HAGAPS: Hierarchical Attentive Graph Neural Networks for Predicting Alternative Polyadenylation Site Quantification

Eleni Giovanoudi (✉) and Dimitrios Rafailidis

University of Thessaly, Volos, Greece
{egiovanoudi,draf}@uth.gr

Abstract. Alternative polyadenylation (APA) is a critical process that enables genes to generate mRNA transcripts with different $3'$ untranslated regions. Notably, during a transcription event, only one polyadenylation (poly(A)) site is used. Thus, estimating the relative usage of alternative poly(A) sites within a gene, known as the poly(A) site quantification problem, is crucial for unraveling the regulatory mechanisms of APA. However, existing approaches either frame the problem as a non-quantitative binary classification task or ignore the RNA structural information. To address these limitations, we propose a novel Hierarchical Attentive Graph Neural Network model for alternative poly(A) site quantification prediction, namely HAGAPS. To the best of our knowledge, we are the first to leverage Graph Neural Networks and RNA secondary structures to quantitatively predict the usage of multiple alternative poly(A) sites. In particular, our model employs a poly(A) site-level message passing network, incorporating RNA secondary structure information. In addition, to account for the competing interactions among poly(A) sites, HAGAPS integrates a gene-level message passing network combined with a nucleotide attention mechanism. Our experimental evaluation on publicly available datasets demonstrates that the proposed HAGAPS model significantly outperforms several state-of-the-art methods. Finally, for reproduction purposes, we make the implementation of HAGAPS publicly available at `https://github.com/egiovanoudi/HAGAPS`.

Keywords: Hierarchical graph neural networks · Attention mechanism · Polyadenylation site quantification · Alternative polyadenylation · RNA secondary structure.

1 Introduction

The central dogma of molecular biology describes the fundamental flow of genetic information in an eukaryotic gene through transcription, post-transcriptional modification, and translation processes [9]. In particular, genetic information is first transcribed into pre-mRNA, which undergoes post-transcriptional modifications to become mature mRNA, and is then translated into the corresponding

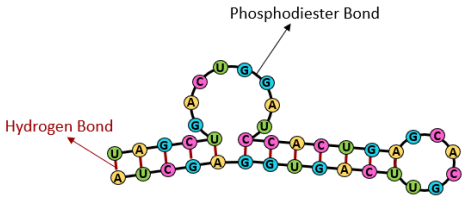


Fig. 1: Example of a RNA secondary structure.

protein. One critical post-transcriptional process is polyadenylation, responsible for creating the mature 3′ ends of nearly all eukaryotic mRNAs by adding a polyadenylation (poly(A)) tail at the 3′ end of the premRNA [34]. It is a two-step reaction that involves an endonucleolytic cleavage near the 3′ end of the pre-mRNA and synthesis of the poly(A) tail at the cleavage site, commonly referred to as the poly(A) site. Importantly, studies have shown that polyadenylation influences multiple aspects of mRNA metabolism, including stability, translation efficiency, transcription termination, and localization [2,8,20,30,35].

A key regulatory feature of polyadenylation is the presence of multiple poly(A) sites within a gene. Alternative poly(A) sites generate different mRNA transcripts with distinct 3′ untranslated regions (3′ UTRs), a process known as alternative polyadenylation (APA) [32]. APA is controlled by interactions between cis-regulatory elements located in the vicinity of poly(A) sites and the associated trans factors [13]. Among these cis-elements, the most well-known is the hexamer AAUAAA and its variants. In mammalian genes, APA is highly prevalent, with more than half of human genes undergoing this process, playing a crucial role in modulating gene regulation dynamics [12,31]. Furthermore, various human diseases, including cancer, alpha-thalassemia, and IPEX syndrome, have been linked to dysregulation of APA [10]. Hence, comprehensive understanding of poly(A) sites and the regulatory mechanisms governing APA are essential for unraveling its role in normal physiology and disease pathology.

Research has indicated that RNA secondary structures near poly(A) sites also impact APA by determining the accessibility of cis-elements to the polyadenylation machinery [1,7]. The primary structure of RNA refers to its linear sequence of nucleotides, connected by phosphodiester bonds along the RNA backbone. The secondary structure arises when complementary bases within the RNA strand form hydrogen bonds, creating structural motifs such as hairpins, bulges, stems, and internal loops. Base-pairing hydrogen bonds occur between A-U and C-G pairs, as well as less stable G-U pairs [16]. An example of a RNA structure illustrating these two types of bonds is shown in Fig. 1. Meanwhile, Graph Neural Networks (GNNs) have been developed to analyze graph-structured data by leveraging the relationships and topological information among nodes. In RNA secondary structure analysis, GNN-based methods have been employed to address various biological problems, including mRNA subcellular localization, RNA-protein binding, and gene silencing [22,25,26,39]. However, these studies are not incorporated into APA analysis.

There has been a long-standing interest in identifying poly(A) sites within genomic sequences. Various models have been developed to distinguish sequences that contain a poly(A) site from those that do not, defining the poly(A) site recognition problem [5,18,19,27,37]. Beyond recognition, a key aspect of APA is that during each transcription event, only a single poly(A) site within the gene is utilized. Consequently, the selection of alternative poly(A) sites within a gene is intrinsically competitive, where usage of one poly(A) site over another is often attributed to its relative strength. This challenging task of estimating the relative usage of alternative poly(A) sites within the same gene is referred to as the poly(A) site quantification problem [21]. Several computational approaches have emerged to address this problem using RNA-seq data [6,14,15,36,40]. Significant progress has also been made in inferring the relative strength of poly(A) sites based on genomic sequences. Early methods focus on predicting the stronger poly(A) site from a pair, framing the poly(A) site quantification problem as a binary classification problem [1,21]. However, this approach has notable limitations, as it fails to account for the competition between multiple poly(A) sites within a gene and does not provide quantitative predictions of usage. Subsequently, regression-based methods were developed to address the challenge of multiple competing poly(A) sites [23,24]. Despite their advantages, these methods do not take advantage of the RNA secondary structure information.

