Bioinformatics, 37(18), 2021, 2955–2962
doi: 10.1093/bioinformatics/btab174
Advance Access Publication Date: 14 March 2021
Original Paper

Data and text mining

Polypharmacy side-effect prediction with enhanced interpretability based on graph feature attention network

Sunjoo Bang , Jong Ho Jhee and Hyunjung Shin *

Department of Industrial Engineering, Ajou University, Suwon 443-749, South Korea

*To whom correspondence should be addressed.

Associate Editor: Jonathan Wren

Received on September 2, 2020; revised on March 2, 2021; editorial decision on March 8, 2021; accepted on March 12, 2021

Abstract

Motivation: Polypharmacy side effects should be carefully considered for new drug development. However, considering all the complex drug–drug interactions that cause polypharmacy side effects is challenging. Recently, graph neural network (GNN) models have handled these complex interactions successfully and shown great predictive performance. Nevertheless, the GNN models have difficulty providing intelligible factors of the prediction for biomedical and pharmaceutical domain experts.

Method: A novel approach, graph feature attention network (GFAN), is presented for interpretable prediction of polypharmacy side effects by emphasizing target genes differently. To artificially simulate polypharmacy situations, where two different drugs are taken together, we formulated a node classification problem by using the concept of line graph in graph theory.

Results: Experiments with benchmark datasets validated interpretability of the GFAN and demonstrated competitive performance with the graph attention network in a previous work. And the specific cases in the polypharmacy side-effect prediction experiments showed that the GFAN model is capable of very sensitively extracting the target genes for each side-effect prediction.

Availability and implementation: https://github.com/SunjooBang/Polypharmacy-side-effect-prediction.

Contact: smalsunjoo@ajou.ac.kr or shin@ajou.ac.kr

1 Introduction

Polypharmacy, the use of multiple medications, is a therapeutic approach to treat many diseases, such as heart failure, metabolic syndrome and diabetes (Lien *et al.*, 2002). Because these diseases occur in complex mechanisms, it is effective to target multiple risk factors via polypharmacy (Grundy, 2006). However, side effects of polypharmacy should be carefully considered. Side effects of polypharmacy emerge from drug–drug interactions, in which the activity of one drug may change, favorably or unfavorably, if taken with another drug (Jia *et al.*, 2009; Lehár *et al.*, 2009). However, considering all the complex interactions is challenging, so the machine-learning can be an efficient approach to deal with this problem. Nowadays, the artificial neural network, which is a basis of deep learning, has been used in diverse and complex problems, including pharmaceutical drug-discovery research and drug mono- and combo side-effect prediction (Cao *et al.*, 2018; Shahid *et al.*, 2019; Sutariya *et al.*, 2013).

Also, the polypharmacy side effects prediction problem which is a main problem of our research has been dealt in several research. Zitnik proposed Decagon to predict polypharmacy side effects with a multi-relational link prediction model in multimodal networks (Zitnik *et al.*, 2018). In the model, a graph neural network (GNN) is employed, which includes drugs as nodes and drug–drug interactions as edges in the graph. Each node in the graph has node features representing target genes. The GNN-based convolutional encoder structure trains a model to compress the information into an incomprehensible vector. Also, there are research dealing with the drug–drug interactions in terms of knowledge graph (Malone *et al.*, 2018; Nováček and Mohamed, 2020; Wang *et al.*, 2020). They also commonly used embedding models to represent the graph structured data including entities, and relations. Then, the embedded low-dimensional vector was input to convolution neural network based models for the prediction. However, this embedding process makes the model blind when it comes to explaining the prediction results. To convince the domain experts about the resulting side effects, intelligible factors which significantly contribute to the prediction should be provided.

Interpretable prediction pursuing explainable artificial intelligent (XAI) has been a crucial issue lately in machine-learning research

(Gunning, 2017; Ribeiro *et al.*, 2016; Xu *et al.*, 2015). This especially applies in the field of biomedical informatics because the domain experts such as clinicians or new drug-development researchers can be convinced by the outcomes of the machine learning models when valid grounds are supported (Holzinger *et al.*, 2017; Tjoa and Guan, 2019). Even though the predictive model was revealed to be accurate by performance measures, intelligible factors contributing to the prediction must be provided to enlighten them, and moreover to assist their decision-making.

In this study, we aim to provide interpretable polypharmacy side-effect predictions. The proposed method is based on graph attention networks (GAT), which was developed to deal with graph-represented data (Veličković *et al.*, 2017). But it is not enough to have function of interpretability. Technically, interpretability means that a model provides intelligible factors with outputs, which makes connections between the predictive result and the input features. In the proposed method, input features are assigned differentiated importance, and those of significance are regarded as intelligible factors. This implements interpretability for the predicted results of the model. Hereafter, the proposed model is denoted as Graph Feature Attention Network (GFAN) emphasizing the novel function of interpretability.

Section 2 explains the background and the models of explainable artificial neural network (Holzinger *et al.*) and GNN, which will ease understanding of our model in the following section. Section 3 introduces GFAN. Experiments on polypharmacy side effects are described in Section 4, following validation of GFAN. Section 5 concludes with limitations and future work for this topic.

2 Background

2.1 Explainable ANNs

Generally, it is still challenging to make neural network models to be interpretable because the non-linearity makes the model as a black box. There have been previous works about explaining the neural network in three categories. Gradient-based methods, such as deep learning important features (DeepLIFT) and SHapley Additive exPlanations (SHAP) are used to provide fundamental solution by examining the values in the neural network structure (Lundberg and Lee, 2017; Shrikumar *et al.*, 2017). However, there exist drawbacks, as the problems are limited to the specific functional cases. On In contrast, model-agnostic methods, such as Local Interpretable Model-agnostic Explanations (LIME), and Randomized Input Sampling for Explanation of Black-box Models (RISE), treat the original predictive model as a black box (Petsiuk *et al.*, 2018; Ribeiro *et al.*, 2016). Because they only observe how inputs affect to the outputs after predictions, they can be applied to any machine learning model. Recently, attention mechanism-based methods have stimulated many studies. The attention mechanism was first presented for machine translation with a recurrent neural network encoder-decoder structure (Bahdanau *et al.*, 2014; Xu *et al.*, 2015).

