SpliceSelectNet: A Hierarchical Transformer-Based Deep Learning Model for Splice Site Prediction

Yuna Miyachi[1] and Kenta Nakai[1, 2, *]

[1]Department of Computer Science, Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, 113-8656, Tokyo, Japan and [2]Human Genome Center, The Institute of Medical Science,, The University of Tokyo, 4-6-1 Shirokanedai, Minato-ku, 108-8639, Tokyo, Japan
[*]Corresponding author. knakai@ims.u-tokyo.ac.jp

Abstract

Accurate RNA splicing is essential for gene expression and protein function, yet the mechanisms governing splice site recognition remain incompletely understood. Aberrant splicing caused by mutations can lead to severe diseases, including cancer and genetic disorders, underscoring the need for accurate computational tools to predict splice sites and detect disruptions. Existing methods have made significant advances in splice site prediction but are often limited in handling long-range dependencies, a factor critical to splicing regulation. Moreover, many models lack interpretability, hindering efforts to elucidate the underlying biological mechanisms. Here, we present SpliceSelectNet (SSNet), a novel deep-learning model that predicts splice sites directly from DNA sequences. This model is capable of handling long-range dependencies (up to 100 kb) using a hierarchical Transformer-based architecture with both local and global attention mechanisms. SSNet offers interpretability at the single nucleotide level, making it particularly effective for identifying aberrant splicing caused by mutations. Our model surpasses the state-of-the-art (SoTA) in splice site prediction on the Gencode test dataset and demonstrates superior performance in aberrant splicing prediction on the BRCA dataset and deep intronic dataset.

Key words: RNA splicing, aberrant splicing, Transformer, deep learning

Introduction

RNA splicing is a fundamental process in eukaryotic gene expression that allows for the removal of non-coding introns and the joining of coding exons to generate mature mRNA. This intricate mechanism is essential for producing functional proteins and is tightly regulated by various cis-acting elements and trans-acting factors. Misregulation of splicing can lead to aberrant splicing events, which are involved in numerous diseases, including multiple types of cancer [1], neurodegenerative disorders [2], and genetic syndromes [3]. For example, in the BRCA1 and BRCA2 genes, which are known to be associated with breast and ovarian cancer, aberrant alternative splicing is highly associated with the development of breast cancer [4]. As an additional example, Duchenne muscular dystrophy is also caused by an variant in the dystrophin gene [5]. Exon skipping occurs, thereby causing a frameshift that results in the loss of functions important for maintaining muscle cell membrane integrity. In response to this, exon-skipping drugs have been developed to avoid the frameshift by skipping neighboring exons, resulting in the production of a protein that is shorter than normal but still retains its function [6]. Consequently, understanding and predicting splicing is critical for both basic research and clinical applications.

To better understand splicing events, comprehensive databases and systematic analyses have been developed. One of the earliest examples is the creation of a database of aberrant splicing mutations in human genes [7]. This database has provided a foundation for identifying common patterns in splicing errors and understanding their mechanisms. Building upon such foundational work, researchers have developed computational tools to analyze and predict splicing events. For instance, tools like MaxEntScan [8] utilize motif-based approaches and sequence-based statistical models to predict splice site strength. This method relies heavily on predefined features and statistical assumptions, which can limit their accuracy and flexibility when dealing with complex genomic variations.

Recent advances in computational methods for the prediction of splice sites have leveraged machine learning and deep learning techniques to enhance predictive accuracy. The advent of deep learning has provided a new avenue for addressing these challenges, allowing models to learn complex patterns from large datasets.

Among the notable approaches in this field is SpliceAI [9], a deep learning model that utilizes a neural network architecture to predict splice sites based solely on nucleotide sequences. SpliceAI has set new benchmarks in splice site prediction accuracy; however, it only consists of convolutional layers and may lack high interpretability. Another prominent tool is Pangolin [10], which integrates RNA-seq data from multiple tissues to improve splice site predictions. It employs a similar deep learning framework but emphasizes splice site usage rates rather than binary classification, providing a more nuanced view of splice site strength. Although Pangolin has shown improvements over previous models, it does not explicitly address the interpretability of its predictions.

DNABERT [11], a variant of the BERT [12] model adapted for DNA sequences, has also been introduced to the field. By leveraging the transformer architecture, DNABERT captures contextual relationships between sequences, allowing the model to perform various genomic tasks, including splice site prediction. However, like many existing models,

DNABERT primarily focuses on sequence representation without specialized mechanisms for handling the long-range dependencies inherent in splicing. Since DNABERT just adopted the same architecture of BERT, used in natural language processing, its limitation of the input sequence is 512 tokens, which is too short for DNA sequences. Regulatory elements like splicing enhancers and silencers are located far from splice sites and play a critical role in determining splicing outcome. In the scenario of aberrant splicing, intronic mutations several kilobases away from canonical splice sites can activate cryptic splice sites or disrupt regulatory motifs, causing aberrant splicing. Thus, considering long-range interactions is essential for accurate splice site prediction.

To address these issues, we propose SpliceSelectNet (SSNet), a hierarchical Transformer [13]-based model designed specifically for splice site prediction from DNA sequences. SSNet uniquely integrates local and global attention mechanisms to capture long-range dependencies (up to 100 kb) while maintaining interpretability. This model aims not only to accurately predict splice sites but also to identify aberrant splicing events caused by mutations, classifying them as either pathogenic or benign.

The objectives of this study are twofold: (1) to develop a highly accurate and interpretable model for splice site prediction using dense attention mechanisms and (2) to evaluate the model's performance in predicting aberrant splicing in disease-related datasets. By addressing the limitations of existing models and providing insights into splicing mechanisms, we aim to contribute valuable knowledge to the field of genomics and personalized medicine.

Methods

Training Data

For this study, we utilized three primary datasets for training: Gencode, GTEx, and Pangolin, each contributing unique features for splice site prediction. We downloaded the DNA sequence hg19 assembly from the UCSC Genome Browser [14], and a DNA sequence between the start and end sites of the transcript was extracted for each gene. We finally padded examples shorter than the input length by zero value and split examples longer than the input length to align with it. When dividing an example, we split it sequentially from front to back, ensuring the last part matched the input length with overlap allowed. To split this dataset into one for training and evaluation, we used the genes in chromosomes 2, 4, 6, 8, 10-22, X, and Y for training, and the genes in chromosomes 1, 3, 5, 7, and 9 that have no paralogs for evaluation.

Gencode Dataset

This dataset is based on the Gencode V24lift37 gene annotations [15]. First, only protein-coding genes were extracted from the database. There were multiple isoforms for one gene, therefore a single principal transcript was used to make the acceptor/donor label. We removed transcripts that consist of only one exon, such that any transcript in this dataset has at least one intron. Using the exon region information from Gencode annotations, we defined acceptor/donor sites as the first/last nucleotide position of each exon, excluding the first/last exon, respectively. We assigned a non-splice site label for the nucleotide position that was neither acceptor nor donor site. This process assigned each nucleotide an acceptor, donor, or non-splice site label. This dataset is identical to that used in SpliceAI, but this study also used additional exon/intron labels.

GTEx Dataset

To enhance our model's ability to predict alternative splice sites, we incorporated additional splice sites from the GTEx dataset [16, 17], which includes intron junction information. Possible splice sites were identified from the end position of each junction seen in more than four individuals, and we ignored the end position beyond the ends of the transcript used in the Gencode dataset. The Gencode dataset primarily contains constitutive splice sites, while the GTEx dataset provides alternative splice sites. This integration resulted in a total number of splice sites that is approximately 1.5 times greater than the Gencode dataset alone. This dataset is also used in SpliceAI. As for exon/intron labels, the same labels as the Gencode dataet were used.

