omicsGAT: Graph Attention Network for Cancer Subtype Analyses

Sudipto Baul, Khandakar Tanvir Ahmed, Joseph Filipek and Wei Zhang

Department of Computer Science, Genomics and Bioinformatics Cluster, University of Central Florida, Orlando, FL 32816, USA

Motivation: The use of high-throughput omics technologies is becoming increasingly popular in all facets of biomedical science. The mRNA sequencing (RNA-seq) method reports quantitative measures of more than tens of thousands of biological features. It provides a more comprehensive molecular perspective of studied cancer mechanisms compared to traditional approaches. Graph-based learning models have been proposed to learn important hidden representations from gene expression data and network structure to improve cancer outcome prediction, patient stratification, and cell clustering. However, these graph-based methods cannot rank the importance of the different neighbors for a particular sample in the downstream cancer subtype analyses. In this study, we introduce omicsGAT, a graph attention network (GAT) model to integrate graph-based learning with an attention mechanism for RNA-seq data analysis. The multi-head attention mechanism in omicsGAT can more effectively secure information of a particular sample by assigning different attention coefficients to its neighbors.

Results: Comprehensive experiments on The Cancer Genome Atlas (TCGA) breast cancer and bladder cancer bulk RNA-seq data, and primary diffuse gliomas single-cell RNA-seq data validate that (1) the proposed model can effectively integrate neighborhood information of a sample and learn an embedding vector to improve disease phenotype prediction, cancer patient stratification, and cell clustering of the sample. (2) The attention matrix generated from the multi-head attention coefficients provides more useful information compared to the sample correlation-based adjacency matrix. From the results, we can conclude that some neighbors play a more important role than others in cancer subtype analyses of a particular sample based on the attention coefficient.

Availability and implementation: Source code is available at: https://github.com/CompbioLabUCF/omicsGAT

Supplementary information: Supplementary data are available at *BioRxiv* online.

Graph Attention Network | Single-cell RNA-seq | Patient Stratification | Cancer Outcome Prediction

Correspondence: ***wzhang.cs@ucf.edu***

Introduction

Cancer is a complex and heterogeneous disease with hundreds of types and subtypes spanning across different organs, tissues and have origins in various cell types (1, 2). For example, breast cancer is highly heterogeneous with different subtypes that lead to varying clinical outcomes including prognosis, response to treatment, and changes of recurrence and metastasis (3–5). Hence, cancer subtype prediction and cancer patient stratification have been the subject of interest to clinicians and patients for many decades. Powered by the high-throughput genomic technologies, the mRNA sequencing (RNA-seq) method is capable of measuring transcriptome-wide mRNA expressions and molecular activities in cancer cells (6, 7). Bulk RNA-seq data provides a view of the average gene expression level of an entire tissue sample instead of differentiating among cell types within the sample. Whereas, single-cell RNA-seq (scRNA-seq) provides opportunities to explore gene expression profiles at the single-cell level. These will enable predicting the changes of expression level at a large scale so as to better understand the biological mechanism that leads to cancer.

The high-throughput RNA-seq datasets show quantitative measures of more than tens of thousands of mRNA isoforms for a cohort of hundreds or thousands of samples (e.g., patients, cells). However, due to the unavoidable sample heterogeneity or experimental noise in the data, extracting biological valuable information and discovering the underlying patterns from the data is becoming a serious challenge to computational biologists (8). While hundreds of computational methods have been developed for cancer subtype prediction/identification (9, 10) and patient stratification (11) using RNA-seq data (12), network analysis of sample similarities has largely been ignored in most methods. Graph-based neural network (GNN) and network-based embedding models recently have shown remarkable success in learning network topological structures from large-scale biological data (13–15). On another note, the self-attention mechanism has been extensively used in different applications including bioinformatics (16–18). This mechanism allows inputs to interact with each other and permits the model to utilize the most relevant parts of the inputs to improve the performance of the deep learning models. The self-attention mechanism was combined with the graph-structured data by Veličković et al. (19) in Graph Attention Networks (GAT). This GAT model calculates the representation of each node in the network by attending to its neighbors, and it uses the multi-head attention to further increase the representation capability of the model (20). It applies varied attentions to the neighbors; therefore, find the most important neighbors of a sample rather than giving all of them the same importance. This model has been successfully applied on various tasks including text classification (21), node classification (22), social influence analysis (23), recommendation system (24), etc. The GAT model has also been applied to bioinformatics applications including drug-target interaction prediction (25), drug-microbe interaction prediction (26), gene essentiality prediction (27), etc.

Inspired by the GAT for capturing node dependencies in a wide range of domains, we proposed omicsGAT model and applied it on cancer samples with RNA-seq data. First, we introduced the model in Methods section. Next, we tested omicsGAT on The Cancer Genome Atlas (TCGA) breast invasive carcinoma (BRCA) data collections (28) and urothelial bladder carcinoma (BLCA) data collections (29) for cancer subtype prediction and cancer patient stratification, respectively (Section F). Then, omicsGAT was applied on 2,458 cells from six primary diffuse gliomas with K27M histone mutations (H3K27M) for cell clustering (Section G). Last, we discussed and interpreted the results based on the sample-by-sample attention matrix generated from the omicsGAT model in the Discussion section.

Methods

In this section, we first introduce our proposed framework, omicsGAT, which generates embeddings from gene expression data to be used in downstream classification and clustering. We extended the GAT model (19) to better fit our tasks of disease outcome prediction and subtype stratification. Then, we discuss the baseline models used to compare and validate the performance of omicsGAT followed by the details of evaluation metrics used in this study.

A. Graph Attention Network. The omicsGAT model architecture builds on the concept of the self-attention mechanism. In omicsGAT, embedding is generated from the gene expression data assuming that the samples (i.e., patients or cells) with similar features (gene expressions) are expected to have similar disease outcomes or cell types, and connected to each other. Hence, network information is injected into the model using the adjacency matrix to take these connections into consideration. However, all connected neighbors of a target sample should not get equal attention in generating the embedding for that sample. A particular neighbor of a target sample can contribute more to its subsequent prediction or clustering which cannot be accurately apprehended by similarity metrics. Therefore, to capture the importance of each neighbor on a sample, the omicsGAT model automatically assigns different attentions to the neighbors of that sample for singular head while generating the embedding. Moreover, to consider the impact of different types of information secured from the neighbors and stabilize the learning process, the above procedure is repeated multiple times in parallel employing several heads (independent attention mechanisms) in a multi-head framework.