In summary, the current challenges in poly(A) site quantification prediction from genomic sequences are i) the lack of consideration for the competition among multiple poly(A) sites within a gene, and ii) the omission of the RNA secondary structure information. To address the shortcomings of existing approaches, we propose the HAGAPS model, making the following contributions:

- We present a novel architecture of Hierarchical GNNs to improve alternative poly(A) site quantification prediction. In particular, we introduce a hierarchical design of custom Message Passing Networks (MPNs), that is the Site and the Gene MPN, to predict the usage of all poly(A) sites within a gene, regardless of the number of competing sites.
- We design the Site MPN, leveraging the RNA secondary structure information. In doing so, the proposed model learns a more representative depiction of the poly(A) sites, capturing both sequential and structural patterns, thereby enabling communication between nucleotides within a site.
- We facilitate communication between poly(A) sites within a gene through the Gene MPN in combination with a nucleotide attention mechanism. This approach ensures that the model accounts for the competing interactions among alternative poly(A) sites.

The rest of the paper is organized as follows, in Section 2 we provide an overview of related work, and Section 3 details the architecture of the proposed HAGAPS model. In Section 4, we present the experimental evaluation of our model against baseline methods, and Section 5 concludes our work.

2 Related Work

Initially, computational methods were proposed to address the poly(A) site recognition problem based on genomic sequences. For instance, Kalkatawi et al. [19] present an artificial neural network and a random forest model that leverage human genomic sequence properties. These properties include thermodynamic, physico-chemical, and statistical features. Magana-Mora et al. [27] introduce a new set of hand-crafted features combined with a recognition model. Their approach employs multiple classifiers in a tree-like decision structure, optimized using genetic algorithms. Xia et al. [37] design a a deep learning model based on Convolutional Neural Networks (CNNs) with group normalization. Additionally, they employ transfer learning to adapt the model for a different species. Kalkatawi et al. [18] propose a CNN-based method for recognizing various genomic signals and regions, including poly(A) signals and translation initiation sites. The model relies on genomic neighborhoods and spatial correlations. However, these methods cannot predict the relative strength of poly(A) sites.

The first study to tackle the poly(A) site quantification problem based on genomic sequences was conducted by Leung et al. [21]. Their approach employs a CNN-based model to predict the stronger poly(A) site from a given pair of competing sites. Arefeen et al. [1] also cast poly(A) site quantification as a pairwise comparison task, incorporating RNA secondary structure features in the DeepPASTA model. Nevertheless, both methods define quantification as a binary classification problem, failing to provide quantitative usage predictions. Restricting competition to only two poly(A) sites is a significant limitation, as studies indicate that a substantial proportion of mammalian genes have more than two alternative poly(A) sites [11,38]. Moreover, DeepPASTA does not utilize GNNs, and its one-hot encoding representation of secondary structures loses valuable structural information [25]. Subsequently, Li et al. [23] formulate poly(A) site quantification as a regression task, considering all competing sites within a gene. Their approach employs CNNs, followed by a Bidirectional Long Short Term Memory (BiLSTM) network to capture interactions between competing poly(A) sites. In addition, Linder et al. [24] present a sequence-based residual neural network with dilated convolutions. Nontheless, these models do not take into account any RNA secondary structure information.

GNN-based models incorporating RNA secondary structures have been proposed to address various biological problems. For example, Li et al. [22] integrate RNA sequences and secondary structures to predict mRNA subcellular localization. Four parallel feature extractors are constructed using Multi-Layer Perceptrons (MLPs), multi-head attention mechanisms, and GNNs. Yan et al. [39] design a GNN-based approach that learns the RNA sequence and secondary structure information using a recurrent GNN and a BiLSTM for RNA-protein binding prediction. Long et al. [26] present a GNN framework for siRNA efficacy prediction. A variety of siRNA and mRNA features are extracted, including sequence encodings and base-pairing probabilities. However, since these methods are not designed for APA analysis, they do not consider the relationships among alternative poly(A) sites.

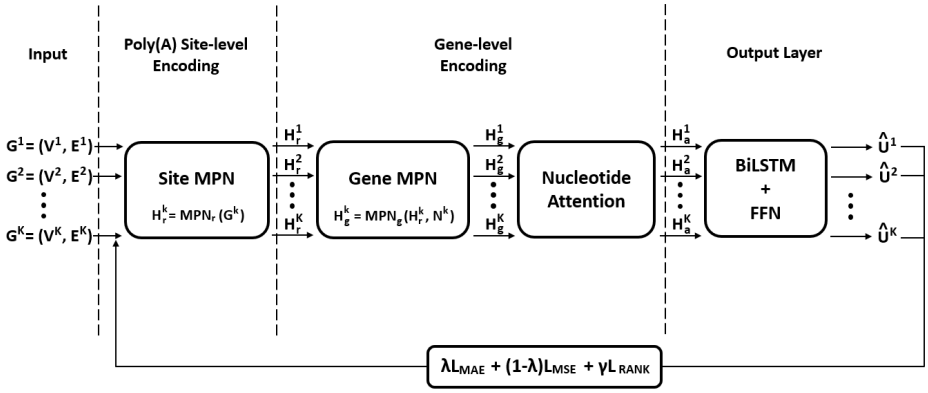


Fig. 2: Overview of the proposed HAGAPS model.