Pangolin Dataset

We employed RNA-seq data from seven human tissues: brain, cerebellum, heart, kidney, liver, ovary, and testis [18]. This dataset was generated from RNA-seq data used in Pangolin. First, RNA-seq data was mapped to their genomes using STAR 2.7.11 [19] based on Gencode V24lift37 gene annotations. Next, we assigned multimapped reads to a single location using MMR [20]. Instead of using binary labels (0/1), we calculated splice site usage rates (continuous values between 0 and 1) as labels. Splice-Site Strength Estimate (SSE) values were calculated using SpliSER 0.1.7 [21]. While Pangolin calculated SSE for each tissue, this study made no distinction and used the average value of each tissue as the label. Notably, we did not utilize exon/intron labels for training on this dataset.

Model Architecture

Our SSNet model is built on a hierarchical Transformer architecture designed to efficiently capture both local and global dependencies in nucleotide sequences. The architecture consists of the following components (Figure 1 a):

- Convolutional Layer: This layer extracts local features, allowing the model to recognize short-range interactions crucial for splice site identification, such as the GT-AG rule.
- Local Attention Mechanism: This layer is a novel aspect of our proposed architecture. It is designed to focus on nearby nucleotides (within 160bp), providing high-resolution attention to regions essential for donor and acceptor site recognition. By prioritizing the local context, the model can identify patterns and signals that indicate the presence of splice sites, enhancing prediction accuracy near the target sites.
- Global Attention Mechanism: Complementing the local attention, the global attention layer captures dependencies over long distances (up to 100 kb). This is particularly important for splicing regulation, as regulatory elements can be located far from the splice sites they influence. The integration of global attention allows the model to incorporate distant interactions and contextual information, which traditional models often overlook.

First, input sequences are transformed to a matrix by one-hot encoding, A, C, G, T, and padding mapped to [1, 0, 0, 0], [0, 1, 0, 0], [0, 0, 1, 0], [0, 0, 0, 1], and [0, 0, 0, 0]. Second, the encoded matrix is fed to the convolutional block with a

residual connection. In Figure 1 a, Conv(N, W) represents a convolutional layer, where N is the number of kernels and W is the kernel size. Third, the extracted local information through convolutional blocks is handled in the local attention block. The input matrix is compressed so that the global attention mechanism can handle it at once. The process of computation in the local attention mechanism is explained in detail in Section Local Attention. Then, the compressed information is fed to the global attention block, and the model learns an end-to-end global mutual relationship. The global attention mechanism is the multi-head self-attention mechanism with 8 heads. Finally, the global information is fed to each dense layer, reshaped to length 100k, and activated with a softmax function to predict the possibility of an acceptor/donor/non-splice site and that of an exon/intron for each nucleotide position. The loss value is calculated for each prediction and the sum of them is used for optimization. The detailed loss function is explained in Section Loss Function.

In the conventional attention mechanism, the computational complexity is the square of the input length, so the limit of dense attention is a few hundred tokens. In this architecture, a two-stage hierarchical attention mechanism is combined to reduce the input length to a few hundred tokens in a single attention computation, thus preserving computational efficiency while learning long-range array interactions in the model as a whole. Figure 1 c illustrates the concept of the hierarchical attention mechanism. In local attention, the model divides the input into blocks consisting of short sequences and calculates the interactions within those blocks. Then, in global attention, the model computes the interaction between each block.

While sparse attention, such as linear attention [22], is often used to reduce the computational complexity to a polynomial input length to handle long-range sequences in the attention mechanism, this study proposes a new method to handle long-range sequences while maintaining dense attention. The dense matrix of local attention and global attention weights makes it possible to visualize which regions the model focuses on for prediction, thus preserving high interpretability.

Local Attention

Figure 1 b illustrates a process of local attention in SSNet-100k. First, input matrices that have a length of 100k are divided into 625 blocks with a length of 160 nt. Next, each block is fed to multi-head self-attention mechanisms with 2 heads using relative positional encoding [23, 24] in parallel. The output of each mechanism is of the same shape as the input, 160 nt × 64. Through a flatten layer and a dense layer, we have a 512-dimension vector from each output. We now have 625 vectors with 512 dimensions as a whole. Finally, we concatenate all of them and obtain an output matrix of size 625 × 512. Since the length of the matrix becomes 625 from 100k, the model can compute an end-to-end relationship in the global attention mechanism.

For models with input lengths of 1k and 10k, the first division into blocks was changed to 50 blocks of length 20 and 50 blocks of length 200, respectively. Other than that, the same parameters were used for the 100k input length model.

Let $M = (M_1, M_2, \ldots, M_B)$ be the input of the local attention, where B is the number of blocks. Using L as the length, D as the dimension of the input, the shape of M is represented as $L \times D$ and M_i as the $B \times L_B \times D$. ($L_B = L/B$) Using H as the number of heads, the local attention is defined by the following equations.

$$LocalAttention(M) = Concat(f(M_1), f(M_2), \ldots, f(M_B)) \cdot W^O$$

$$f(M_i) = Concat(head_{i_1}, head_{i_2}, \ldots, head_{i_H})$$

$$head_{i_j} = softmax\left(\frac{(M_i W_j^Q) \cdot (M_i W_j^K)^{\mathrm{T}} + S_{ij}^{rel}}{\sqrt{D_K}}\right) \cdot (M_i W_j^V)$$

$$S_{ij}^{rel} = (M_i W_j^Q) \cdot R^{\mathrm{T}}$$

This is almost the same as the equations in Multi-Head Self-Attention. $W^O \in \mathbb{R}^{(L_B \times D) \times D_O}, W_j^Q, W_j^K, W_j^V \in \mathbb{R}^{D \times D_K}$ are the matrices for the linear projection. D_O is the output dimension and D_K is the output dimension of each head ($D_K = D/H$). S_{ij}^{rel} represents the relative positional embeddings interacted with queries. $R \in \mathbb{R}^{L_B \times L_B \times D_K}$ is a matrix of relative positional encodings. When use R_{kl} as the element of matrix R, R_{kl} is defined as E_{l-k}, where $E = (E_1, E_2, \ldots, E_{2L_B-1}) \in \mathbb{R}^{2L_B-1 \times D_K}$. Here, E_{l-k} represents the relative positional encoding according to the distance from query k to query l. For the effective computation of relative positional encoding, we used the skew algorithm proposed in [24] in code implementation.

Training Procedure

We trained four variations of the SSNet model depending on the training dataset. All models were initially trained on the Gencode dataset for 400 epochs. Variations were as follows:

- SSNet_pangolin: Trained on the Pangolin dataset for an additional 50 epochs.
- SSNet_gtex: Trained on the GTEx dataset for 50 epochs.
- SSNet_gtex_pangolin: Trained on the GTEx dataset followed by the Pangolin dataset, each for 50 epochs.
- SSNet_pangolin_gtex: Trained on the Pangolin dataset followed by the GTEx dataset, each for 50 epochs.

We trained the model 5 times for each variation, randomly dividing the training datasets into 90% training dataset and 10% validation dataset. The validation data was used for hyperparameter optimization and early stopping, which was to stop the training when it converges. We used the Tensorflow library and Adam as the optimization method. The training was performed using Tesla V100, NVIDIA A100, and NVIDIA H100 with batch size 16. The learning rate was set to warm up linearly to 0.00005 in the first 80000 steps and then decreased linearly to 0.0 by 800000 steps. Mixed precision, using both 16-bit and 32-bit floating-point types, was used to train the model faster and use less memory. However, for the part that uses softmax as an activation function, we used a 32-bit floating-point type because softmax is a numerically unstable computation.

The source code and the trained models are publicly available at `https://github.com/yuna728/SSNet`.

