BIG DATA MINING AND ANALYTICS
ISSN 2096-0654 14/25 pp765–780
DOI: 10.26599/BDMA.2024.9020002
Volume 7, Number 3, September 2024

SGCL-LncLoc: An Interpretable Deep Learning Model for Improving lncRNA Subcellular Localization Prediction with Supervised Graph Contrastive Learning

Min Li, Baoying Zhao, Yiming Li, Pingjian Ding, Rui Yin, Shichao Kan, and Min Zeng*

Abstract: Understanding the subcellular localization of long non-coding RNAs (lncRNAs) is crucial for unraveling their functional mechanisms. While previous computational methods have made progress in predicting lncRNA subcellular localization, most of them ignore the sequence order information by relying on k-mer frequency features to encode lncRNA sequences. In the study, we develope SGCL-LncLoc, a novel interpretable deep learning model based on supervised graph contrastive learning. SGCL-LncLoc transforms lncRNA sequences into de Bruijn graphs and uses the Word2Vec technique to learn the node representation of the graph. Then, SGCL-LncLoc applies graph convolutional networks to learn the comprehensive graph representation. Additionally, we propose a computational method to map the attention weights of the graph nodes to the weights of nucleotides in the lncRNA sequence, allowing SGCL-LncLoc to serve as an interpretable deep learning model. Furthermore, SGCL-LncLoc employs a supervised contrastive learning strategy, which leverages the relationships between different samples and label information, guiding the model to enhance representation learning for lncRNAs. Extensive experimental results demonstrate that SGCL-LncLoc outperforms both deep learning baseline models and existing predictors, showing its capability for accurate lncRNA subcellular localization prediction. Furthermore, we conduct a motif analysis, revealing that SGCL-LncLoc successfully captures known motifs associated with lncRNA subcellular localization. The SGCL-LncLoc web server is available at http://csuligroup.com:8000/SGCL-LncLoc. The source code can be obtained from https://github.com/CSUBioGroup/SGCL-LncLoc.

Key words: supervised contrastive learning; long non-coding RNA (lncRNA); subcellular localization prediction; deep learning; Graph Convolutional Network (GCN)

• Min Li, Baoying Zhao, Yiming Li, Shichao Kan, and Min Zeng are with School of Computer Science and Engineering, Central South University, Changsha 410083, China. E-mail: limin@mail.csu.edu.cn; baoyingzhao@csu.edu.cn; lym1998@csu.edu.cn; kanshichao@csu.edu.cn; zengmin@csu.edu.cn.
• Pingjian Ding is with Center for Artificial Intelligence in Drug Discovery, Case Western Reserve University, Cleveland, OH 44106, USA. E-mail: pxd210@case.edu.
• Rui Yin is with Department of Health Outcomes and Biomedical Informatics, University of Florida, Gainesville, FL 32603, USA. E-mail: ruiyin@ufl.edu.
* To whom correspondence should be addressed.
Manuscript received: 2023-11-22; revised: 2023-12-28; accepted: 2024-01-02

1 Introduction

With the rapid development of genomics and transcriptomics, long non-coding RNAs (lncRNAs) have attracted extensive attention as an important class of transcriptional products[1, 2]. Unlike mRNAs, lncRNAs do not encode proteins. They participate in crucial biological processes, such as gene expression regulation, cell cycle regulation, cell differentiation, and the onset and progression of diseases by interacting with proteins, DNAs or other RNAs[3]. lncRNAs in the nucleus serve various functions. Firstly, they can interact with regulatory proteins in enhancer region and participate in gene transcriptional regulation, thereby influencing gene expression levels[4]. Secondly, they act as guide molecules to guide processes, such as chromatin remodeling, transcription factor localization, and chromatin interactions[5]. Additionally, they can interact with proteins or other nucleic acid molecules, competitively binding to interfere with or transfer the binding of other molecules, thereby regulating gene expression and genome stability[6]. Furthermore, they are involved in the regulation of three-dimensional chromatin structure, and affect gene expression regulation[7]. In the cytoplasm, lncRNAs also play crucial roles. Firstly, they interact with mRNA, regulating translation level and stability, thereby influencing protein synthesis[8]. Secondly, they act as sponges, interacting with miRNAs or proteins to regulate their activity and stability in cellular processes[9]. Additionally, they possess functional sequences that can be translated into small peptides, potentially participating in biological processes, such as cell cycle regulation, cell proliferation or apoptosis[10]. Therefore, studying the subcellular localization of lncRNA is of great significance to reveal their functional mechanisms and their effects on biological processes and diseases[11].

Several biology experimental techniques are available to identify the subcellular localization of lncRNAs. Fluorescence in situ hybridization is a widely used technique to accurately detect the localization of lncRNAs within cells. By labeling the lncRNA with a complementary probe, the exact location of the lncRNA in the nucleus or cytoplasm can be directly visualized. Despite existing biological experimental techniques play important roles in the study of lncRNA subcellular localization, they are expensive, time-consuming, and technically challenging. Given these limitations, it is of great interest for biologists to develop accurate and reliable computational methods to predict lncRNA subcellular localization.

Nowadays, many computational methods have been proposed for predicting lncRNA subcellular localization. Among these studies, the widely used feature is *k*-mer frequency features. For example, iLoc-lncRNA[12], Locate-R[13], and iLoc-lncRNA(2.0)[14], extract *k*-mer frequency features from lncRNA sequences, and employ feature selection methods. The selected optimal features are then fed into Support Vector Machine (SVM) to obtain the final classification probability. LncLocator[15] uses 4-mer frequency features and high-level features extracted by autoencoders, followed by SVM and Random Forest (RF) for predictions. RNAlight[16] develops a machine learning model based on light gradient boosting machine, using *k*-mer frequency features for identifying subcellular localization of both mRNAs and lncRNAs. GM-LncLoc[17] calculates the cosine similarity of *k*-mer frequency features to construct a lncRNA sequence similarity network, and preforms subcellular localization prediction through Graph Convolutional Networks (GCNs). In addition to the exclusive use of *k*-mer frequency features, some researches have started exploring the fusion of *k*-mer frequency features with other features. For example, DeepLncRNA[18] combines *k*-mer frequency features with known RNA-binding protein motif sites and genomic characteristics. lncLocPred[19] combines *k*-mer frequency features with Triplet and PseDNC features. lncLocation[20] integrates multi-source heterogeneous features, including *k*-mer frequency features, physical-chemical properties, and secondary structure features. TACOS[21] combines *k*-mer frequency features with dinucleotide physicochemical properties and PseKNC features. LightGBM-LncLoc[22] incorporates a variant of *k*-mer frequency features and position-specific trinucleotide propensity. With the rapid development in deep learning and natural language processing, researches have begun to exploring the utilization of word embedding techniques. For instance, lncLocator 2.0[23] uses the GloVe technique to learn word embedding vector for each 6-mer. DeepLncLoc[24] and LncLocFormer[25] use the Word2Vec technique to learn word embedding vector for each 3-mer.

Although previous methods for lncRNA subcellular localization prediction have made some progress, they often ignore the crucial sequence order information by utilizing *k*-mer frequency features to encode lncRNA sequences. This limitation hampers their ability to accurately predict the subcellular localization of lncRNAs. In our previous work, GraphLncLoc[26], we address this issue by transforming lncRNA sequences into de Bruijn graphs. This transformation allows us to convert the sequence classification problem into a graph classification problem. By transforming lncRNA sequences into de Bruijn graphs, GraphLncLoc keeps the local order information of lncRNA sequences, resulting in more distinguishable features and thus yielding improved prediction results. In the study, building upon the success of GraphLncLoc, we further propose SGCL-LncLoc, an interpretable deep learning model based on supervised graph contrastive learning.

SGCL-LncLoc and GraphLncLoc share a common approach of transforming lncRNA sequences into graphs, and utilizing graph neural networks for feature extraction. However, SGCL-LncLoc differs from GraphLncLoc in two main aspects. Firstly, SGCL-LncLoc incorporates supervised contrastive learning, which enables the model to learn the relationships between different samples by comparing the similarity of samples within the same category and the dissimilarity between samples from different categories. This approach leverages both the relationships between different samples and the label information of samples, guiding the model to enhance representation learning for lncRNAs. In contrast, conventional graph neural network frameworks in GraphLncLoc primarily rely on the graph's topology (nodes and edges) for learning, without explicitly considering the similarities and differences between samples of the same or different categories. The second difference is that SGCL-LncLoc serves as an interpretable deep learning model. SGCL-LncLoc introduces a global attention pooling mechanism, allowing the model to learn the weights of nodes in the graph. Furthermore, SGCL-LncLoc proposes a computational method for mapping the weights of nodes in the graph to the weights of each nucleotide in the sequence. In contrast, GraphLncLoc does not consider the visualization of the weights of nucleotides in the sequence, and thus it cannot provide nucleotide-level interpretability. By incorporating supervised contrastive learning and the global attention pooling mechanism, SGCL-LncLoc enhances both the discriminative power and interpretability.