3 The Proposed HAGAPS Model

The HAGAPS model is designed to predict the usage values of the different poly(A) sites within a gene. The input is a graph $\mathcal{G}^k = (\mathcal{V}^k, \mathcal{E}^k)$ for each poly(A) site $k \in [1, K]$, where the node set $\mathcal{V}^k$ is derived from the encoded RNA sequence and the edge set $\mathcal{E}^k$ is determined by the RNA secondary structure. Notably, genes do not have a uniform number of poly(A) sites, that is K is not a constant. To accommodate this variability, HAGAPS is designed to handle inputs of different sizes, ensuring flexibility across genes. As illustrated in Fig. 2, the model adopts a two-level hierarchical encoding structure, poly(A) site-level and gene-level encoding, leveraging hierarchical GNNs with two custom MPNs. Specifically, the poly(A) site level consists of the Site MPN and the gene level consists of the Gene MPN and a nucleotide attention mechanism. Firstly, each $\mathcal{G}^k$ is processed by the Site MPN, which encodes information at the poly(A) site level, that is each site is independently encoded based on its sequence and structure. Nodes are updated via message passing with their neighborhoods, defined from both sequential and structural relationships. This process results in the node embedding $\mathbf{H}_r^k \in \mathbb{R}^{l \times h}$, where l is the length of the RNA sequence and h is the hidden embedding size. To capture interactions between alternative poly(A) sites within the same gene, the model then performs gene-level encoding. More specifically, the Gene MPN first updates eack poly(A) site's embedding $\mathbf{H}_r^k$ based on messages from its neighborhood, that is the rest $K-1$ poly(A) sites, producing $\mathbf{H}_g^k \in \mathbb{R}^{l \times h}$. Then, the nucleotide attention integrates both $\mathbf{H}_r^k$ and $\mathbf{H}_g^k$ to enhance poly(A) site representation. These interactions ensure that the prediction for each poly(A) site is influenced by the entire gene context, yielding the embedding $\mathbf{H}_a^k$. Finally, $\mathbf{H}_a^k$ is passed through an output layer to generate the predicted usage value $\hat{\mathbf{U}}^k \in [0, 1]$. During the training process, HAGAPS minimizes the error between predicted and true usage values using the $\mathcal{L}_{MAE}$ and $\mathcal{L}_{MSE}$ losses, while also optimizing poly(A) site ranking via the $\mathcal{L}_{RANK}$ loss.

3.1 Sequence & Structure Representation

Each poly(A) site k is represented as a graph $\mathcal{G}^k = (\mathcal{V}^k, \mathcal{E}^k)$, where $\mathcal{V}^k$ is the set of node features derived from the RNA sequence and $\mathcal{E}^k$ is the set of weighted edges determined by the RNA secondary structure. Given an initial RNA sequence $S = \{s_i | i = 1, 2, \ldots, l\}$, we encode S using a mapping function:

$$m(s_i) = \begin{cases} 1, & \text{if } s_i = \text{A} \\ 2, & \text{if } s_i = \text{T or U} \\ 3, & \text{if } s_i = \text{C} \\ 4, & \text{if } s_i = \text{G} \end{cases}. \tag{1}$$

Thus, we obtain the node set $\mathcal{V} = \{v_i | i = 1, 2, \ldots, l\}$, where $v_i = m(s_i)$.

The structure of a poly(A) site is defined by two types of bonds between nucleotides: i) phosphodiester bonds between consecutive nucleotides and ii) hydrogen bonds between complementary bases. The edge set regarding phosphodiester bonds is defined as:

$$\mathcal{E}_{cov} = \{(i, i+1, w_{cov}(i, i+1)) | 0 \leq i \leq l-1\}, \tag{2}$$

where $w_{cov}(i, i+1)$ is the edge weight, set to a constant value of 1, as these bonds always do exist. To obtain hydrogen bonds, we use the RNAplfold package [3], which, given an RNA sequence, outputs probable RNA secondary structures based on thermodynamic principles. Rather than relying solely on the minimum free energy structure, we retain an ensemble of probable structures to capture the inherent uncertainty and dynamics of RNA folding. The probability of a given structure X for sequence S is defined as:

$$p(X|S) = \frac{1}{Z} e^{-\beta E(X,S)}, \tag{3}$$

where Z is a partition function and $E(X, S)$ represents the free energy of S under X [28,39]. Considering all possible secondary structures in a thermodynamic equilibrium, the base-pairing probability for nucleotides i and j is then computed as:

$$p([i, j]|S) = \sum_{[i,j] \in X} p(X|S). \tag{4}$$

Based on these probabilities, we define the edge set for hydrogen bonds as:

$$\mathcal{E}_{base} = \{(i, j, w_{base}(i, j)) | p([i, j]|S) > 0\}, \tag{5}$$

where $w_{base}(i, j) = p([i, j]|S)$ is the edge weight, reflecting the likelihood of base pairing. Finally, the overall weighted edge set is the union of the phosphodiester and hydrogen bond edge sets:

$$\mathcal{E} = \mathcal{E}_{cov} \cup \mathcal{E}_{base}, \tag{6}$$

ensuring a comprehensive structural representation of the poly(A) site.

3.2 Site MPN

Firstly, each poly(A) site is encoded independently to ensure that the model learns a representation based on its sequence and structure. The encoding process begins by passing $\mathcal{V}^k$ through an embedding layer that maps each distinct nucleotide to a dense vector representation in a h-dimensional space, resulting in $\mathcal{G}'^k = (\mathcal{V}'^k, \mathcal{E}^k)$. Next, $\mathcal{G}'^k$ is passed to a custom MPN, namely the Site MPN, as shown in Fig. 2, consisting of a GNN and a CNN. This way, we obtain the node embedding $\mathbf{H}_r^k \in \mathbb{R}^{l \times h}$ from MPN_r for the k^{th} poly(A) site:

$$\mathbf{H}_r^k = \text{MPN}_r(\mathcal{G}'^k) = \text{CNN}(\text{GNN}(\mathcal{G}'^k)). \tag{7}$$

This approach allows the model to effectively integrate both structural and sequential information. Our GNN employs auto-regressive moving average graph convolutional (ARMAConv) layers [4], as follows:

$$\mathbf{H}^k = \frac{1}{C} \sum_{c=1}^{C} \mathbf{H}_c^{k(T)} \tag{8}$$

with $\mathbf{H}_c^{k(T)}$ being recursively defined by:

$$\mathbf{H}_c^{k(t+1)} = \sigma(\mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}\mathbf{H}_c^{k(t)}\mathbf{W}_1 + \mathbf{H}^{k(0)}\mathbf{W}_2), \quad 1 \le t \le T, \ 1 \le c \le C. \tag{9}$$

Here, $\mathbf{A} \in \mathbb{R}^{l \times l}$ is the weighted adjacency matrix, $\mathbf{D}$ is the degree matrix, $\mathbf{W}_1, \mathbf{W}_2 \in \mathbb{R}^{h \times h}$ are trainable parameters, T is the number of ARMAConv layers, C is the number of parallel stacks of layers, and $\sigma(\cdot)$ is the non-linear activation function ReLU. Then, the node embedding $\mathbf{H}^k \in \mathbb{R}^{l \times h}$ is further refined by the CNN, which is a two-layer convolutional network with batch normalization, max-pooling, and the non-linear activation function LeakyReLU:

$$\mathbf{H}_r^k = \text{CNN}(\mathbf{H}^k). \tag{10}$$

This design allows MPN_r to effectively capture both local and long-range dependencies in the RNA sequence. Regarding local interactions, each nucleotide embedding is updated using hydrogen and phosphodiester bond information using Eq. (8) and (9). As for long-range dependencies, messages from more distant nucleotides along the RNA backbone are aggregated via Eq. (10).