Loss Function

Cross entropy was used as the loss function. Let y be the correct answer labels represented by categorical variables and p the probability distribution of output. Suppose that the value of the correct answer label y was i, let the i-th probability p_i be denoted by p_t. Cross entropy loss $CE(p_t)$ is defined as follows;

$$CE(p_t) = -\log p_t$$

Our dataset was imbalanced because the number of acceptor/donor sites was much smaller than that of non-splice

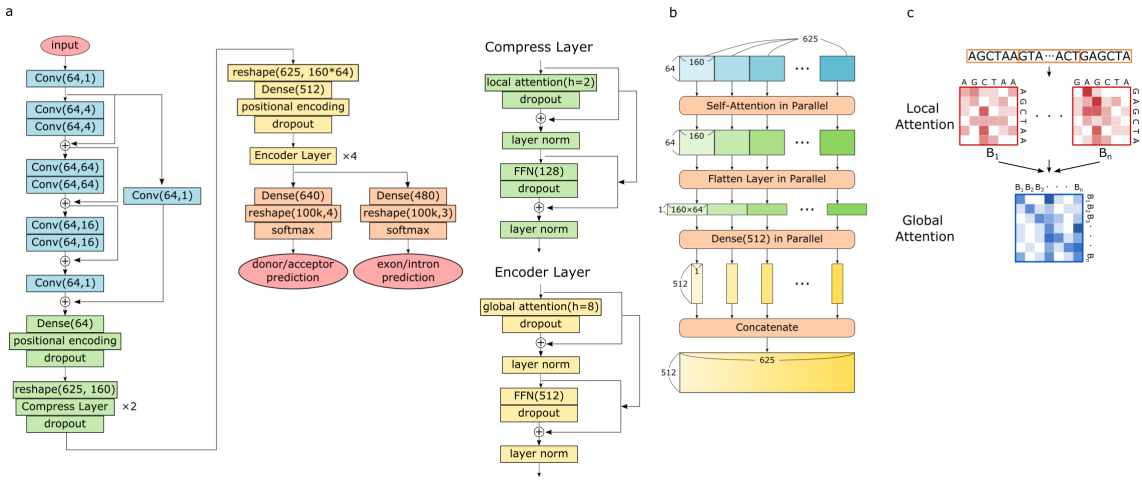


Fig. 1. a. The overall architecture of SSNet. It consists of convolutional layers (blue), local attention layers (green), and global attention layers (yellow). b. The detailed architecture of local attention layers. c. The concept of hierarchical attention mechanism.

sites (donor/acceptor site labels only accounted for 0.033% on the Gencode dataset) and the length of exon was much shorter than that of intron (exon labels accounted for 5.82% on the Gencode dataset). Thus, to improve that imbalanced situation, we combined balanced cross entropy and focal loss [25].

Balanced cross entropy $BCE(p_t)$ was defined in the following equation. By introducing α and setting larger values to fewer labels, the impact of fewer label examples on loss values increases. Suppose that the value of the correct answer label y was i, let the i-th α value α_i be denoted by α_t.

$$BCE(p, q) = -\alpha_t \log p_t$$

Focal loss function $FL(p_t)$ was represented as follows. By introducing $\gamma \geq 0$, the loss value for more difficult examples accounts for a larger proportion of the total loss.

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

When p_t is close to 0, which is an example where the difference between the output and the correct label is large, and the model is not well-trained, $1 - p_t$ is close to 1, so the effect of introducing γ on the loss is small. On the other hand, when p_t is close to 1, this is when the model can produce an output close to the correct label, and in this case, $1 - p_t$ is small, so the loss becomes small as $\gamma \geq 0$.

Combining these two loss functions defined by α and γ, the loss function used in training $BFL(p_t)$ was defined as follows:

$$BFL(p_t) = -\alpha_t (1 - p_t)^{\gamma} \log p_t$$

For the α value, we used 8.0/8.0/3.0/1.0 for donor/acceptor/non-splice/padding labels and 0.7/0.25/0.05 for exon/intron/padding labels in the training on the Gencode Dataset. For additional learning on the GTEx dataset, 6.0/6.0/5.0/3.0 for donor/acceptor/nons-plice/padding labels and 0.7/0.25/0.05 for exon/intron/padding labels were used. For additional learning on the Pangolin dataset, 8.0/8.0/3.0/1.0 for donor/acceptor/non-splice/padding and 0.0 for all exon/intron/padding labels were used. As for the γ value, we used 2.0 in both phases of training because it showed the best result in [25].

Validation Dataset

For model validation, we utilized two datasets:

- BRCA Dataset: This dataset includes variants in the BRCA1 and BRCA2 genes, which are known to be involved in breast cancer. Data were sourced from BRCA Exchange [26].
- Deep Intronic Dataset: This dataset was curated to include variants classified as Pathogenic, Likely Pathogenic, Benign, or Likely Benign in ClinVar [27]. Only intronic variants in protein-coding genes were retained based on Ensembl VEP annotations, excluding exonic variants from other genes. To address the class imbalance, all benign variants with ≥ 2 stars were included and one-star variants were randomly sampled up to a maximum of 50,000. The final dataset comprised 18,446 pathogenic and 49,343 benign variants, categorized based on their distance from splice sites. This dataset was made in [28].

Both datasets contain information about mutations and labels as benign or pathogenic. To evaluate the performance of models, the difference between the model's predictions for reference sequences without a mutation and those for alt sequences with a mutation was taken, and if the maximum difference was greater than a threshold value, the variant was judged pathogenic, otherwise benign. A threshold value of 0.2 was used for SSNet, SpliceAI, and Pangolin.

Results

SSNet outperformed SoTA on the Gencode test dataset.

The performance of SSNet was verified on the Gencode test dataset and compared to SpliceAI and Pangolin in terms of both precision and F1 score. SSNet_base is SSNet trained on the Gencode dataset only. Compared to the two SoTA models, SSNet_base achieves higher precision and thus higher F1 and has fewer false positives (Table 1).

Table 1. The performance comparison on the Gencode test dataset.

Model	Recall	Precision	F1	Topk	auPRC
SpliceAI	**0.9439**	0.8557	0.8976	0.9278	**0.945**
Pangolin	0.9273	0.9030	0.9149	0.9366	0.9447
SSNet_base	0.9368	0.9337	**0.9352**	**0.9400**	0.9429
SSNet_da	0.9205	0.7792	0.8439	0.8778	0.9034
SSNet_a0	0.9197	0.9290	0.9243	0.9291	0.9369
SSNet_g0	0.9300	**0.9345**	0.9322	0.9366	0.9380
SSNet_1k	0.8044	0.9146	0.8559	0.8701	0.8866
SSNet_10k	0.9080	0.9296	0.9186	0.9252	0.9307

Ablation study on the Gencode test dataset.

In the previous section, we have shown that SSNet can learn spliced site features at a level comparable to or better than SoTA. We conducted an ablation study to investigate how SSNet's learning process contributes to this high performance.

First, we compared the SSNet training without intron/exon labels (SSNet_da). In this case, there was a significant performance drop in all metrics, with the largest drop in Precision. When training a language model as large as 100 kb, acceptor/donor labels alone are not sufficient information, and the addition of intron/exon labels may have improved performance by allowing learning of sequence features within the exon. In addition, the decrease in Precision was much larger than the decrease in Recall. This indicates that the intron/exon labels are helping to reduce false positives, while the acceptor/donor labels can learn the features of highly conserved motifs such as the GT-AG rule, but cannot learn the differences between spliced sites and non-spliced GT, AG dinucleotides. The addition of intron/exon labels would facilitate learning the context of whether or not a site is used as a splice site.

Next, we compared the performance of alpha and gamma, which are innovations in the loss function, with that of the case in which they were not used. Except for precision in SSNet_g0, which did not use gamma, SSNet_base performed the best in all indices, indicating that the weighting in the loss function contributes to the model performance. The difference between SSNet_a0 and SSNet_base is larger, indicating that weighting that resolves label imbalance is important for training data sets with very large label bias, such as spliced sites.

Then, a comparison was made for different input lengths. When the input length was set to 1k and 10k, the performance was worse for all indicators compared to when the input length was 100k. The performance improvement was more pronounced when the input length increased from 1k to 10k compared to 10k to 100k. This may be because more introns are fully included in the input with the increase from 1k to 10k.

SSNet showed superior auPRC on BRCA dataset.

