling tools, such as heterogeneous graph attention networks, are required to efficiently capture and utilise the complex interactions among diverse data for predictive modelling. Our objective is to enhance readers' comprehension of the study setting and the difficulties associated with managing intricate data by presenting a comprehensive dataset description. A comparison between other methods and GATNet is given in Table 1.

Two distinct graph models were constructed in this work, each based on unique topologies and analytical objectives. The first graph was generated using Neo4j to integrate patient information into an EMR system. This allows for the efficient visualization of the patient's medical history and facilitates the retrieval of relevant data points through simple query input. The construction of our property graph involved the use of semantics mapping on superficial network embedding. The second graph was generated using the Stellar graph, and neural network predictions were conducted.

**Table 1.** Comparison table for various methods with GATNet

| Method | Adapting Heterogeneity | Attention Mechanism | Scalability | Therapeutic emphasis |
|---|---|---|---|---|
| GCN | No | No | Moderate | Low |
| Graph SAGE | Partial | No | High | Low |
| R-GCN | Yes | No | Moderate | Moderate |
| HGNN | Yes | Limited | Moderate | Moderate |
| GATNet | Yes | Yes | High | High |

### 3.1 Steps in constructing heterogeneous graph

Determining the various node types that represent the entities or concepts in the dataset is the first stage in building a heterogeneous network. The diversity of the network is reflected in the many node types, which will be used as a starting point for further modelling.

Next, it's crucial to define edge types, or the many possible connections between nodes, after the node types have been created. These connections between nodes in the dataset often have several meanings, showing the complexity of the relationships between them. Associations between friends, followers, or genres in a movie recommendation system are all instances of such linkages.

An essential step, data collecting involves accumulating information from many resources such databases, application programming interfaces (APIs), and text files. The next step is data preparation, which includes activities like data cleansing, filtering, and transformation to bring the raw data into line with the intended network layout. The building of the heterogeneous graph entails the generation of nodes and edges. Each entity or notion is represented by a node, and each node has its own type and properties that are essential to its function. Relationships between nodes are represented by edges, which may have many edge kinds and other properties to capture subtle ties.


**Figure 1.** Overall process for graph construction includes the graph schema representation for the database

Graphs can be represented with the help of certain graph libraries or frameworks, which improves the graph's storage and management. NetworkX, Neo4j, and Gephi are three well-known programmers with specialized features for different purposes [24]. A wide range of graph analytics operations, such as node categorization, connection prediction, community discovery, and more, may be built upon the generated heterogeneous graph. Tools for visualizing data help us make sense of the network of links, illuminating hidden patterns and insights. Regular maintenance and updates, including the inclusion of new data and the removal of old information, are necessary to keep the graph up-to-date and accurate.

In inference, the development of heterogeneous graphs is a vital step for the modelling of complex, heterogeneous data in a wide variety of fields. It provides a versatile and context-aware method of data representation and analysis, allowing analysts and researchers to extract useful insights and information from complex datasets.

(1) Identify Node Types: Figure out what kinds of nodes will be included in your medical recommendation network. Patients, medical problems, therapies, healthcare providers, and pharmaceuticals are all examples of possible node types in this context. The recommendation system requires that different types of nodes each stand for unique entities or ideas.

(2) Define Edge Types: Identify the different kinds of connections that may be made, or "edges," between nodes. Patient-doctor linkages, patient-treatment ties, drug-condition ties, and more are all possible edge types for a medical recommendation system. Each type of edge should capture a different facet of the medical data.

(3) Data Collection: Collect information from a wide range of places, such as patient files, medical databases, clinical trial data, and electronic health records (EHRs). Patients' medical histories, diagnoses, treatments, and drugs are all examples of what should be included in this data.

(4) Data Preprocessing: To verify that the data can be used to generate graphs, you will need to prepare and clean it. Data cleansing, duplication detection, and format conversion to make the information suitable for graph modelling are all tasks that may be required.