2.2 Explainable GNNs

After the previous trials described above, GAT was developed to deal with the graph-represented data (Veličković *et al.*, 2017). The GAT model performs node classification for a targeted node by attending its neighborhoods' features while assigning different weights to different nodes in a neighborhood, which has opened the possibility of model interpretability. However, it remains at attending neighbor nodes, which are the same level of the targeted prediction, whereas the node features are at a deeper level. Previously, an interpretable graph convolutional neural network, GNN explainer, was available. It is a model-agnostic method for nodes, graph classification and link prediction. GNN explainer used mutual information formulation to measure the importance of each features and presented subgraph and sub-node features (Ying *et al.*, 2019).

3 Materials and methods

To handle the polypharmacy side-effect prediction problem, we proposed novel neural network based model GFAN with enhanced interpretability. Main function of the proposed method GFAN is to interpret the predictive results comparing to the previous general GAT. Shortly, GFAN model can tell which target genes are significantly contributing to the polypharmacy side-effect prediction. In mathematical formulation, portion of target genes (important features) can be highlighted by the δ which is defined by reflecting changes of prediction error when each feature removed step by step.Figure 1 depicts the overall process. We constructed a drug–drug interaction network based on polypharmacy side effects. Then, to artificially simulate polypharmacy situations, where two different drugs are taken together, we combined the two drug nodes into the one polypharmacy node. Finally, the GFAN model predicts the polypharmacy side effect of the two nodes.

3.1 Drug network construction

A drug network $G = (\mathcal{V}, \mathcal{R})$ consists of nodes $v_i \in \mathcal{V}$ indicating drugs, where each node has a node feature vector $\vec{h}_i$ $(\in \mathbb{R}^{1\times F})$ that represents drug target genes, as shown in Figure 1a, where F is the number of target genes. When a drug has a set of target genes, the node feature vector has value 1 at the location corresponding to the target genes and 0 at the others. Two different nodes, v_i and v_j, can be connected by an edge $r_{ij} \in \mathcal{R}$ if they have polypharmacy side effects. This work is a multi-label prediction problem because multiple combo side effects can exist simultaneously.

3.2 Polypharmacy network construction

Polypharmacy side-effect prediction has been considered generally as a link-prediction problem. Because the polypharmacy side effect is represented as an edge in the drug network. In this study, however, we transform the drug network into a polypharmacy network to formulate this problem as a node classification. This strategy is for the enhanced interpretability when the model predicts polypharmacy side effects. To find significant target genes for predicting a certain polypharmacy side effect, we should compare the contributions of them. In the polypharmacy network, the contributions can be measured because the target genes (node features) naturally represent the properties of the polypharmacy (node). Therefore, we transform the drug network to a polypharmacy network by implementing the concept of line graphs in graph theory. A line graph $L(G)$ includes the nodes of graph $L(G)$ are the edges of graph G (Harary and Norman, 1960; Mason and Verwoerd, 2007). That is, polypharmacy edges in the drug network become nodes in the polypharmacy network. Then node features are combined, and the nodes are connected with edges when they have common drugs. For example, as you can see in Figure 1a and b, polypharmacy of the two drugs v_1 and v_3 becomes a new single node $v_{[1,3]}$ with node feature $h_{[1,3]} = [2, 1, 0, 1, 0]$ from the combination of the two node features $h_1 = [1, 1, 0, 0, 0]$ and $h_3 = [1, 0, 0, 1, 0]$. Node $v_{[1,3]}$ has a label of polypharmacy side effect $r_{[1,3]}$. In this way, we can artificially simulate the situation of taking two different drugs together while conjoining the individual drug properties (target genes). Simultaneously, we can facilitate interpretable prediction by explaining each drug's properties in the node features. Sections 3.3 and 3.4 discuss the novel proposed method GFAN, which is used for the node classification problem (Fig. 1c).

3.3 Graph attention network

Previous research on GAT by Petar Veličković et al. (2017) motivated our work. The GAT model performs node classification for a targeted node by attending over their neighborhoods' features while assigning different weights to different nodes in a neighborhood (Veličković *et al.*, 2017). As an expanded research, we propose GFAN to facilitate interpretable prediction by emphasizing node features differently during the model training process. We begin by explaining the structure of GAT. In a layer of the GNN, node feature $h = \{\vec{h}_1, \vec{h}_2, \ldots \vec{h}_N\}, \vec{h}_i \in \mathbb{R}^F$ with feature dimension F is fed

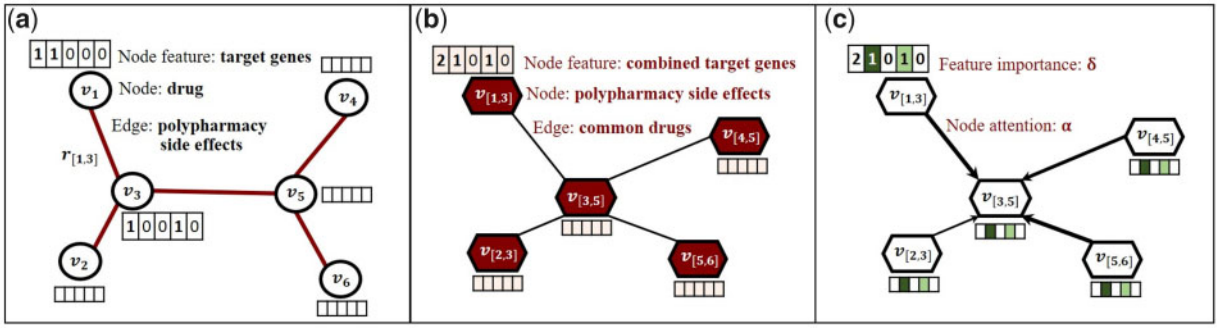

Fig. 1. Overall process: In this study, we propose a novel approach for polypharmacy side-effect prediction with enhanced interpretability based on GFAN. (**a**) Drug network construction: For the first time, we constructed drug–drug interaction network, including polypharmacy side effects information on edges ($r_{[1,3]}$) between drug nodes (v_1, v_3), where each node has node feature of target genes. (**b**) Polypharmacy network construction: Secondly, we transform the drug network to a polypharmacy network by implementing the concept of line graphs in graph theory. Every edge in the drug network becomes nodes in the polypharmacy network. That is, polypharmacy nodes ($v_{[1,3]}$) is created and their node features are combined target genes. And they are connected with edges when they have common drugs. (**c**) Then, the proposed method GFAN predicts the multi-label for the polypharmacy side effects, and extracts features that contribute significantly to the prediction

