OXFORD

Briefings in Bioinformatics, 2025, **26(4)**, bbaf409
https://doi.org/10.1093/bib/bbaf409
Problem Solving Protocol

DeepGraphMut: a graph-based deep learning method for cancer prognosis using somatic mutation profile

Aswin Jose, Akansha Srivastava, Ariba Ansari, Palakkad Krishnanunni Vinod*

Centre for Computational Natural Sciences and Bioinformatics, IIIT Hyderabad, Prof. C R Rao Road, Gachibowli, Hyderabad 500032, India
*Corresponding author. Centre for Computational Natural Sciences and Bioinformatics, IIIT Hyderabad, Prof. C R Rao Road, Gachibowli, Hyderabad 500032, India. E-mail: vinod.pk@iiit.ac.in

Abstract

Cancer remains a leading cause of morbidity and mortality worldwide. Despite advances in genomics, identifying clinically relevant subtypes of cancer remains challenging due to its complex and heterogeneous nature. In this work, we propose DeepGraphMut (DGM), a novel graph-based deep-learning pipeline that integrates somatic mutation data with protein–protein interaction (PPI) networks. By employing a graph autoencoder with a graph attention layer and a node-level attention decoder, DGM generates patient-specific clinically relevant encodings for unsupervised and supervised tasks. We demonstrate the effectiveness of DGM across 16 cancer types comprising of 7352 samples from The Cancer Genome Atlas (TCGA). Unsupervised clustering reveals distinct subtypes with significant survival differences in 11 cancer types. In supervised analysis using a Cox regression model, DGM demonstrates excellent performance in predicting survival outcomes, achieving a high concordance index (C-index) value in the range of 0.7 across most cancers, underscoring its robust predictive performance using only somatic mutation data. Furthermore, DGM outperforms its lightweight variant and network-based stratification methods in both unsupervised and supervised analyses. In summary, this study presents a promising approach for cancer subtype identification and prognosis, especially in resource-limited settings where multi-omics data may not be readily available. By leveraging the strengths of graph learning and network biology, DGM offers a valuable tool for advancing personalized medicine.

Keywords: cancer subtype identification; graph neural network; somatic mutation; protein–protein interaction network; survival prediction

Introduction

According to GLOBOCAN, there were close to 20 million new cancer cases and 9.7 million cancer deaths worldwide in 2022, with approximately one in five people developing cancer [1]. Traditionally, cancer classification relies on the site of origin and histological characteristics [2]. The current treatment landscape is often constrained by a one-size-fits-all approach [3, 4] due to an incomplete understanding of the underlying molecular and regulatory changes driving cancer progression. While cost-effective and the best option available given current knowledge, this strategy is far from ideal for addressing the diverse and complex nature of cancer. It is important to use molecular biomarkers for cancer stratification to help guide personalized treatment strategies and improve patient outcomes.

Cancer cells evade signals that control cell behavior due to various DNA abnormalities, such as somatic mutations, alterations in copy number, and changes in DNA methylation patterns [5]. The advent of high-throughput sequencing (HTS) technology and its increasing accessibility have revolutionized our understanding of the human genome, ushering in a new era of data abundance [6]. Somatic mutation profiles, obtained by comparing the genome or exome of a patient's tumor with that of the germ line using HTS, are a promising source of data for cancer stratification. These profiles are presumed to contain the causal drivers of cancer progression [7]. However, somatic mutation profiles are extremely sparse and heterogeneous, making stratification challenging.

Network Biology enables the interpretation and modeling of complex biological systems through the integration of omics data and biological interactions [8, 9]. Recently, the application of biological networks has proven instrumental in unraveling biological mechanisms, understanding disease origins, and forecasting responses to therapies at both molecular and systemic levels [10]. Network-based approaches, particularly network diffusion (ND) or propagation, have gained prominence in analyzing HTS datasets by leveraging known or inferred gene relationships. The network-based stratification (NBS) method proposed by Hofree *et al.* [11] is widely utilized to integrate protein–protein interaction (PPI) network with tumor mutation data. Patient encodings are obtained by propagating mutational information across the PPI network. Based on these encodings, patients are further stratified into clinically relevant groups [11]. Zhong *et al.* [12] applied NBS to

Received: November 21, 2024. **Revised:** July 01, 2025. **Accepted:** July 19, 2025

stratify 13 cancers using gene panels, discovering survival-related subtypes in five of them, while PyNBS further refined the NBS approach by utilizing a more compact PPI network for improved outcomes [13]. Other works have integrated different genomic data profiles with PPI networks at various stages, classified based on the point of data integration (ND first, ND during, and ND after) [14]. Additionally, the network embedding method has been introduced as another approach for patient stratification. Notably, the network embedding stratification (NES) approach combines the human PPI network with genome-scale somatic mutation profiles, hypothesizing that patients with mutations in similar network regions are more likely to be of the same subtype. This method involves gene vectorization through network embedding using struc2vec, patient feature construction by integrating somatic mutation profiles with gene vectors, and patient stratification using machine learning approaches [15].

Building on diffusion-based and embedding techniques like NES, graph learning goes a step further by applying machine learning directly to graph-structured data. Instead of relying on precomputed embeddings or dimensionality reduction, graph learning models jointly leverage graph topology and node features to generate fixed-dimensional embeddings. These vector representations capture meaningful structural and feature information in a continuous space, aiding in downstream analysis [16]. In recent years, various approaches have been developed to understand omics data through graph-based deep learning [17]. A notable example of graph-based stratification is the Consensus-Guided Graph Autoencoder (CGGA) method, which effectively identifies cancer subtypes by integrating structure information and node features. It employs graph autoencoders and iteratively refines the learned representations using omic-specific similarity matrices to enhance subtype separation [18]. Another approach, omicsGAT, leverages the self-attention mechanism to generate embeddings from gene expression data. It constructs an adjacency matrix by linking samples with similar expression profiles and applies a multi-head self-attention mechanism to weigh the importance of neighboring samples [19]. These existing graph-based deep learning approaches primarily focus on node-level tasks, beginning with the creation of a patient similarity network and then performing either supervised or unsupervised analysis.

In this work, we introduce a novel graph-level framework, DeepGraphMut (DGM), leveraging a graph autoencoder equipped with a graph attention layer and a decoder featuring node-level attention to generate clinically relevant patient-wise encodings in an unsupervised manner. This approach integrates prior biological knowledge with somatic mutation information for cancer subtype identification and prognosis, harnessing the power of graph learning and modern computational capabilities. By treating each patient as an individual graph, our method enables a comprehensive and personalized analysis of somatic mutation data.

Results

Overview of the pipeline

Figure 1 illustrates the overall workflow of the DGM pipeline, a graph-based deep learning model designed to integrate somatic mutation data with prior biological knowledge for cancer subtype identification and prognosis prediction. The pipeline begins with construction of cancer-specific subnetwork (NCG network) from the human PPI network (see Methods). Each patient is represented as a graph with mutation values encoded at the node level, while the underlying network structure remains constant across all cancer samples. This network is then processed through the DeepGraphMut (DGM) pipeline, which includes encoding and decoding modules within an autoencoder framework. In the encoding module, GraphNorm is applied to optimize the normalization procedure by leveraging graph structure information. This technique offers notable advantages over traditional normalization methods, particularly in managing heterogeneous data types and complex network structures. In the decoding module, rather than using a traditional edge decoder, we employ a node decoder that focuses on learning variations in node values, representing patient-specific mutation profiles. The node decoder transforms the latent representation back into a series of node values, which are directly compared with the original omics data input to the encoder. This node-focused approach enables the extraction of salient features from the latent space, enhancing the analysis of the omics data. The model is trained until the optimal weights are learned, and the weights corresponding to the minimal validation loss are used for generating the encodings of input data. The encoded data then undergo processing through a mean pooling layer, facilitating subsequent analyses using both supervised and unsupervised approaches. The DGM pipeline was used to analyze somatic mutation data from 16 different cancer types in the TCGA.