The mathematical notations used to explain omicsGAT are summarized in Table 1. Let n be the number of samples (e.g., patients, cells) and m be the number of features (e.g., genes) representing each sample. The input feature matrix is given by $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n]$, where $\boldsymbol{x} \in \mathbb{R}^{1\times m}$ represents a sample vector. $\boldsymbol{A}$ be the $n \times n$ adjacency matrix (includes self-connections) built based on the pairwise correlation between the samples. Suppose, the set of neighbors for a sample $\boldsymbol{x}_i$ is denoted by $\mathcal{N}_i$. Depending on the number of neighbors $|\mathcal{N}_i|$ to be kept for a sample, the connections with high correlation scores are kept (assigned a value of 1) and the others are discarded (assigned a value of 0). The adjacency matrix is binarized as it will be used to mask the attention coefficients in later part of the model. Self-connections are applied to integrate the information from the samples themselves in their embeddings. While generating the embedding of sample $\boldsymbol{x}_i$, the attention given to it from its neighbor $\boldsymbol{x}_j$ for a single head can be calculated as

Table 1. Mathematical notations for omicsGAT

Name	Definition
n	number of samples (i.e., patients or cells)
m	number of features (i.e., genes)
p	embedding size for a single head
h	number of heads
$\boldsymbol{X} \in \mathbb{R}^{n\times m}$	input feature matrix
$\boldsymbol{A} \in \mathbb{R}^{n\times n}$	correlation-based adjacency matrix of samples
$\boldsymbol{W} \in \mathbb{R}^{m\times p}$	weight matrix of a single head
$\boldsymbol{a} \in \mathbb{R}^{2p\times 1}$	attention weight matrix of a single head
$\boldsymbol{\alpha} \in \mathbb{R}^{n\times n}$	attention coefficients of a single head
$\boldsymbol{Z} \in \mathbb{R}^{n\times ph}$	embedding matrix learned from the model

$$c_{ij} = \boldsymbol{a}^T[\boldsymbol{W}\boldsymbol{x}_i||\boldsymbol{W}\boldsymbol{x}_j] \qquad \textbf{(1)}$$

where $\boldsymbol{W} \in \mathbb{R}^{p\times m}$ and $\boldsymbol{a} \in \mathbb{R}^{2p\times 1}$ are learnable weight parameters of a single head which are shared across all the samples and p is the embedding size. $||$ and $.^T$ symbols denote the concatenation and transposition operations of the matrices respectively. The calculated attention values are passed through a *LeakyReLU* activation function. Then the structural information of the network is introduced by masking the attention values using the adjacency matrix. Only the attention values where a connection is present between the nodes (samples) in the adjacency matrix $\boldsymbol{A}$ are kept and all the remaining values are made zero. After that, the attention coefficient for a neighbor $\boldsymbol{x}_j$ is calculated using *Softmax* function which follows the equation below:

$$\alpha_{ij} = \frac{\exp(\mathit{LeakyReLU}(\boldsymbol{a}^T[\boldsymbol{W}\boldsymbol{x}_i||\boldsymbol{W}\boldsymbol{x}_j]))}{\sum_{r\in\mathcal{N}_i}\exp(\mathit{LeakyReLU}(\boldsymbol{a}^T[\boldsymbol{W}\boldsymbol{x}_i||\boldsymbol{W}\boldsymbol{x}_r]))} \qquad \textbf{(2)}$$

The attention coefficients calculated for all of the neighbors of $\boldsymbol{x}_i$ using equation (2) are leveraged to calculate its final embedding for a single head

$$\boldsymbol{x}'_i = \sigma(\sum_{j\in\mathcal{N}_i}\alpha_{ij}\boldsymbol{W}\boldsymbol{x}_j) \qquad \textbf{(3)}$$

where σ is a non-linear activation function. Note that the sample $\boldsymbol{x}_i$ is also included in its neighbors since self-connections are used in the adjacency matrix.

In a multi-head attention network, each head has a separate attention mechanism with its own weight matrix $\boldsymbol{W}$ and attention vector $\boldsymbol{a}$. Outputs generated by all the heads for one particular sample are concatenated to generate the final embedding vector of that sample. This is done to stabilize the learning process while generating the embedding. It is similar to the mechanism used by Vaswani et al. (16) in self-attention. Hence, the output embedding from the first part of

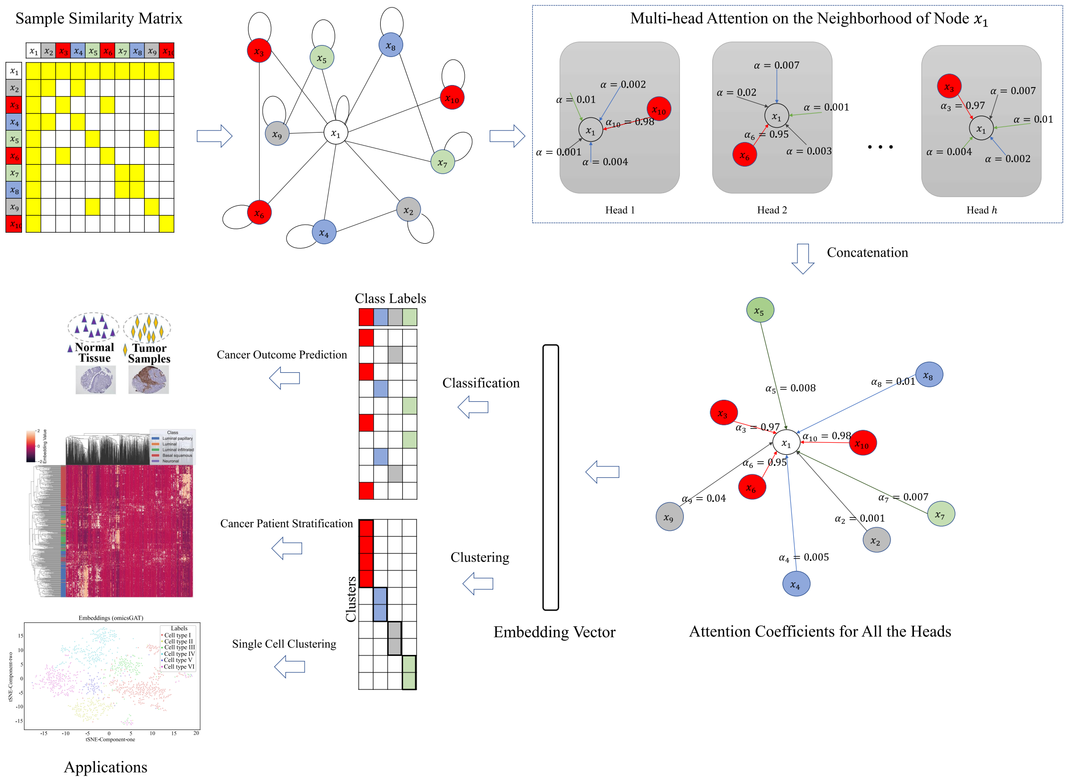

Fig. 1. Workflow of omicsGAT. For a sample $\boldsymbol{x}_1$, based on the input feature matrix and adjacency matrix, each head calculates the attention given to $\boldsymbol{x}_1$ from its neighbors separately. The embeddings produced by all heads are concatenated together to generate the final embedding for $\boldsymbol{x}_1$ which is then used for classification or clustering of $\boldsymbol{x}_1$.