into the network and new node feature $h' = \left\{\vec{h}'_1, \vec{h}'_2, \ldots \vec{h}'_N\right\}, \vec{h}'_i \in \mathbb{R}^{F'}$ is produced as an layer output. Then, attention function $a : \mathbb{R}^F \times \mathbb{R}^{F'} \rightarrow \mathbb{R}$ is applied to $\mathbf{W}\vec{h}_i$ to obtain the attention coefficients. Let $e_{ij} = a(\mathbf{W}\vec{h}_i, \mathbf{W}\vec{h}_j) = \vec{a}^T[\mathbf{W}\vec{h}_i || \mathbf{W}\vec{h}_j]$ denotes an attention coefficient with learnable linear transformation weight matrix $\mathbf{W} \in \mathbb{R}^{F' \times F}$ and $\vec{a}$ denotes a parameterized weight vector. The normalized attention coefficient using the softmax function is defined as follows:

$$\alpha_{ij} = softmax_i(e_{ij}) = \frac{exp\left(LeakyReLU\left(\vec{a}^T[\mathbf{W}\vec{h}_i || \mathbf{W}\vec{h}_j]\right)\right)}{\sum_{k \in \mathcal{N}_i} exp\left(LeakyReLU\left(\vec{a}^T[\mathbf{W}\vec{h}_i || \mathbf{W}\vec{h}_j]\right)\right)}. \quad (1)$$

As a result, we can have layer output $\mathbf{h}'$ in a shared attentional mechanism, as follows:

$$\vec{h}'_i = \sigma\left(\sum_{j \in \mathcal{N}_i} \alpha_{ij} \mathbf{W}\vec{h}_j\right), \quad (2)$$

where $\mathcal{N}_i$ denotes neighborhoods of node i and σ is a non-linear activation function.

3.4 Graph feature attention network

In GFAN, the model is trained while emphasizing several node features but all features with *feature importance* $\delta = \{\delta_i^k\}$ with datapoint i and feature k ($\delta \in \mathbb{R}^{N \times F}$). The feature importance δ is a sort of masking parameter which is multiplied by original feature matrix H, resulting in the following output representation:

$$\vec{t}'_i = \sigma\left(\sum_{j \in \mathcal{N}_i} \alpha_{ij} \mathbf{W}(\vec{h}_j \circ \delta_{ij})\right) \rightarrow T' = \sigma(\text{Å}(H \circ \delta)W), \quad (3)$$

where $\circ$ represents element-wise multiplication. $H \circ \delta$ refers the modified input such that important features are highlighted while unimportant features are downplayed for prediction. The modified input produces emphasized output T after the next training epochs. To define the feature importance δ, the statistical concept of stepwise feature selection has been borrowed (Steyerberg *et al.*, 1999). That is, we observe the changes in the error when one feature is removed from the original input node feature. If the error increases in comparison with a base result which has the full feature input, the removed feature can be considered as important one for the prediction, and vice versa. To reflect this concept, we calculated the proportion of the errors between E_{-k} and E_0:

$$\delta = \| \, k = 1F\left(\frac{E_{-k}}{E_0}\right), \quad (4)$$

where $\|$ represents concatenation, so we can have a vector of all features. E_0 denotes error which is calculated after a feedforward GAT model with full input node features (a base result). Furthermore, E_{-k} is achieved by the same GAT model (with the same initialization) with input excluding the kth feature. $S^k \in \mathbb{R}^{N \times F}$ indicates matrix of ones with zeros in k_{th} column. The effect of the kth feature is removed by assigning 0 values to the kth column of S. For example,

$$S^1 = \begin{Bmatrix} 0 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{Bmatrix} \text{ in case of } k = 1. \quad (5)$$

The output $\hat{y}_{-k}$ is

$$\hat{y}_{-k} = \sigma\left(\text{Å}(H \circ S^k)W\right), \quad (6)$$

the error E_{-k} and E_0 are

$$E_{-k} = L(\hat{y}_{-k}, y), \; k = \{1, 2, \ldots, F\} \quad (7)$$

and $E_0 = L\left(\sigma(\text{Å}\boldsymbol{HW}), y\right)$. $\hat{y}_{-k}$ is a predicted output while y is an actual output. After calculation of δ from errors, we apply the modified input $\boldsymbol{H} \circ \delta$ to the model again, as follows:

$$E = L(T', y). \quad (8)$$

The error is calculated after every epoch. Categorical cross-entropy loss with softmax output is used for the classification model. In the multi-label classification task (our polypharmacy side effect case), we need to modify the activation function of the model and loss function. With the sigmoid activation function at the output layer, the neural network models the probability of each class as a Bernoulli distribution. We use binary cross-entropy loss, which is $L = -\sum_i^C y_i log(t_i)$, y_i is a ground truth and t_i is the ith element of the output logit vector with C classes, to penalize each output independently. Finally, we define the feature importance score I_k as

$$I_k = \sum_{i=1}^{N} \left(\delta_i^k - 1\right)^2 \; for \; \delta_i^k > 1. \quad (9)$$

δ_i^k represents the feature importance for each class and each data point, and the sum of variance of δ_i^k is expressed as a measure of feature importance score. An important condition for feature importance score is that the feature importance is distinguishable according to the classes. This condition can be reflected by checking the variance of feature importance $\delta_i^k > 1$. A feature with a large variance among feature importance values that are larger than one has the largest feature importance score. Thus, we are able to present the relatively important features from the model in order.