We evaluated the performance of aberrant splicing prediction on the BRCA dataset. First of all, we made predictions only for variants labeled by experts (4909 positive, 2504 negative). Since the F1 score and accuracy are greatly affected by the threshold settings, we mainly look at auROC and auPRC to compare the performance. Figure 2 shows that the proposed model outperforms SpliceAI and Pangolin by a wide margin. In particular, SSNet_pangolin consistently showed the best performance. Second, we included predictions that were "Not Yet Reviewed" in the expert labeling but were labeled in other databases (7959 positive, 6566 negative). In this case, as before, SSNet outperforms the two SoTA models in auROC and auPRC. In this case, SSNet_pangolin continued to perform the best. Comparing SpliceAI and Pangolin, which have the same architecture, Pangolin shows higher performance in both cases, and SSNet_pangolin shows the best performance when compared within SSNet, followed by SSNet_gtex_pangolin. The last model trained on the pangolin dataset, which takes into account the strength of the splice, tends to perform better than the other models that were not trained on the pangolin dataset at the end of the training.

We also compared the distribution of predictions between SSNet_pangolin and SpliceAI, which performed best among the SSNet variants, plotting only those variants present in the BRCA1 gene that were labeled pathogenic in Figure 3. The x-axis represents position, the y-axis represents predictions in SSNet_pangolin, and the color represents benign/pathogenic predictions made by SpliceAI. The blue plots show variants predicted as benign and the orange plots show one as pathogenic. Focusing on exon10, the largest exon in the BRCA1 gene, we see that it is correctly predicted as pathogenic by SSNet_pangolin with a predicted value of 0.2 or higher, while in SpliceAI variants predicted as benign (predicted values are 0.2 or lower) are concentrated in this exon (in the red circle of Figure 3). A similar trend was observed when comparing the predictions of SSNet_pangolin and Pangolin. It is known that exon10 is a region where many variants related to breast cancer pathogenicity are clustered, and it is also known to be a place where complex splicing regulation takes place in relatively long exons. Therefore, we examined in detail several pathogenic variants in exon10, which is above 0.2 in SSNet_pangolin and below 0.2 in SpliceAI, as described earlier.

We found that a common cryptic acceptor site was activated in many of the variants. Figure 4 shows the change in SSNet predictions in the chr17:g.41244217:G>A variant. The cryptic acceptor site in exon10 is a very weak splice site in Ref Seq but changes to a relatively strong splice site in Alt Seq. To examine the change in attention of this variant, we visualized the Local Attention at Block 209, where the variant and cryptic acceptor site are located, and at Block 208 before it (Figure 5 a,b). The global attention between the surrounding Blocks is also shown (Figure 5 c). We also found that specific sites upstream of the variants were activated in the attention map, even though the position of the variants varied. In this variant, we see that the location about 160 bases upstream from the variant is strongly noted in Block 208. The downstream region of the variant also shows a large change in attention, while the cryptic acceptor site itself shows a relatively small change in Block 209. In addition, focusing on global attention, the effect is noticeably stronger from Block 208, where a strongly activated region was present, to Block 209, where a cryptic acceptor site is present. This suggests that the activation region of Block 208 is involved in some kind of splicing regulation, which activates

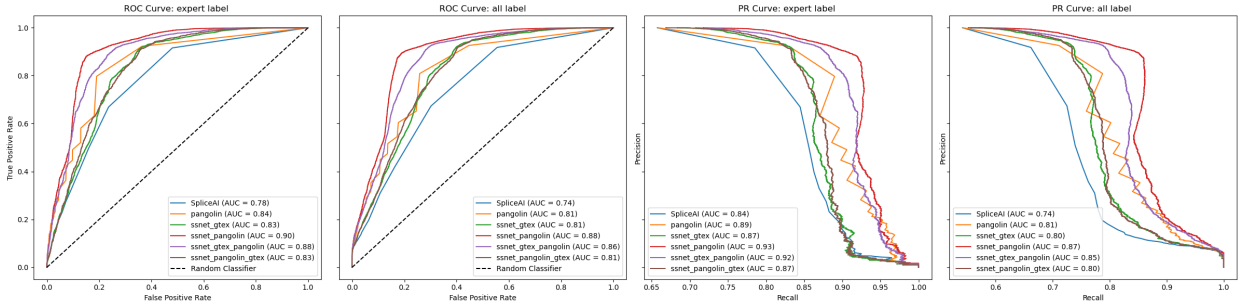


Fig. 2. ROC curves and PR curves of SpliceAI, Pangolin, and SSNet for expert label and all label.

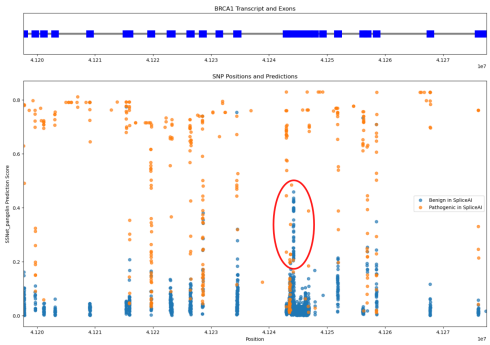


Fig. 3. Prediction value of SSNet and SpliceAI of pathogenic variants located in BRCA1 gene. The area circled in red is the cluster of mutations in exon10 that have a prediction greater than 0.2 in SSNet but less than 0.2 in SpliceAI.

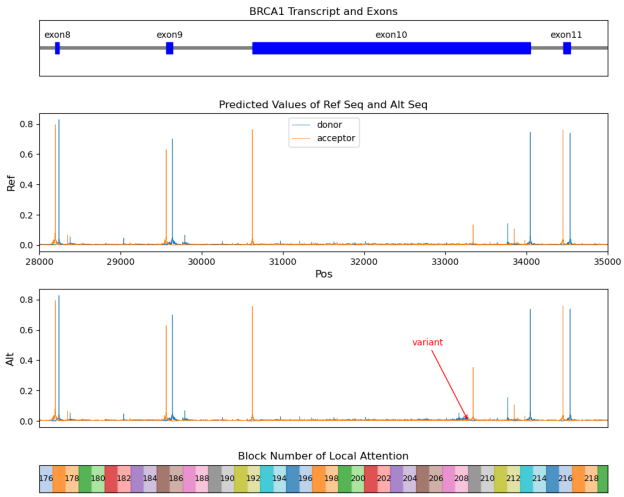


Fig. 4. The probability of splice sites of variant chr17:g.41244217:G>A. The top row shows the BRCA1 gene structure information, the second row shows the SSNet predictions for the Ref Seq, the third row shows the predictions for the Alt Seq, and the forth row shows the corresponding block number of local attention. The x-axis shows the relative distance from the start of the BRCA1 transcript. The orange line shows the probability of being used as an acceptor site and the blue line shows that of the donor site. The position of the variant is indicated by the red arrow, and the cryptic acceptor site downstream of it (in Block 209) is active in the Alt Seq.

the cryptic acceptor site in the downstream region as a splice site. Although no splicing regulators have been reported for these regions in previous studies to the best of our knowledge, it is possible that the exonic splicing enhancer is activated by the variant or that the exonic splicing silencer is destroyed and the cryptic acceptor site downstream of the variant is activated.

SSNet can predict better for deep intronic mutation.

It is more difficult to predict the impact of a mutation when the mutation is located far from the splice site than when the mutation is located near the splice site. Here, we focused on the prediction of aberrant splicing in the deep intronic dataset, especially in the case of deep intronic regions. Figure 6 presents the auPRC values for each variant distance from the nearest splice site. SSNet consistently shows comparable performance to SpliceAI and Pangolin. In particular, SSNet achieves an auPRC up to 4 times higher than SpliceAI and Pangolin when the variants are at distances of 41-200 bp, where the auPRC of SpliceAI and Pangolin is less than 0.2. Thus, SSNet can accurately capture long-range effects that are not captured by conventional models due to the global attention mechanism.

Discussion

We introduced SpliceSelectNet, the first hierarchical Transformer model capable of handling long-range dependencies for splice site prediction.