our model for $\boldsymbol{x}_i$ is given by:

$$\boldsymbol{z}_i = \|_{k=1}^{h} \sigma\Big(\sum_{j \in \mathcal{N}_i} \alpha_{ij}^{k} \boldsymbol{W}^{k} \boldsymbol{x}_j\Big) \tag{4}$$

where h is the number of heads. The output projected in the embedding space is represented by $\boldsymbol{Z} \in \mathbb{R}^{n \times ph}$ and embedding for one sample is $\boldsymbol{z} \in \mathbb{R}^{1 \times ph}$. The generated embeddings are then used in separate models for classification and clustering tasks. The overall framework of our proposed pipeline is illustrated in Figure 1.

B. omicsGAT Classifier.

omicsGAT Classifier is a unified model that passes the embedding $\boldsymbol{Z}$ generated from the first part of our pipeline described in Section A through three fully connected (FC) layers. Let the number of classes for the classification task be c. The first two layers converts $\boldsymbol{Z} \in \mathbb{R}^{n \times ph}$ to $\boldsymbol{Z}_{cls_1} \in \mathbb{R}^{n \times \frac{ph}{2}}$ and then to $\boldsymbol{Z}_{cls_2} \in \mathbb{R}^{n \times 2c}$ respectively. The output layer transforms $\boldsymbol{Z}_{cls_2}$ into $\boldsymbol{Y}_{cls} \in \mathbb{R}^{n \times c}$, where $\boldsymbol{Y}_{cls} = [\boldsymbol{y}_{cls_1}, \boldsymbol{y}_{cls_2}, ..., \boldsymbol{y}_{cls_n}]$ represent the classification outcomes. Each layer can be formulated as

$$\boldsymbol{Z}_{cls} = \sigma(\boldsymbol{W}_{cls} \boldsymbol{Z}_{in} + \boldsymbol{b}_{cls}) \tag{5}$$

where $\boldsymbol{Z}_{cls}$ and $\boldsymbol{Z}_{in}$ are the output and input matrices, $\boldsymbol{W}_{cls}$ is the learnable weight, and $\boldsymbol{b}_{cls}$ is the bias vector of a particular layer. σ denotes the activation function which is *ReLU* for the first two layers and *Softmax* for the output layer.

Let the ground truth labels for n samples be $\boldsymbol{Y} = [y_{in_1}, y_{in_2}, ..., y_{in_n}]$. In order to calculate the overall loss of the model, Negative Log Likelihood (NLL) loss function is applied, formulated as follows:

$$\mathcal{L}_{cls} = -\sum_{i=1}^{n} \log(\mathit{Likelihood}(\boldsymbol{y}_{cls_i} | y_{in_i})). \tag{6}$$

$\mathcal{L}_{cls}$ is minimized to train the unified omicsGAT Classifier framework.

C. omicsGAT Clustering.

For clustering, we propose a two-step omicsGAT Clustering framework. The first step is an autoencoder that generates the gene expression embedding in an unsupervised approach, and the second step is a hierarchical clustering model. omicsGAT described in Section A serves as the encoder in the autoencoder architecture whereas a four layers fully connected neural network is constructed as the decoder. The output $\boldsymbol{Z} \in \mathbb{R}^{n \times ph}$ from the omicsGAT encoder is fed into the first layer of the decoder. The output of the consecutive FC layers are $\boldsymbol{Z}_{clr_1} \in \mathbb{R}^{n \times \frac{ph}{2}}$, $\boldsymbol{Z}_{clr_2} \in \mathbb{R}^{n \times \frac{m}{4}}$, $\boldsymbol{Z}_{clr_3} \in \mathbb{R}^{n \times \frac{m}{2}}$, and $\boldsymbol{Y}_{clr} \in \mathbb{R}^{n \times m}$ respectively. Each layer can be formulated as

$$\boldsymbol{Z}_{clr} = \sigma(\boldsymbol{W}_{clr} \boldsymbol{Z}_{in} + \boldsymbol{b}_{clr}) \tag{7}$$

where $\boldsymbol{Z}_{clr}$ and $\boldsymbol{Z}_{in}$ are the output and input matrices respectively, $\boldsymbol{W}_{clr}$ is the learnable weight, and $\boldsymbol{b}_{clr}$ is the bias vector of a particular layer of the decoder. For the first three layers, σ denotes the activation function *ReLU*, and no activation function is used in the final layer.

The output, projected back to the input feature space by the decoder, is given by $\boldsymbol{Y}_{clr} = [\boldsymbol{y}_{clr_1}, \boldsymbol{y}_{clr_2}, ..., \boldsymbol{y}_{clr_n}]$. Mean squared error (MSE) is employed to calculate the reconstruction loss as follows:

$$\mathcal{L}_{clr} = \sum_{i=1}^{n} (\boldsymbol{x}_i - \boldsymbol{y}_{clr_i})^2. \tag{8}$$

$\mathcal{L}_{clr}$ is minimized to train the autoencoder and an embedding is generated as output from the trained encoder. The embedding is then fed into the second step of omicsGAT Clustering, a hierarchical clustering model implemented using the *scikit-learn* package (30). It stratifies the input samples into the defined number of clusters by assigning each sample to a group based on the similarity of the generated embedding of that sample with that of the other samples in the group.

D. Baseline Models used for Comparison.

D.1. Baselines for Classification Tasks. Support Vector Machine (SVM), Random Forest (RF), Deep Neural Network (DNN), and Graph Convolutional Network (GCN) are used as baselines to evaluate and compare the performance of omicsGAT Classifier. The baselines are built using several Python open-source library packages including *Scikit-learn* (30) and *Pytorch* (31).

SVM and RF are two of the most widely used machine learning models. In this study, ‘rbf’ kernel is applied for SVM. Hyperparameters for RF, including the number of trees, split criterion, maximum depth of the tree, maximum number of features considered for split, are also tuned. The Deep Neural Network model consists of three fully connected linear layers with first two of them followed by the *ReLU* activation function. For better evaluation of our model by comparing it to a similar graph-based deep learning model, we follow the Graph Convolution Network (GCN) proposed by Kipf and Welling (32). The GCN model is composed of four graph convolution layers. The correlation-based adjacency matrix $\boldsymbol{A}$ is used as neighborhood information in the GCN model. The hyperparameters for all of these models were tuned on the validation set using grid search.