Stratification of patients into subtypes with distinct survival outcomes

The gene representations learned from the DGM were used for clustering patients. Consensus clustering was applied to these representations to stratify cancer patients into distinct subtypes. We employed two clustering algorithms: Partitioning Around Medoids (PAM) and K-means. Survival analysis was then performed to evaluate the differences in survival among the identified cancer subtypes. The optimal number of clusters for each cancer type was determined based on silhouette scores, cophenetic correlation coefficients (CCC), and *P*-values. The DGM pipeline consistently performed well in stratifying patients into subtypes across the 11 cancers studied. Each subtype included a substantial number of patients, exhibited high CCC, and yielded significant *P*-values (Table 1 and Fig. 2). These results highlight the pipeline's broad applicability and effectiveness in addressing diverse cancer types. The PAM clustering algorithm demonstrated superior stability and statistical significance in clustering across most datasets. Notably, PAM achieved a balanced distribution of patients within each cluster, enhancing the interpretability and potential clinical relevance of the clustering outcomes (Table S1). In contrast, while K-means clustering identified some significant clusters, it often resulted in less distinct or significant groupings compared with PAM. An exception was noted in the analysis of GBM, where K-means outperformed PAM by producing significant clusters. Additionally, the optimal number of clusters varied across the 11 cancer types: two subtypes were identified for KIRC, GBM, and UCEC; three subtypes for BLCA, HNSC, OV, and STAD; four subtypes for LIHC; and five subtypes for LGG, LUSC, and SKCM (Table 1). The Kaplan–Meier plots illustrate the differences in survival probabilities among the cancer subtypes (Fig. 2). Notably, for BLCA, GBM, LGG, and SKCM, the observed *P*-values were very low, indicating highly significant differences in survival outcomes of the identified subtypes. The consensus maps and the corresponding Kaplan–Meier plots for the other cancers are provided in the Supplementary Material (Figure S1).

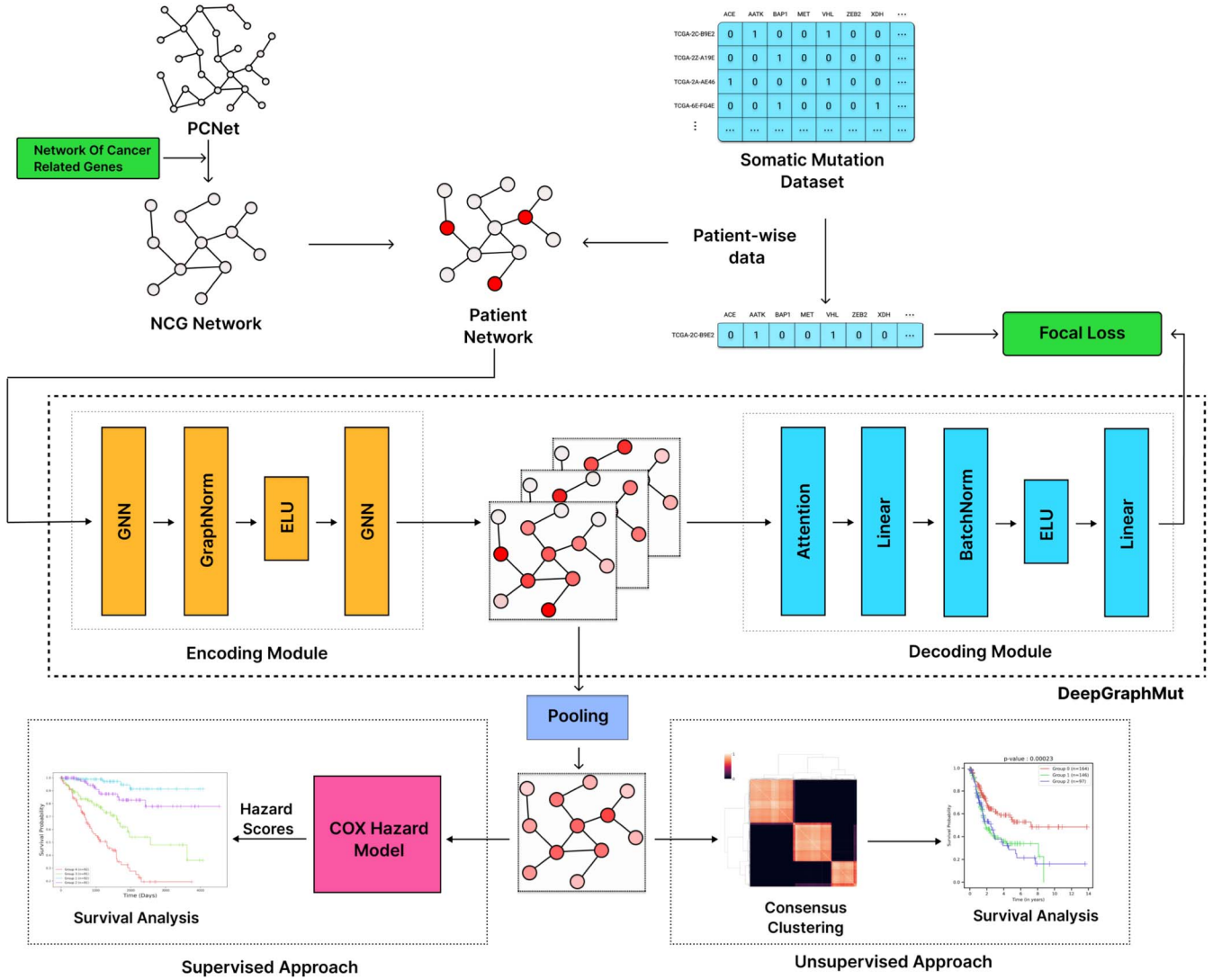

Figure 1. Overview of DeepGraphMut pipeline.

Table 1. Unsupervised clustering results based on encodings obtained using the DGM model with NCG network

Cancer	Clustering algorithm	Clusters	Silhouette score	P-value	CCC
BLCA	pam	3	0.45	.00023	0.99
GBM	kmeans	2	0.06	.00013	0.995
HNSC	pam	3	0.42	.02546	0.985
KIRC	pam	2	0.35	.0168	0.996
LGG	pam	5	0.29	7.36e-14	0.978
LIHC	pam	4	0.29	.008	0.984
LUSC	pam	5	0.39	.02044	0.969
OV	pam	3	0.31	.00254	0.992
SKCM	pam	5	0.46	6.56e-8	0.988
STAD	pam	3	0.48	.00961	0.99
UCEC	pam	2	0.6	.00727	0.996

We further characterized the identified subtypes using somatic mutation profile and clinical data, such as histological types and grades. Tumor mutational burden (TMB) was calculated for every sample to explore how mutational load varies across subtypes. A Kruskal–Wallis test was used to assess variation in TMB among subtypes within each cancer type. TMB was significantly associated with identified subtypes across all cancers except GBM. In KIRC, LGG, HNSC, and LIHC, we observed that subtypes with higher TMB tended to have poor survival outcomes (Fig. 3 and S2). For instance, in KIRC, Group 0 showed lower TMB and better prognosis, whereas Group 1 had higher TMB and worse survival (Fig. 3A). In LGG, Group 0, which had the poor survival, was enriched for EGFR mutations but had fewer IDH or TP53 alterations (Fig. 4A). However, in other cancers like BLCA, UCEC,

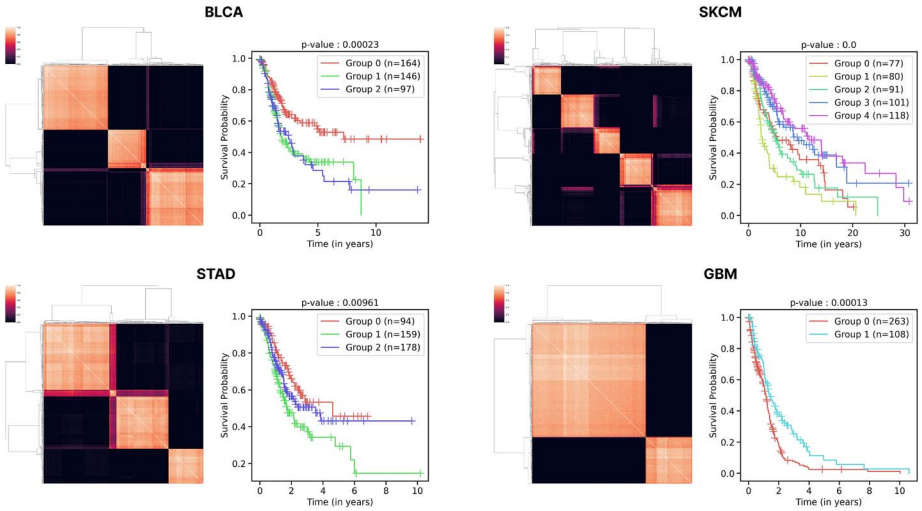

Figure 2. Unsupervised clustering for four cancer types (BLCA, STAD, SKCM, and GBM) with consensus maps (left) showing clustering stability and Kaplan-Meier plots (right) illustrating survival differences between clusters.