We conduct extensive experiments to evaluate the effectiveness of SGCL-LncLoc. We compare it with various machine learning and deep learning baseline models, and existing predictors. The results clearly demonstrate that SGCL-LncLoc outperforms these models. Furthermore, we conducte an ablation study to analyze the effects of different components on the prediction performance. The results confirm the contribution of each component. To provide further insights, we present t-SNE visualizations of the graph vectors obtained from SGCL-LncLoc both with and without contrastive learning. These visualizations clearly highlight the advantages of employing contrastive learning. In addition, we demonstrat SGCL-LncLoc's ability to capture both the most frequently occurring motifs and known nucleus localization motifs, enabling a nucleotide-level interpretation. Finally, to facilitate the accessibility and usability of SGCL-LncLoc, we develop a user-friendly web server. We hope that SGCL-LncLoc will contribute to the field by enabling researchers to make more accurate predictions for lncRNA subcellular localization.

2 Material and Method

2.1 Dataset

In order to collect comprehensive data for lncRNA subcellular localization, we integrate lncRNA entries from different databases. First, we acquire a lncRNA dataset consisting of 3792 samples from RNAlight[16]. The dataset encompasses data from three different sources: lncATLAs[27], CeFra-seq[28], and APEX-seq[29]. Second, we collect 653 lncRNA sequences from the RNALocate v1.0[30] database. It should be noted that, in this study, we focus on lncRNAs in two major subcellular locations, the nucleus and cytoplasm, for model training and prediction. Then we merge the two datasets and remove redundant sequences using the CD-HIT-EST[31] tool with a cut-off rate of 80%. Finally, we obtain a benchmark dataset comprising 4334 lncRNA sequences, including 2105 lncRNAs localized in the cytoplasm and 2229 lncRNAs localized in the nucleus. The flowchart describing the process of obtaining the benchmark dataset is shown in Fig. S1, which is in the Electronic Supplementary Material

(ESM) of the online version of this article. Furthermore, we conduct an analysis of the sequence length distribution across subcellular localization compartments, as shown in Fig. 1a. Notably, the length of lncRNA sequences in the benchmark dataset range from approximately 200 nt to 200 000 nt. Upon thorough examination, we found that most sequences fall within the range of approximately 200 nt to 10 000 nt. Therefore, we set the lower boundary of the sequence length as 200 nt and the upper boundary as 10 000 nt, then remove the outliers outside the boundary. In addition, Fig. 1a shows that lncRNA sequences localized in the nucleus tend to be generally longer than those found in the cytoplasm.

To evaluate the prediction performance of our model, we generate an independent test set by collecting lncRNA localization entries from the RNALocate v2.0 database[32]. The process of generating the independent test set is outlined as follows:

(1) 38 677 RNA subcellular localization entries are retrieved from the RNALocate v2.0 database.

(2) From these entries, we screen and select a total of 816 lncRNA subcellular localization data pertaining to the cytoplasm and nucleus.

(3) We remove lncRNA subcellular localization data with blank Gene IDs, resulting in a set of 573 lncRNAs.

(4) According to the Gene IDs, we search for corresponding transcript sequences in the sequence dataset provided by the RNALocate v2.0 database. We merge sequences with the same Gene ID, resulting in 158 lncRNA transcript sequences.

(5) The CD-HIT-EST tool with a cutoff of 80% is used to eliminate redundant lncRNA sequences. Finally, we obtain 151 lncRNA sequences, comprising 65 lncRNAs localized in the cytoplasm and 86 lncRNAs localized in the nucleus.

Thus far, we have collected a benchmark dataset consisting of 4334 lncRNAs, including 2105 lncRNAs localized in the cytoplasm and 2229 lncRNAs localized in the nucleus. Additionally, we collect an independent test set of 151 lncRNA sequences from the RNALocate v2.0 database. Similarly, we conducted an analysis of the sequence length distribution across subcellular localization compartments for the independent test set, as illustrated in Fig. 1b. Furthermore, we summarize the distributions of both the benchmark dataset and the test set in Table 1.

2.2 Model architecture

Figure 2 illustrates the architecture of the SGCL-LncLoc model. SGCL-LncLoc takes lncRNA sequences as input, and produces predicted probabilities for each subcellular localization of the lncRNA, along with the attention score of each nucleotide in the sequence. The main idea of GraphLncLoc is to transform lncRNA sequences into de Bruijn graphs, then use the Word2Vec technique to encode node features. Subsequently, GCN are applied to extract high-level features from the graph. Building upon this foundation, SGCL-LncLoc incorporates the supervised contrastive learning strategy and a global attention pooling mechanism to establish an interpretable deep learning model, so as to improve the prediction performance of lncRNA subcellular localization prediction.

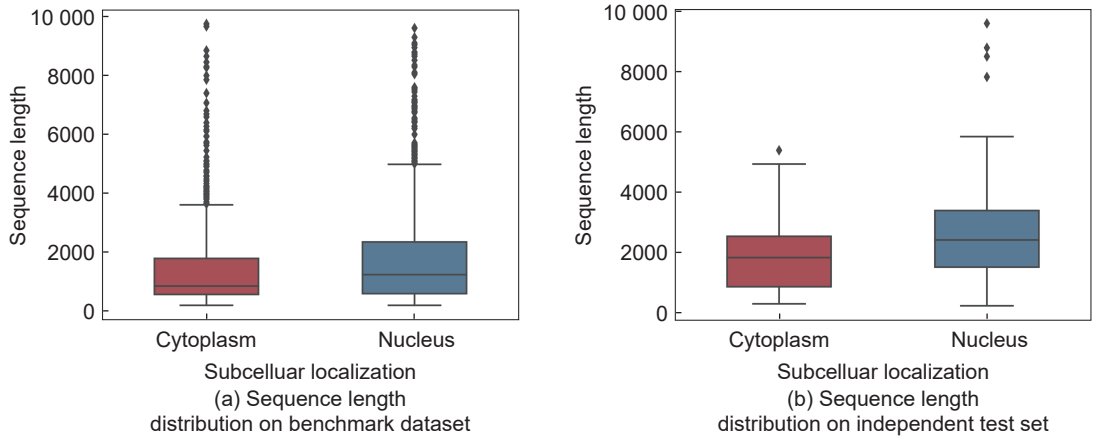


Fig. 1 Distribution of sequence lengths across subcellular localization compartments on the benchmark dataset and independent test set.

Table 1 Distributions of the benchmark dataset and independent test set.

Subcellular localization	Number of lncRNAs in benchmark dataset	Number of lncRNAs in independent test set
Cytoplasm	2105	65
Nucleus	2229	86
Total	4334	151

2.3 Graph construction

Before feed a lncRNA sequence into the GCN layer, SGCL-LncLoc transforms it into a de Bruijn graph. The de Bruijn graph construction method used in SGCL-LncLoc follows the original GraphLncLoc[26]. The method keeps the local order information of the sequence and can capture patterns and motifs of different lengths. The steps of transforming a lncRNA sequence into a de Bruijn graph are as follows:

(1) Define a k-mer as a substring of length k within the lncRNA sequence. Typically, k is set to a specific value, such as $k = 5$.

(2) Construct nodes in the de Bruijn graph based on the k-mers present in the lncRNA sequence. Each k-mer represents a node in the graph.

(3) Connect the nodes in the de Bruijn graph by adding edges that represent the overlapping relationship between adjacent k-mers. If two k-mers share a common (k−1)-mer suffix and prefix, an edge is added between the corresponding nodes in the graph.

(4) Repeat Steps (2) and (3) for all k-mers in the lncRNA sequence, thereby creating additional nodes and edges in the de Bruijn graph.