3.3 Gene MPN & Nucleotide Attention

The gene-level network facilitates communication between each poly(A) site k and its neighborhood within the gene, that is the rest of the poly(A) sites of the gene. This interaction ensures that the predicted usage value of a given poly(A) site is influenced by the alternative sites, effectively capturing poly(A) site competition within the gene. To achieve this, we construct the neighborhood of poly(A) site k as:

$$\mathbf{N}^k = \sum_{\substack{d=1 \\ d \neq k}}^{K} \mathbf{H}_r^d. \tag{11}$$

This formulation allows the model to handle the variable number of poly(A) sites across genes. Then, we feed the poly(A) site-level representation $\mathbf{H}_r^k$, along with its neighborhood $\mathbf{N}^k$, into the Gene MPN (Fig. 2):

$$\mathbf{H}_g^k = \text{MPN}_g(\mathbf{H}_r^k, \mathbf{N}^k) = \text{LSTM}(\mathbf{H}_r^k, \mathbf{N}^k), \tag{12}$$

where MPN_g is composed of a one-layer LSTM network with ReLU activation. The output $\mathbf{H}_g^k \in \mathbb{R}^{l \times h}$ provides an updated representation of the poly(A) site, determined by messages from the other sites in the gene. To further refine the representation, we employ a nucleotide attention mechanism, as illustrated in Fig. 2, with the following architecture:

$$\mathbf{a}^k = \mathbf{H}_r^{k\top} \mathbf{H}_g^k \tag{13a}$$

$$\mathbf{w}_i^k = \frac{\exp(\mathbf{a}_i^k)}{\sum_{j=1}^{l} \exp(\mathbf{a}_j^k)} \tag{13b}$$

$$\mathbf{b}^k = \sum_{i=1}^{l} \mathbf{w}_i^k \mathbf{H}_{i,r}^k \tag{13c}$$

$$\mathbf{H}_a^k = [\mathbf{H}_g^k; \mathbf{b}^k]. \tag{13d}$$

Thus, we obtain the attended context embedding $\mathbf{H}_a^k \in \mathbb{R}^{l \times 2h}$, which incorporates sequence-dependent interactions along the spatial axis of the RNA [33].

In summary, each poly(A) site k is initially influenced by the remaining $K-1$ competing sites using Eq. (12). Then, the attention mechanism utilizes both the individual poly(A) site-level embedding and the gene-level updated embedding to enhance the representation through Eq. (13a) - (13d).

Finally, $\mathbf{H}_a^k$ is passed to a BiLSTM layer with LeakyReLU to learn a more global nucleotide embedding, and a two-layer Feed-Forward Network (FFN) with LeakyReLU and Softmax respectively, to obtain the final predicted usage value $\hat{\mathbf{U}}^k$.

3.4 Model Optimization

The loss function of the model consists of three parts:

- $\mathcal{L}_{MAE}$ is the Mean Absolute Error (MAE) loss between the predicted $\hat{\mathbf{U}}$ and true $\mathbf{U}$ usage values:

$$\mathcal{L}_{MAE} = \frac{1}{K} \sum_{k=1}^{K} |\hat{\mathbf{U}}^k - \mathbf{U}^k| \tag{14}$$

- $\mathcal{L}_{MSE}$ is the Mean Squared Error (MSE) loss between the predicted $\hat{\mathbf{U}}$ and true $\mathbf{U}$ usage values:

$$\mathcal{L}_{MSE} = \frac{1}{K} \sum_{k=1}^{K} (\hat{\mathbf{U}}^k - \mathbf{U}^k)^2 \tag{15}$$

– $\mathcal{L}_{RANK}$ penalizes deviations from the expected ranking of poly(A) sites within a gene. It is computed as:

$$\mathcal{L}_{RANK} = 1 - corr(\hat{r}, r), \tag{16}$$

where *corr* denotes the Spearman correlation coefficient, and $\hat{r}$, r are the predicted and true rankings, respectively. Both rankings are determined by placing the poly(A) sites in descending order based on their predicted and true usage values.

Thus, we formulate the joint loss function:

$$\mathcal{L} = \lambda\mathcal{L}_{MAE} + (1-\lambda)\mathcal{L}_{MSE} + \gamma\mathcal{L}_{RANK}, \tag{17}$$

where γ and λ are regularization parameters that control the influence of the $\mathcal{L}_{RANK}$ and $\mathcal{L}_{MAE}$, $\mathcal{L}_{MSE}$ losses, respectively. More specifically, λ governs the balance between $\mathcal{L}_{MAE}$ and $\mathcal{L}_{MSE}$.

Optimization Our proposed HAGAPS model was developed in PyTorch. Since HAGAPS is designed to process all alternative poly(A) sites within a gene simultaneously, we group poly(A) sites by their corresponding gene before passing them to the model. This means that the batch size refers to the number of gene groups rather than individual poly(A) sites. During training, in each epoch, the model learns a low-dimensional representation of the initial poly(A) sites via the joint loss function $\mathcal{L}$ of Eq. (17). By minimizing the $\mathcal{L}_{MAE}$ and $\mathcal{L}_{MSE}$ losses of Eq. (14) and (15), HAGAPS outputs usage values that are closer to the ground truths. Furthermore, optimizing the $\mathcal{L}_{RANK}$ loss of Eq. (16) improves the model's ability to correctly capture the relative strength of poly(A) sites. Optimization is achieved through backpropagation using the Adam optimizer.