D.2. Baselines for Clustering Tasks. To evaluate the embedding learned from omicsGAT, we use the clustering results of raw features and their PCA components as baselines. Hierarchical and k-means clusterings are employed for the baselines, i.e., components learned using PCA or the raw features are fed into the clustering models as input. Moreover, for a better interpretation of our model, the attention coefficients from each of the heads are extracted to build up the attention matrix which will be described in the Discussion section. The correlation-based adjacency matrix $\boldsymbol{A}$ is used as baseline to evaluate the attention matrix. Hierarchical clustering is applied on the attention matrix and adjacency matrix, both of which represent the relation among the samples.

E. Evaluation Metrices. In this section, we define three evaluation metrics used in this study implemented using the *scikit-learn* library of Python. The Area Under the Receiver Operating Characteristic Curve (AUC) is used for comparison of the classification models. It is defined as the area under the curve plotted using True Positive Rate (*precision*) along the y-axis and False Positive Rate (1-*specificity*) along the x-axis. The Normalized Mutual Information (NMI) and Adjusted Rand Index (ARI) are applied to evaluate the clustering methods both of which have a range from 0 to 1, where 1 means perfect clustering and 0 means totally random.

Experiments

We carried out experiments on TCGA RNA-seq datasets and H3K27M gliomas scRNA-seq data to evaluate the performance of omicsGAT in this section. In the first part, we performed experiments with omicsGAT for cancer outcome prediction on TCGA breast cancer dataset and cancer patient stratification on TCGA bladder cancer dataset (Section F). In the later part, omicsGAT was applied on scRNA-seq data for single cell clustering analysis (Section G).

F. Experiments on TCGA Cancer Patient Samples.

F.1. Datasets and Preprocessing. The proposed framework, omicsGAT, was tested on TCGA breast invasive carcinoma (BRCA) (28) and urothelial bladder carcinoma (BLCA) (29) datasets. The RNA-seq mRNA expression dataset of each cancer type was downloaded from UCSC Xena Hub (33). $log2(x+1)$ transformed mRNA expression was used in the analyses. The clinical information of the two cancer studies was downloaded from cBioPortal (34). The BRCA dataset consists of 411 patient samples and 20,351 genes for each sample. Similarly, the BLCA dataset consists of 426 patient samples and 20,531 genes for each sample.

Table 2. The classification performance on TCGA breast cancer (BRCA) dataset. The mean AUROC scores and standard deviation (SD) of classifying patients in breast cancer subtypes are reported. *Denotes the difference between the results of omicsGAT and baseline method to be statistically significant (p-*value* < 0.001)

Cancer Subtype	Method	AUC score	SD
ER	SVM	0.9155	0.4868
	Random Forest	0.9206	0.0436
	DNN	0.8705*	0.0586
	GCN	0.8835*	0.0612
	omicsGAT	**0.9407**	**0.0360**
TN	SVM	0.8800*	0.0722
	Random Forest	0.8567*	0.0663
	DNN	0.8226*	0.0819
	GCN	0.8560*	0.1023
	omicsGAT	**0.9368**	**0.0375**

F.2. omicsGAT Improved Overall Cancer Outcome Prediction. We designed two tasks on TCGA BRCA mRNA expression data to evaluate the performance of omicsGAT Classifier on cancer outcome prediction. There are 331 Estrogen Receptor positive (ER+) and 80 ER negative (ER-) samples, 65 Triple-negative (TN) and 346 non-TN samples in the dataset. The two tasks were to predict the ER and TN statuses of the breast cancer patients. omicsGAT Classifier was compared with SVM, RF, DNN, and GCN. First, the dataset was divided into pre-train and test set containing 80% and 20% of the total samples respectively. Then the pre-train set was divided into training and validation set containing 80% and 20% samples of the pre-train set respectively. The hyperparameters of the proposed model used in these two tasks are listed in Supplementary Table S1. They were selected through grid search on the validation set. The same validation set was also applied to select the best model for DNN and GCN. We ran omicsGAT Classifier and baseline methods with above mentioned dataset splitting 50 times. The average AUROC scores for both omicsGAT and baseline methods are reported in Table 2. As we can see, our proposed model outperforms all the baselines for both ER and TN status predictions. Moreover, the gain in AUROC caused by omicsGAT is significant for all baselines, except SVM and RF for ER prediction. omicsGAT Classifier also offers a lower standard deviation which signifies a more consistent and stable prediction compared to the baselines. The stability of our proposed model can be pertained to the use of several heads which can secure information from different directions and the model can effectively combine them by learning distinct attention parameters for each head.

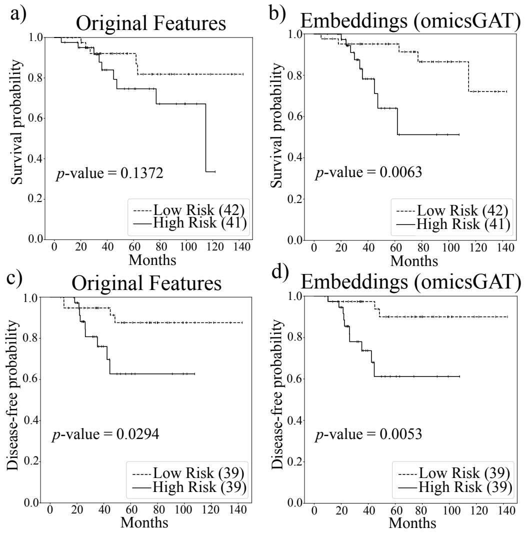

Fig. 2. Survival and disease-free time predictions on breast cancer patients with original gene expression and the embeddings generated by omicsGAT. Kaplan-Meier plots for low (dashed line) and high (solid line) risk groups generated by a) original gene expression and b) omicsGAT learned embeddings for survival analysis; c) original gene expression and d) omicsGAT learned embeddings for disease-free analysis. The number in the parenthesis indicates the number of samples in low or high risk group. The p-value is calculated by the log-rank test to compare the overall survival or disease-free probability of two groups of breast cancer patients.

To evaluate the performance of omicsGAT in greater depth, the patient's overall survival time and disease-free time were predicted on the breast cancer dataset. The Cox proportional hazards model with elastic net penalty (35) evaluated the correlation between the patient's overall survival time or disease-free time and genomic features, i.e., the original gene expression and the omicsGAT learned embeddings. 80% of the patient samples were applied to train the model and the performance was tested on 20% test samples. The low and high risk groups on the independent test set were generated based on the prognostic index (36). The survival and disease-free prediction were visualized by Kaplan-Meier plots and compared by the log-rank test. The Kaplan-Meier plots in Figure 2 illustrates the improved patient survival time and disease-free time prediction on breast cancer patients using the embeddings generated by omicsGAT compared to the original gene expression. The log-rank test p-values clearly demonstrate a strong additional prediction power of the learned embeddings beyond the gene expression.