Once the de Bruijn graph is constructed, SGCL-LncLoc employs the Word2Vec technique to learn continuous distributed word vector for each k-mer (i.e.,

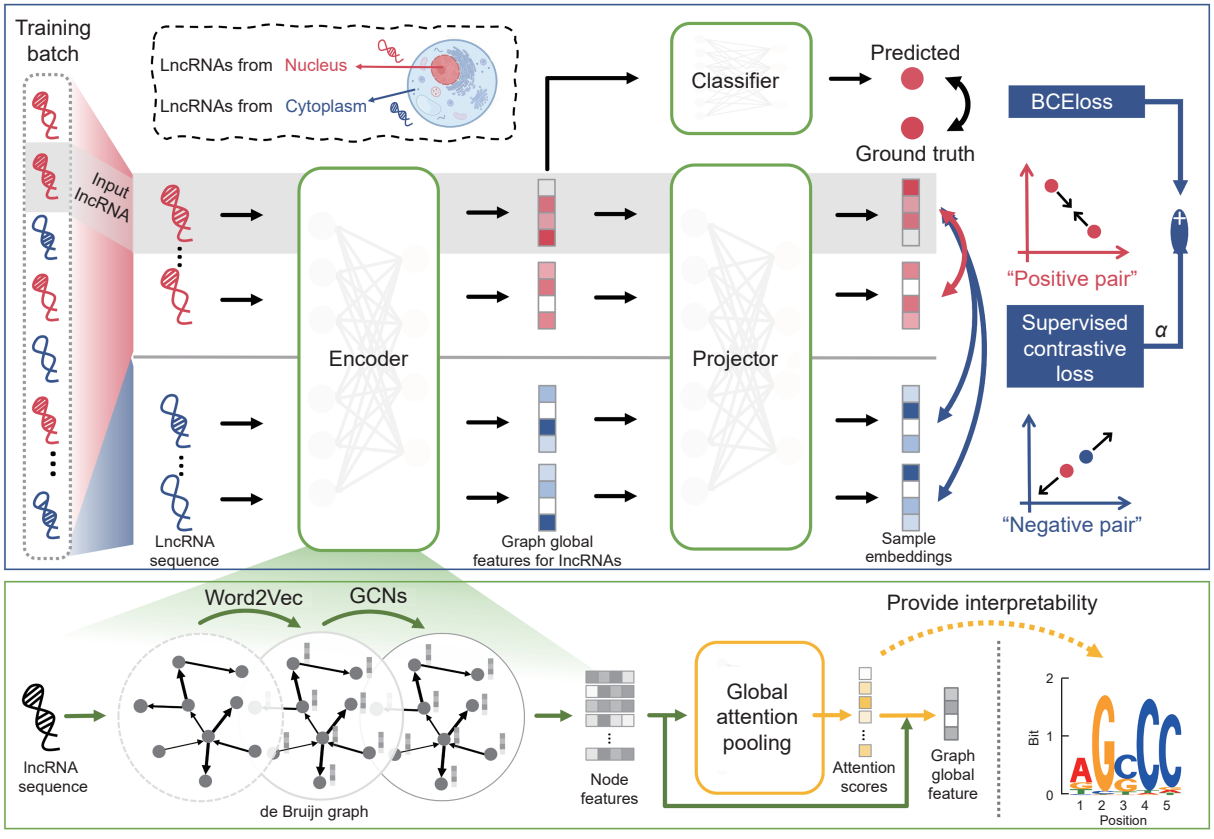


Fig. 2 SGCL-LncLoc model architecture. First, the input lncRNA sequence is transformed into a de Bruijn graph. Then, the graph convolutional network acts as the encoder to capture the node features of the de Bruijn graph. The global attention pooling mechanism is then applied to obtain the node weights and the representation of the entire graph. The resulting graph representation is simultaneously fed into two downstream modules, classifier and projector. In the classifier module, a linear layer with a sigmoid function is applied to output the predicted probability of subcellular localization. In the projector module, a linear layer is applied to project the representation of the entire graph to another embedding space. Here, supervised contrastive learning is applied to obtain better representations for lncRNA sequences belonging to different subcellular localizations.

each node in the de Bruijn graph). The entire process of de Bruijn graph construction is illustrated in Fig. S2 in the ESM. We refer to the original publication[26] on de Bruijn graph construction for more details.

2.4 Graph convolutional networks

After constructing the de Bruijn graphs, we trained a GCN model as an encoder to extract graph features. The main idea of GCN is to learn node representation by using the relationship information among nodes and their neighbors. Thus, this approach enables GCN to capture both local and global information, achieved through iterative updates to the node representation layer by layer[33–35]. These updates are performed through a convolution operation, which can be represented as follows:

$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right) \tag{1}$$

where $\tilde{D}$ represents the degree matrix of $\tilde{A}$, $\tilde{A} = A + I$, I is an identity matrix, $\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$ denotes a symmetric normalization of the adjacency matrix A. $H^{(l)}$ corresponds to the feature matrix of the l-th layer. $W^{(l)}$ represents the weight of the l-th layer, and σ denotes the nonlinear activation function.

By employing the GCN model, we can effectively encode and extract useful information from the de Bruijn graphs. This enables the extraction of meaningful features that encompass the topological characteristics and relationships between nodes within the graph. We refer to the original publication on graph convolutional networks[26] for more details.

2.5 Global attention pooling mechanism

SGCL-LncLoc uses a global attention pooling mechanism to obtain node weights and the overall graph representation. Firstly, each node in the graph is assigned an attention score that represents its contribution to subcellular localization prediction. Then, the encoding vector of each node is multiplied by its corresponding attention score, resulting in a representation of the entire graph. The attention score α is calculated according to

$$\alpha = \mathrm{softmax}\left(HW^{\mathrm{T}} + b\right) \tag{2}$$

where H denotes the output matrix of the last layer of the GCNs, as referenced in Eq. (1), representing the node feature matrix with size of $n\times h$, where n represents the number of nodes in the graph, and h represents the dimension of the node features. The size of attention score α is $n\times 1$. W is the weight matrix and b is the bias.

To obtain the representation of the entire graph, the attention score is multiplied by the encoding vector of the corresponding node, as shown in the following:

$$v_{\mathrm{G}} = \alpha^{\mathrm{T}}H \tag{3}$$

where α is the attention score with size of $n\times 1$. v_{G} is the vector of the entire graph, with size of $1\times h$.

2.6 Mapping node weights in graphs to nucleotide weights in sequences

In our study, we employ a global attention mechanism to obtain attention weights for each node in the graph. However, the attention weights are specific to graph nodes and do not directly reflect the weights of nucleotides in the sequence. To address this, we propose a computational method to map the attention weights of graph nodes to nucleotide weights in the lncRNA sequence.

In order to map the node weights to the nucleotide weights, we borrow the idea of convolution operations employed in convolutional neural networks. Specifically, we use a sliding window with a window size of 5 nucleotides and a stride of 1 nucleotide, and then cross the entire sequence from left to right. For a given lncRNA sequence, for nucleotide t at position j (where $5 \leqslant j \leqslant L-4$, L denotes the sequence length) in the sequence, nucleotide t appears in 5 sliding windows, as illustrated in Fig. 3. Consequently, we calculate the average value of the attention weights from these 5 sliding windows (corresponding to 5 nodes in the graph) as the attention score for nucleotide t at position j.

In other scenarios, when $1 \leqslant j < 5$ (that is, when nucleotide t appears in the first four nucleotides of the sequence), nucleotide t only appears in j sliding windows. In this case, we calculate the average value of these j sliding windows as the attention score for nucleotide t at position j. Similarly, when $L-4 < j \leqslant L$ (that is, when nucleotide t appears in the last four nucleotides of the sequence), nucleotide t also appears in $L-j+1$ sliding windows. The three cases in which the sliding window computes the attention weights during the sliding process are shown in the following:

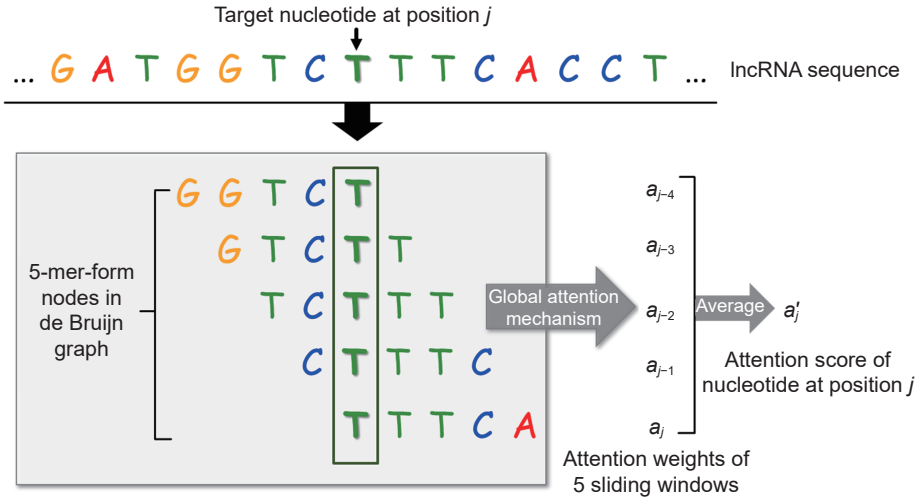


Fig. 3 Illustration of weight calculation for nucleotide t at position j ($5 \leqslant j \leqslant L-4$).

$$a_j' = \begin{cases} \dfrac{\sum_{k=1}^{j} a_k}{j}, & \text{if } 1 \leqslant j < 5; \\ \dfrac{\sum_{k=j-4}^{j} a_k}{5}, & \text{if } 5 \leqslant j \leqslant L-4; \\ \dfrac{\sum_{k=j-4}^{L-4} a_k}{L-j+1}, & \text{if } L-4 < j \leqslant L \end{cases} \tag{4}$$

where j refers to the j-th nucleotide in the sequence, $j \in \{1, 2, \ldots, L\}$, k refers to the k-th 5-mer in the sequence, $k \in \{1, 2, \ldots, L-4\}$, a_k is the attention weight of the k-th 5-mer node, and a_j' is the attention weight of the j-th nucleotide.