STAD, OV, and LUSC, subtypes with higher TMB were associated with better outcomes. (Fig. 3 and S2). For example, in BLCA, Group 0 exhibited high TMB and better survival, while Group 2 had lower TMB and poorer outcomes (Fig. 3C).

In the case of GBM, subtype-specific differences are largely driven by TP53 mutation status: Group 1 is predominantly TP53-mutant, while Group 2 harbors very few TP53 mutations (Fig. 4B). We also observed significant associations between the identified subtypes and histological grades in BLCA, HNSC, LGG, and KIRC. In LGG, the subtypes further correlated with distinct histological types, including astrocytoma, oligodendroglioma, and oligoastrocytoma (Table S2). For UCEC, Group 0 was primarily composed of serous samples, whereas Group 1 mainly consisted of endometrioid tumors, with only a few serous cases.

Graph-learning-based survival prediction model

We also adopted a supervised approach, which involved training a Cox regression model on the embeddings using patient survival data. The model performance was evaluated using the C-index value, which measures the predictive accuracy of survival outcomes. The C-index varied across cancer types, ranging from 0.5 to 0.8. For BRCA, CESC, COAD, KIRC, KIRP, LIHC, and LGG, the C-index exceeded 0.7, indicating strong predictive performance in survival analysis. For BLCA, LUAD, LUSC, HNSC, SKCM, STAD, and UCEC, the C-index ranged between 0.6 and 0.7, reflecting good predictive capability (Table 2). These results demonstrate the model's robust performance across various cancer types. We also divided patients into four groups based on Hazard scores obtained from Cox-model. These four groups showed significant survival differences for all 16 cancer types. Patients in lower hazard score percentiles exhibited significantly better survival rates than those in higher percentiles (Fig. 5 and Figure S3).

Robustness of encoders

To evaluate the impact of attention mechanisms on encoder performance, we repeated our experiments using a simplified, lightweight encoder based on the SAGEConv layer, termed DGMS. The unsupervised clustering and survival analysis revealed significant survival differences between patient groups for DGM compared with DGMS, as evident from the corresponding *P*-value

Table 2. C-index values from Cox regression using embeddings generated by the DGM model with the NCG network

Cancer type	C-Index
BLCA	0.64
BRCA	0.72
CESC	0.78
COAD	0.74
GBM	0.56
HNSC	0.66
KIRC	0.73
KIRP	0.73
LGG	0.8
LIHC	0.71
LUAD	0.63
LUSC	0.62
OV	0.59
SKCM	0.61
STAD	0.64
UCEC	0.69

(Table S3). Unsupervised clustering of DGMS encodings yielded significant results in 10 out of the 16 cancers studied. However, DGM clusters exhibited superior separation compared with those of DGMS, which is evident from the silhouette score and the log-rank *P*-value. In supervised analysis, the DGMS model encodings achieved a C-index over 0.7 for only four cancers and between 0.6 and 0.7 for seven cancers. In contrast, the DGM model outperformed DGMS, achieving C-index values above 0.7 for 7 cancers (Figure S4).

In our experiments, both autoencoder models generated relevant encodings across most cancer types. However, the DGM model exhibited greater robustness and consistency in capturing clinically meaningful patterns compared with its counterpart, DGMS. The attention-based architecture of DGM also led to faster convergence and a lower minimum loss during training, indicating more efficient learning. Nonetheless, these advantages come with increased computational cost, primarily due to the graph attention mechanism, which requires more resources than the simpler model without attention.

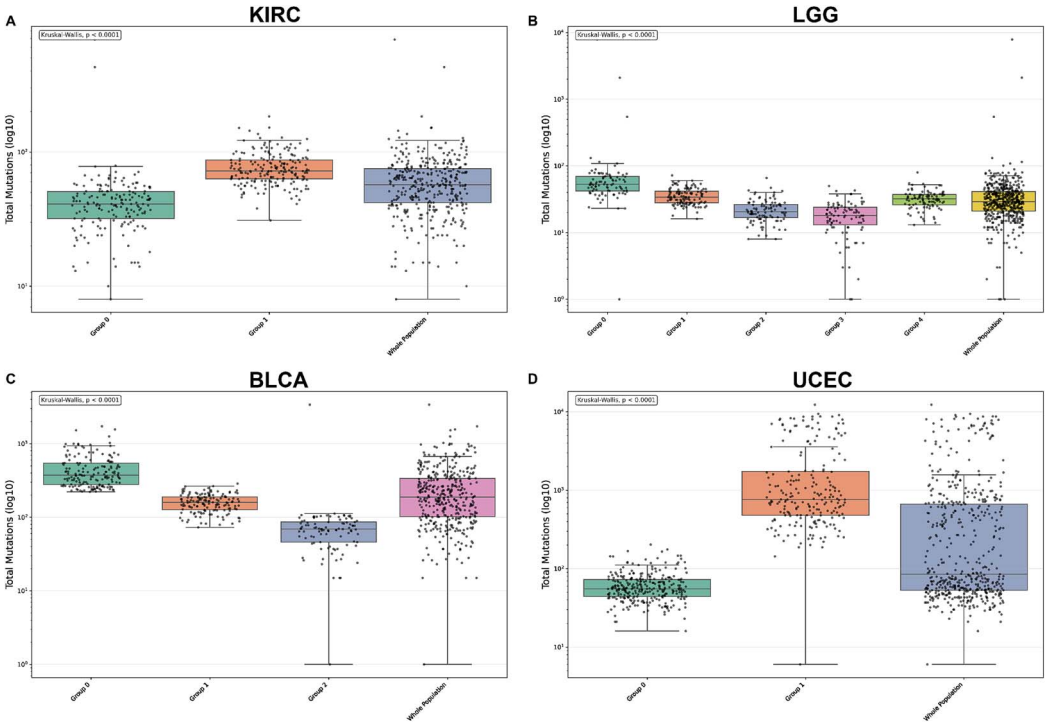

Figure 3. Distribution of tumor mutation burden across identified subtypes and whole cohort. (A) KIRC, (B) LGG, (C) BLCA, and (D) UCEC.

Network comparison

The additional networks were used to investigate model performance with different network sizes and node/edge compositions. We compared the results obtained using the NCG network with another reported-cancer-specific network, CancerReferenceNetwork (CRN) [20]. Due to the usage of different gene sets for the creation of the networks, there are significant disparities in the nodes (genes) and edges encompassed by each network (NCG and CRN) (Figure S5). The CRN yielded significant performance for 10 cancers (Table S4), while the unified network of CRN and NCG achieved notable results in 11 of the 16 cancer types (Table S5). However, their *P*-values were not as good as those of NCG-based encodings in most cases (Table S3). These networks also showed cancer-specific differences in performance. For example, KIRC was not accurately clustered using the CRN, suggesting that different networks capture distinct cancer-specific feature landscapes. This highlights the importance of selecting the appropriate network to accurately interpret the complex biological information presented by omics data. Notably, larger network sizes were associated with reduced clustering stability and increased computational demands, primarily due to the expansion of the attention matrix, despite no change in the number of learnable model parameters. The simpler DGMS model, which lacks an attention mechanism, yielded satisfactory clustering results in 10 of 16 cancer types when using the NCG network, compared with six using the CRN and eight using the unified network. Moreover, supervised Cox regression on the model encodings showed that both NCG and Unified network-based embeddings achieved higher C-index values than CRN (Figure S6). Collectively, these results emphasize the trade-offs between network size, computational cost, and performance, with the NCG network offering the best overall balance for cancer subtype identification.