The success of SSNet's in three benchmark datasets can be attributed to several innovations. The first one is the usage of intron/exon label in the first step of training. In large DNA language models, having the model perform the subtasks associated with donor/acceptor site predictions simultaneously increases the generality of the model and reduces overfitting. It also made it possible to capture sequence features within the exon, and the model was able to learn the context in which the splice site was present. This improved the model's ability to distinguish between a real splice site and just a GT, AG dinucleotide, and reduced false positives. The second one is loss function design. The use of balanced cross-entropy and focal loss addressed the class imbalance issue inherent in splicing datasets. Since we often face the class imbalance problem in the field of bioinformatics, this technique can be applied to other biological prediction tasks. The third one is data augmentation. The incorporation of GTEx and Pangolin datasets enhanced the model's ability to predict both constitutive and alternative splice sites. The fourth one is longer input lengths. By extending the input length to 100 kb, the model effectively captured long-range dependencies. An ablation study showed that each of these factors contributed significantly to the

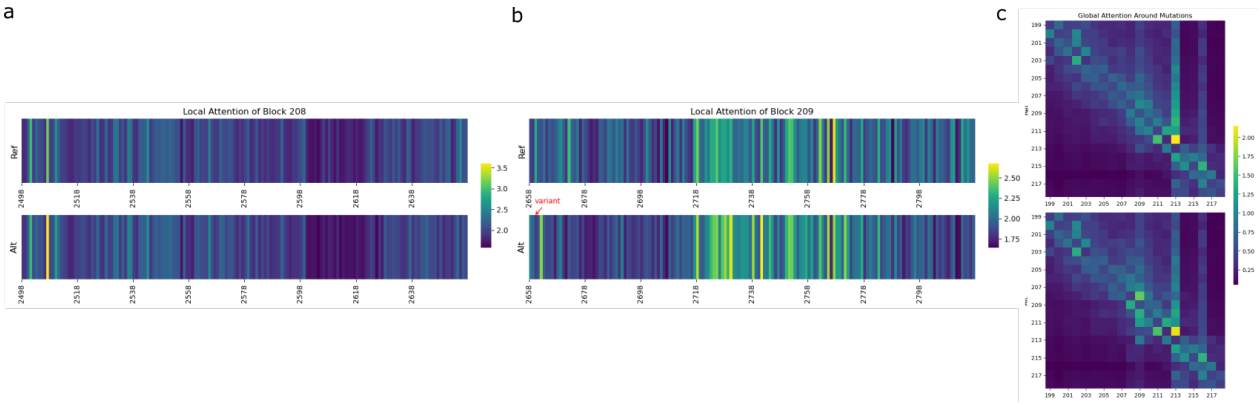


Fig. 5. The heatmap of attention weights of variant chr17:g.41244217:G>A. The upper row is the heatmap for Ref Seq and the lower row is the heatmap for Alt Seq. a.The local attention of block 208. The x-axis shows the relative position from exon10, and this figure represents the value of local attention from the 2498th nucleotide to the 2657th. There is a strongly activated area upstream. b. The local attention of block 209. The x-axis shows the relative position from exon10, and this figure represents the value of local attention from the 2658th nucleotide to the 2817th. The variant exists in the 2661th nucleotide. A large change in attention weights is observed downstream from the variant. c. The global attention around the mutation. It can be seen that the effect from Block 208 to Block 209 becomes stronger by mutation.

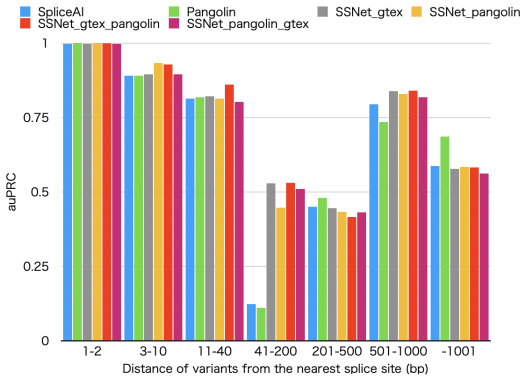


Fig. 6. The comparison of auPRC values between SpliceAI, Pangolin, and SSNet. The x-axis is categorized based on the distance from the nearest canonical splice site of each variant.

overall performance, highlighting their importance in model development.

SSNet's hierarchical attention mechanism proved essential for handling long-range dependence, an important aspect of splicing control. Comparisons with SpliceAI and Pangolin revealed that SSNet's dense attention allows more context information and enables better performance for mutations involved in complex splicing control mechanisms and for long-distance mutations. SSNet is also, to our knowledge, the first attempt in bioinformatics to use a hierarchical attention mechanism. This structure solves the limitation to the input length, which is a problem when dealing with DNA sequences while keeping the computational complexity low. Although the number of parameters is much larger than in SpliceAI or Pangolin, which only consist of convolutional layers, the introduction of the hierarchical attention mechanism allows inference at almost the same speed.

Furthermore, the interpretability of SSNet by attention maps paves the way for understanding splicing mechanisms. For example, in BRCA1 exon10, attention weighting revealed a potential regulatory region upstream of the cryptic splice site, providing new insight into splicing activity by mutation. Since the model uses dense attention weights, visualization and interpretation of the attention are intuitive, making the model easy to understand for experts in the fields of biology and medicine. The advantage is that the attention weights are the result of intermediate calculations during inference, so there is no need to perform additional calculations for interpretation. In addition to understanding splicing mechanisms, attention maps can also be leveraged to explore sequence motifs that are important for splicing regulation. By identifying which regions of the sequence receive higher attention weights, we can uncover previously unknown motifs or enhance the understanding of known splicing regulatory elements. For variants of uncertain significance, attention allows us to identify how the mutation affects splicing regulation, contributing to the elucidation of disease mechanisms.

The architecture of SSNet is versatile and could be extended to other genomic tasks, such as the prediction of transcription factor binding sites, chromatin accessibility, and epigenomic modifications. In particular, since it can take input sequences as long as 100k, it is expected to perform well in prediction tasks where long-range interactions are important. One promising future direction is to enhance the prediction of alternative splicing events in a tissue-specific manner. By integrating tissue-specific RNA-seq datasets and splicing regulatory element annotations, SSNet could identify key elements that drive differential splicing across tissues. This could provide novel insights into the regulatory mechanisms underlying splicing and uncover potential therapeutic targets for diseases associated with aberrant splicing. Another future plan is to develop a general-purpose DNA language model using this architecture. A DNA language model with high interpretability and practical computational speed could be applied to a variety of downstream tasks.

In this paper, our model outperforms the current SoTA in both splice site prediction and aberrant splicing prediction. With its interpretable attention mechanism, SpliceSelectNet is poised to be a valuable tool for researchers studying gene regulation and disease-associated splicing variants. Moreover, its architecture can be generalized to other DNA sequence tasks, opening the door to broader applications as a versatile DNA language model.

Competing interests

No competing interest is declared.

Author contributions statement

Y.M. conducted the majority of the research, including conceptualization, methodology development, data analysis, and manuscript drafting. K.N. supervised the project, provided critical guidance throughout the research process, and contributed to the manuscript's revision and finalization. Both authors reviewed and approved the final version of the manuscript.