Accordingly, we calculate the average value of these $L-j+1$ sliding windows as the attention score for nucleotide t at position j. This mapping method allows us to precisely identify the important nucleotide fragments.

2.7 Supervised contrastive learning

Contrastive learning is a kind of self-supervised learning, has gained significant attention in recent years[36]. Unlike traditional supervised learning methods that rely on labeled data, contrastive learning leverages unlabeled data to acquire knowledge. It has emerged as a research frontier in the field of computer vision and natural language processing[37]. Moreover, contrastive learning has been extended and applied to supervised learning tasks, resulting in improved classification results[38, 39]. In supervised learning task, contrastive learning focuses on learning common features among samples of the same category while emphasizing differences between samples of different categories. By making the encoding features of different categories more distinguishable and distinct, supervised contrastive learning contributes to improving the classification performance of lncRNA subcellular localization.

Specifically, for a given lncRNA, we consider it has multiple positive and negative samples. The definition of "positive" and "negative" is determined by the labels of the samples. For example, suppose we randomly sample a batch of de Bruijn graphs. For a given lncRNA i in the nucleus, other lncRNAs in the nucleus in the batch would be regarded as the positive samples of lncRNA i. Thus, any pair consisting of lncRNA i and another lncRNA in the nucleus forms a positive pair. On the other hand, lncRNAs located in the cytoplasm would serve as negative samples of lncRNA i. Consequently, any pair consisting of lncRNA i and another lncRNA in the cytoplasm forms a negative pair. The supervised contrastive loss function is formulated as follows:

$$\text{SupContrastLoss} = \sum_{i \in S} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp\left(\frac{v_i v_p}{\tau}\right)}{\sum_{a \in A(i)} \exp\left(\frac{v_i v_a}{\tau}\right)} \tag{5}$$

where for each sample i, $P(i)$ represents the positive sample set of sample i, $|P(i)|$ denotes the number of positive samples for sample i, p represents one of the positive samples. $A(i)$ represents the negative sample set of sample i, and a is one of the negative samples. v_i is the vector representation of sample i, and v_p and v_a refer to the vector representations of sample p and sample a, respectively. The parameter $\tau \in \mathbf{R}^+$, represents the temperature coefficient and is set to 0.2

in the study. S represents the entire sample set.

2.8 Implementation details

All training, testing, and evaluation experiments are performed using NVIDIA Corporation Device 2204 Graphics Processing Units (GPU). The Computed Unified Device Architecture (CUDA) version is 11.4. SGCL-LncLoc is implemented based on the PyTorch deep learning framework and Deep Graph Library (DGL). The k value of the k-mer node is set to 5. For each node, we use the Word2Vec technique to pretrain the encoding vectors, with a dimension of 128. The encoder consists of two-layer convolutional networks with 256 output neurons. The projector module consists of a linear layer with 256 input and output neurons. The classifier module uses the sigmoid activation function to calculate the output, which ranges from 0 to 1 and represents the predicted probability. During the model training process, an early stop mechanism is employed. The mechanism monitors the Area Under the receiver operator characteristic Curve (AUC) of the validation set and halts training if the AUC does not improve for 100 consecutive epochs. The batch size is set to 16. We perform 5-fold Cross-Validation (CV) on the benchmark dataset, and evaluate the generalization ability of the model using an independent test set.

The training process of SGCL-LncLoc utilizes two loss functions. The supervised contrastive loss function, as described in Eq. (5), is employed to guide the learning process. Additionally, we utilized the Binary Cross Entropy Loss (BCELoss) as the classification loss function. The equation for BCELoss is shown as follows:

$$\text{BCELoss} = \sum_{i=1}^{N} \left(-y_i \log \hat{y}_i - (1-y_i)\log(1-\hat{y}_i)\right) \quad (6)$$

where N is the total number of samples, y_i is the true label of sample i, while $\hat{y}_i$ is the corresponding predicted label. Since the contrastive loss function generally yields larger values than that of BCELoss, a scaling factor of $\alpha = 0.1$ is introduced to the contrastive loss function. Finally, the overall loss function used for model optimization is shown as follows:

$$\text{SumLoss} = \text{BCELoss} + \alpha \times \text{SupContrastLoss} \quad (7)$$

In addition, SGCL-LncLoc applies an exponential decay warm up strategy for the learning rate. As shown in Fig. S3 in the ESM, during the initial step_w (step_w = 150) epochs of model training, the learning rate follows a linear increase from 0 to a pre-defined base learning rate (lr_{base} = 0.0005). After the step_w epoch, the learning rate decays. We formalize the two processes of learning rate growth and decay as follows:

$$\text{lr} = \begin{cases} \dfrac{\text{lr}_{\text{base}}(\text{epoch}+1)}{\text{step}_w}, & \text{if epoch} \leqslant \text{step}_w; \\ \text{lr}_{\text{base}} e^{-((\text{epoch}+1-\text{step}_w)\gamma)}, & \text{if epoch} > \text{step}_w \end{cases} \quad (8)$$

where lr_{base} represents the base learning rate, which is the value at the turning point in the learning rate change process and is set to 0.0005. "epoch" refers to the number of completed training rounds. step_w represents the epoch at which the learning rate begins exponential decay and is set to 150. The coefficient γ controls the rate of exponential decay of the learning rate and is set to 0.005.

Importantly, we employ the Optuna[40] framework to automatically optimize the model hyperparameters. Table S1 in the ESM shows the optimal hyperparameters of SGCL-LncLoc and the corresponding search space. Optuna is a powerful tool for hyperparameter optimization in machine learning or deep learning models. It leverages the tree-structured Parzen estimator to efficiently explore the hyperparameter space. By leveraging Optuna, SGCL-LncLoc enhances the efficiency of hyperparameter optimization, ultimately leading to improved model performance.

3 Result

3.1 Evaluation metrics

In order to evaluate the predictive performance of SGCL-LncLoc, we use some evaluation metrics. These evaluation metrics include accuracy (ACC), precision, recall, F1-score, AUC, and Matthews Correlation Coefficient (MCC). By utilizing these evaluation metrics, we are able to evaluate the model's performance from different perspectives,

$$\text{ACC} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{FN}+\text{TN}+\text{FP}} \quad (9)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} \quad (10)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}} \quad (11)$$

$$\text{F1-score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}} \quad (12)$$

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP}+\text{FP})(\text{TP}+\text{FN})(\text{TN}+\text{FP})(\text{TN}+\text{FN})}} \quad (13)$$

In the above equations, TP, TN, FP, and FN represent the numbers of true positives, true negatives, false positives, and false negatives, respectively.

3.2 Comparison with machine learning and deep learning baseline models

In order to illustrate the prediction ability of SGCL-LncLoc, we conduct a comparison with several machine learning and deep learning baseline models using 5-fold CV. The experimental results are shown in Table 2.

(1) Machine learning baseline models: We use 5-mer frequency features as inputs for traditional machine learning models. The employed machine learning classifiers include SVM, RF, Logistic Regression (LR), and Multi-Layer Perceptron (MLP). All of these machine learning baseline models are implemented using the scikit-learn library.

(2) Deep learning baseline models: we develope two deep learning baseline models, namely Word2Vec + Convolutional Neural Network (CNN) + MLP and Word2Vec + Long Short-Term Memory (LSTM) + MLP. First, we use the Word2Vec technique to obtain the encoding vector for each 5-mer. Subsequently, we employ either CNN or LSTM to extract high-level features from the sequences. Finally, we employ an MLP for classification. Considering that the majority of lncRNA sequences in our dataset have a length of less than approximately 4000 nucleotides, we set a fixed sequence length of 4000. If the length of a sequence exceeds 4000, the subsequent portion is discarded. Conversely, if the length of a sequence is shorter than 4000, we pad it with zeros at the end. This processing step ensures that all lncRNA sequences are standardized to a consistent length for using the deep learning models.

Based on the results presented in Table 2, it is evident that the 5-mer + RF model outperforms other machine learning baseline models in terms of ACC, Precision, AUC, and MCC. On the other hand, the 5-mer + LR model demonstrates superiority over other machine learning baseline models in terms of F1-score and Recall. Regarding the deep learning baseline models, the Word2Vec + CNN + MLP model exhibits higher performance in ACC, Precision, AUC, and MCC. However, the Word2Vec + LSTM + MLP model performs better in terms of F1-score and Recall. Notably, SGCL-LncLoc significantly outperforms all machine learning and deep learning baseline models in terms of ACC, F1-score, Recall, AUC, and MCC. It is only slightly lower than the 5-mer + RF model in Precision. These results clearly highlight the superior predictive performance of SGCL-LncLoc in lncRNA subcellular localization prediction.