4 Experiments

4.1 Datasets

For the model evaluation, we use two publicly available poly(A) site quantification datasets[1]. The data were derived from the fibroblast cell lines of the C57BL/6J (BL) and SPRET/EiJ (SP) mouse strains [23,38]. To build a poly(A) site reference for the two strains, the total RNA was extracted from their fibroblasts, and then subjected to $3'$ region extraction and deep sequencing [17]. Subsequently, $3'$ mRNA sequencing was performed to quantify poly(A) site usage, with usage values computed based on sequencing read counts. The sequence surrounding each cleavage site was extracted. While both datasets contain the same poly(A) sites associated with the same genes, they differ in their sequences, due to single nucleotide polymorphisms and indels, as well as in their poly(A) site usage values.

[1] Datasets

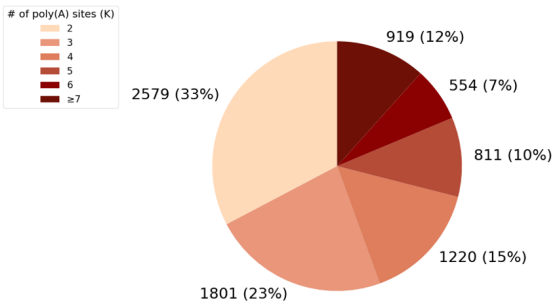


Fig. 3: Distribution of genes based on the number of alternative poly(A) sites K in the BL and SP datasets.

In the preprocessing phase, following the evaluation protocol of [23], we excluded poly(A) sites with missing usage values, thus retaining $30,940$ poly(A) sites belonging to $7,884$ genes for each dataset. As mentioned in Section 3, the number of alternative poly(A) sites K varies across genes. Fig. 3 illustrates the distribution of genes according to the number of associated poly(A) sites in the two datasets. Next, we randomly split the genes into training, validation and testing sets with a 6:2:2 ratio, yielding $4,730$ genes for training, $1,577$ for validation, and $1,577$ for testing. The dataset was split at the gene level rather than at the poly(A) site level to ensure that all poly(A) sites within the same gene remained in the same set, preserving their competing interactions. This splitting process was repeated five times for each dataset, and we report the average experimental results with the standard deviation.

4.2 Evaluation Protocol

To evaluate the models' ability to predict poly(A) site quantification, we report the following evaluation metrics:

- *Mean Absolute Error (MAE)* between the predicted $\hat{\mathbf{U}}$ and ground truth $\mathbf{U}$ usage values:

$$\text{MAE} = \frac{1}{F}\sum_{f=1}^{F}|\hat{\mathbf{U}}^f - \mathbf{U}^f|, \tag{18}$$

  where F is the total number of poly(A) sites across all genes in the testing set.
- *Root Mean Squared Error (RMSE)* between the predicted $\hat{\mathbf{U}}$ and ground truth $\mathbf{U}$ usage values:

$$\text{RMSE} = \sqrt{\frac{1}{F}\sum_{f=1}^{F}(\hat{\mathbf{U}}^f - \mathbf{U}^f)^2}. \tag{19}$$

– *Highest Usage Prediction Accuracy (HUPA)* evaluates the ability of predicting the poly(A) site with the highest usage within a gene. It is defined as the percentage of genes whose strongest poly(A) site is correctly predicted:

$$\text{HUPA} = \frac{M_{ch}}{M}, \tag{20}$$

where M is the total number of genes in the testing set, and M_{ch} is the number of genes with correctly identified highest-usage poly(A) sites [23].

In doing so, we assess the models on the direct regression task of poly(A) site usage prediction (MAE and RMSE), as well as the important task of identifying the strongest poly(A) site within a gene (HUPA).

4.3 Compared Methods

– **Allocator**[2] consists of four parallel feature extractors, two for RNA sequences and two for RNA secondary structures. The model utilizes MLPs, multi-head attention mechanisms and GNNs [22].
– **RNASSR-Net**[3] utilizes a CNN to extract RNA sequence features and a GCN to capture RNA secondary structure features. In addition, the spatial importance learned by the CNN guides the GCN training process [25].
– **RPI-Net**[4] learns the RNA sequence and secondary structure information with a recurrent GNN and a BiLSTM. Moreover, a Set2Set model is employed to pool the node embeddings along the spatial axis of the RNA [39].
– **APARENT2**[5] employs a sequence-based residual neural network incorporating dilated convolutions [24].
– **DeeReCT-APA**[6] utilizes a CNN-BiLSTM architecture. The CNN extracts RNA sequence features and the BiLSTM models the competing interactions between poly(A) sites [23].
– **HAGAPS-Seq** is a variant of the proposed model, omitting the GNN in the Site MPN and the RNA secondary structure information.
– **HAGAPS-Site** is a variant of the proposed model, without the Gene MPN.
– **HAGAPS** is the proposed model.

The parameters of the examined models have been determined via cross-validation and we report the best results in our experiments. For the proposed method, the learning rate is set to 1e-3 with a batch size of 32 across both datasets. The parameter analysis of HAGAPS is further investigated in Section 4.5.

[2] Allocator code
[3] RNASSR-Net code
[4] RPI-Net code
[5] APARENT2 code
[6] DeeReCT-APA code

Table 1: Average MAE, RMSE, and HUPA of the proposed HAGAPS method, when compared with its variants and the baselines on the BL and SP datasets. Bold values indicate the best method.