(5) Node and Edge creation: Make a node for each item the system is meant to propose. Types of nodes can be assigned to define their functions. Nodes representing people, medical problems, and therapies, for instance, would each have a specific type. Create links (or edges) between nodes to show connections. Relationships between nodes in a network should be represented by edges with names like "treated by," "diagnosed with," and "prescribed."

(6) Graph Representation: Create the heterogeneous graph using a graph library or framework that allows for different types of nodes and edges. Verify that the graph can accommodate different sorts of nodes and edges.

(7) Attribute Assignment: Give labels to nodes and labels to edges. Examples of patient characteristics include age, gender, and medical history; examples of therapy characteristics include efficacy and potential for adverse effects. These details improve the graph and aid in making suggestions.

(8) Integration of External Knowledge: Integrating external medical information sources into the graph, such as medical ontologies, medication databases, or clinical guidelines, can help to improve its quality. Because of this, the precision of the suggestions may be improved.

(9) Recommendation Model Integration: Incorporate methods of machine learning or recommendation that make use of the heterogeneous graph. These models can make use of the graph structure and the characteristics in order to give patients with personalized medicinal suggestions.

(10) Evaluation and Validation: Execute an analysis of the performance of the recommendation system making use of the necessary metrics and validation methods. Make sure that the suggestions are in line with the medical guidelines and that they offer helpful insights to both patients and the professionals who care for them.

(11) Continual Updates: To keep the recommendation system up to date and relevant, you will need to maintain the heterogeneous graph and frequently update it with fresh data, therapies, and medical knowledge.

Constructing a heterogeneous graph for a medical recommendation system is a complicated but necessary task. If beneficial, this can result in more precise and individualized suggestions for medical care, which will eventually be of value to both patients and healthcare professionals [25].

### 3.2 Link prediction in bipartite network

In order to anticipate the connection, a network is defined as a graph. The information inside the network is depicted using nodes, while the connections between them are depicted using links. Predictions are made about the future of unconnected linkages between pairs of nodes. A score is computed for every pair of nodes that are probable to be linked. As the estimated score between two nodes increases, the likelihood of a future connection between those nodes also increases. Bipartite networks have nodes spread over two distinct clusters. Links exist solely between nodes located in distinct clusters. There is a lack of connections between nodes within the same cluster. Several societal networks in our surroundings have a bipartite network topology [26].

The majority of link prediction algorithms are designed for networks with a single mode. Consequently, conventional link prediction techniques are not suitable for direct use in bipartite networks. In order to forecast links in bipartite social networks, it is common practice to convert these networks into single-mode social networks. Common methods to find the link prediction are given below:

### 3.3 Jaccard coefficient

The Jaccard coefficient is a normalized version of the common neighborhood measure. One of the common neighbours of the pair of nodes a and b is selected randomly from the collection of neighbours. There are multiple nodes. This metric increases as the number of similar neighbours increases.

The equation is shown below:

$$J(a,b) = (r(a) \cap r(b)) / (r(a) \cup r(b)) \quad (1)$$

### 3.4 Preferential attachment (P)

The likelihood of one of the endpoints of a future connection being made in the network being a is directly related to the quantity of neighboring nodes connected to node a. Nodes that have a higher number of neighbours are more prone to establishing additional connections. Newman states that the likelihood of collaboration between a and b in a

cooperative network is directly related to the number of collaborations involving a and b.

Below is the mathematical equation:

$$P(a,b) = \left| r(a) \right| . \left| r(b) \right| \quad (2)$$

### 3.5 Common neighbors (C)

It is a metric that operates on the idea that the likelihood of two nodes becoming linked in the future is directly related to the quantity of neighbors they have. The likelihood of a connection being established between two nodes increases as the number of common nodes they share increases. Due to its simplicity, it is one of the most commonly employed measures in the field of link prediction. The mathematical equation is shown below:

$$C(a,b) = | r(a) \cap r(b) | \quad (3)$$

Deep Neural Networks have been utilised in link prediction, with techniques like KBAT and CapsE demonstrating impressive performance. However, their effectiveness varied when applied to different benchmarks. A study conducted by revealed that the observed behaviour was a result of an inadequate assessment process, and the performance of these assessment process, and the performance of these models declined once the underlying biases were addressed. According to their analysis, shallow Knowledge Graph Embedding (KGE) models such as TransE, RotatE, ComplEx, and QuatE demonstrate consistent performance across many assessment methods.