Acknowledgments

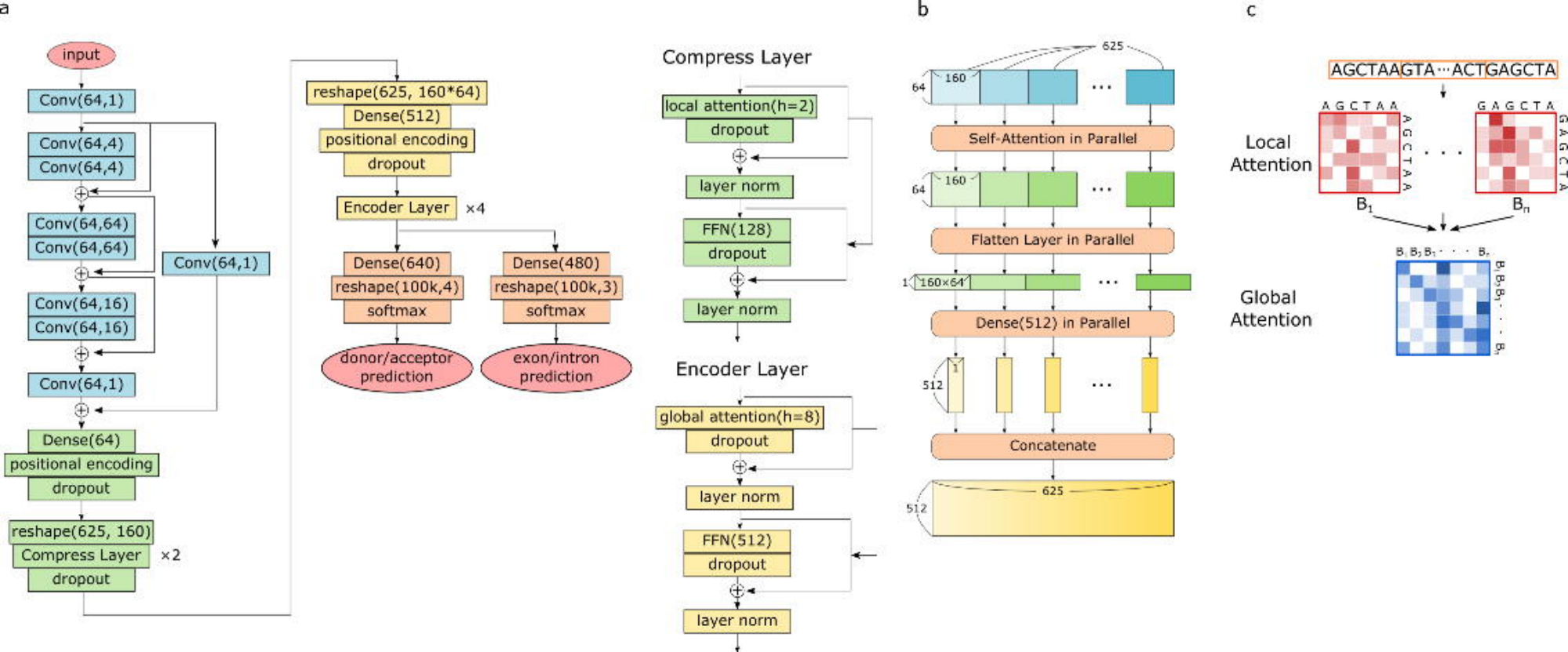
This work was supported by JST SPRING, Grant Number JPMJSP2108. We also acknowledge the Human Genome Center, the Institute of Medical Science, the University of Tokyo, for providing the super-computing resource (`http://sc.hgc.jp/shirokane.html`).