Datasets	Methods	MAE	RMSE	HUPA
BL	Allocator	0.2387 ± 0.0021	0.3851 ± 0.0054	0.4159 ± 0.0105
	RNASSR-Net	0.2350 ± 0.0032	0.3778 ± 0.0062	0.4814 ± 0.0077
	RPI-Net	0.2118 ± 0.0044	0.3369 ± 0.0085	0.5978 ± 0.0144
	APARENT2	0.2290 ± 0.0043	0.3231 ± 0.0105	0.4685 ± 0.0147
	DeeReCT-APA	0.2081 ± 0.0272	0.2755 ± 0.0215	0.5646 ± 0.0590
	HAGAPS-Seq	0.1824 ± 0.0035	0.2717 ± 0.0039	0.5910 ± 0.0070
	HAGAPS-Site	0.1751 ± 0.0047	0.2648 ± 0.0025	0.6086 ± 0.0058
	HAGAPS	$\mathbf{0.1696 \pm 0.0050}$	$\mathbf{0.2599 \pm 0.0043}$	$\mathbf{0.6257 \pm 0.0059}$
SP	Allocator	0.2456 ± 0.0193	0.3762 ± 0.0170	0.4183 ± 0.0295
	RNASSR-Net	0.2328 ± 0.0021	0.3770 ± 0.0073	0.4888 ± 0.0200
	RPI-Net	0.2157 ± 0.0023	0.3403 ± 0.0048	0.5698 ± 0.0232
	APARENT2	0.2313 ± 0.0044	0.3325 ± 0.0104	0.4632 ± 0.0124
	DeeReCT-APA	0.2133 ± 0.0267	0.2839 ± 0.0196	0.5362 ± 0.0674
	HAGAPS-Seq	0.1811 ± 0.0052	0.2767 ± 0.0044	0.5990 ± 0.0086
	HAGAPS-Site	0.1808 ± 0.0026	0.2740 ± 0.0043	0.5905 ± 0.0072
	HAGAPS	$\mathbf{0.1770 \pm 0.0026}$	$\mathbf{0.2688 \pm 0.0036}$	$\mathbf{0.6159 \pm 0.0067}$

4.4 Performance Evaluation

In Table 1, we show the experimental results of the examined models on the BL and SP datasets, in terms of average MAE, RMSE, and HUPA. The results indicate that the proposed HAGAPS model and its variants outperform the baseline models across all metrics in both datasets. In particular, the GNN-based models, that is Allocator, RNASSR-Net, and RPI-Net, overlook the relationships between competing poly(A) sites. Ignoring the co-existence of multiple poly(A) sites within a gene, which determines how the total usage is distributed among them, results in lower prediction accuracy. Meanwhile, DeeReCT-APA and APARENT2 do not incorporate RNA secondary structure information, relying solely on RNA sequences. Therefore, these methods do not take advantage of the poly(A) site structural data, lacking a more meaningful representation. Regarding the variants of the HAGAPS model, that is HAGAPS-Seq and HAGAPS-Site, they outperform the other baselines, highlighting the valuable contributions of the Gene and Site MPNs, respectively. Nevertheless, the HAGAPS-Seq variant underperforms compared to HAGAPS-Site and HAGAPS, emphasizing the significance of RNA secondary structures and GNNs in the analysis of poly(A) sites. Likewise, HAGAPS-Site performs worse than HAGAPS, underscoring the importance of the custom attention-based modeling of poly(A) site interactions. Overall, the proposed HAGAPS model consistently achieves the best performance across all metrics on the two datasets. By leveraging hierarchical GNNs through two levels of MPNs, poly(A) site and gene level, HAGAPS delivers the

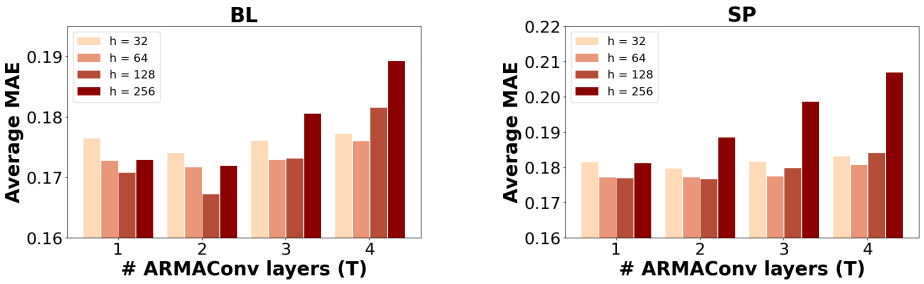


Fig. 4: Average MAE of HAGAPS on the BL and SP datasets when varying the number of ARMAConv layers T and the hidden embedding size h.

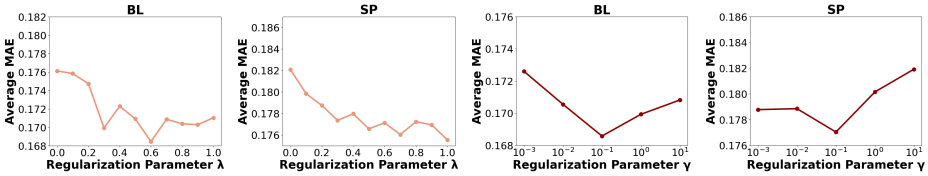
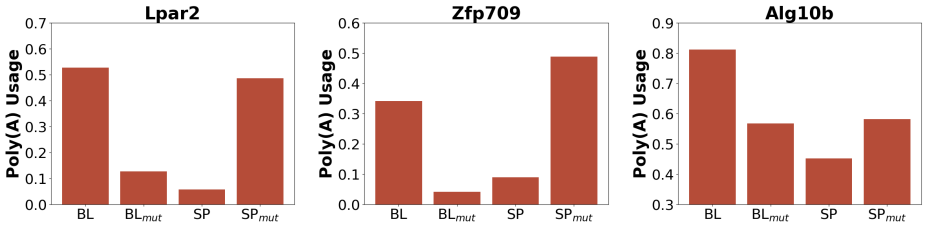


Fig. 5: Average MAE of HAGAPS on the BL and SP datasets when varying the regularization parameters λ and γ.

most accurate predictions for both poly(A) site usage, expressed by the lowest MAE and RMSE, and the identification of the strongest poly(A) site in a gene, corresponding to the highest HUPA values.