A community-aware high-order proximity may be used to optimise node embedding in an intriguing family of methodologies called community embeddings. An example of such a model is vGraph, which is a probabilistic generative model that jointly learns community membership and node representation. ComE+ is an alternative method for embedding communities that can address situations when the number of communities is uncertain. Nevertheless, these techniques primarily concentrate on generating node and community embeddings by considering intra-group connections for clustering and node classification purposes, rather than for predicting links.

## 4. PROPOSED MODEL

Nodes in the graph represent entities, and edges reflect interactions between them; the Graph Attention Network (GATNet) is a neural network architecture tailored to processing such graph-structured data. GATNet makes use of attention techniques to gather data from nearby nodes efficiently. Table 2 provides the detailed pseudo code used to experiment and Figure 2. Represents the procedure to construct the GATNet algorithm used for the training purpose.

Adam optimizer with learning rate of 0.005 is used. Cross entropy classification is utilized to calculate the loss function with the batch size of 128 for 200 epochs is used for the implementation process. The early stopping technique is implemented with a patience of 10 epochs, using the validation loss as a criterion to avoid overfitting.

**Table 2.** Pseudo-code outlining the GATNet training method

| | |
|---|---|
| **1** | Set the initial values of the GATNet parameters as follows: weights W and attention coefficients a. |
| **2** | Iterate through each epoch from 1 to num epochs. |
| **3** | Randomise the sequence of the training data |
| **4** | Iterate over each batch B in the training data. |
| **5** | Calculate the node embeddings H: |
| **6** | $h_m$ = LeakyReLU((X * $w_i$) ⊕ (a_1 * [X * W_1])) |
| **7** | $h_n$ = LeakyReLU(($h_m$ * $w_j$) ⊕ (a_2 * [$h_m$ * $w_j$])) |
| **8** | deploy dropout in $h_n$ |
| **9** | Calculate the logits Z by applying the softmax function on $h_n$ |
| **10** | Calculate the loss L by employing cross-entropy loss with the labels Y. |
| **11** | Utilise the Adam optimizer to backpropagate and alter the parameters (w, a). |
| **12** | Evaluate the model using the validation set |
| **13** | For patient epochs, if validation loss is unchanged, then |
| **14** | End training. |
| **15** | Return-trained GATNet model |