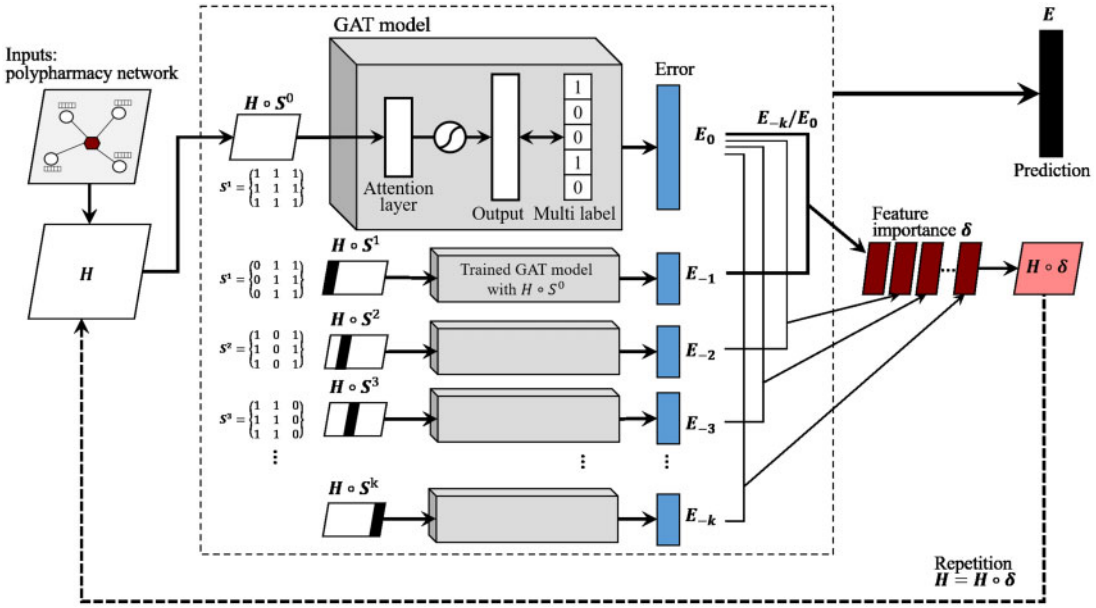

Fig. 2. GFAN architecture: GFAN obtains a subgraph H for a single polypharmacy side effect. H is inputted to a general GAT model to produce error E_0. At the same time, the H is multiplied by S^k to eliminate features one by one ($k = 1, \ldots, F$), then the produced F inputs are inserted into the GAT model to produce F error E_{-k}. Then, we calculate feature importance δ by the proportion of error change E_{-k}/E_0. Then, the modified input $H \circ \delta$ is repeatedly inserted into the procedure to determine feature importance. We defined the epoch inside of the dotted box as the inner epoch, and the repetition (dotted blue arrow) as the outer epoch

4 Experiments

We conducted two types of experiments. Preferentially, we carried out benchmark experiments for model validation. Before applying the proposed method GFAN to the polypharmacy side-effect prediction problem, it is essential to verify whether the model is adequate and reasonable for the interpretable prediction. GFAN was designed for the node classification problem. Therefore, it is proper to validate its performance (for training, and interpretability) with the benchmark datasets for the node classification problem such as IRIS. After the model validation experiment, we applied the model to polypharmacy side-effect prediction experiment. As described in Figure 1b, we changed the whole 'drug network' into one 'polypharmacy network' to consider the polypharmacy as nodes. Therefore, validating GFAN with the node classification datasets is proper for the polypharmacy problem (Fig. 2).

4.1 Benchmark experiments for model validation

For model validation, we used three well-known benchmark data: IRIS, Digit and USPS. IRIS dataset has 150 data points of iris and four features of petal length, petal width, sepal length and sepal width. We constructed IRIS network which has 150 nodes with 4-dimensional node features. To construct the connection between the nodes with edges, we used Gaussian kernel function for similarity measure among 150 data points and made $\mathbb{R}^{150 \times 150}$ similarity matrix. Then, we ranked the similarity values of the pairs in descending order and left top 28% to make sparse network (3130 edges among 11 175 possible edges). In the same vein, we constructed Digit and USPS networks, as shown in Table 1.

4.1.1 Training trend of GFAN using IRIS comparing to the GAT

To validate the GFAN model, we monitored the trend of the training error along with the GAT model which is proved to be well-perform on network data. Generally, if the training error gradually decreases according to the training epoch, the model is considered well-trained. As shown in Figure 3, we plotted training error results in two ways: the averages of experiments repeated 100 times (thick line) and the best cases (thin line) for both GFAN and GAT, which were applied to the IRIS network. The GAT model was trained for 300 epochs with a full batch. The GFAN model was trained with the same initial weights and input; however, unlike that for

Table 1. Network information of benchmark dataset

	IRIS	DIGIT	USPS
Node	150	1500	1500
Node feature	4	241	241
Edge	1490	120 802	118 704
Class	3	2	2

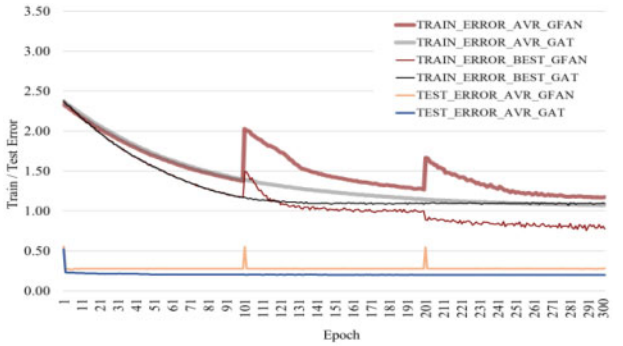

Fig. 3. Training loss for IRIS data classification

GAT, the input was changed every 100 epochs. In more detail, after the GFAN model was trained for 100 epochs (same as the GAT model), the resulting model was trained for another 100 epochs with modified input, which was multiplied by the feature importance. AVR_GAT showed stable decreasing trends. Even though AVR_GFAN showed two picks meaning that the GFAN model was needed to adapt to the modified input, it was clear that the training error of the GFAN model converges to the point near training error of GAT model eventually. Also, test error is shown to be stable for both models. Additionally, we point out that an interesting result was drawn from the one of our experiments (depicted as best case). At the second pick of the BEST_GFAN, the training error decreased dramatically. This means that the GFAN model could escape from