Performance comparison with NBS methods

Our framework focuses on a graph-level task, distinguishing it from most deep learning approaches in this domain, which have primarily focused on multi-omics data integration [17]. Given the differences in data types, direct comparison with these methods may not be appropriate. To ensure a fair and rigorous evaluation, we initially benchmarked our method using the SAGEConv encoder (Table S3 and Figure S4), a widely recognized model in genomic research. Subsequently, we evaluated the performance of other network-based methods: PyNBS [13], struc2vec [21], and network-based Autoencoder (RWR-AE), across 16 cancer types using the NCG network, allowing for a direct comparison with DGM (see Methods). Figure 6 shows the performance of different models in the unsupervised task, reporting *P*-values and CCC. PyNBS yielded significant *P*-values in only three cancer types, while both RWR-AE and struc2vec achieved significance in eight cancer types. RWR-AE showed its best performance in UCEC and BLCA, whereas struc2vec performed best in KIRC and LIHC. In comparison, DGM achieved statistically significant subtype separations in 11 cancer types and outperformed RWR-AE in GBM, HNSC, LGG, LIHC, OV, SKCM, and STAD, as well as struc2vec in GBM, HNSC, LGG, OV, SKCM, and STAD. In terms of CCC values, DGM achieved the highest values in 15 cancer types, indicating highly consistent and robust patient stratification. In the supervised task, DGM consistently outperformed PyNBS, RWR-AE, and struc2vec, achieving higher C-index values across cancer types (Fig. 7). These results highlight the superior encoding and predictive capabilities of our DGM model.

Discussion

Somatic mutation profiling is becoming important for precision diagnostics and treatment. Heterogeneity in somatic mutation profile of specific cancers raises the challenge in accurately stratifying patients for cancer prognosis and treatment. Biological network provides an alternative view to molecular world. In this study, we have proposed a novel graph-based deep learning model called DGM for integrating patient mutation profile with

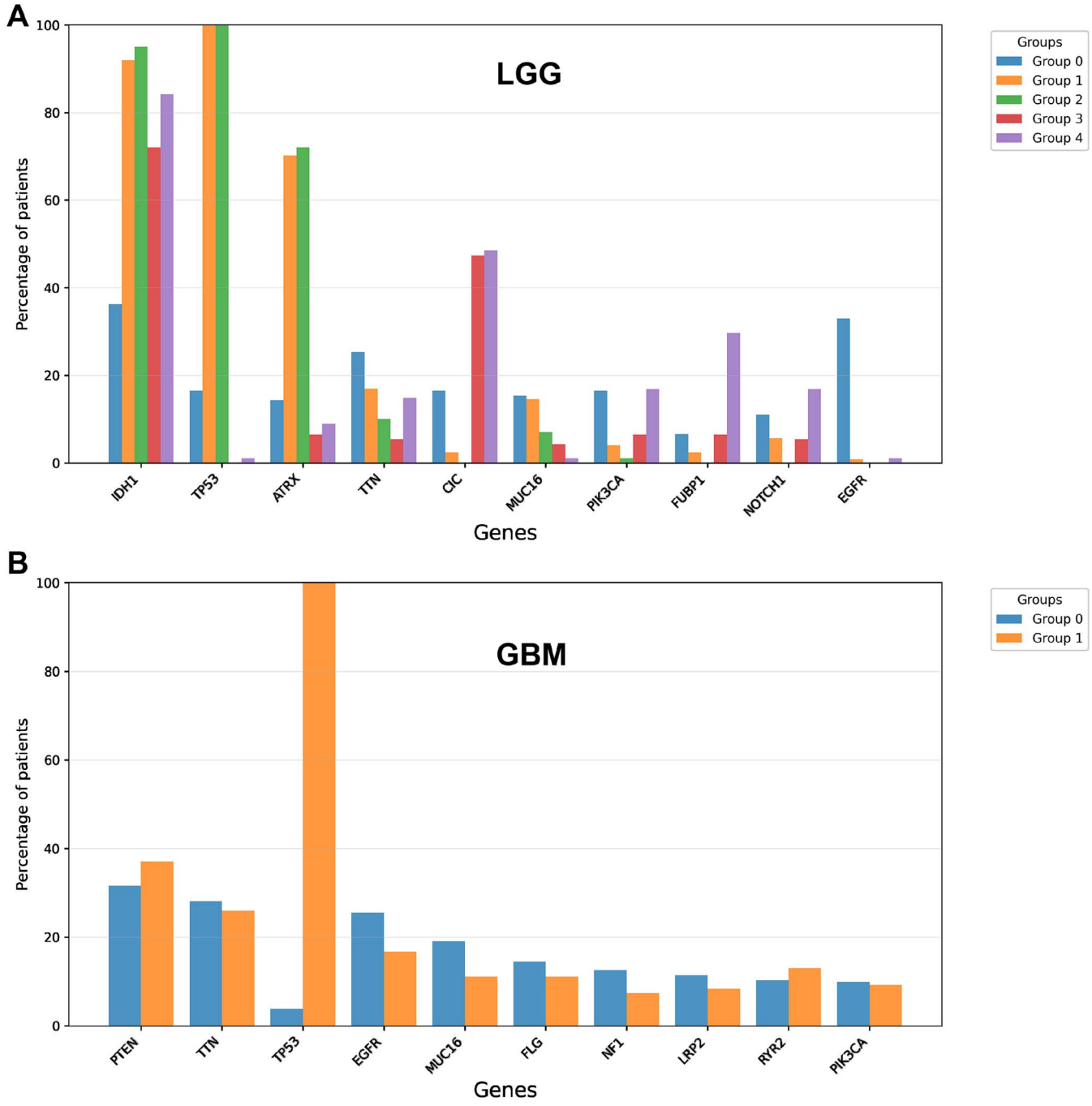

Figure 4. Distribution of mutated genes in each subtype of (A) LGG, (B) GBM.

prior network information for cancer subtype identification and prognosis across multiple cancer types.

A key strength of DGM lies in its robustness and generalizability. It uses a fixed, cancer-relevant NCG network for all cancer types, allowing the same graph structure to support diverse tasks such as unsupervised subtype discovery and supervised survival prediction. Patient-specific variation is captured through node attributes derived from somatic mutation profiles, while graph attention mechanisms in both the encoder and decoder focus learning on the most relevant edges and nodes. The incorporation of GraphNorm helps to process the gene embeddings effectively and stabilizes the learning process, accelerating convergence during the training. Additionally, the use of focal loss effectively addresses mutation sparsity, improving the accuracy of reconstructing patient-specific mutation profiles. In unsupervised analyses, DGM identified clinically significant subtypes in 11 of 16 cancers, while in supervised settings, the model's embeddings showed strong predictive power for patient survival across all cancer types. Among the tested networks, the NCG-based model consistently yielded the most informative encodings, likely due to its focus on cancer-relevant genes and interactions. Clustering via PAM also resulted in balanced subtype distributions and distinct survival separations.

Biological interpretability of the identified subtypes extends beyond survival stratification and is supported by associations with TMB, histological features, and known driver mutations. We observed cancer-type-specific patterns in TMB across subtypes.