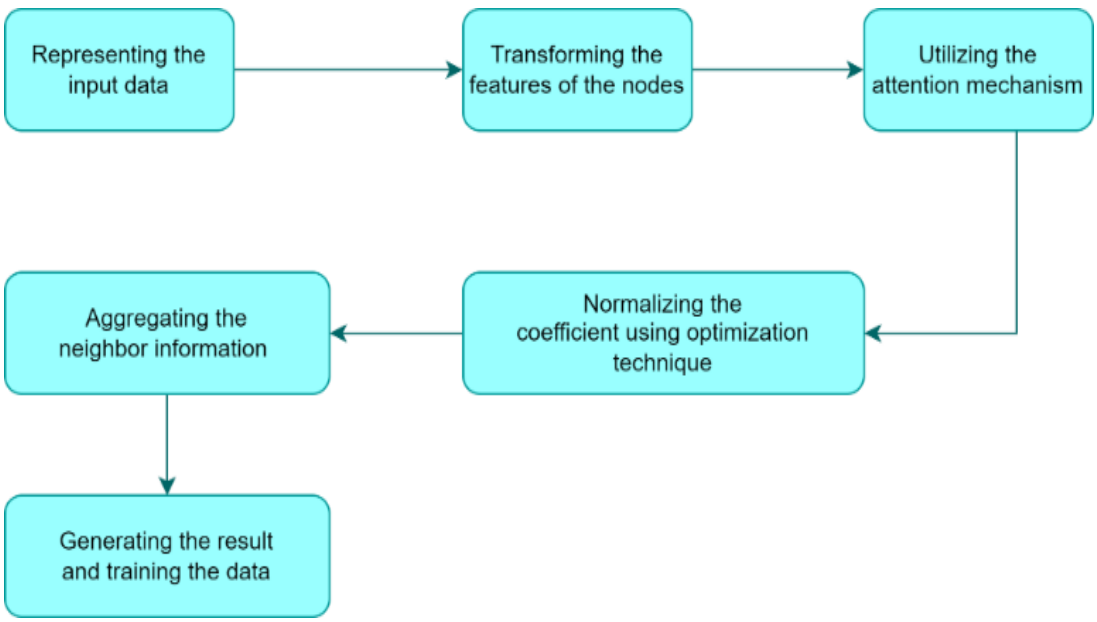



**Figure 2.** Procedure to construct and deploy Graph Attention Network (GATNet)

### 4.1 Representing the input data

Considering a graph,

$$g = (v, e) \tag{4}$$

where v are nodes and e are edges. Every vertex $u_i$ is the feature vector associated with $v_i$ at the outset, this can be a representation with encryption or an embedded representation.

### 4.2 Transforming the features of the nodes

To create the first representations of the nodes, a linear transformation is applied to each one.

$$T_i^{(o)} = w_o u_i \tag{5}$$

$w_o$ is the weight of the matrix.

### 4.3 Utilizing the attention mechanism

To determine attention coefficients for each node's neighbors, GATNet makes use of attention mechanisms.

To calculate the attention coefficient $e_{mn}$ between the nodes $v_m$ and $v_n$ using the attention mechanism.

$$e_{mn} = \text{LeakyReLu}\left(a^T \left[wh_m \| wh_n\right]\right) \tag{6}$$

$a$ is the shared attention mechanism.

$\|$ represents the concatenation.

LeakyReLu is the activation function of a leaky rectified linear unit.

### 4.4 Normalizing the coefficient using optimization technique

To the calculated attention coefficients, apply a SoftMax operation $e_{mn}$ over each node's neighboring nodes $v_i$:

$$a_{mn} = \frac{\exp(e_{mn})}{\sum_{k \in N_i} (\exp(e_{mk}))} \tag{7}$$

$N_i$ denotes the set of neighbors of node $v_i$.

### 4.5 Aggregating the neighbor information

Construct a novel representation for each node by aggregating neighbour node characteristics using the attention coefficients:

$$r_i^{(j+1)} = \sigma\left(\sum_{l \in N_i} a_{mn} w^{(j)} h_l^{(j)}\right) \tag{8}$$

$h_l^{(j)}$ represents the node $v_i$ at layer $l$.

$w^{(j)}$ weight matrix for layer $l$.

$\sigma$ is an activation function like LeakyReLU.

## 5. RESULT AND DISCUSSION

Generating the result and training the data: The final node models acquired through the equations [4-8] after numerous layers of aggregating and manipulation can be utilized for node categorization or other downstream activities. Training the GATNet using an appropriate loss function (for classification tasks, such as cross-entropy loss) then optimize the model parameters with gradient-based optimization methods such as stochastic gradient descent (SGD) or Adam. Table 3 provides the comparison data of the existing and proposed model which is depicted as a graph in Figure 3.

Case Study 1:

Warfarin and antibiotics can interact with one other.

Hypothesis: GATNet effectively forecasts a probable interaction between Warfarin (an anticoagulant) and specific antibiotics.