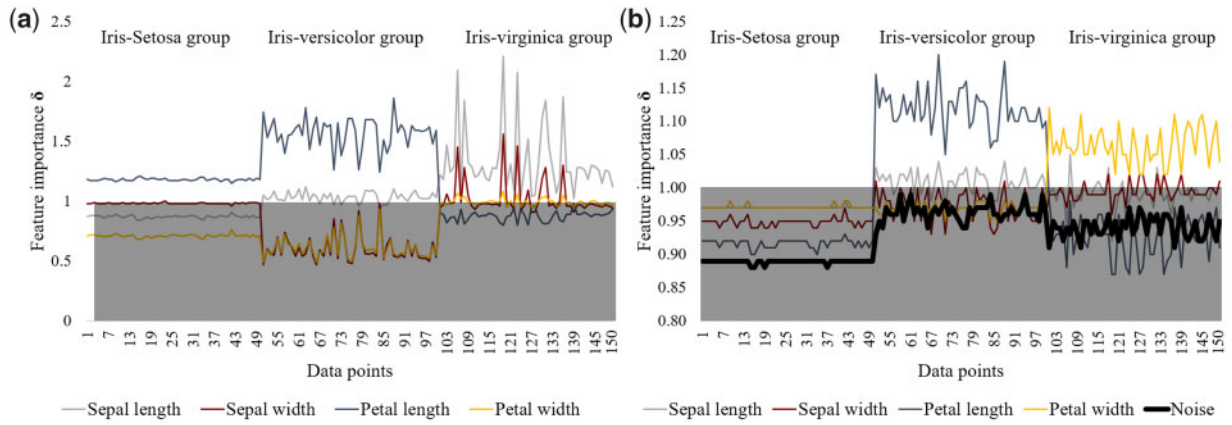

Fig. 4. Feature importance trends for three groups (setosa, versicolor and virginica) of IRIS data

Table 2. Details for the drug network and the polypharmacy network

	Drug network		Polypharmacy network	
Node	Drug	284	Polypharmacy	14 247
Node label	–	–	Polypharmacy side effects	14 247 × 1308
Node feature	Target genes	284 × 3648	Combined target genes of two drugs	14 248 × 3648
Edge	Polypharmacy	14 247 × 2	Relation of common drugs	14 248 × 14 248
Edge label	Polypharmacy side effects	14 247 × 1308	–	–

the local optimum and search for a more optimal solution. This is one of the future works to improve GFAN model as a better node classification problem.

4.1.2 Interpretability of GFAN

Figure 4a shows feature importance of four features in IRIS dataset: sepal length, sepal width, petal length and petal width. We plotted the feature importance δ after second outer epoch. δ are centered around 1 because we defined the feature importance as the ratio of two errors in Equation (4). That is, if the δ is larger than 1, the corresponding feature is relatively more important to the prediction. Reversely, if the δ is equal to or smaller than 1, the corresponding feature is relatively unimportant or less important than other features. As a result, we can see that petal length is the most important feature to classify the data of the setosa group (1–50 data points) from other data. The petal length and sepal length are both important for the data of versicolor group. Lastly, sepal length, sepal width and petal width are revealed to be important for virginica group. Thus, the GFAN model can define feature importance scores per datum, and eventually per class. The feature importance score I of four features for the IRIS dataset is 0.1234 for petal length, 0.0001 for petal width, 0.0629 for sepal length and 0.0068 for sepal width. The petal length was the most important feature for classification among these four features. Figure 4b is the result of a noise experiment conducted by adding one noise feature to the original IRIS dataset. The noise feature was generated by adding Gaussian noise to the original sepal length feature. The importance score of the noise feature is expected to be less than 1, as the noise feature should not affect the prediction of IRIS classes. As a result, we could see that the GFAN model successfully achieved an importance score of less than 1 for all data points of the noise feature (thick black line).

4.1.3 Performance of GFAN for benchmark datasets

To validate the general performance of GFAN model of benchmark datasets, we measured the test accuracy. We performed five cross-validation for each experiment and the test accuracy of the three datasets was 0.9068 for IRIS, 0.8940 for DIGIT and 0.955 for USPS. As a result, we could ensure that the proposed GFAN model is valid for the node classification problem based on the benchmark experiment.

4.2 Polypharmacy side-effect prediction

The goal of this experiments is to predict polypharmacy side effects by applying the proposed GFAN model to the polypharmacy network and to provide intelligible factors which are target genes for the prediction. The problem was formulated as a node classification in implementing line graph concept.

4.2.1 Datasets

The original polypharmacy side effect graph datasets were downloaded from http://snap.stanford.edu/decagon from Zitnik's work. They had been organized after collection from the Side effect Resource (SIDER), OFFSIDES and TWOSIDES databases (Kuhn *et al.*, 2016; Tatonetti *et al.*, 2012). In brief, there are over 4 649 441 combo side effects among 639 drugs. Multiple combo side effects can exist simultaneously among 1308 types of side effects. Also, there are over 18 689 interaction data between drug and target genes, for 284 drugs and 3648 target genes. Practically, among the datasets provided by Zitnik's work, we utilized two datasets 'bio-decagon-combo' and 'bio-decagon-targets' to construct the drug network (see Section 3.1 and Fig. 1a). The details of the constructed drug network are shown in Table 2. The bio-decagon-combo includes polypharmacy (edges) and polypharmacy side effects (edge label) data. And the bio-decagon-targets includes target genes (node feature) data and listed drugs in this file to define nodes in the drug network. Then we transfer this drug network to polypharmacy network (see Section 3.2 and Fig. 1b). Therefore, from here on out, nodes are polypharmacy, node label is polypharmacy side effects. And node features are combined target genes of two drugs as explained in Section 3.2. The two polypharmacy nodes connected when they have common drugs. Therefore, the polypharmacy network which is an input of the GFAN had 14 247 nodes, 3648 node features and 1308 classes. For experiments, we used 80% nodes in the network for training, 10% for validation and 10% for test purposes.