3.3 Comparison with existing predictors

In this section, we provide a comprehensive comparison of SGCL-LncLoc with existing predictors. We carefully select a representative set of predictors, including lncLocator, iLoc-lncRNA, Locate-R, iLoc-lncRNA(2.0), TACOS, RNAlight, LightGBM-LncLoc, and GraphLncLoc. Each of these predictors offers an accessible web server or stand-alone tool that exclusively requires lncRNA sequences as input, and generates predicted scores for subcellular localization as the output.

Among these predictors, TACOS and RNAlight provide predictions for two subcellular locations, namely the nucleus and cytoplasm. iLoc-lncRNA, Locate-R, iLoc-lncRNA(2.0), and GraphLncLoc predict four subcellular locations, including the

Table 2 Performance comparison of SGCL-LncLoc with other baseline models using 5-fold CV. The best performance values are highlighted in bold.

Baseline model	ACC	F1-score	Precision	Recall	AUC	MCC
5-mer + SVM	0.576	0.556	0.690	0.465	0.655	0.192
5-mer + RF	0.596	0.561	**0.736**	0.453	0.680	0.247
5-mer + LR	0.576	0.568	0.677	0.488	0.609	0.182
5-mer + MLP	0.556	0.511	0.686	0.407	0.666	0.168
Word2Vec + CNN + MLP	0.623	0.642	0.699	0.593	0.647	0.252
Word2Vec + LSTM + MLP	0.609	0.655	0.659	0.651	0.607	0.205
SGCL-LncLoc	**0.675**	**0.717**	0.713	**0.721**	**0.715**	**0.337**

nucleus, cytoplasm, ribosome, and exosome. lncLocator and LightGBM-LncLoc provide predictions for five subcellular locations, namely the nucleus, cytoplasm, cytosol, ribosome, and exosome. The ribosome is an organelle responsible for protein synthesis, is located within the cytoplasm, while the exosome, a type of extracellular vesicle, is typically formed within the cytoplasm and released into the extracellular space. From a simplified perspective, lncRNAs localized within the ribosome and exosome can be considered as localizing in the cytoplasm. Table 3 presents the performance comparison between SGCL-LncLoc with the existing predictors on the independent test set. Notably, iLoc-lncRNA and iLoc-lncRNA(2.0) only provide results for a single subcellular location with its associated probabilities, making it impractical to calculate the AUC value. Figure S4 in the ESM shows the confusion matrices of SGCL-LncLoc and existing predictors on the test set. Figure S5 in the ESM shows the Receiver Operator Characterstic (ROC) curves of SGCL-LncLoc and existing predictors on the test set. From Table 3, we can observe that SGCL-LncLoc outperforms other predictors, achieving the best results in terms of ACC (0.675), F1-score (0.717), Recall (0.721), AUC (0.715), and MCC (0.337). It is only slightly lower than Locate-R (0.765) in Precision. Remarkably, both iLoc-lncRNA and GraphLncLoc, despite being designed as multi-class predictors, exhibit strong overall performance when applied to binary classification tasks. In summary, SGCL-LncLoc demonstrates superior prediction capabilities for lncRNA subcellular localization.

3.4 Ablation study

In order to evaluate the effects of various components in SGCL-LncLoc, we conduct ablation study from three aspects: node feature encoding, pooling mode, and model training strategy. The experimental details are described below.

(1) One-Hot: This model encodes the 1024 nodes of the graph using the One-Hot encoding method, resulting in a One-Hot encoding vector with a dimension of 1024.

(2) GloVe: This model uses the GloVe technique to encode the node features. The GloVe technique focuses on global statistics, while the Word2Vec technique focuses on local context information. The dimension of the encoding vector generated by the GloVe technique is set to 128, which is consistent with the dimension of the Word2Vec encoding vector in SGCL-LncLoc.

(3) MaxPooling: Instead of global attention pooling, this model uses maximum pooling to convert the graph into a vector representation.

(4) AvgPooling: Instead of global attention pooling, this model uses average pooling to convert the graph into a vector representation.

(5) noSupContrast: This model indicates that the strategy of supervised graph contrastive learning is not utilized during the training process. In this case, the weight value of the contrastive loss α is set to 0.

The performance comparison of SGCL-LncLoc with various ablation models on the independent test set is shown in Table 4. The results highlight the significant influence of the pooling method used for converting graphs to vectors on the predictive performance of the model. When employing average pooling, there is a substantial decrease in ACC, F1-score, AUC, and MCC by 11.70%, 22.59%, 9.09%, and 24.93%, respectively. Contrastive learning is also very important in SGCL-LncLoc. The absence of contrastive learning leads to a decrease of approximately 4.89%, 4.32%, 5.03%, and 19.58% in

Table 3 Performance comparison of SGCL-LncLoc with existing predictors on the test set. The best performance values are highlighted in bold.

Predictor	ACC	F1-score	Precision	Recall	AUC	MCC
lncLocator	0.477	0.288	0.640	0.186	0.564	0.063
iLoc-lncRNA	0.589	0.557	0.722	0.453	–	0.230
Locate-R	0.550	0.433	**0.765**	0.302	0.634	0.212
iLoc-lncRNA(2.0)	0.536	0.417	0.735	0.291	–	0.180
TACOS	0.570	0.575	0.657	0.512	0.621	0.157
RNAlight	0.662	0.691	0.722	0.663	0.670	0.322
LightGBM-LncLoc	0.530	0.599	0.582	0.616	0.542	0.032
GraphLncLoc	0.570	0.511	0.723	0.395	0.600	0.209
SGCL-LncLoc	**0.675**	**0.717**	0.713	**0.721**	**0.715**	**0.337**

Table 4 Performance comparison of SGCL-LncLoc with various ablation models on the test set. The best performance values are highlighted in bold.

Ablation model	ACC	F1-score	AUC	MCC
One-Hot	0.649	**0.731**	0.654	0.266
GloVe	0.649	0.723	0.664	0.267
MaxPooling	0.609	0.593	0.690	0.258
AvgPooling	0.596	0.555	0.650	0.253
noSupContrast	0.642	0.686	0.679	0.271
SGCL-LncLoc	**0.675**	0.717	**0.715**	**0.337**

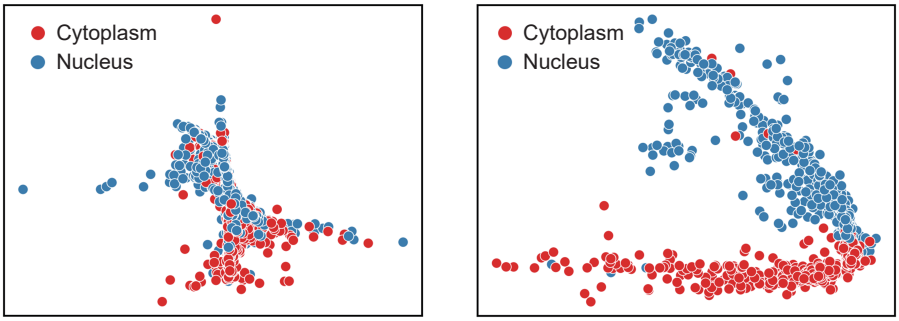
ACC, F1-score, AUC, and MCC, respectively. Conversely, the choice of encoding methods for node features has a relatively minor impact. In summary, SGCL-LncLoc outperforms the other ablation models. This suggests that the integration of these components has a positive impact on lncRNA subcellular localization prediction tasks.

3.5 Effectiveness of supervised contrastive learning

To further observe the effects of supervised contrastive learning, we conduct an analysis comparing the graph vectors obtained from SGCL-LncLoc with its ablation model, denoted as "noSupContrast". The "noSupContrast" model represents SGCL-LncLoc without supervised contrastive learning. Subsequently, we employ t-distributed Stochastic Neighbor Embedding (t-SNE) to project these graph vectors into a two-dimensional space, as shown in Fig. 4, where blue dots represent samples from the nucleus, while red dots represent samples from the cytoplasm. Figures 4a and 4b represent the visualizations of "noSupContrast" and SGCL-LncLoc, respectively. Notably, Fig. 4b shows a significant improvement in the separation of samples from different subcellular localizations. These results highlight the effectiveness of supervised contrastive learning in enhancing the discriminatory ability. By leveraging supervised contrastive learning, SGCL-LncLoc achieves a remarkable enhancement in the spatial separation of samples from different subcellular localizations in the embedding space, thereby facilitating accurate prediction of lncRNA subcellular localizations.