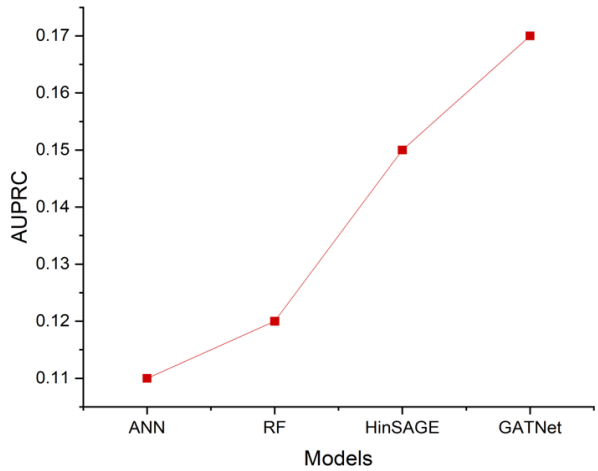
Medical Implication: This interaction may result in a higher likelihood of bleeding. Early detection enables the adjustment of dosage or the prescription of alternative antibiotics, so reducing the occurrence of severe bleeding issues.

Case Study 2:

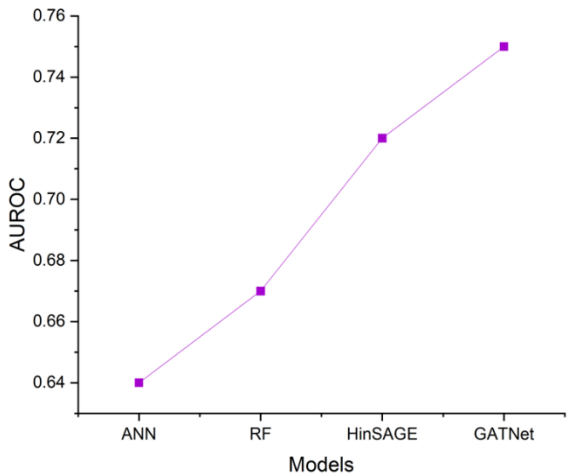
Effect of Statins and Grapefruit Juice Interaction.

Hypothesis: GATNet detects the correlation between Statins (medications that reduce cholesterol levels) and grapefruit juice.

Medical Implication: This combination can heighten the likelihood of muscle poisoning. Notifying healthcare providers allows them to provide nutritional guidance to patients, reducing the danger.




(a)




(b)

**Figure 3.** Performance evaluation of AUPRC

**Table 3.** Model performance comparison for baseline models and GATNet

| Model | AUROC | AUPRC |
|---|---|---|
| **ANN** | 0.64(0.63,0.70) | 0.11(0.09,0.12) |
| **RF** | 0.67(0.63,0.70) | 0.11(0.09,0.12) |
| **HinSAGE** | 0.72(0.71,0.74) | 0.15(0.14,0.16) |
| **GATNet** | 0.75(0.73,0.78) | 0.17(0.15,0.18) |

## 6. CONCLUSION

This paper describes the initial exploration of heterogeneous graph structure learning for GATNet. We provide an approach that simultaneously learns the heterogeneous graph structure and the GATNet parameters to achieve the prediction target. In particular, by leveraging the intricate relationships among diverse networks, we build and combine feature similarity, feature propagation, and semantic graphs to acquire an ideal heterogeneous graph structure for classification. In addition, this study utilized the mechanics of the graph attention network to develop a more advantageous framework in comparison to the prior baseline technique. We have done comprehensive experiments, which involve node categorization and model analysis, to illustrate the efficiency of the suggested framework.

## REFERENCES


[1] Li, Y., Yang, D., Gong, X. (2022). Patient similarity via medical attributed heterogeneous graph convolutional network. IAENG International Journal of Computer Science, 49(4): 1-10. https://www.iaeng.org/IJCS/issues_v49/issue_4/IJCS_49_4_18.pdf.