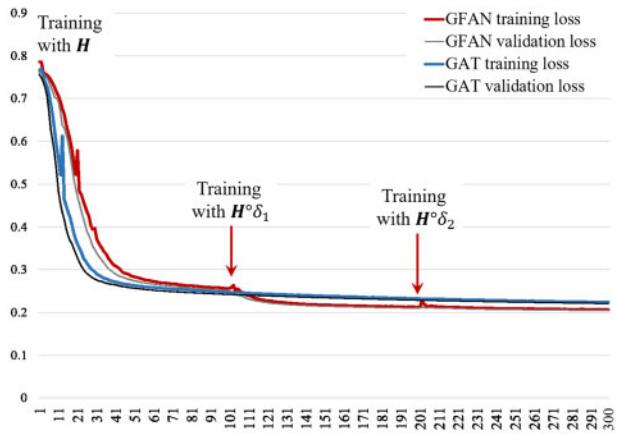

Fig. 5. Training and validation loss for the GFAN and GAT model

4.2.2 Experimental setup

A GFAN model was build based on a general GAT model, which consists of two layers (Veličković *et al.*, 2017). The classification layer followed by one attention layer and the parameters were set as follows: 8 attention heads computes 8 features each (that is total 64 features). Learning rate is 0.001 and dropout rate for each layer is 0.6. After training the general GAT model for 100 epochs, we computed a feature importance δ_1 with the $k + 1$ errors which were computed by applying k inputs eliminating k_{th}feature to the trained model (see Section 3.4). Then we continue to train the model with the changed input $H^{\circ}\delta_1$. We repeated the above process two times so that δ_2 and δ_3 were defined. That is, 100 epochs repeated 3 times (total 300 epochs). For the performance comparison, we trained the general GAT for 300 epochs with the same initial weight. Finally, third δ was used for explaining the predictions.

4.2.3 Performance of GFAN for polypharmacy side-effect prediction

Figure 5 shows training loss and validation loss for the GFAN and GAT model: GFAN training loss (thick red line), GFAN validation loss (thin gray line), GAT training loss (thick blue line) and GAT validation loss (thin black line). We found that the loss got smaller every 100 epochs comparing to the general GAT, when the trained model (up to the previous epoch) continued to be trained with $H^{\circ}\delta_1$ and $H^{\circ}\delta_2$. Eventually, the training and validation loss at 300 epochs were 0.224 and 0.221 for GFAN model, and 0.207 and 0.205 for GAT model.

In multi label classification problem, we can have AUC per multiple classes. In Table 3, side effects which have the highest AUC for the test sets were listed. The last column is the number of polypharmacy (drug pairs) related to the side effects.

4.2.4 Case example

GFAN model defines feature importance related to each classification label, that is polypharmacy side effects. If two pairs of drugs have disparate labels of polypharmacy side effects, the GFAN model could present different sets of target genes. Figure 6 shows how we set up an experiment for this problem. For the specific case, we chose CID 5090 (rofecoxib), which has combo side effects with 466 other drugs. Hereafter, other drugs which have combo side effects with the CID 5090 is denoted as paired drug. First, we calculated the Jaccard distance among the 466 combo side effect vectors ($\in \mathbb{R}^{1317}$). Then, we selected the most different pairs with the largest distance. Simultaneously, we tried to select paired drugs that have a small number of target genes to compare easily influence from the drugs. As shown in Figure 6a, two combo side effects $r_{[2955,\ 5090]}$ and $r_{[1065,\ 5090]}$ among v_{5090}, v_{2955} and v_{1065} (CID 2955; Dapsone and CID 1065: Quinidine, dotted line) are selected. The $r_{[2955,\ 5090]}$ includes 184 combo side effects (back ache, asthma and autonomic

Table 3. AUC per side effects

AUC	Side effects	Side effects name	Number
0.974	C0340274	Pregnancy induced hypertension	50
0.951	C0242786	High risk pregnancy	40
0.946	C0024421	Macroglossia	91
0.943	C0020545	Renovascular hypertension	5
0.943	C0032580	Familial adenomatous polyposis	23
0.916	C0152020	Gastric stasis	36
0.902	C0042376	Vascular headache	7
0.884	C0010930	Dacrocystitis	10
0.874	C0003834	Arterial insufficiency	99
0.841	C0032305	Pneumocystis carinii pneumonia	114

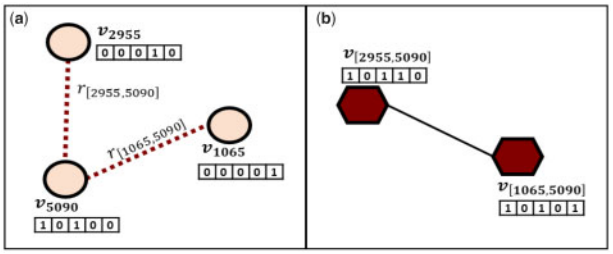

Fig. 6. Illustration of problem definition for a disparate case: (**a**) a drug network including three drug nodes CID 5090, CID 2955 and CID 1065; (**b**) transformed polypharmacy network with two polypharmacy nodes $v_{[2955,\ 5090]}$ and $v_{[1065,\ 5090]}$

neuropathy etc.). $r_{[1065,\ 5090]}$ includes 45 side effects (anxiety, drowsiness and psoriasis etc.). And v_{5090} has 167 target genes and other two nodes (v_{2955} and v_{1065}) have single target gene. As described in Figure 6b, we combined the two pairs of node features into $v_{[2955,\ 5090]}$ and $v_{[1065,\ 5090]}$. $v_{[2955,\ 5090]}$ has 90 neighbor drugs and $v_{[1065,\ 5090]}$ has 186 neighbor drugs. The experimental setup for GFAN model is as follows. Input: compound target genes (637 drugs and one combined node $\in \mathbb{R}^{638 \times 3648}$), label: polypharmacy side effect of combined drugs, inner epoch: 10 000, and outer epoch (feature importance update): 3. Figure 7 shows the resulting feature importance of the specific case. The two resulting feature importance scores are extremely different, even though v_{2955} and v_{1065} each have only one target gene. Therefore, the GFAN model is capable of very sensitively extracting the target genes for each side-effect prediction.