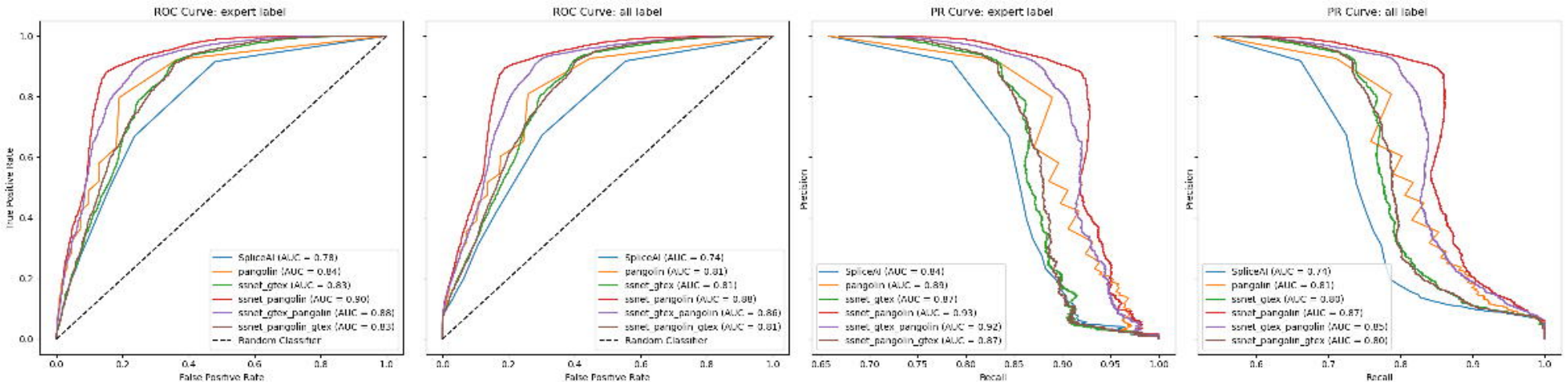
References

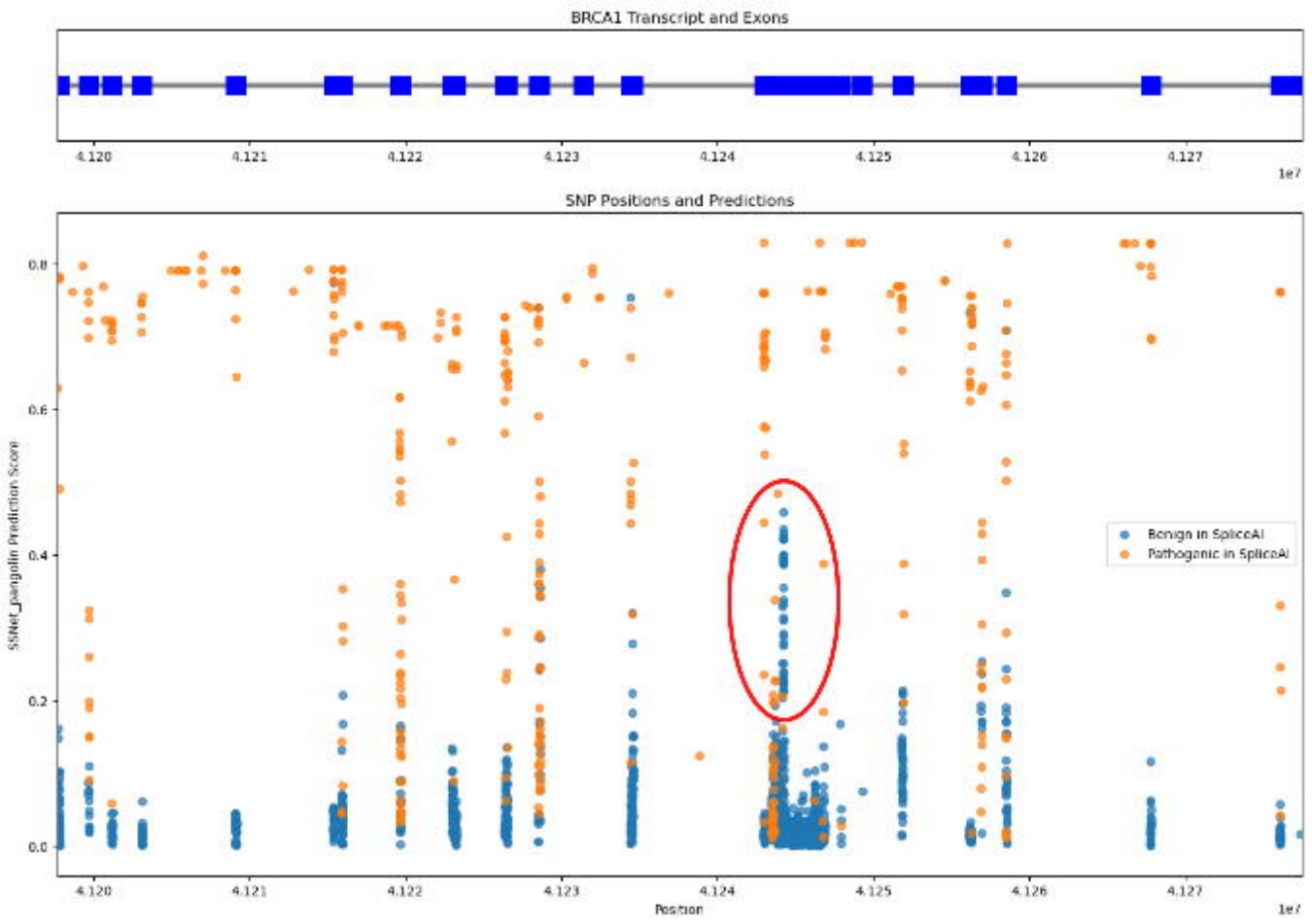
1. Julian P Venables. Aberrant and alternative splicing in cancer. *Cancer research*, 64(21):7647–7654, 2004.
2. Dunhui Li, Craig Stewart McIntosh, Frank Louis Mastaglia, Steve Donald Wilton, and May Thandar Aung-Htut. Neurodegenerative diseases: a hotbed for splicing defects and the potential therapies. *Translational neurodegeneration*, 10(1):16, 2021.
3. Abramowicz Anna and Gos Monika. Splicing mutations in human genetic disorders: examples, detection, and confirmation. *Journal of applied genetics*, 59:253–268, 2018.
4. Quan Yang, Jinyao Zhao, Wenjing Zhang, Dan Chen, and Yang Wang. Aberrant alternative splicing in breast cancer. *Journal of molecular cell biology*, 11(10):920–929, 2019.
5. Sebahattin Cirak, Virginia Arechavala-Gomeza, Michela Guglieri, Lucy Feng, Silvia Torelli, Karen Anthony, Stephen Abbs, Maria Elena Garralda, John Bourke, Dominic J Wells, et al. Exon skipping and dystrophin restoration in patients with duchenne muscular dystrophy after systemic phosphorodiamidate morpholino oligomer treatment: an open-label, phase 2, dose-escalation study. *The Lancet*, 378(9791):595–605, 2011.
6. Ryszard Kole and Arthur M Krieg. Exon skipping therapy for duchenne muscular dystrophy. *Advanced drug delivery reviews*, 87:104–107, 2015.
7. Kenta Nakai and Hiroshi Sakamoto. Construction of a novel database containing aberrant splicing mutations of mammalian genes. *Gene*, 141(2):171–177, 1994.
8. Gene Yeo and Christopher B Burge. Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals. *Journal of computational biology*, 11(2-3):377–394, 2004.
9. Kishore Jaganathan, Sofia Kyriazopoulou Panagiotopoulou, Jeremy F McRae, Siavash Fazel Darbandi, David Knowles, Yang I Li, Jack A Kosmicki, Juan Arbelaez, Wenwu Cui, Grace B Schwartz, et al. Predicting splicing from primary sequence with deep learning. *Cell*, 176(3):535–548, 2019.
10. Tony Zeng and Yang I Li. Predicting rna splicing from dna sequence using pangolin. *Genome biology*, 23(1):1–18, 2022.
11. Yanrong Ji, Zhihan Zhou, Han Liu, and Ramana V Davuluri. DNABERT: pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome. *Bioinformatics*, 37(15):2112–2120, 2021.
12. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
13. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.
14. Donna Karolchik, Galt P Barber, Jonathan Casper, Hiram Clawson, Melissa S Cline, Mark Diekhans, Timothy R Dreszer, Pauline A Fujita, Luvina Guruvadoo, Maximilian Haeussler, et al. The UCSC genome browser database: 2014 update. *Nucleic acids research*, 42(D1):D764–D770, 2014.
15. Jennifer Harrow, Adam Frankish, Jose M Gonzalez, Electra Tapanari, Mark Diekhans, Felix Kokocinski, Bronwen L Aken, Daniel Barrell, Amonida Zadissa, Stephen Searle, et al. GENCODE: the reference human genome annotation for The ENCODE Project. *Genome research*, 22(9):1760–1774, 2012.
16. John Lonsdale, Jeffrey Thomas, Mike Salvatore, Rebecca Phillips, Edmund Lo, Saboor Shad, Richard Hasz, Gary Walters, Fernando Garcia, Nancy Young, et al. The genotype-tissue expression (GTEx) project. *Nature genetics*, 45(6):580–585, 2013.
17. GTEx Consortium, Kristin G Ardlie, David S Deluca, Ayellet V Segrè, Timothy J Sullivan, Taylor R Young, Ellen T Gelfand, Casandra A Trowbridge, Julian B Maller, Taru Tukiainen, et al. The genotype-tissue expression (gtex) pilot analysis: multitissue gene regulation in humans. *Science*, 348(6235):648–660, 2015.
18. Margarida Cardoso-Moreira, Jean Halbert, Delphine Valloton, Britta Velten, Chunyan Chen, Yi Shao, Angélica Liechti, Kelly Ascenção, Coralie Rummel, Svetlana Ovchinnikova, et al. Gene expression across mammalian organ development. *Nature*, 571(7766):505–509, 2019.
19. Alexander Dobin, Carrie A Davis, Felix Schlesinger, Jorg Drenkow, Chris Zaleski, Sonali Jha, Philippe Batut, Mark Chaisson, and Thomas R Gingeras. Star: ultrafast universal rna-seq aligner. *Bioinformatics*, 29(1):15–21, 2013.
20. André Kahles, Jonas Behr, and Gunnar Rätsch. Mmr: a tool for read multi-mapper resolution. *Bioinformatics*, 32(5):770–772, 2016.
21. Craig I Dent, Shilpi Singh, Sourav Mukherjee, Shikhar Mishra, Rucha D Sarwade, Nawar Shamaya, Kok Ping Loo, Paul Harrison, Sridevi Sureshkumar, David Powell, et al. Quantifying splice-site usage: a simple yet powerful approach to analyze splicing. *NAR Genomics and Bioinformatics*, 3(2):lqab041, 2021.
22. Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning*, pages 5156–5165. PMLR, 2020.
23. Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. *arXiv*

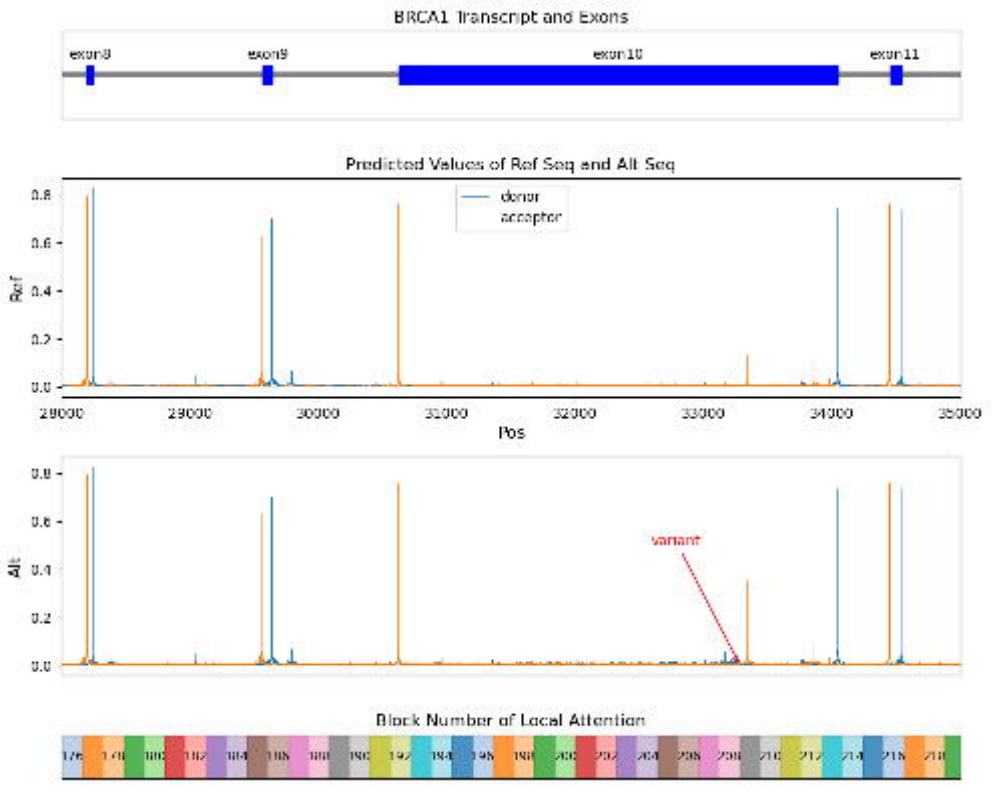
preprint arXiv:1803.02155, 2018.