[2] Xu, J., Kim, S., Song, M., Jeong, M., Kim, D., Kang, J., Rousseau, J.F., Li, X., Xu, W., Torvik, V.I., Bu, Y., Chen, C., Ebeid, I.A., Li, D., Ding, Y. (2020). Building a PubMed knowledge graph. Scientific Data, 7(1): 205. https://doi.org/10.1038/s41597-020-0543-2

[3] Liu, F., Liu, M., Li, M., Xin, Y., Gao, D., Wu, J., Zhu, J. (2023). Automatic knowledge extraction from Chinese electronic medical records and rheumatoid arthritis knowledge graph construction. Quantitative imaging in medicine and surgery, 13(6): 3873-3890. https://doi.org/10.21037/qims-22-1158

[4] Wanyan, T., Honarvar, H., Azad, A., Ding, Y., Glicksberg, B.S. (2021). Deep learning with heterogeneous graph embeddings for mortality prediction from electronic health records. Data Intelligence, 3(3): 329-339. https://doi.org/10.1162/dint_a_00097

[5] Abu-Salih, B., Al-Qurishi, M., Alweshah, M., Al-Smadi, M., Alfayez, R., Saadeh, H. (2023). Healthcare knowledge graph construction: A systematic review of the state-of-the-art, open issues, and opportunities. Journal of Big Data, 10(1): 81. https://doi.org/10.1186/s40537-023-00774-9

[6] Liu, Z., Li, X., Peng, H., He, L., Philip, S.Y. (2020). Heterogeneous similarity graph neural network on electronic health records. In 2020 IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, pp. 1196-1205. https://doi.org/10.1109/BigData50022.2020.9377795

[7] Chang, H., Zan, H., Zhang, S., Zhao, B., Zhang, K. (2023). Construction of cardiovascular information extraction corpus based on electronic medical records. Mathematical biosciences and engineering, 20(7): 13379-13397. https://doi.org/10.3934/mbe.2023596

[8] Qu, Z., Cui, L., Xu, Y. (2022). Disease risk prediction via heterogeneous graph attention networks. In 2022 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Las Vegas, NV, USA, 2022, pp. 3385-3390. https://doi.org/10.1109/BIBM55620.2022.9995491

[9] Junfeng, Y.A.N., Zhihua, W.E.N., Beiji, Z.O.U. (2022). Heterogeneous graph construction and node representation learning method of Treatise on Febrile Diseases based on graph convolutional network. Digital Chinese Medicine, 5(4): 419-428. https://doi.org/10.1016/j.dcmed.2022.12.007

[10] Gündoğan, E., Kaya, B. (2017). A link prediction approach for drug recommendation in disease-drug bipartite network. In 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya, Turkey, pp. 1-4. https://doi.org/10.1109/IDAP.2017.8090219

[11] Wang, X., Bo, D., Shi, C., Fan, S., Ye, Y., Philip, S.Y. (2022). A survey on heterogeneous graph embedding: Methods, techniques, applications and sources. IEEE Transactions on Big Data, 9(2): 415-436. https://doi.org/10.1109/TBDATA.2022.3177455

[12] Zhang, M., Chen, Y. (2017). Weisfeiler-lehman neural machine for link prediction. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, pp. 575-583. https://doi.org/10.1145/3097983.3097996

[13] Nayyeri, M., Cil, G.M., Vahdati, S., Osborne, F., Rahman, M., Angioni, S., Salatino, A., Recupero, D.R., Vassilyeva, N., Motta, E., Lehmann, J. (2021). Trans4E: Link prediction on scholarly knowledge graphs. Neurocomputing, 461: 530-542. https://doi.org/10.1016/j.neucom.2021.02.100

[14] Lu, H., Uddin, S. (2024). A parameterised model for link prediction using node centrality and similarity measure based on graph embedding. Neurocomputing, 593: 127820. https://doi.org/10.1016/j.neucom.2024.127820

[15] Chen, J., Wang, X., Xu, X. (2022). GC-LSTM: Graph convolution embedded LSTM for dynamic network link prediction. Applied Intelligence, 52(7): 7513-7528. https://doi.org/10.1007/s10489-021-02518-9