The three most important genes for predicting polypharmacy side effects of $v_{[1065,\ 5090]}$ are Entrez ID 134, 136 and 9283, which are ADORA1, ADORA2B and GPR37L1, respectively. For $v_{[2955,\ 5090]}$, they are Entrez ID 1816, 4987 and 59340, which are DRD5, OPRL1 and HRH4. Table 4 is evidence that the extracted target genes are related to the predicted polypharmacy side effects from the literatures. For example, we found the report about relations between asthma and ADORA2B. they reported that expression of ADORA2B is increased in monocytes obtained from patients with BA and are associated with the generation of CD14posCD209pos pro-inflammatory cells. A positive correlation between expression of ADORA2B and IL-6 was identified in human monocytes and may explain the increased expression of IL-6 mRNA in asthmatics.

5 Conclusion

In this study, we proposed GFAN, a novel model for interpretable prediction of polypharmacy side effects. This model provides important features that significantly contribute to prediction. Technically, to artificially simulate polypharmacy situation, we used the line graph concept in graph theory. The significant features, which are target genes in this case, are intelligible to convince the

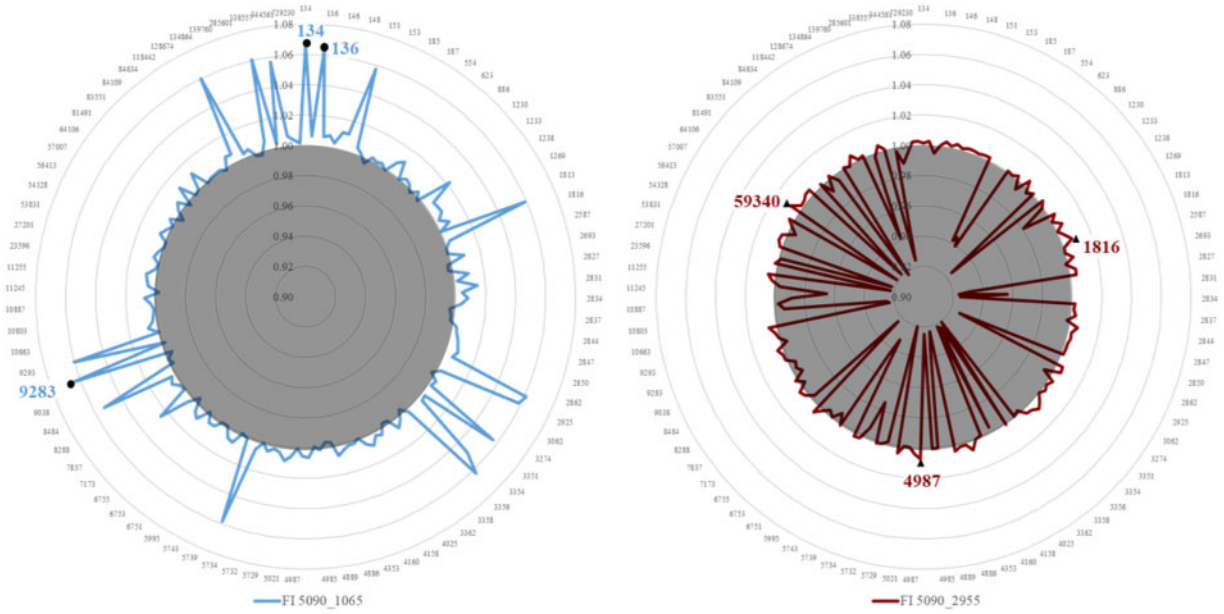

Fig. 7. Case results of target gene importance δ for the two polypharmacy side-effect prediction cases: Blue line is for the combo side effects of 5090 and 1065 drugs. Red line is for the combo side effects of 5090 and 2955 (Entrez gene id)

Table 4. Evidence for the presented target genes

CID	CID	Side effect	Entrez ID	Reference
5090 (rofecoxib)	2955 (dapsone)	Back ache	134 (ADORA1)	–
		Asthma	136 (ADORA2B)	Yuryeva *et al.* (2015)
		Autonomic neuropathy	9283 (GPR37L1)	Liu *et al.* (2017)
5090 (rofecoxib)	1065 (Quinidine)	Anxiety	1816 (DRD5)	Tahir *et al.* (2000)
		Drowsiness	4987 (OPRL1)	Le Merrer *et al.* (2009)
		Psoriasis	59340 (HRH4)	Gschwandtner *et al.* (2011)

domain experts about the resulting polypharmacy side effects. The experiment with benchmark datasets reveals that GFAN had a performance comparable to that of GAT in a previous work. This implies that the GFAN architecture can be more widely used as it exhibits a superior prediction performance and reasoning function for the prediction. GFAN provides an importance value for individual data points. This indicates that each of the polypharmacy side effects is explained with different target genes. This makes the model more beneficial for diverse domain problems. Therefore, an experiment with the specific cases shows that the GFAN model is capable of very sensitively extracting the target genes for each side-effect prediction.