3.6 Motif analysis

In this study, we employ a global attention pooling mechanism to obtain the attention weights of each *k*-mer node in the de Bruijn graph. Additionally, we propose a computational method to map the node weights to the nucleotide weights in the sequence. To evaluate the performance of the global attention pooling mechanism and the mapping method in SGCL-LncLoc, we conduct a comprehensive motif analysis. First, we validate the ability of SGCL-LncLoc to detect the most frequently occurring motifs. Specifically, we utilize the MEME suite[41] to identify motifs in our benchmark dataset. The motifs are identified using a length of 9 and an *E*-value threshold of 0.05. We introduced a threshold in SGCL-LncLoc to distinguish motifs with high attention. This threshold is set to the 90-th percentile of the attention scores calculated from the entire input sequence. The nucleotides with attention scores exceeding the threshold are considered to receive significant attention, while those with attention scores below are regarded as less relevant. Figure 5 presents some typical examples, with the left column displaying the motifs discovered by the MEME suite, the middle column showing the motifs identified by SGCL-LncLoc, and the right indicating the *E*-value of the MEME suite. The comparison demonstrates that



(a) Vectors obtained from the noSupContrast model (b) Vectors obtained from SGCL-LncLoc

Fig. 4 t-SNE visualization of graph vectors obtained from SGCL-LncLoc and noSupContrast models. Each dot represents a sample and its color represents its true class. Samples originating from the nucleus are plotted in blue, while samples from the cytoplasm are marked in red.

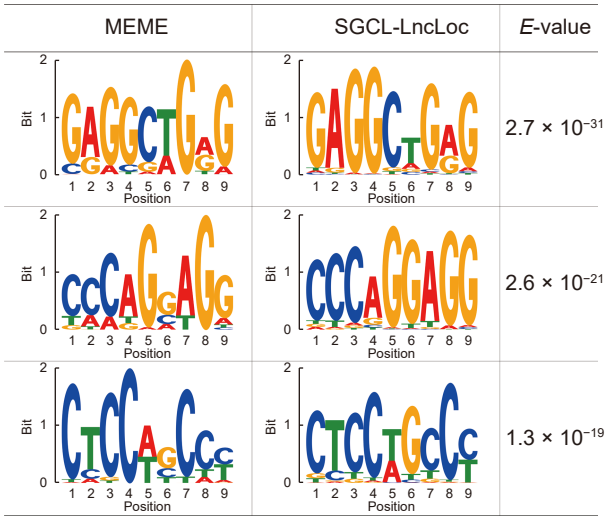


Fig. 5 Comparison of motifs discovered by the MEME suite and SGCL-LncLoc, with corresponding *E*-values of the MEME suite.

SGCL-LncLoc captures motifs similar to those discovered by the MEME suite, indicating its ability to detect frequently occurring motifs[42].

Furthermore, we investigate the ability of SGCL-LncLoc to recognize known motifs relevant to subcellular localization. Specifically, we review recent literature for motifs relevant to subcellular localization. Lubelsky and Ulitsky[43] identified the RCCTCCC motif, which plays a crucial role in nucleus localization. Zhang et al.[44] discovered the AGCCC motif as a common signal for nucleus localization. By using RCCTCCC and AGCCC as examples in Fig. 6, we demonstrate the capability of SGCL-LncLoc to capture nucleus motifs.

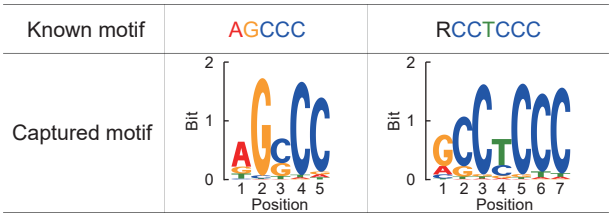


Fig. 6 SGCL-LncLoc captures two known nucleus localization motifs.

3.7 Case study

To better understand the interpretability of SGCL-LncLoc, we conduct a case study using lncRNA AC006160.5 (clone-based (Vega) gene, Ensembl ID: ENSG00000249502), as shown in Fig. 7. The lncRNA AC006160.5 is found in Homo sapiens and localizes in the nucleus. Previous research by Zhang et al.[44] has suggested that the AGCCC motif serves as a common nucleus localization signal. We feed the sequence of lncRNA AC006160.5 into SGCL-LncLoc, and obtain the corresponding prediction results and nucleotide weights for the sequence. Remarkably, SGCL-LncLoc accurately predicts the subcellular localization of lncRNA AC006160.5, correctly identifying its localizes in the nucleus. Considering the extensive length of the lncRNA AC006160.5 sequence, spanning 27 105 nucleotides, we select a specific segment for detailed observation. Specifically, we focus on a small fragment from position 13 376 to position 13 465. Figure 7 shows the corresponding nucleotide weight heatmap, where varying shades of red represent the attention weights. Darker colors indicate higher weights, while lighter colors signify lower weights.

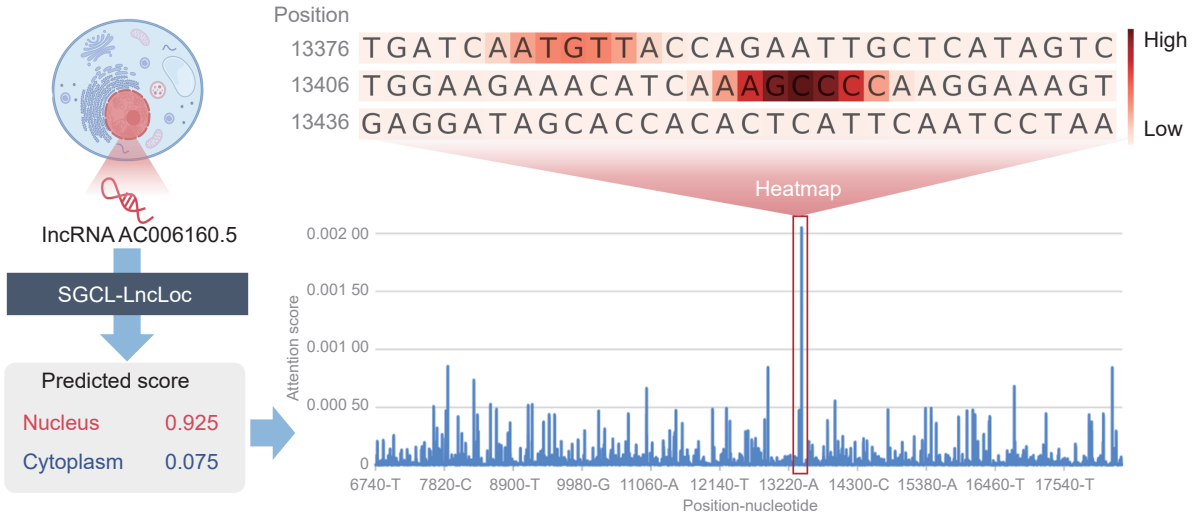


Fig. 7 Attention weight visualization of lncRNA AC006160.5.

The visualization in Fig. 7 clearly demonstrates that SGCL-LncLoc effectively captures the important motif "AGCCC". This finding highlights that SGCL-LncLoc has great potential in motif discovery.

3.8 Web server

To facilitate the usage of SGCL-LncLoc, we develope a user-friendly web server, which can be available at http://csuligroup.com:8000/SGCL-LncLoc. The web server is designed to accept one lncRNA sequence in FASTA format at a time and supports sequence lengths ranging from 200 to 10 000 nucleotides. SGCL-LncLoc typically takes less than 5 seconds to predict the subcellular localization of lncRNA sequences. The results page provides comprehensive information, including the input sequence provided by the user, the corresponding prediction results, and a visualization graph displaying the attention scores of the lncRNA sequence, as shown in Fig. S6 in the ESM, where the prediction results are presented in the table, providing the predicted probabilities for different subcellular localization categories. The visualization graph allows users to gain insights into the importance of different segments within the sequences, as well as to identify important motifs.

4 Conclusion

In the study, we propose SGCL-LncLoc, a novel interpretable deep learning model based on supervised graph contrastive learning, for efficient and accurate prediction of lncRNA subcellular location. Compared to previous work, SGCL-LncLoc has two main innovations: (1) leverage supervised graph contrastive learning to enhance prediction performance by bringing samples of the same category closer together in the embedding space and pushing samples of different categories further apart; (2) employ a global attention mechanism to learn the weights of each node in the graph, and propose a computational method to map node weights to nucleotide weights in the lncRNA sequence. Extensive experimental results demonstrate that SGCL-LncLoc outperforms the machine learning and deep learning baseline models, as well as existing state-of-the-art predictors. Ablation study confirms the effectiveness of multiple components in the model. Finally, the visualization of the attention scores for each nucleotide in the lncRNA sequence, enabling the identification of sequence fragments that play an important role in the prediction process. To facilitate the usage, we developed a user-friendly web server for researchers to access SGCL-LncLoc.