4.5 Parameter Tuning

The most important parameters in our model are i) the hidden embedding size h, ii) the number of ARMAConv layers T in the Site MPN, iii) the regularization parameter λ for the $\mathcal{L}_{MAE}$ and $\mathcal{L}_{MSE}$ losses, and iv) the regularization parameter γ for the $\mathcal{L}_{RANK}$ loss. For $h = \{32, 64, 128, 256\}$ we vary $T \in [1, 4]$ by a step of 1. In Fig. 4, we demonstrate the impact of h and T, reporting the average MAE on the BL and SP datasets. We observe that the best architecture is obtained with $h = 128$ and $T = 2$ for both datasets. Notably, selecting significantly higher or lower values for h and T leads to overfitting or underfitting, preventing the model from effectively capturing poly(A) site representations. Fig. 5 illustrates the effect of the regularization parameters λ and γ in Eq.(17), on the performance of our model in terms of MAE across the two datasets. In particular, we vary $\lambda \in [0, 1]$ by a step of 0.1, where lower values emphasize the $\mathcal{L}_{MSE}$ loss and higher values prioritize the $\mathcal{L}_{MAE}$ loss. Aiming for a balance between the two loss functions, we fix λ to 0.3 for both datasets. Moreover, we vary γ in $\{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}$ and observe that the best performance is achieved when setting $\gamma = 10^{-1}$ for both datasets. Lower values of γ result in decreased performance, highlighting the importance of the $\mathcal{L}_{RANK}$ loss in capturing the



(a) (b) (c)

Fig. 6: Predicted usage values of the original BL, SP and the mutated BL$_{mut}$, SP$_{mut}$ poly(A) sites. BL$_{mut}$ and SP$_{mut}$ denote the respective BL and SP sequences with a variation. (a) BL$_{mut}$ has the G substitution and SP$_{mut}$ has the canonical A instead of G for the *Lpar2* gene. (b) BL$_{mut}$ has the T substitution and SP$_{mut}$ has the canonical A instead of T for the *Zfp709* gene. (c) BL$_{mut}$ has the UUUU insertion and SP$_{mut}$ lacks the insertion for the *Alg10b* gene.

relative strength of competing poly(A) sites. However, excessively high γ values also degrade performance, as they place disproportionate emphasis on the $\mathcal{L}_{RANK}$ loss, diminishing the contributions of $\mathcal{L}_{MAE}$ and $\mathcal{L}_{MSE}$.

4.6 Case Study: Impact of sequence variations on APA

To further investigate the ability of HAGAPS in understanding APA regulation, we leverage experimental findings from [38]. The study demonstrates that specific sequence variations in the SP strain relative to the BL strain (Section 4.1) in the distal poly(A) site of three genes, that is *Lpar2*, *Zfp709*, and *Alg10b* (Ensembl Gene IDs: ENSMUSG00000031861, ENSMUSG00000056019, ENSMUSG00000075470), result in reduced poly(A) site usage in the SP strain compared to BL. Specifically, these variations include a substitution from A to G in *Lpar2*, a substitution from A to T in *Zfp709*, and an insertion of UUUU in *Alg10b*. To assess whether HAGAPS can predict the impact of these sequence alterations on poly(A) site usage, we generate two additional mutated sequences for each gene, along with their respective structures, by swapping the sequence differences between BL and SP [23,38]. Our analysis in Fig. 6 showcases the ability of the proposed model to capture the effects of the variations on poly(A) site usage. In all three genes, the SP strain exhibits lower predicted usage than BL. Moreover, consistent with experimental observations, the mutated BL$_{mut}$ shows reduced usage relative to BL, whereas the mutated SP$_{mut}$ exhibits higher usage compared to SP. Consequently, these results highlight the potential of HAGAPS to contribute to the study of APA, offering valuable insights that may advance gene regulation research.

5 Conclusion

In this study, we presented HAGAPS, a hierarchical GNN-based approach for alternative poly(A) site quantification prediction. The two key factors of the proposed model are i) the poly(A) site-level MPN that integrates RNA secondary structure information, and ii) the gene-level MPN coupled with a nucleotide attention mechanism to capture the competing interactions between multiple alternative poly(A) sites. Our experimental evaluation demonstrates the superiority of HAGAPS compared to several state-of-the-art methods. In addition, we conducted a case study, showcasing the model's ability in uncovering the underlying mechanisms of APA. An interesting future direction is the incorporation of a parameter-evolving strategy between alternative poly(A) sites, enhancing communication within the gene [29].

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Arefeen, A., Xiao, X., Jiang, T.: Deeppasta: deep neural network based polyadenylation site analysis. Bioinformatics **35**(22), 4577–4585 (2019)
2. Barreau, C., Paillard, L., Osborne, H.B.: Au-rich elements and associated factors: are there unifying principles? Nucleic acids research **33**(22), 7138–7150 (2005)
3. Bernhart, S.H., Hofacker, I.L., Stadler, P.F.: Local rna base pairing probabilities in large sequences. Bioinformatics **22**(5), 614–615 (2006)
4. Bianchi, F.M., Grattarola, D., Livi, L., Alippi, C.: Graph neural networks with convolutional arma filters. IEEE transactions on pattern analysis and machine intelligence **44**(7), 3496–3507 (2021)
5. Bogard, N., Linder, J., Rosenberg, A.B., Seelig, G.: A deep neural network for predicting and engineering alternative polyadenylation. Cell **178**(1), 91–106 (2019)
6. Bonfert, T., Friedel, C.C.: Prediction of poly (a) sites by poly (a) read mapping. PLoS One **12**(1), e0170914 (2017)
7. Brown, P.H., Tiley, L.S., Cullen, B.R.: Effect of rna secondary structure on polyadenylation site selection. Genes & development **5**(7), 1277–1284 (1991)
8. Colgan, D.F., Manley, J.L.: Mechanism and regulation of mrna polyadenylation. Genes & development **11**(21), 2755–2766 (1997)
9. Crick, F.: Central dogma of molecular biology. Nature **227**(5258), 561–563 (1970)
10. Danckwardt, S., Hentze, M.W., Kulozik, A.E.: 3′ end mrna processing: molecular mechanisms and implications for health and disease. The EMBO journal **27**(3), 482–498 (2008)
11. Derti, A., Garrett-Engele, P., MacIsaac, K.D., Stevens, R.C., Sriram, S., Chen, R., Rohl, C.A., Johnson, J.M., Babak, T.: A quantitative atlas of polyadenylation in five mammals. Genome research **22**(6), 1173–1183 (2012)
12. Di Giammartino, D.C., Nishida, K., Manley, J.L.: Mechanisms and consequences of alternative polyadenylation. Molecular cell **43**(6), 853–866 (2011)
13. Elkon, R., Ugalde, A.P., Agami, R.: Alternative cleavage and polyadenylation: extent, regulation and function. Nature Reviews Genetics **14**(7), 496–506 (2013)