Table 3. Hyperparameter selection for omicsGAT Clustering

Hyperparameter	Selection Set
No. of PCA components (features) selected	$[50, 100, 200, \mathbf{400}]$
Embedding size of a head	$[4, 8, 16, 32, \mathbf{64}]$
No. of heads	$[4, 8, 16, 32, \mathbf{64}]$
Network density of adjacency matrix	$[0.02, 0.04, 0.1, \mathbf{0.2}]$
No. of FC layers	$[2, \mathbf{3}, 4]$

F.3. omicsGAT Improved Cancer Patient Stratification. To evaluate the generalization of our embedding mechanism, we employed omicsGAT Clustering to stratify bladder cancer (BLCA) patients. The dataset consisted of five cancer subtypes and our task was to cluster the patients into these five categories. Embeddings were generated following the first step of omicsGAT Clustering, i.e., the autoencoder described in Section C. First, the dimension of the raw gene expression data was reduced using PCA implemented through *sklearn.decomposition.PCA* package. The top 400 PCA components were then used as input in the omicsGAT pipeline, and the generated embeddings were fed to the second step of omicsGAT Clustering, a hierarchical clustering model. We show two findings in this experiment: (1) clustering of the embeddings demonstrates cluster-specific patterns in the embeddings, and (2) high-quality embeddings enhance the performance of clustering cancer patients into cancer subtypes. The embedding clustering result is illustrated in Figure 3 where each row represents a patient sample and columns represent embeddings. The patient samples were grouped together according to their cancer subtypes. The distinct pattern can be observed for the embeddings generated for a particular cancer subtype signifying the ability of omicsGAT to effectively integrate neighborhood information into the embedding for a better predictive signature.

Next, we compared the performance of omicsGAT Clustering with the baselines for clustering patient samples into cancer subtypes. Assigned cluster values of the samples by omicsGAT Clustering and the true cancer subtype of the samples were matched to calculate NMI and ARI scores. The NMI and ARI scores which were calculated after employing hierarchical clustering and k-means clustering on raw gene expressions, and the 400 PCA components were used as baselines. Additionally, we clustered the sample adjacency matrix

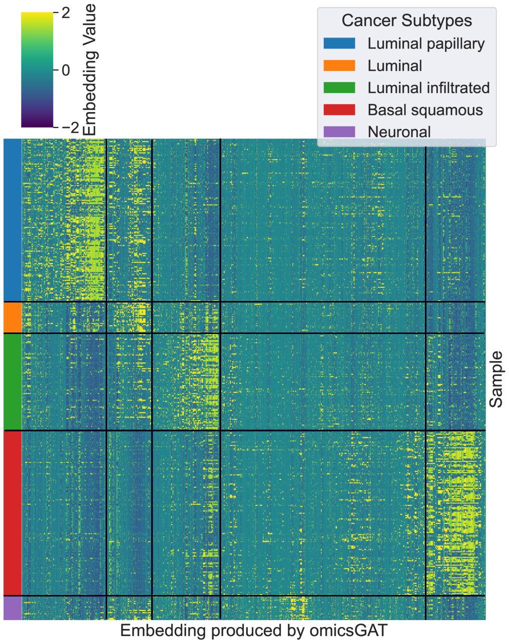

Fig. 3. Embeddings generated by omicsGAT clustered into the corresponding cancer subtypes

A as another baseline. The results are reported in Table 4. It can be observed that both NMI and ARI scores are highest for omicsGAT Clustering followed by the clustering of the adjacency matrix. The scores for the PCA components and the raw gene expression features are lower which can be attributed to the absence of sample similarity information in the datasets, whereas the embeddings from omicsGAT and the adjacency matrix consider the relations between samples. omicsGAT used the information from the neighbors more effectively by assigning different attention coefficients to the neighbors of a sample, thereby capturing the hidden relations between samples in the embeddings. This influx of information caused by the attention mechanism in embedding generation enabled omicsGAT Clustering to outperform all baselines by a considerable margin.

To visualize the clustering performance, tSNE plots (Python *seaborn* package) were created on the PCA components and the embeddings generated by omicsGAT in Figure 4 (a) and (b) respectively. Figure 4 (a) illustrates that PCA components cannot properly separate the five clusters. Although there is some separation among the patient samples in 'Basal squamous', 'Luminal Papillary', and 'Luminal infiltrated' subtypes, the samples in 'Luminal' and 'Neuronal' subtypes were randomly scattered on the plot. On the other hand, Figure 4 (b) shows that omicsGAT Clustering can effectively separate all five clusters, revealing the meaningful neighborhood information contained within the embeddings. Moreover, 'Luminal' and 'Neuronal' are the subtypes with the smallest number of samples which means our proposed method particularly excels at clustering rare subtypes.

Table 4. The clustering performance on TCGA bladder cancer (BLCA) dataset. The NMI and ARI scores of omicsGAT Clustering and baseline methods are reported in the table. Hierarchical clustering was computed with 'Manhattan' distance and 'Average' linkage. Mean NMI and ARI scores with standard deviation are reported for k-means clustering (run 10 times).

Input Data (Clustering Method)	NMI	NMI SD	ARI	ARI SD
gene expression (hierarchical)	0.0515	-	0.0153	-
gene expression (k-means)	0.4944	0.0171	0.4468	0.0548
PCA components (hierarchical)	0.1222	-	0.0353	-
PCA components (k-means)	0.4883	0.0176	0.4338	0.0388
adjacency matrix (hierarchical)	0.5448	-	0.5505	-
omicsGAT embeddings (hierarchical)	**0.6147**	-	**0.6698**	-

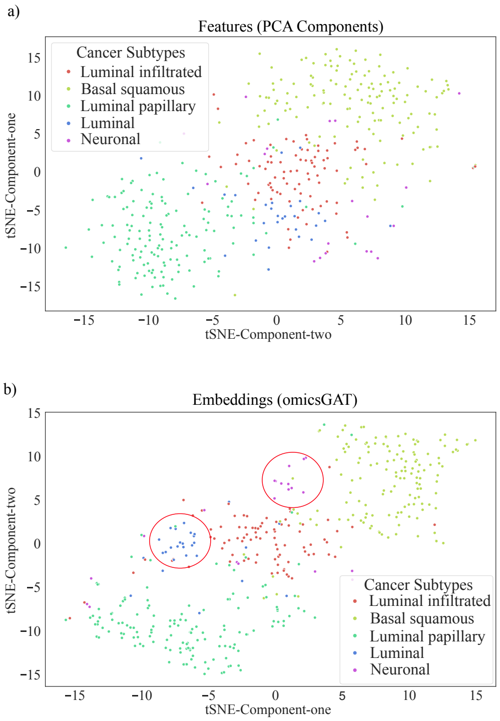

Fig. 4. tSNE plots of the (a) PCA components generated from the BLCA data and (b) omicsGAT generated embeddings for bladder cancer patients stratification.