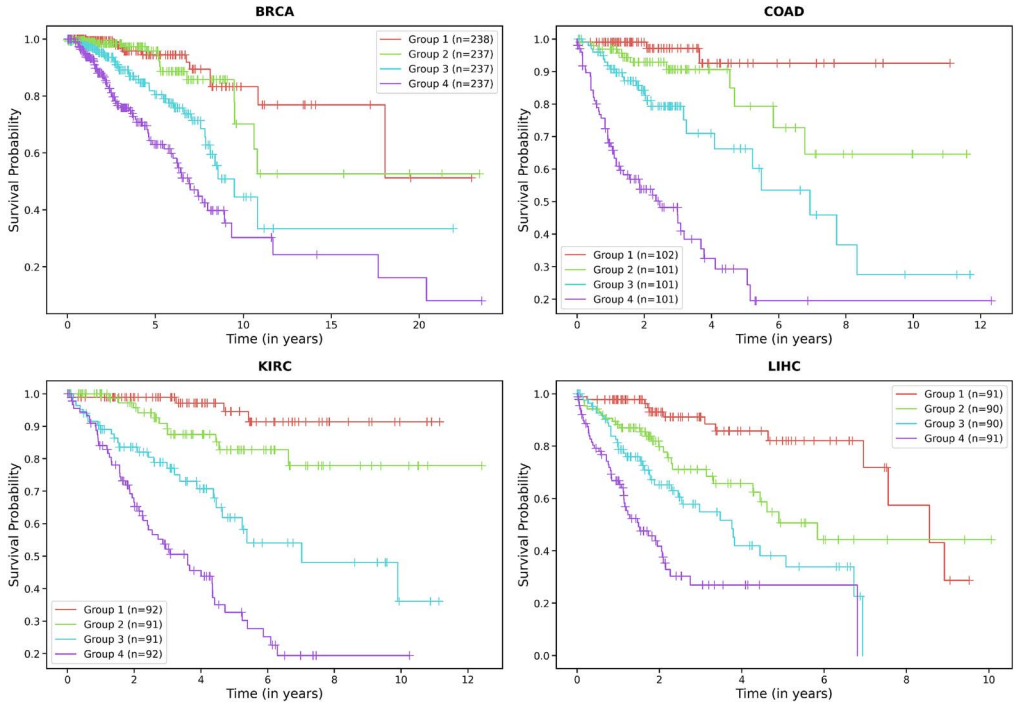

Figure 5. Kaplan-Meier plots showing survival outcomes for BRCA, COAD, KIRC, and LIHC patients stratified into four groups by quartiles of hazard scores predicted by Cox proportional hazards model.

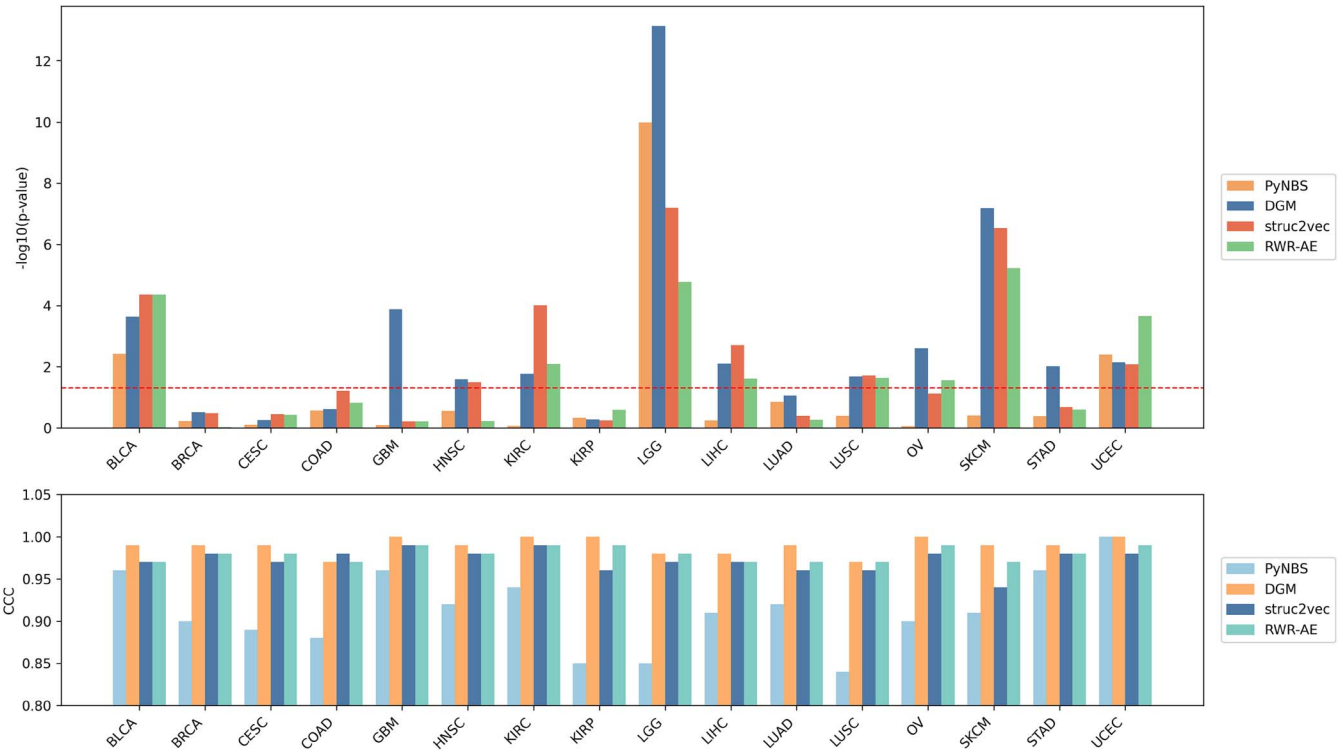

Figure 6. Comparative analysis with network-based approaches in unsupervised task.

In some cancers such as LGG, HNSC, LIHC, and KIRC, higher TMB was associated with poorer survival, aligning with prior studies linking genomic instability to aggressive tumor behavior [22]. Conversely, in cancers such as BLCA, STAD, UCEC, and LUSC, higher TMB was linked to better prognosis, potentially reflecting enhanced tumor immunogenicity and treatment responsiveness [23–25]. These divergent trends highlight the complex, context-dependent role of TMB in cancer biology. Significant associations were observed between subtype assignments and histological grades or subtypes in cancers such as LGG, BLCA, KIRC, HNSC, and UCEC. At the molecular level, we identified subtype-enriched mutations in specific cancers. In LGG, the group associated with poor survival was enriched for EGFR mutations, which can reshape the tumor immune microenvironment,

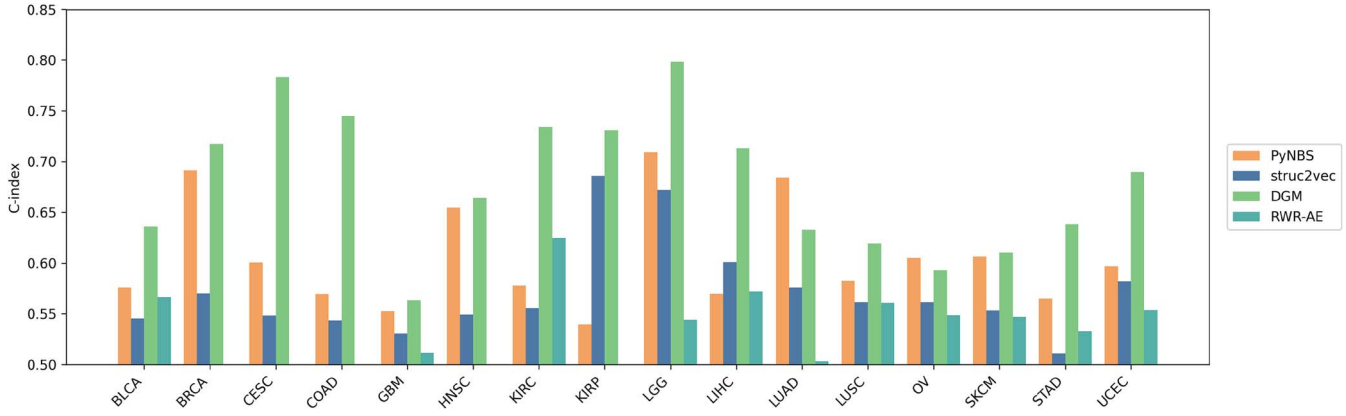

Figure 7. Comparison of performance of DGM and network-based approaches in supervised task.

contributing to treatment resistance and poor prognosis [26]. In GBM, subtype differences were primarily explained by TP53 mutation status, with Group 1 harboring frequent TP53 mutations. However, TP53 mutations in GBM have not been consistently associated with differences in patient survival [27]. In ongoing work, we are focusing on applying network interpretability methods to gain further insights into the molecular characteristics underlying the identified subtypes.