14. Gruber, A.J., Schmidt, R., Ghosh, S., Martin, G., Gruber, A.R., van Nimwegen, E., Zavolan, M.: Discovery of physiological and cancer-related regulators of 3′ utr processing with kapac. Genome biology **19**, 1–17 (2018)
15. Ha, K.C., Blencowe, B.J., Morris, Q.: Qapa: a new method for the systematic analysis of alternative polyadenylation from rna-seq data. Genome biology **19**, 1–18 (2018)
16. Higgs, P.G.: Rna secondary structure: physical and computational aspects. Quarterly reviews of biophysics **33**(3), 199–253 (2000)
17. Hoque, M., Ji, Z., Zheng, D., Luo, W., Li, W., You, B., Park, J.Y., Yehia, G., Tian, B.: Analysis of alternative cleavage and polyadenylation by 3′ region extraction and deep sequencing. Nature methods **10**(2), 133–139 (2013)
18. Kalkatawi, M., Magana-Mora, A., Jankovic, B., Bajic, V.B.: Deepgsr: an optimized deep-learning structure for the recognition of genomic signals and regions. Bioinformatics **35**(7), 1125–1132 (2019)
19. Kalkatawi, M., Rangkuti, F., Schramm, M., Jankovic, B.R., Kamau, A., Chowdhary, R., Archer, J.A., Bajic, V.B.: Dragon polya spotter: predictor of poly (a) motifs within human genomic dna sequences. Bioinformatics **28**(1), 127–129 (2012)
20. Lau, A.G., Irier, H.A., Gu, J., Tian, D., Ku, L., Liu, G., Xia, M., Fritsch, B., Zheng, J.Q., Dingledine, R., et al.: Distinct 3′ utrs differentially regulate activity-dependent translation of brain-derived neurotrophic factor (bdnf). Proceedings of the National Academy of Sciences **107**(36), 15945–15950 (2010)
21. Leung, M.K., Delong, A., Frey, B.J.: Inference of the human polyadenylation code. Bioinformatics **34**(17), 2889–2898 (2018)
22. Li, F., Bi, Y., Guo, X., Tan, X., Wang, C., Pan, S.: Advancing mrna subcellular localization prediction with graph neural network and rna structure. Bioinformatics **40**(8), btae504 (2024)
23. Li, Z., Li, Y., Zhang, B., Li, Y., Long, Y., Zhou, J., Zou, X., Zhang, M., Hu, Y., Chen, W., et al.: Deerect-apa: prediction of alternative polyadenylation site usage through deep learning. Genomics, Proteomics and Bioinformatics **20**(3), 483–495 (2022)
24. Linder, J., Koplik, S.E., Kundaje, A., Seelig, G.: Deciphering the impact of genetic variation on human polyadenylation using aparent2. Genome biology **23**(1), 232 (2022)
25. Liu, Z., Luo, F., Du, B.: Rna secondary structure representation network for rna-proteins binding prediction. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 362–370 (2021)
26. Long, R., Guo, Z., Han, D., Liu, B., Yuan, X., Chen, G., Heng, P.A., Zhang, L.: sirnadiscovery: a graph neural network for sirna efficacy prediction via deep rna sequence analysis. Briefings in Bioinformatics **25**(6), bbae563 (2024)
27. Magana-Mora, A., Kalkatawi, M., Bajic, V.B.: Omni-polya: a method and tool for accurate recognition of poly (a) signals in human genomic dna. BMC genomics **18**, 1–13 (2017)
28. McCaskill, J.S.: The equilibrium partition function and base pair binding probabilities for rna secondary structure. Biopolymers: Original Research on Biomolecules **29**(6-7), 1105–1119 (1990)
29. Pareja, A., Domeniconi, G., Chen, J., Ma, T., Suzumura, T., Kanezashi, H., Kaler, T., Schardl, T., Leiserson, C.: Evolvegcn: Evolving graph convolutional networks for dynamic graphs. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 5363–5370 (2020)
30. Richard, P., Manley, J.L.: Transcription termination by nuclear rna polymerases. Genes & development **23**(11), 1247–1269 (2009)

31. Tian, B., Hu, J., Zhang, H., Lutz, C.S.: A large-scale analysis of mrna polyadenylation of human and mouse genes. Nucleic acids research **33**(1), 201–212 (2005)
32. Tian, B., Manley, J.L.: Alternative polyadenylation of mrna precursors. Nature reviews Molecular cell biology **18**(1), 18–30 (2017)
33. Vinyals, O., Bengio, S., Kudlur, M.: Order matters: Sequence to sequence for sets. In: Proceedings of the International Conference on Learning Representations (2016)
34. Wahle, E., Kühn, U.: The mechanism of 3′ cleavage and polyadenylation of eukaryotic pre-mrna. Progress in nucleic acid research and molecular biology **57**, 41 (1997)
35. Wickens, M., Anderson, P., Jackson, R.J.: Life and death in the cytoplasm: messages from the 3′ end. Current opinion in genetics & development **7**(2), 220–232 (1997)
36. Xia, Z., Donehower, L.A., Cooper, T.A., Neilson, J.R., Wheeler, D.A., Wagner, E.J., Li, W.: Dynamic analyses of alternative polyadenylation from rna-seq reveal a 3′-utr landscape across seven tumour types. Nature communications **5**(1), 5274 (2014)
37. Xia, Z., Li, Y., Zhang, B., Li, Z., Hu, Y., Chen, W., Gao, X.: Deerect-polya: a robust and generic deep learning method for pas identification. Bioinformatics **35**(14), 2371–2379 (2019)
38. Xiao, M.S., Zhang, B., Li, Y.S., Gao, Q., Sun, W., Chen, W.: Global analysis of regulatory divergence in the evolution of mouse alternative polyadenylation. Molecular systems biology **12**(12), 890 (2016)
39. Yan, Z., Hamilton, W.L., Blanchette, M.: Graph neural representational learning of rna secondary structures for predicting rna-protein interactions. Bioinformatics **36**(Supplement_1), i276–i284 (2020)
40. Ye, C., Long, Y., Ji, G., Li, Q.Q., Wu, X.: Apatrap: identification and quantification of alternative polyadenylation sites from rna-seq data. Bioinformatics **34**(11), 1841–1849 (2018)