[16] Wang, X., Ji, H., Shi, C., Wang, B., Ye, Y., Cui, P., Yu, P.S. (2019). Heterogeneous graph attention network. In WWW '19: The World Wide Web Conference, San Francisco, CA, USA, pp. 2022-2032. https://doi.org/10.1145/3308558.3313562

[17] Amirahmadi, A., Ohlsson, M., Etminani, K. (2023). Deep learning prediction models based on EHR trajectories: A systematic review. Journal of Biomedical Informatics, 144: 104430. https://doi.org/10.1016/j.jbi.2023.104430

[18] Al Hasan, M., Chaoji, V., Salem, S., Zaki, M. (2006). Link prediction using supervised learning. In SDM06: workshop on link analysis, counter-terrorism and security, 30: 798-805. https://www.cs.rpi.edu/~zaki/PaperDir/LINK06.pdf.

[19] Zhao, C., Jiang, J., Guan, Y., Guo, X., He, B. (2018). EMR-based medical knowledge representation and

inference via Markov random fields and distributed representation learning. Artificial Intelligence in Medicine, 87: 49-59. https://doi.org/10.1016/j.artmed.2018.03.005

[20] Li, L., Wang, P., Yan, J., Wang, Y., Li, S., Jiang, J., Sun, Z., Tang, B., Chang, T., Wang, S., Liu, Y. (2020). Real-world data medical knowledge graph: Construction and applications. Artificial Intelligence in Medicine, 103: 101817. https://doi.org/10.1016/j.artmed.2020.101817

[21] Rotmensch, M., Halpern, Y., Tlimat, A., Horng, S., Sontag, D. (2017). Learning a health knowledge graph from electronic medical records. Scientific Reports, 7(1): 5994. https://doi.org/10.1038/s41598-017-05778-z

[22] Johnson, D., Connor, A.J., McKeever, S., Wang, Z., Deisboeck, T.S., Quaiser, T., Shochat, E. (2014). Semantically linking in silico cancer models. Cancer Informatics, 13(S1): 133-143. https://doi.org/10.4137/CIN.S13895

[23] Sun, H., Xiao, J., Zhu, W., He, Y., Zhang, S., Xu, X., Hou, L., Li, J., Ni, Y., Xie, G. (2020). Medical knowledge graph to enhance fraud, waste, and abuse detection on claim data: Model development and performance evaluation. JMIR Medical Informatics, 8(7): e17653. https://doi.org/10.2196/17653

[24] Soulakis, N.D., Carson, M.B., Lee, Y.J., Schneider, D.H., Skeehan, C.T., Scholtens, D.M. (2015). Visualizing collaborative electronic health record usage for hospitalized patients with heart failure. Journal of the American Medical Informatics Association, 22(2): 299-311. https://doi.org/10.1093/jamia/ocu017

[25] Xiu, X., Qian, Q., Wu, S. (2020). Construction of a digestive system tumor knowledge graph based on chinese electronic medical records: Development and usability study. JMIR Medical Informatics, 8(10): e18287. https://doi.org/10.2196/18287

[26] Gündoğan, E., Kaya, B. (2017). A recommendation method based on link prediction in drug-disease bipartite network. In 2017 2nd International Conference on Advanced Information and Communication Technologies (AICT), Lviv, Ukraine, pp. 125-128. https://doi.org/10.1109/AIACT.2017.8020081

## NOMENCLATURE

| | |
|---|---|
| a, b | Random Nodes |
| J(a, b) | Jaccard Coefficient |
| P(a, b) | Preferential attachment |
| C(a, b) | Common neighbors |
| g | graph |
| v | nodes |
| e | edges |
| $u_i$ | Vertex |
| $T_i$ | Linear Transformation |
| $w_o$ | Weight of the Matrix |
| $h_l^{(j)}$ | Node $v_i$ at layer l |
| σ | Activation function |
| $e_{mn}$ | Attention coefficient between nodes $v_m$ and $v_n$ |
| $a_{mn}$ | Shared attention mechanism |