Our methodology extends the previously established NBS approach, which has been successfully applied in multiple studies using different prior networks and omics data types to obtain clinically relevant encodings [14]. Notably, our single-omics approach achieves C-index values that are comparable with or better than those reported in multi-omics studies for the same cancer types [28]. We also simplified the pipeline by employing a lightweight model DGMS (SAGEConv encoder), which achieved reasonable performance across a few cancer types. By focusing solely on somatic mutation data within the network of cancer-related genes, our models reduce computational complexity and offer a more streamlined solution. Although struc2vec and RWR-AE models performed well in specific cancer types, our proposed DGM model consistently outperformed them in both clustering stability and survival prediction, achieving higher C-index scores across multiple cancer types. By leveraging network topology, our approach effectively stratifies patients even in the absence of canonical driver mutations, which are often used as clinical markers. This capacity to uncover molecular subtypes highlights the complementary role of DGM in existing clinical workflows.

A key challenge in network-based modeling is managing the number of nodes and addressing the incompleteness of PPI networks. Our findings indicate that network size alone does not determine clinical relevance; instead, the choice of biologically meaningful gene sets and reliable interaction maps is more crucial for generating informative representations. While our current implementation of DGM is optimized for somatic mutation data, the framework is flexible and can be extended to integrate other omics modalities for improved patient stratification. DGM's ability to model the functional impact of sparse mutations using network context makes it applicable to other areas, such as rare disease subtyping and drug response prediction. Overall, the proposed approach demonstrates strong performance in capturing clinically relevant information from sparse mutation profiles, offering a practical solution for cancer subtype identification and prognosis, especially in resource-constrained settings where multi-omics data may not be available.

Methods

Omics data

Somatic mutation data for 16 distinct cancer types, along with the corresponding clinical information, were systematically retrieved from the Genomic Data Commons (GDC https://portal.gdc.cancer.gov) [29] data portal using the TCGABiolinks (2.25.0) R package [30] (Table 3). Only cancers with a cohort size exceeding 250 patient samples were considered. The somatic mutation data were then processed into patient-by-gene matrices, where "1" indicated the presence of a mutation and "0" its absence, irrespective of mutation type (e.g. missense, frameshift, etc.).

Network assembly

The primary network used in our study was assembled from genes in the Network of Cancer Genes (NCG) database [31], which includes both cancer-driver genes and noncancer clonal expansion genes. To capture functional interactions among these genes, we retrieved their edges from PCNet, a high-confidence parsimonious PPI network compiled by Huang *et al.* [20]. The resulting graph, referred to as the NCG Network, consisted of 3217 protein-coding genes and their corresponding interactions. This network was used uniformly across all cancer types, with patient-specific mutation data assigned to the node features. Additionally, we utilized two other networks to examine the impact of network choice on the performance. The first network, referred to as the CancerReference Network (CRN), included genes used in the PyNBS study [13], with edges extracted from PCNet. The second network, referred to as the unified network, was constructed by combining the edge lists from both the CRN and NCG Networks. The number of nodes, edges, and the extent of overlap between the NCG and CRN networks are shown in Figure S5.

Model architecture

Encoder

This study proposes two encoder designs, each featuring the same backbone architecture but employing different graph layers (Fig. 1). The first encoder, DGM, integrates graph attention layers, specifically employing a graph layer called TransformerConv [32]. In contrast, the second model, DGMS, utilizes the SAGEConv graph layer [33] with a mean aggregator. The incorporation of graph attention layers enables the model to prioritize crucial edges by computing attention weights, thereby enhancing the encoding process. Additionally, GraphNorm [34], a normalization strategy that leverages graph structure information, is applied to optimize the normalization procedure. This strategy presents

Table 3. Total number of samples considered in each cancer type

Abbreviation	Cancer type	Patient count
BLCA	Bladder urothelial carcinoma	407
LGG	Brain lower grade glioma	509
BRCA	Breast invasive carcinoma	968
CESC	Cervical squamous cell carcinoma	287
COAD	Colon adenocarcinoma	428
GBM	Glioblastoma multiforme	371
HNSC	Head and neck squamous cell carcinoma	508
KIRC	Kidney renal clear cell carcinoma	370
KIRP	Kidney renal papillary cell carcinoma	278
LIHC	Liver hepatocellular carcinoma	368
LUAD	Lung adenocarcinoma	557
LUSC	Lung squamous cell carcinoma	485
OV	Ovarian serous cystadenocarcinoma	407
SKCM	Skin cutaneous melanoma	467
STAD	Stomach adenocarcinoma	431
UCEC	Uterine corpus endometrial carcinoma	511

significant advantages over traditional normalization techniques, particularly in handling heterogeneous data types and complex network structures. The combination of these methods enhance the overall encoding process, leading to more robust and accurate representations of graph-structured data. For subsequent downstream analysis, the mean pooled output from the encoder layer was used to generate a feature list representing gene-wise information. The message-passing in the TransformerConv and the SAGEConv layers is governed by the following equations:

TransformerConv:

$$\mathbf{x}_i' = \mathbf{w}_1\mathbf{x}_i + \sum_{j\in\mathcal{N}(i)} \alpha_{i,j}\mathbf{w}_2\mathbf{x}_j, \tag{1}$$

where the attention coefficients $\alpha_{i,j}$ are computed via multi-head dot product attention:

$$\alpha_{i,j} = \text{softmax}\left(\frac{(\mathbf{w}_3\mathbf{x}_i)^\top(\mathbf{w}_4\mathbf{x}_j)}{\sqrt{d}}\right) \tag{2}$$

SageConv:

$$\mathbf{x}_i' = \mathbf{w}_1\mathbf{x}_i + \mathbf{w}_2 \cdot \frac{1}{n_i}\sum_{j\in\mathcal{N}(i)} \mathbf{x}_j \tag{3}$$

Here, x_i and x_j denote the feature vectors of node i and its neighbor j, respectively, while x_i' represents the updated feature vector of node i after applying the convolution. The learnable weight matrices w_1 and w_2 transform the features of the central node and its neighbors, while w_3 and w_4 are used to compute the attention coefficients $\alpha_{i,j}$, which determine the importance of neighbor j when updating the features of node i in the TransformerConv layer. The dimensionality of the feature vectors is represented by d, and the softmax function is used to normalize the attention coefficients. $\mathcal{N}(i)$ represents the set of neighbors of node i, and n_i is the number of neighbors used for normalization in SAGEConv.

Decoder

We propose a node decoder architecture with an integrated multi-head attention mechanism to overcome the limitations of traditional graph autoencoders that use edge decoders [35]. Traditional edge decoders generate an adjacency matrix by performing an inner product on the encodings, which is not suitable for our specific application. In our analysis, the graph structure remains consistent across all patient samples, making the adjacency matrix invariant and limiting the effectiveness of edge decoders in capturing discriminative representations. In contrast, our attention-augmented node decoder is designed to effectively capture node features while accommodating the static nature of the edge list across different patient samples.

Loss function

We used Focal Loss [36] to address the challenge of handling highly sparse node feature data. Focal loss is a modification of cross-entropy loss that allows for the modulation of importance given to 0s and 1s within the dataset through the adjustment of the hyperparameter gamma. The formulation of loss function is detailed in Equation (4). Traditional loss functions such as cross-entropy and mean squared error prove to be ineffective for our datasets as they predominantly account for the presence of mutations (1s) without adequately addressing the absence of mutation (0s). By employing Focal Loss, we enhance the model's ability to prioritize and accurately classify both the 0s and 1s, thereby improving the model's performance on our sparse dataset.

$$FL(p_t) = -\alpha_t(1-p_t)^\gamma \log(p_t) \tag{4}$$

In the given equation, (p_t) signifies the probability predicted for the actual class. The parameter (α_t) acts as a balancing factor between positive and negative instances, often assigned the value of the inverse class frequency. Additionally,(γ) serves as a focusing parameter.