Despite the substantial progress made in this study, there are still some limitations that can be improved. In future research, the following directions could be explored: (1) Integration of multi-source biological data. To enhance the prediction performance of lncRNA subcellular localization prediction, it would be beneficial to integrate lncRNA sequence information with other relevant biological data[45, 46]. For example, incorporating RNA binding motifs and functional annotations of lncRNAs could provide valuable insights and potentially leading to improved predictions. (2) Exploration of advanced sequence encoding models. In this study, we utilized the Word2Vec technique for encoding lncRNA sequence features. With the advancement of large-scale language models, their potential application in lncRNA research remains largely unexplored. Therefore, in the future, it would be worthwhile to investigate the use of more sophisticated models like BERT or Transformer for encoding lncRNA sequences[47], as they may capture intricate patterns and relationships within lncRNA sequences, hold the potential to further enhance prediction performance.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (No. 62102457), the Hunan Provincial Natural Science Foundation of China (No. 2023JJ40763), the Hunan Provincial Science and Technology Program (No. 2021RC4008), the Fundamental Research Funds for the Central Universities of Central South University (No. CX20230271). This work is carried out in part using computing resources at the High Performance Computing Center of Central South University. We thank Dr. Fuhao Zhang for his assistance in the analysis of the results. Additionally, we are thankful to Master Jingwei Lu for his efforts in the development of our web server.

Electronic Supplementary Material

Supplementary materials including:

- Figure S1 Data processing flowchart,
- Figure S2 Graph construction steps,
- Figure S3 Change of learning rate under exponential decaywarm up strategy,

• Table S1 Optimal hyper-parameters of SGCL-LncLocsearched by Optuna framework and the corresponding search space,

• Figure S4 Confusion matrices of SGCL-LncLoc with existing predictors on the test set,

• Figure S5 ROC curves of SGCL-LncLoc and existing predictors on the test set, and

• Figure S6 Results page of SGCL-LncLoc web server for lncRNA subcellular localization prediction are available in the online version of this article at https://doi.org/10.26599/BDMA.2024.9020002.

References

[1] C.-C. Hon, J. A. Ramilowski, J. Harshbarger, N. Bertin, O. J. L. Rackham, J. Gough, E. Denisenko, S. Schmeier, T. M. Poulsen, J. Severin et al., An atlas of human long non-coding RNAs with accurate 5' ends, *Nature*, vol. 543, no. 7644, pp. 199–204, 2017.

[2] M. Zeng, C. Lu, Z. Fei, F.-X. Wu, Y. Li, J. Wang, and M. Li, DMFLDA: A deep learning framework for predicting lncRNA–disease associations, *IEEE/ACM Trans. Comput. Biol. Bioinform.*, vol. 18, no. 6, pp. 2353–2363, 2021.

[3] J. J. Quinn and H. Y. Chang, Unique features of long non-coding RNA biogenesis and function, *Nat. Rev. Genet.*, vol. 17, no. 1, pp. 47–62, 2016.

[4] U. A. Ørom, T. Derrien, M. Beringer, K. Gumireddy, A. Gardini, G. Bussotti, F. Lai, M. Zytnicki, C. Notredame, Q. Huang, et al., Long noncoding RNAs with enhancer-like function in human cells, *Cell*, vol. 143, no. 1, pp. 46–58, 2010.

[5] R. A. Gupta, N. Shah, K. C. Wang, J. Kim, H. M. Horlings, D. J. Wong, M. C. Tsai, T. Hung, P. Argani, J. L. Rinn, et al., Long non-coding RNA HOTAIR reprograms chromatin state to promote cancer metastasis, *Nature*, vol. 464, no. 7291, pp. 1071–1076, 2010.

[6] F. Zhang, W. Shi, J. Zhang, M. Zeng, M. Li, and L. Kurgan, PROBselect: Accurate prediction of protein-binding residues from proteins sequences via dynamic predictor selection, *Bioinformatics*, vol. 36, no. Supplement_2, pp. i735–i744, 2020.

[7] E. Hacisuleyman, L. A. Goff, C. Trapnell, A. Williams, J. Henao-Mejia, L. Sun, P. McClanahan, D. G. Hendrickson, M. Sauvageau, D. R. Kelley, et al., Topological organization of multichromosomal regions by the long intergenic noncoding RNA Firre, *Nat. Struct. Mol. Biol.*, vol. 21, no. 2, pp. 198–206, 2014.

[8] C. Carrieri, L. Cimatti, M. Biagioli, A. Beugnet, S. Zucchelli, S. Fedele, E. Pesce, I. Ferrer, L. Collavin, C. Santoro, et al., Long non-coding antisense RNA controls Uchl1 translation through an embedded SINEB2 repeat, *Nature*, vol. 491, no. 7424, pp. 454–457, 2012.

[9] F. Karreth, M. Reschke, A. Ruocco, C. Ng, B. Chapuy, V. Léopold, M. Sjoberg, T. Keane, A. Verma, U. Ala, et al., The BRAF pseudogene functions as a competitive endogenous RNA and induces lymphoma InVivo, *Cell*, vol. 161, no. 2, pp. 319–332, 2015.

[10] D. M. Anderson, K. M. Anderson, C.-L. Chang, C. A. Makarewich, B. R. Nelson, J. R. McAnally, P. Kasaragod, J. M. Shelton, J. Liou, R. Bassel-Duby, et al., A micropeptide encoded by a putative long noncoding RNA regulates muscle performance, *Cell*, vol. 160, no. 4, pp. 595–606, 2015.

[11] M. Zeng, C. Lu, F. Zhang, Y. Li, F.-X. Wu, Y. Li, and M. Li, SDLDA: lncRNA-disease association prediction based on singular value decomposition and deep learning, *Methods*, vol. 179, pp. 73–80, 2020.

[12] Z. D. Su, Y. Huang, Z. Y. Zhang, Y. W. Zhao, D. Wang, W. Chen, K. C. Chou, and H. Lin, iLoc-lncRNA: Predict the subcellular location of lncRNAs by incorporating octamer composition into general PseKNC, *Bioinformatics*, vol. 34, no. 24, pp. 4196–4204, 2018.

[13] A. Ahmad, H. Lin, and S. Shatabda, Locate-R: Subcellular localization of long non-coding RNAs using nucleotide compositions, *Genomics*, vol. 112, no. 3, pp. 2583–2589, 2020.

[14] Z. Y. Zhang, Z. J. Sun, Y. H. Yang, and H. Lin, Towards a better prediction of subcellular location of long non-coding RNA, *Front. Comput. Sci.*, vol. 16, no. 5, p. 165903, 2022.

[15] Z. Cao, X. Pan, Y. Yang, Y. Huang, and H. B. Shen, The lncLocator: A subcellular localization predictor for long non-coding RNAs based on a stacked ensemble classifier, *Bioinformatics*, vol. 34, no. 13, pp. 2185–2194, 2018.

[16] G. H. Yuan, Y. Wang, G. Z. Wang, and L. Yang, RNAlight: A machine learning model to identify nucleotide features determining RNA subcellular localization, *Brief. Bioinform.*, vol. 24, no. 1, p. bbac509, 2023.

[17] J. Cai, T. Wang, X. Deng, L. Tang, and L. Liu, GM-lncLoc: LncRNAs subcellular localization prediction based on graph neural network with meta-learning, *BMC Genom.*, vol. 24, no. 1, p. 52, 2023.

[18] B. L. Gudenas and L. Wang, Prediction of LncRNA subcellular localization with deep learning from sequence features, *Sci. Rep.*, vol. 8, no. 1, p. 16385, 2018.

[19] Y. Fan, M. Chen, and Q. Zhu, lncLocPred: Predicting LncRNA subcellular localization using multiple sequence feature information, *IEEE Access*, vol. 8, pp. 124702–124711, 2020.

[20] S. Feng, Y. Liang, W. Du, W. Lv, and Y. Li, LncLocation: Efficient subcellular location prediction of long non-coding RNA-based multi-source heterogeneous feature fusion, *Int. J. Mol. Sci.*, vol. 21, no. 19, p. 7271, 2020.

[21] Y. J. Jeon, M. M. Hasan, H. W. Park, K. W. Lee, and B. Manavalan, TACOS: A novel approach for accurate prediction of cell-specific long noncoding RNAs subcellular localization, *Brief. Bioinform.*, vol. 23, no. 4, p. bbac243, 2022.

[22] J. Lyu, P. Zheng, Y. Qi, and G. Huang, LightGBM-LncLoc: A LightGBM-based computational predictor for recognizing long non-coding RNA subcellular localization, *Mathematics*, vol. 11, no. 3, p. 602, 2023.