Limitations and further research are recommended as follows. One of the limitations of our study is that we only considered side effects for two drugs. However, patients are prescribed more than 2 drugs in many cases and that is very important. When we implement the line graph concept as mentioned in method section, modeling polypharmacy side-effect prediction for more than 2 drugs seems quite possible by combining more than 2 drugs into one polypharmacy node. However, labeling the combined node with polypharmacy side effects for more than 2 drugs is practically difficult and even impossible in our datasets. If we can have enough labels, the model could be more practical in real world. And the GFAN model which is a wrapper approach of feature selection method was predestined to have heavy time complexity. Also, the process of defining important features takes more than F times over general GAT model (F is the number of features). Finally, the quantitative evaluation results must be developed so that the predicted interpretability can be supported. Especially, in this specific case, validating the risk factors causing the side effects are very difficult and requires careful approach.

Also, the node feature can be diversified with drug properties, such as classification of health, physical and environmental hazard data from globally harmonized system of classification and labeling of chemicals (GHS), and anatomical therapeutic chemical (ATC) data from the WHO. Diverse features will enrich the interpretability of side-effect prediction as well as candidate drug–drug relationships for new drug development. lastly, the methodology of the GFAN model can be improved to make it more stable from the modified inputs to reach to the optimal solutions more quickly.

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT [2017R1E1A1A03070345]. Also, this research was supported by a grant [21182MFDS265] from Ministry of Food and Drug Safety in 2021 and the Ajou University research fund.

Conflict of Interest: none declared.

References

Bahdanau,D. *et al.* (2014) Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*

Cao,C. *et al.* (2018) Deep learning and its applications in biomedicine. *Genomics Proteomics Bioinf.*, **16**, 17–32.

Grundy,S.M. (2006) Drug therapy of the metabolic syndrome: minimizing the emerging crisis in polypharmacy. *Nat. Rev. Drug Discov.*, **5**, 295–309.

Gschwandtner,M. *et al.* (2011) The histamine H4 receptor is highly expressed on plasmacytoid dendritic cells in psoriasis and histamine regulates their

cytokine production and migration. *J. Investig. Dermatol.*, **131**, 1668–1676.
Gunning,D. (2017) Explainable artificial intelligence (XAI). *Defense Adv. Res. Projects Agency (DARPA)*, **2**, 2017.
Harary,F. and Norman,R.Z. (1960) Some properties of line digraphs. *Rendiconti del Circolo Matematico di Palermo*, **9**, 161–168.
Holzinger,A. *et al.* (2017) What do we need to build explainable AI systems for the medical domain? *arXiv preprint arXiv:1712.09923*.
Jia,J. *et al.* (2009) Mechanisms of drug combinations: interaction and network perspectives. *Nat. Rev. Drug Discov.*, **8**, 111–128.
Kuhn,M. *et al.* (2016) The SIDER database of drugs and side effects. *Nucleic Acids Res.*, **44**, D1075–D1079.
Le Merrer,J. *et al.* (2009) Reward processing by the opioid system in the brain. *Physiol. Rev.*, **89**, 1379–1412.
Lehár,J. *et al.* (2009) Synergistic drug combinations tend to improve therapeutically relevant selectivity. *Nat. Biotechnol.*, **27**, 659–666.
Liu,B. *et al.* (2017) Astroglia as a cellular target for neuroprotection and treatment of neuro-psychiatric disorders. *Glia*, **65**, 1205–1226.
Lundberg,S.M. and Lee,S.-I. (2017) A unified approach to interpreting model predictions. *Adv. Neural Inf. Process. Syst.*, 4765–4774.
Malone,B. *et al.* (2018) Knowledge graph completion to predict polypharmacy side effects. In: *International Conference on Data Integration in the Life Sciences*. Springer, Berlin. pp. 144–149.
Mason,O. and Verwoerd,M. (2007) Graph theory and networks in biology. *IET Syst. Biol.*, **1**, 89–119.
Nováček,V. and Mohamed,S.K. (2020) Predicting polypharmacy side-effects using knowledge graph embeddings. *AMIA Summits Transl. Sci. Proc.*, **2020**, 449–458.
Petsiuk,V. *et al.* (2018) Rise: randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*.
Ribeiro,M.T. *et al.* (2016) "Why should i trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. pp. 1135–1144.
Shahid,N. *et al.* (2019) Applications of artificial neural networks in health care organizational decision-making: a scoping review. *PLoS One*, **14**, e0212356.
Shrikumar,A. *et al.* (2017) Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning*, Vol. 70. JMLR.org. pp. 3145–3153.
Steyerberg,E.W. *et al.* (1999) Stepwise selection in small data sets: a simulation study of bias in logistic regression analysis. *J. Clin. Epidemiol.*, **52**, 935–942.
Sutariya,V. *et al.* (2013) Artificial neural network in drug delivery and pharmaceutical research. *Open Bioinf. J.*, **7**, 49–62.
Tahir,E. *et al.* (2000) Association and linkage of DRD4 and DRD5 with attention deficit hyperactivity disorder (ADHD) in a sample of Turkish children. *Mol. Psychiatry*, **5**, 396–404.
Tatonetti,N.P. *et al.* (2012) Data-driven prediction of drug effects and interactions. *Sci. Transl. Med.*, **4**, 125ra31.
Tjoa,E. and Guan,C. (2019) A survey on explainable artificial intelligence (xai): towards medical xai. arXiv preprint arXiv:1907.07374.
Veličković,P. *et al.* (2018) Graph attention networks. arXiv preprint arXiv: 1710.10903, *ICLR, 1-12*.
Wang,R. *et al.* (2020) Predicting polypharmacy side effects based on an enhanced domain knowledge graph. In: *International Conference on Applied Informatics*. Springer, Berlin. pp. 89–103.
Xu,K. *et al.* (2015) Show, attend and tell: neural image caption generation with visual attention. In: *International Conference on Machine Learning*. pp. 2048–2057.
Ying,R. *et al.* (2019) Gnn explainer: a tool for post-hoc explanation of graph neural networks. *arXiv preprint arXiv:1903.03894*.
Yuryeva,K. *et al.* (2015) Expression of adenosine receptors in monocytes from patients with bronchial asthma. *Biochem. Biophys. Res. Commun.*, **464**, 1314–1320.
Zitnik,M. *et al.* (2018) Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics*, **34**, i457–i466.