Model hyperparameter tuning and training

In this study, 80% of the data were used for training and the rest 20% for validation. The optimal output channel size was identified as 10 through grid search analysis, which revealed that increasing the number of channels beyond this point did not significantly reduce the loss, and a smaller channel size would lead to over-squashing [37]. We set the learning rate at 1e-4 after testing to optimize model performance without compromising efficiency. The Gamma parameter in the focal loss was set to 2. The Adam optimizer was chosen for its effectiveness in optimizing the loss function and training was expedited using the PyTorch Lightning

and PyTorch Geometric frameworks. The number of parameters in both the encoder and decoder depends solely on the number of output channels considered in the study and remains constant across all cancers and networks considered. Experiments were conducted on NVIDIA GTX 2080Ti GPUs over 200 epochs. To optimize resource use, we employed a batch size of 2, accumulating gradients over four batches before backpropagation.

Unsupervised analysis of encodings

In our study, we used a consensus clustering framework to analyze the encodings from the graph autoencoder, with the goal of achieving robust and reproducible clustering results. This approach is implemented through two distinct clustering algorithms: the PAM algorithm [38], which is effective in handling outliers by selecting actual data points as cluster centers, and the K-means algorithm [39], which offers computational efficiency for large datasets despite being sensitive to initial conditions. The adoption of consensus clustering [40] serves to mitigate the variability inherent in clustering outcomes, ensuring that the clusters identified are stable and consistent across different iterations of the experiment.

The evaluation of cluster effectiveness and significance in this study was based on three critical metrics. Firstly, the CCC was used to assess the integrity of the consensus clustering. This coefficient measures the correlation between the original distances among data points and the distances represented in the clustering dendrogram, providing insight into how well the clustering solution reflects the original dataset's structure. A higher CCC indicates that the clustering arrangement more accurately represents the inherent relationships among data points, thus validating the effectiveness of consensus clustering. Secondly, the Silhouette score was calculated to evaluate how well the patients fit into the predicted clusters [41]. Lastly, the log-rank test[42] was used as the primary method to assess differences in survival probabilities across clusters. We applied a *P*-value threshold of <0.05 to identify significant clusters. These survival analyses were performed using the "lifelines" package in Python [43].

Supervised analysis of encodings

The Cox proportional hazards model was trained using the encodings for survival prediction [44]. We employed a split-sample methodology(70:30) and performed a five-fold cross-validation 100 times. We reported the mean C-Index, which is a reliable indicator of the model's performance. We also calculated the hazard score for each patient and categorized them into four groups based on the quantile hazard ratios. Survival differences between these groups were analyzed using Kaplan–Meier plot.

Performance benchmarking

We benchmarked DGM against three network-based methods: network propagation (PyNBS) [13], a network-based autoencoder, and a deep learning-based network embedding method (struc2vec) [15, 21]. PyNBS is a well-established method based on the concept of random walk with restart (RWR) to smooth sparse mutation profiles by integrating somatic mutation data with a PPI network. It then applies non-negative matrix factorization and consensus clustering on the smoothed profiles to stratify patients into subtypes.

A network-based autoencoder is a deep learning extension of the PyNBS approach, where smoothed mutation profiles are further processed through an autoencoder architecture (RWR-AE). The encoder consists of two fully connected layers: a dense layer with 500 neurons and ReLU activation, followed by a bottleneck layer with 100 neurons and ReLU activation. The decoder mirrors this structure and ends with an output layer that matches the input dimension, using a Sigmoid activation function. The model is trained using the Adam optimizer with a learning rate of 0.0001, reducing the high-dimensional smoothed mutation profiles to a compact 100-dimensional latent space. These low-dimensional representations are subsequently used to cluster patient samples using PAM, with consensus clustering applied to ensure robust subtype identification.

struc2vec, an extension of the DeepWalk framework [45], is a graph-based structural embedding model used to generate vector representations of nodes in a PPI network. It first computes pairwise node similarities based on a hierarchical structure, followed by the construction of a multilayer weighted graph. The resulting graph is then used to generate structural contexts for the nodes through biased random walks. These node sequences are used to train a Skip-Gram model, resulting in 128 embedding vectors for each gene node. This approach captures genes that are distant in the network but structurally similar by embedding them close together in the feature space. These embeddings were used to construct patient-level features by mapping each patient's mutated genes onto these embedding vectors. The embedded genes were then grouped into 10 bins based on network degree, using degree-based thresholds such that each bin represented ~10% of genes according to their degree distribution (Figure S7). Genes with very high degrees (above the 90th percentile) were excluded. For each patient, the embeddings of mutated genes within each bin were summed, resulting in ten 128-dimensional vectors. These were concatenated to form a 1280-dimensional feature vector, which was subsequently used for clustering using PAM with consensus clustering. Benchmarking was done for both unsupervised and supervised tasks using three metrics: CCC (for clustering), log-rank *P*-value (for clinical relevance), and C-index (for survival prediction).

Characterization of subtypes

To characterize differences among the identified subtypes, we analyzed TMB across cancer types. For each sample, mutation counts were calculated and log10 transformed to account for the skewed distribution of mutation data. Subtype-level differences in TMB were assessed using the Kruskal–Wallis test, a nonparametric method suitable for comparing multiple groups without assuming normality. Further, associations between subtypes and clinical variables, including histological type and tumor grade, were assessed using the chi-square test. A *P*-value <0.05 was considered statistically significant.

Key Points

- DeepGraphMut (DGM) is a novel graph-based deep learning pipeline to identify cancer subtype and predict survival using only somatic mutation data.
- It integrates somatic mutation data with cancer-specific protein–protein interaction network to generate personalized patient encodings.
- DGM has been successfully applied to thousands of patient samples across 16 cancer types, uncovering distinct cancer subtypes with significant survival differences.
- It serves as a valuable tool for enhancing precision medicine.

Author contributions

Aswin Jose (Formal analysis, Investigation, Methodology, Writing original draft, Writing review & editing), Akansha Srivastava (Formal analysis, Investigation, Writing original draft, Writing review & editing), Ariba Ansari (Formal analysis, Investigation, Writing review & editing), and P.K. Vinod (Funding acquisition, Supervision, Conceptualization, Writing review & editing)

Supplementary data

Supplementary data is available at *Briefings in Bioinformatics* online.

Competing interests

The authors have no competing interests to declare.

Funding

This work was supported by iHUB-Data, International Institute of Information Technology, Hyderabad, India.

Data availability

The codes used in this study is available from the GitHub page: https://github.com/CancerDiag/DeepGraphMut