G. Experimentation on Single-cell RNA-seq data.

Single-cell RNA-seq (scRNA-seq) reveals heterogeneity at the cell level and offers a larger number of samples (i.e., cells) compared to bulk RNA-seq data (e.g., number of patient samples). We applied omicsGAT Clustering on scRNA-seq data and clustered cells to evaluate the generalization of our proposed model.

G.1. Dataset and Preprocessing. scRNA-seq data from six primary H3K27M-gliomas (H3 lysine27-to-methionine mutations) was used in the following experiment. This type of gliomas (malignant tumors) primarily arise in the midline of the central nervous system of young children (37). Early detection of tumors may improve disease prognosis; hence, stratifying the tumor cells into the correct gliomas could be very helpful for clinicians. Gene expression and label information of 2,458 cells was used for this experiment. The dataset was downloaded from the Single Cell Portal (38) and the cells were generated from six different gliomas: BCH836, BCH869, BCH1126, MUV1, MUV5, MUV10. $log2(x+1)$ transformed TPM (Transcripts-per-million) value was used in the analysis.

Table 5. The clustering performance on scRNA-seq H3K27M-gliomas data. The NMI and ARI scores of omicsGAT Clustering and baseline methods are reported in the table. Hierarchical clustering was computed with 'Cosine' distance and 'Average' linkage. Mean NMI and ARI scores with standard deviation are reported for k-means clustering (run 10 times).

Matrix Type (Clustering Type)	NMI	NMI SD	ARI	ARI SD
gene expression (hierarchical)	0.0055	-	0.0010	-
gene expression (k-means)	0.5052	0.0176	0.4410	0.0145
PCA components (hierarchical)	0.6146	-	0.5339	-
PCA components (k-means)	0.5010	0.0016	0.4640	0.0013
adjacency matrix (hierarchical)	0.5757	-	0.3982	-
omicsGAT embeddings (hierarchical)	**0.6584**	-	**0.6366**	-

G.2. Single Cell Clustering. The same omicsGAT Clustering method described in Section C is followed to cluster the cells with scRNA-seq data. The top 200 PCA components were selected as the input of the omicsGAT Clustering to generate embeddings. The omicsGAT's hyperparameters for this experiment are listed in Table S2 in the Supplementary document. The autoencoder was trained following the same steps as explained in Section F.3. Embeddings generated from the autoencoder were then fed into the hierarchical clustering model. Hierarchical and k-means clustering methods on raw gene expression and 200 PCA components were considered as the baselines along with hierarchical clustering on the adjacency matrix. As reported on Table 5, omicsGAT Clustering outperforms all the baselines which means the cluster assignments resulting from the omicsGAT generated embeddings are more similar to the true label information. This result is corroborated by the tSNE plots in Figure 5 (a) and (b) which are drawn on the PCA components and the embeddings generated by omicsGAT respectively. The tSNE plot for omicsGAT Clustering shows more separation among the clusters as compared to the PCA components. Specifically, for the 'MUV1' group, our model formed a single cluster containing all the cells belonging to that type (red circle in Figure 5 (b)), whereas the tSNE plot using PCA components shows two different clusters for the cells in 'MUV1'. Based on the results, we can conclude that in the case of scRNA-seq data analysis, omicsGAT Clustering can take advantage of the detailed cellular level information and uses the attention mechanism on the cell-cell similarity network to better cluster the samples.

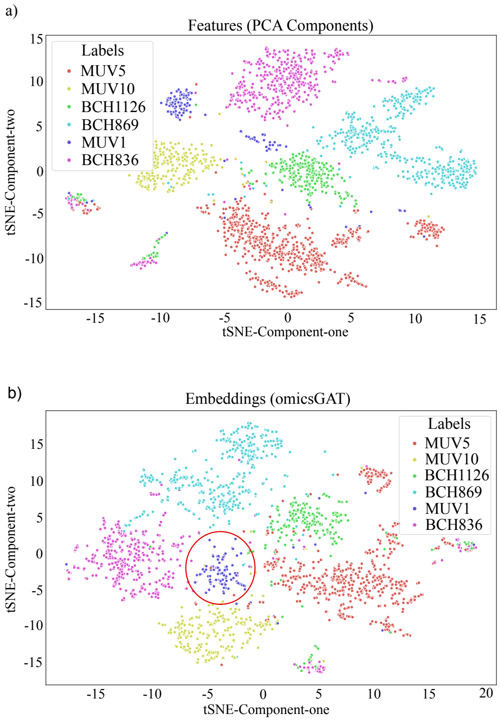

Fig. 5. tSNE plots of the (a) PCA components generated from the scRNA-seq data and (b) omicsGAT generated embeddings for cell clustering.

Discussion

omicsGAT can successfully integrate the structural information within gene expression data into sample embeddings enabling better classification and clustering performance compared to the original dataset. The stronger predictive ability of the embeddings is contributed by the self-attention mechanism in omicsGAT. A binary adjacency matrix is applied to define neighborhoods in omicsGAT that includes self-connections to ensure that the information of a sample itself is also considered in the embedding. The performance is reduced when we ran the same classification task with just the adjacency matrix. The adjacency matrix is calculated using correlation only, which keeps track of the pairwise linear relations between samples. However, using the attention mechanism, omicsGAT can capture complex nonlinear relations by accounting for the importance of neighboring samples on the classification or clustering of a target sample. The captured relations among samples are represented in the generated embeddings which enables the model to perform better on classification and clustering tasks.