[23] Y. Lin, X. Pan, and H. B. Shen, lncLocator 2.0: A cell-line-specific subcellular localization predictor for long non-coding RNAs with interpretable deep learning, *Bioinformatics*, vol. 37, no. 16, pp. 2308–2316, 2021.

[24] M. Zeng, Y. Wu, C. Lu, F. Zhang, F. X. Wu, and M. Li, DeepLncLoc: A deep learning framework for long non-coding RNA subcellular localization prediction based on subsequence embedding, *Brief. Bioinform.*, vol. 23, no. 1, p. bbab360, 2022.

[25] M. Zeng, Y. Wu, Y. Li, R. Yin, C. Lu, J. Duan, and M. Li, LncLocFormer: A Transformer-based deep learning model for multi-label lncRNA subcellular localization prediction by using localization-specific attention mechanism, *Bioinformatics*, vol. 39, no. 12, p. btad752, 2023.

[26] M. Li, B. Zhao, R. Yin, C. Lu, F. Guo, and M. Zeng, GraphLncLoc: Long non-coding RNA subcellular localization prediction using graph convolutional networks based on sequence to graph transformation, *Brief. Bioinform.*, vol. 24, no. 1, p. bbac565, 2023.

[27] D. Mas-Ponte, J. Carlevaro-Fita, E. Palumbo, T. Hermoso Pulido, R. Guigo, and R. Johnson, LncATLAS database for subcellular localization of long noncoding RNAs, *RNA*, vol. 23, no. 7, pp. 1080–1087, 2017.

[28] L. P. B. Bouvrette, N. A. L. Cody, J. Bergalet, F. A. Lefebvre, C. Diot, X. Wang, M. Blanchette, and E. Lécuyer, CeFra-seq reveals broad asymmetric mRNA and noncoding RNA distribution profiles in Drosophila and human cells, *RNA*, vol. 24, no. 1, pp. 98–113, 2018.

[29] F. M. Fazal, S. Han, K. R. Parker, P. Kaewsapsak, J. Xu, A. N. Boettiger, H. Y. Chang, and A. Y. Ting, Atlas of subcellular RNA localization revealed by APEX-seq, *Cell*, vol. 178, no. 2, pp. 473–490.e26, 2019.

[30] T. Zhang, P. Tan, L. Wang, N. Jin, Y. Li, L. Zhang, H. Yang, Z. Hu, L. Zhang, C. Hu, et al., RNALocate: A resource for RNA subcellular localizations, *Nucleic Acids Res*., vol. 45, no. D1, pp. D135–D138, 2017.

[31] Y. Huang, B. Niu, Y. Gao, L. Fu, and W. Li, CD-HIT Suite: A web server for clustering and comparing biological sequences, *Bioinformatics*, vol. 26, no. 5, pp. 680–682, 2010.

[32] T. Cui, Y. Dou, P. Tan, Z. Ni, T. Liu, D. Wang, Y. Huang, K. Cai, X. Zhao, D. Xu, et al., RNALocate v2.0: An updated resource for RNA subcellular localization with increased coverage and annotation, *Nucleic Acids Res*., vol. 50, no. D1, pp. D333–D339, 2022.

[33] Y. Wu, M. Gao, M. Zeng, J. Zhang, and M. Li, BridgeDPI: A novel Graph Neural Network for predicting drug-protein interactions, *Bioinformatics*, vol. 38, no. 9, pp. 2571–2578, 2022.

[34] S. Kan, Y. Cen, Y. Li, M. Vladimir, and Z. He, Local semantic correlation modeling over graph neural networks for deep feature embedding and image retrieval, *IEEE Trans. Image Process.*, vol. 31, pp. 2988–3003, 2022.

[35] M. Chen, Y. Jiang, X. Lei, Y. Pan, C. Ji, W. Jiang, and H. Xiong, Drug-target interactions prediction based on signed heterogeneous graph neural networks, *Chin. J. Electron.*, vol. 33, no. 1, pp. 231–244, 2024.

[36] S. Kan, Z. He, Y. Cen, Y. Li, V. Mladenovic, and Z. He, Contrastive Bayesian analysis for deep metric learning, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 6, pp. 7220–7238, 2023.

[37] J. Chen, R. Zhang, Y. Mao, and J. Xu, Contrastnet: A contrastive learning framework for few-shot text classification, in *Proc. AAAI Conference on Artificial Intelligence*, https://doi.org/10.1609/aaai.v36i10.21292, 2023.

[38] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, Supervised contrastive learning, *Advances in Neural Information Processing Systems*, vol. 33, pp. 18661–18673, 2020.

[39] S. Chen and C. Geng, A comprehensive perspective of contrastive self-supervised learning, *Front. Comput. Sci.*, vol. 15, no. 4, p. 154332, 2021.

[40] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, A next-generation hyperparameter optimization framework, in *Proc. 25th ACM SIGKDD International Conference on Knowledge Discovery Data Mining*, https://doi.org/10.1145/3292500.3330701, 2023.

[41] T. L. Bailey, M. Boden, F. A. Buske, M. Frith, C. E. Grant, L. Clementi, J. Ren, W. W. Li, and W. S. Noble, MEME suite: Tools for motif discovery and searching, *Nucleic Acids Res*., vol. 37, no. suppl_2, pp. W202–W208, 2009.

[42] Y. Guo, X. Lei, Y. Pan, and R. Su, An encoding-decoding framework based on CNN for circRNA-RBP binding sites prediction, *Chin. J. Electron.*, vol. 33, no. 1, pp. 256–263, 2024.

[43] Y. Lubelsky and I. Ulitsky, Sequences enriched in Alu repeats drive nuclear localization of long RNAs in human cells, *Nature*, vol. 555, no. 7694, pp. 107–111, 2018.

[44] B. Zhang, L. Gunawardane, F. Niazi, F. Jahanbani, X. Chen, and S. Valadkhan, A novel RNA motif mediates the strict nuclear localization of a long noncoding RNA, *Mol. Cell. Biol.*, vol. 34, no. 12, pp. 2318–2329, 2014.

[45] X. Yang, X. Lei, and J. Zhao, Essential protein prediction based on shuffled frog-leaping algorithm, *Chin. J Electronics*, vol. 30, no. 4, pp. 704–711, 2021.

[46] Y. Zhang, X. Lei, Z. Fang, and Y. Pan, CircRNA-disease associations prediction based on metapath2vec++ and matrix factorization, *Big Data Mining and Analytics*, no. 4, pp. 280–291, 2020.

[47] Y. Li, M. Zeng, F. Zhang, F. X. Wu, and M. Li, DeepCellEss: Cell line-specific essential protein prediction with attention-based interpretable deep learning, *Bioinformatics*, vol. 39, no. 1, p. btac779, 2023.

Min Li received the PhD degree in computer science from Central South University, China in 2008. She is currently the dean and a professor at School of Computer Science and Engineering, Central South University, Changsha, China. Her research interests include computational biology, systems biology, and bioinformatics. She has published more than 100 technical papers in refereed journals, such as *Nature Communications*, *Genome Research, Genome Biology*, *Nucleic Acids Research*, and *Bioinformatics*, and conference proceedings.

Yiming Li received the MEng degree in computer science and technology from Central South University, China in 2022. She is currently a PhD candidate at Central South University, China. Her research interests include bioinformatics and deep learning.

Rui Yin received the PhD degree from Nanyang Technological University, Singapore in 2020. He completed his postdoctoral training at Harvard Medical School, USA in 2022. He is currently an assistant professor at Department of Health Outcomes and Biomedical Informatics, University of Florida, USA. His research interests mainly focus on AI-driven precision medicine to improve public health outcomes and equity.

Min Zeng received the BEng degree from Lanzhou University, China in 2013, the MEng and PhD degrees from Central South University, China in 2016 and 2020, respectively. He is currently an associate professor at School of Computer Science and Engineering, Central South University, China. His main research interests include bioinformatics, machine learning, and deep learning.

Baoying Zhao received the BEng degree from Guizhou University, China in 2017. She is currently a master student in bioinformatics at Central South University, China. Her current research interests include bioinformatics, lncRNA subcellular localization prediction, and deep learning.

Pingjian Ding received the PhD degree in computer science from Hunan University, China in 2019. He is currently a research associate at Center for Artificial Intelligence in Drug Discovery, Case Western Reserve University, USA. His research primarily focuses on bridging machine learning and deep learning to biostatistical methodologies, with specific applications in statistical genetics, real-world evidence, bioinformatics, and Alzheimer's disease.

Shichao Kan received the BEng, MEng, and PhD degrees from Beijing Jiaotong University, China in 2014, 2016, and 2021, respectively. From 2019 to 2020, he was a visiting student researcher at Department of Computer Science, University of Missouri, Columbia, MO, USA. He is currently a lecturer at School of Computer Science and Engineering, Central South University, China. His research interests include multimodal large language model, metric learning, large-scale object retrieval, and deep learning.