References

1. Bray F, Laversanne M, Sung H. *et al*. Global cancer statistics 2022: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA Cancer J Clin* 2024;**74**:229–63. https://doi.org/10.3322/caac.21834
2. Carbone A. Cancer classification at the crossroads. *Cancers* 2020;**12**:980.
3. Vargo-Gogola T, Rosen JM. Modelling breast cancer: one size does not fit all. *Nat Rev Cancer* 2007;**7**:659–72. https://doi.org/10.1038/nrc2193
4. Kulavi S, Ghosh C, Saha M. *et al*. One size does not fit all: an overview of personalized treatment in cancer. *J Pharm Res Int* 2021;**33**:87–103. https://doi.org/10.9734/jpri/2021/v33i28A31513
5. Cooper GM. *The Cell: A Molecular Approach*2nd edn. Sunderland (MA): Sinauer Associates, 2000.
6. Reuter JA, Spacek DV, Snyder MP. High-throughput sequencing technologies. *Mol Cell* 2015;**58**:586–97. https://doi.org/10.1016/j.molcel.2015.05.004
7. Carter H, Chen S, Isik L. *et al*. Cancer-specific high-throughput annotation of somatic mutations: computational prediction of driver missense mutations. *Cancer Res* 2009;**69**:6660–7. https://doi.org/10.1158/0008-5472.CAN-09-1133
8. Barabasi A-L, Oltvai ZN. Network biology: understanding the cell's functional organization. *Nat Rev Genet* 2004;**5**:101–13. https://doi.org/10.1038/nrg1272
9. Redhu N, Thakur Z. Chapter 23 - Network biology and applications. *Bioinformatics*. Academic Press, 2022;381–407. https://doi.org/10.1016/B978-0-323-89775-4.00024-9
10. Zhang P, Itan Y. Biological network approaches and applications in rare disease studies. *Genes* 2019;**10**:797. https://doi.org/10.3390/genes10100797
11. Hofree M, Shen JP, Carter H. *et al*. Network-based stratification of tumor mutations. *Nat Methods* 2013;**10**:1108–15. https://doi.org/10.1038/nmeth.2651
12. Zhong X, Yang H, Zhao S. *et al*. Network-based stratification analysis of 13 major cancer types using mutations in panels of cancer genes. *BMC Genomics* 2015;**16**:1–8. https://doi.org/10.1186/1471-2164-16-S7-S7
13. Huang JK, Jia T, Carlin DE. *et al*. pyNBS: a python implementation for network-based stratification of tumor mutations. *Bioinformatics* 2018;**34**:2859–61. https://doi.org/10.1093/bioinformatics/bty186
14. Di Nanni N, Bersanelli M, Milanesi L. *et al*. Network diffusion promotes the integrative analysis of multiple omics. *Front Genet* 2020;**11**:488641.
15. Liu C, Han Z, Zhang Z-K. *et al*. A network-based deep learning methodology for stratification of tumor mutations. *Bioinformatics* 2021;**37**:82–8. https://doi.org/10.1093/bioinformatics/btaa1099
16. Xia F, Sun K, Yu S. *et al*. Graph learning: a survey. *IEEE Trans Artif Intell* 2021;**2**:109–27. https://doi.org/10.1109/TAI.2021.3076021
17. Wekesa JS, Kimwele M. A review of multi-omics data integration through deep learning approaches for disease diagnosis, prognosis, and treatment. *Front Genet* 2023;**14**:1199087. https://doi.org/10.3389/fgene.2023.1199087
18. Liang C, Shang M, Luo J. Cancer subtype identification by consensus guided graph autoencoders. *Bioinformatics* 2021;**37**:4779–86. https://doi.org/10.1093/bioinformatics/btab535
19. Baul S, Ahmed KT, Filipek J. *et al*. omicsGAT: graph attention network for cancer subtype analyses. *Int J Mol Sci* 2022;**23**:10220. https://doi.org/10.3390/ijms231810220
20. Huang JK, Carlin DE, Yu MK. *et al*. Systematic evaluation of molecular networks for discovery of disease genes. *Cell Syst* 2018;**6**:484–495.e5. https://doi.org/10.1016/j.cels.2018.03.001
21. Ribeiro LF, Saverese PH, Figueiredo DR. struc2vec: learning node representations from structural identity. *KDD'17: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining* 2017, 385–394. https://doi.org/10.1145/3097983.3098061
22. Li L, Bai L, Lin H. *et al*. Multiomics analysis of tumor mutational burden across cancer types. *Comput Struct Biotechnol J* 2021;**19**:5637–46. https://doi.org/10.1016/j.csbj.2021.10.013 Open access under CC BY-NC-ND 4.0
23. Chalmers ZR, Connelly CF, Fabrizio D. *et al*. Analysis of 100,000 human cancer genomes reveals the landscape of tumor mutational burden. *Genome Med* 2017;**9**:34. https://doi.org/10.1186/s13073-017-0424-2
24. Voutsadakis IA. Urothelial bladder carcinomas with high tumor mutation burden have a better prognosis and targetable molecular defects beyond immunotherapies. *Curr Oncol* 2022;**29**:1390–407. https://doi.org/10.3390/curroncol29030117
25. Meng A, Yuile A, Sim H-W. *et al*. A systematic review and meta-analysis of the impact of tumour mutational burden on survival outcomes in solid tumours. *medRxiv [preprint]* 2025.
26. Hao Z, Guo D. EGFR mutation: novel prognostic factor associated with immune infiltration in lower-grade glioma; an exploratory study. *BMC Cancer* 2019;**19**:1184. https://doi.org/10.1186/s12885-019-6384-8
27. Zhang Y, Dube C, Gibert M. *et al*. The p53 pathway in glioblastoma. *Cancers* 2018;**10**:297. https://doi.org/10.3390/cancers10090297
28. Cheerla A, Gevaert O. Deep learning with multimodal representation for pancancer prognosis prediction. *Bioinformatics* 2019;**35**:i446–54. https://doi.org/10.1093/bioinformatics/btz342
29. Grossman RL, Heath AP, Ferretti V. *et al*. Toward a shared vision for cancer genomic data. *N Eng J Med* 2016;**375**:1109–12. https://doi.org/10.1056/NEJMp1607591

30. Huber W, Carey VJ, Gentleman R. *et al*. Orchestrating high-throughput genomic analysis with bioconductor. *Nat Methods* 2015;**12**:115–21. https://doi.org/10.1038/nmeth.3252
31. Repana D, Nulsen J, Dressler L. *et al*. The network of cancer genes (NCG): a comprehensive catalogue of known and candidate cancer genes from cancer sequencing screens. *Genome Biol* 2019;**20**:1–12. https://doi.org/10.1186/s13059-018-1612-0
32. Shi Y, Huang Z, Feng S. *et al*. Masked label prediction: unified message passing model for semi-supervised classification. *IJCAI'21: Proceedings of the 30th International Joint Conference on Artificial Intelligence* 2021;1548–1554. https://doi.org/10.24963/ijcai.2021/214
33. Hamilton W, Ying Z, Leskovec J. Inductive representation learning on large graphs. *NIPS'17: Proceedings of the 31st Conference on Neural Information Processing Systems* 2017;1025–1035. https://dl.acm.org/doi/10.5555/3294771.3294869
34. Cai T, Luo S, Xu K. *et al*. GraphNorm: a principled approach to accelerating graph neural network training. *ICML'21: Proceedings of the 38th International Conference on Machine Learning* 2021;**139**:1204–1215.
35. Kipf TN, Welling M. Variational graph auto-encoders. *arXiv[preprint]* 2016. https://doi.org/10.48550/arXiv.1611.07308
36. Lin T-Y, Goyal P, Girshick R. *et al*. Focal loss for dense object detection. *ICCV'17: Proceedings of the IEEE international conference on computer vision* 2017;2999–3007. https://doi.org/10.1109/ICCV.2017.324
37. Alon U, Yahav E. On the bottleneck of graph neural networks and its practical implications. *ICLR'21: Proceedings of the 9th International Conference on Learning Representations* 2021. https://openreview.net/forum?id=i80OPhOCVH2
38. Park H-S, Jun C-H. A simple and fast algorithm for K-medoids clustering. *Expert Syst Appl* 2009;**36**:3336–41. https://doi.org/10.1016/j.eswa.2008.01.039
39. Jin X, Han J. K-means clustering. In: Sammut C, Webb GI (eds) *Encyclopedia of Machine Learning*. Boston, MA: Springer, 2011. https://doi.org/10.1007/978-0-387-30164-8_425
40. Strehl A, Ghosh J. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *J Mach Learn Res* 2002;**3**:583–617.
41. Rousseeuw PJ. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J Comput Appl Math* 1987;**20**:53–65. https://doi.org/10.1016/0377-0427(87)90125-7
42. Kaplan EL, Meier P. Nonparametric estimation from incomplete observations. *J Am Stat Assoc* 1958;**53**:457–81. https://doi.org/10.1080/01621459.1958.10501452
43. Davidson-Pilon C. Lifelines: survival analysis in python. *J Open Source Software* 2019;**4**:1317. https://doi.org/10.21105/joss.01317
44. Cox DR. Regression models and life-tables. *J R Stat Soc B Methodol* 1972;**34**:187–202. https://doi.org/10.1111/j.2517-6161.1972.tb00899.x
45. Perozzi B, Al-Rfou R, Skiena S. DeepWalk: online learning of social representations. *KDD'14: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* 2014, 701–710. https://doi.org/10.1145/2623330.2623732

Briefings in Bioinformatics, 2025, **26(4)**, bbaf409
https://doi.org/10.1093/bib/bbaf409
Problem Solving Protocol