In order to verify the effect of the multi-head attention mechanism, a $sample \times sample$ attention matrix was constructed

Table 6. NMI and ARI scores of the Hierarchical Clustering applied on attention and adjacency matrices

Dataset	Input Matrix	NMI	ARI
BLCA	adjacency matrix	0.5448	0.5505
	attention matrix	0.5743	0.6373
scRNA	adjacency matrix	0.5757	0.3982
	attention matrix	0.5788	0.4821

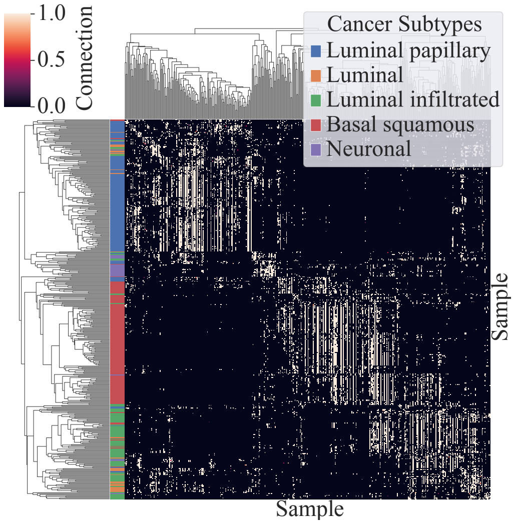

Fig. 6. Clustermap of the Attention Matrix generated from the trained omicsGAT model on BLCA data

by extracting the attention coefficients from a trained omicsGAT model following the method used by Ullah and Ben-Hur (39). For a target sample, each of the h heads assigns different attention coefficients to its neighbors, and only the highest among the h attention coefficients was considered for each neighbor to represent its relation with the target sample. The same procedure is repeated to generate the full attention matrix. This process was applied to build the attention matrix for both BLCA and cell clustering tasks described in Section F.3 and Section G.2 respectively. This attention matrix reveals the importance of combining the attention mechanism with the network information received through the adjacency matrix. As seen in Table 6, clustering on the attention matrix outperforms the clustering on the adjacency matrix for both datasets. Moreover, the clustermap of the attention matrix obtained from the trained model on BLCA data, illustrated in Figure 6, shows a distinct pattern of the cancer subtypes specifically for 'Luminal papillary' and 'Basal squamous'. From these results, we can conclude that some neighbors play a more important role than others in classification or clustering of a sample, and omicsGAT can effectively inject this information into the model along with the graph structure to generate more meaningful embeddings for better downstream analyses. An important aspect of omicsGAT is the use of multiple heads. The learnable weight parameters ($\boldsymbol{W}$ and $\boldsymbol{a}$) of each head were initialized separately using the *xavier normal* library function in *Pytorch* (31).

For the clustering tasks, the NMI and ARI scores of the baselines were relatively low with hierarchical clustering which can be observed in Table 4 and Table 5. Therefore, we also applied k-means clustering to them in order to compare them with omicsGAT. Since the performance of k-means clustering depends on the initialization of the cluster-centers, clustering was conducted 10 times and the mean scores along with standard deviations were reported in the tables.

Conclusion

Powered by high-throughput genomic technologies, the RNA-seq method is capable of measuring transcriptome-wide mRNA expressions and molecular activities in cancer cells. Hundreds of computational methods have been developed for cancer outcome prediction, patient stratification, and cancer cell clustering. Some of these methods consider sample-sample similarities in the analysis, and some of them do not. These sample similarity-based methods cannot distinguish the importance of the neighbors for a particular sample in the downstream prediction or clustering tasks. Therefore, we introduced omicsGAT in this study which leverages a self-attention mechanism consisting of multiple heads to assign proper attention weights to the neighbors of a sample in the network. Experiments on cancer subtype analyses show the superior performance of the model in every aspect compared to the baseline methods. We also show the generalization of omicsGAT's performance on both bulk RNA-seq and scRNA-seq data. As a future objective, we would like to extend omicsGAT to include metapath selection which would consider the best paths in a network to perform a certain task on a particular sample.

Bibliography

1. Sarah CP Williams. News feature: Capturing cancer's complexity. *Proceedings of the National Academy of Sciences*, 112(15):4509–4511, 2015.
2. Paulina Krzyszczyk, Alison Acevedo, Erika J Davidoff, Lauren M Timmins, Ileana Marrero-Berrios, Misaal Patel, Corina White, Christopher Lowe, Joseph J Sherba, Clara Hartmanshenn, et al. The growing role of precision and personalized medicine for cancer treatment. *Technology*, 6(03n04):79–100, 2018.
3. A Spitale, P Mazzola, D Soldini, L Mazzucchelli, and A Bordoni. Breast cancer classification according to immunohistochemical markers: clinicopathologic features and short-term survival analysis in a population-based study from the South of Switzerland. *Annals of oncology*, 20(4):628–635, 2009.
4. Ping Tang, Jianmin Wang, and Patria Bourne. Molecular classifications of breast carcinoma with similar terminology and different definitions: are they the same? *Human pathology*, 39 (4):506–513, 2008.
5. Fiona M Blows, Kristy E Driver, Marjanka K Schmidt, Annegien Broeks, Flora E Van Leeuwen, Jelle Wesseling, Maggie C Cheang, Karen Gelmon, Torsten O Nielsen, Carl Blomqvist, et al. Subtyping of breast cancer by immunohistochemistry to investigate a relationship between subtype and short and long term survival: a collaborative analysis of data for 10,159 cases from 12 studies. *PLoS medicine*, 7(5):e1000279, 2010.
6. John C Marioni, Christopher E Mason, Shrikant M Mane, Matthew Stephens, and Yoav Gilad. RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome research*, 18(9):1509–1517, 2008.
7. Zhong Wang, Mark Gerstein, and Michael Snyder. RNA-Seq: a revolutionary tool for transcriptomics. *Nature reviews genetics*, 10(1):57–63, 2009.
8. Quan Zou, Jiancang Zeng, Liujuan Cao, and Rongrong Ji. A novel features ranking metric with application to scalable visual and bioinformatics data classification. *Neurocomputing*, 173:346–354, 2016.
9. Feng Gao, Wei Wang, Miaomiao Tan, Lina Zhu, Yuchen Zhang, Evelyn Fessler, Louis Vermeulen, and Xin Wang. Deepcc: a novel deep learning-based framework for cancer molecular subtype classification. *Oncogenesis*, 8(9):1–12, 2019.
10. Khandakar Tanvir Ahmed, Jiao Sun, Sze Cheng, Jeongsik Yong, and Wei Zhang. Multi-omics data integration by generative adversarial network. *Bioinformatics*, 38(1):179–186, 2022.
11. Xianxue Yu, Guoxian Yu, and Jun Wang. Clustering cancer gene expression data by projective clustering ensemble. *PloS one*, 12(2):e0171429, 2017.
12. Therese Sørlie, Charles M Perou, Robert Tibshirani, Turid Aas, Stephanie Geisler, Hilde Johnsen, Trevor Hastie, Michael B Eisen, Matt Van De Rijn, Stefanie S Jeffrey, et al. Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proceedings of the National Academy of Sciences*, 98(19):10869–10874, 2001.

13. M. Gori, G. Monfardini, and F. Scarselli. A new model for learning in graph domains. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 2, pages 729–734 vol. 2, 2005. doi: 10.1109/IJCNN.2005.1555942.
14. Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The Graph Neural Network Model. *IEEE Transactions on Neural Networks*, 20(1): 61–80, 2009. doi: 10.1109/TNN.2008.2005605.
15. Khandakar Tanvir Ahmed, Sunho Park, Qibing Jiang, Yunku Yeu, TaeHyun Hwang, and Wei Zhang. Network-based drug sensitivity prediction. *BMC medical genomics*, 13(11):1–10, 2020.
16. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.
17. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
18. Yifeng Tao, Chunhui Cai, William W Cohen, and Xinghua Lu. From genome to phenome: Predicting multiple cancer phenotypes based on somatic genomic alterations via the genomic impact transformer. In *PACIFIC SYMPOSIUM ON BIOCOMPUTING 2020*, pages 79–90. World Scientific, 2019.
19. Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph Attention Networks. *International Conference on Learning Representations*, 2018.
20. Jinlong Hu, Lijie Cao, Tenghui Li, Shoubin Dong, and Ping Li. GAT-LI: a graph attention network based learning and interpreting method for functional brain network classification. *BMC bioinformatics*, 22(1):1–20, 2021.
21. Hu Linmei, Tianchi Yang, Chuan Shi, Houye Ji, and Xiaoli Li. Heterogeneous graph attention networks for semi-supervised short text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4821–4830, 2019.
22. Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. Heterogeneous graph attention network. In *The world wide web conference*, pages 2022–2032, 2019.
23. Jiezhong Qiu, Jian Tang, Hao Ma, Yuxiao Dong, Kuansan Wang, and Jie Tang. Deepinf: Social influence prediction with deep learning. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2110–2119, 2018.
24. Qitian Wu, Hengrui Zhang, Xiaofeng Gao, Peng He, Paul Weng, Han Gao, and Guihai Chen. Dual graph attention networks for deep latent representation of multifaceted social effects in recommender systems. In *The World Wide Web Conference*, pages 2091–2102, 2019.
25. Haiyang Wang, Guangyu Zhou, Siqi Liu, Jyun-Yu Jiang, and Wei Wang. Drug-Target Interaction Prediction with Graph Attention networks. *arXiv preprint arXiv:2107.06099*, 2021.
26. Yahui Long, Min Wu, Yong Liu, Chee Keong Kwoh, Jiawei Luo, and Xiaoli Li. Ensembling graph attention networks for human microbe–drug association prediction. *Bioinformatics*, 36(Supplement_2):i779–i786, 2020.
27. João Schapke, Anderson Tavares, and Mariana Recamonde-Mendoza. Epgat: Gene essentiality prediction with graph attention networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2021.
28. DCFR Koboldt, Robert Fulton, Michael McLellan, Heather Schmidt, Joelle Kalicki-Veizer, Joshua McMichael, Lucinda Fulton, David Dooling, Li Ding, Elaine Mardis, et al. Comprehensive molecular portraits of human breast tumours. *Nature*, 490(7418):61–70, 2012.
29. Cancer Genome Atlas Network TCGA et al. Comprehensive molecular characterization of urothelial bladder carcinoma. *Nature*, 507(7492):315, 2014.
30. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
31. Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.
32. Thomas N Kipf and Max Welling. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv preprint arXiv:1609.02907*, 2016.
33. Mary J Goldman, Brian Craft, Mim Hastie, Kristupas Repečka, Fran McDade, Akhil Kamath, Ayan Banerjee, Yunhai Luo, Dave Rogers, Angela N Brooks, et al. Visualizing and interpreting cancer genomics data via the Xena platform. *Nature Biotechnology*, pages 1–4, 2020.
34. Jianjiong Gao, Bülent Arman Aksoy, Ugur Dogrusoz, Gideon Dresdner, Benjamin Gross, S Onur Sumer, Yichao Sun, Anders Jacobsen, Rileen Sinha, Erik Larsson, et al. Integrative analysis of complex cancer genomics and clinical profiles using the cBioPortal. *Science signaling*, 6(269):pl1–pl1, 2013.
35. Yi Yang and Hui Zou. A cocktail algorithm for solving the elastic net penalized cox's regression in high dimensions. *Statistics and its Interface*, 6(2):167–173, 2013.
36. Wei Zhang, Takayo Ota, Viji Shridhar, Jeremy Chien, Baolin Wu, and Rui Kuang. Network-based survival analysis reveals subnetwork signatures for predicting outcomes of ovarian cancer treatment. *PLoS computational biology*, 9(3):e1002975, 2013.
37. Mariella G Filbin, Itay Tirosh, Volker Hovestadt, McKenzie L Shaw, Leah E Escalante, Nathan D Mathewson, Cyril Neftel, Nelli Frank, Kristine Pelton, Christine M Hebert, et al. Developmental and oncogenic programs in H3K27M gliomas dissected by single-cell RNA-seq. *Science*, 360(6386):331–335, 2018.
38. Mariella G. Filbin, Itay Tirosh, Volker Hovestadt, McKenzie L. Shaw, Leah E. Escalante, Nathan D. Mathewson, Cyril Neftel, Nelli Frank, Kristine Pelton, Christine M. Hebert, Christine Haberler, Keren Yizhak, Johannes Gojo, Kristof Egervari, Christopher Mount, Peter van Galen, Dennis M. Bonal, Quang-De Nguyen, Alexander Beck, Claire Sinai, Thomas Czech, Christian Dorfer, Liliana Goumnerova, Cinzia Lavarino, Angel M. Carcaboso, Jaume Mora, Ravindra Mylvaganam, Christina C. Luo, Andreas Peyrl, Mara Popović, Amedeo Azizi, Tracy T. Batchelor, Matthew P. Frosch, Maria Martinez-Lage, Mark W. Kieran, Pratiti Bandopadhayay, Rameen Beroukhim, Gerhard Fritsch, Gad Getz, Orit Rozenblatt-Rosen, Kai W. Wucherpfennig, David N. Louis, Michelle Monje, Irene Slavc, Keith L. Ligon, Todd R. Golub, Aviv Regev, Bradley E. Bernstein, and Mario L. Suvà. Developmental and oncogenic programs in H3K27M gliomas dissected by single-cell RNA-seq. *Science*, 360(6386): 331–335, April 2018. doi: 10.1126/science.aao4750.
39. Fahad Ullah and Asa Ben-Hur. A self-attention model for inferring cooperativity between regulatory features. *Nucleic acids research*, 49(13):e77–e77, 2021.

Baul *et al.* | omicsGAT bioRχiv | 9