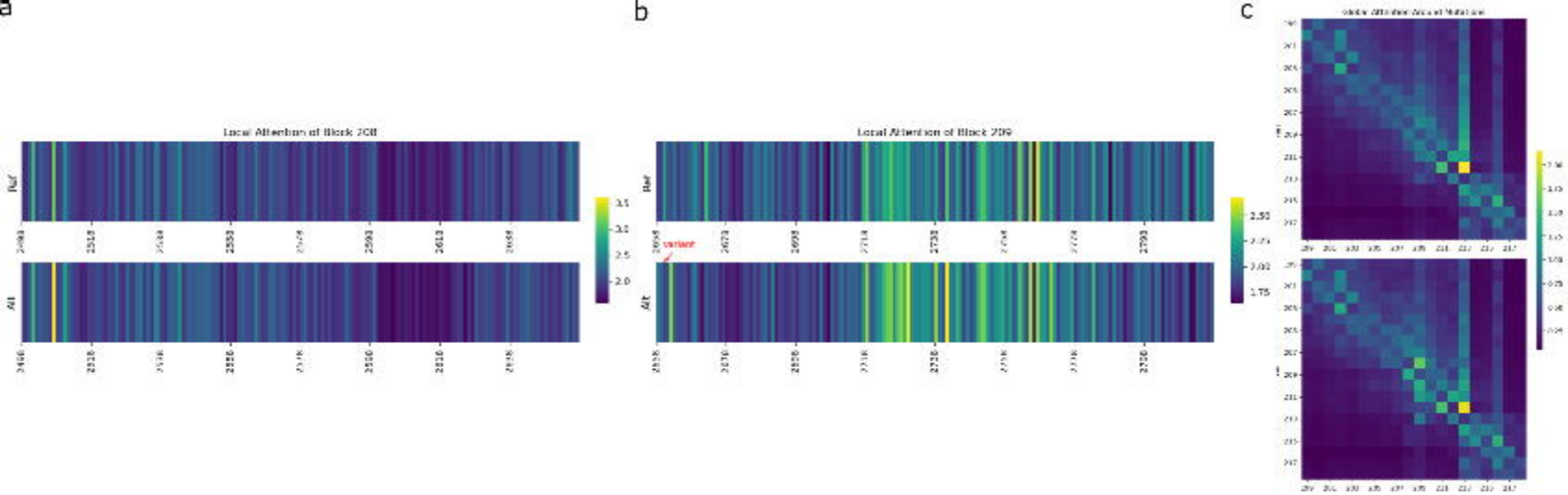
24. Cheng-Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Noam Shazeer, Ian Simon, Curtis Hawthorne, Andrew M Dai, Matthew D Hoffman, Monica Dinculescu, and Douglas Eck. Music transformer. *arXiv preprint arXiv:1809.04281*, 2018.
25. Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.
26. Melissa S Cline, Rachel G Liao, Michael T Parsons, Benedict Paten, Faisal Alquaddoomi, Antonis Antoniou, Samantha Baxter, Larry Brody, Robert Cook-Deegan, Amy Coffin, et al. Brca challenge: Brca exchange as a global resource for variants in brca1 and brca2. *PLoS genetics*, 14(12):e1007752, 2018.
27. Melissa J Landrum, Jennifer M Lee, Mark Benson, Garth R Brown, Chen Chao, Shanmuga Chitipiralla, Baoshan Gu, Jennifer Hart, Douglas Hoffman, Wonhee Jang, et al. Clinvar: improving access to variant interpretations and supporting evidence. *Nucleic acids research*, 46(D1):D1062–D1067, 2018.
28. Pedro Barbosa, Rosina Savisaar, Maria Carmo-Fonseca, and Alcides Fonseca. Computational prediction of human deep intronic variation. *Gigascience*, 12:giad085, 2023.

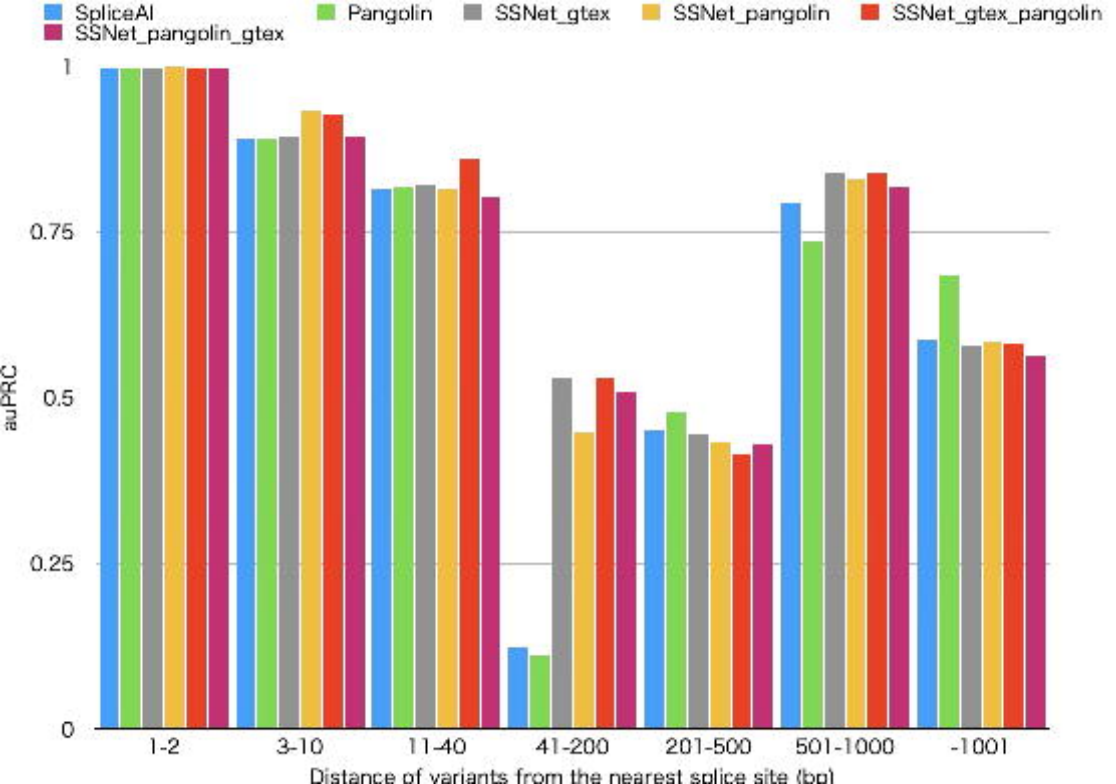

a
input
Conv(64,1)
Conv(64,4)
Conv(64,4)
Conv(64,64)
Conv(64,64)
Conv(64,16)
Conv(64,16)
Conv(64,1)
Conv(64,1)
Dense(64)
positional encoding
dropout
reshape(625, 160)
Compress Layer
×2
dropout
reshape(625, 160*64)
Dense(512)
positional encoding
dropout
Encoder Layer
×4
Dense(640)
reshape(100k,4)
softmax
Dense(480)
reshape(100k,3)
softmax
donor/acceptor prediction
exon/intron prediction
Compress Layer
local attention(h=2)
dropout
layer norm
FFN(128)
dropout
layer norm
Encoder Layer
global attention(h=8)
dropout
layer norm
FFN(512)
dropout
layer norm
b
625
64
160
Self-Attention in Parallel
64
160
Flatten Layer in Parallel
1
160×64
Dense(512) in Parallel
512
1
Concatenate
512
625
c
AGCTAAGTA···ACTGAGCTA
Local Attention
A G C T A A
G A G C T A
B1
Bn
Global Attention


ROC Curve: expert label
True Positive Rate
False Positive Rate
SpliceAI (AUC = 0.78)
pangolin (AUC = 0.84)
ssnet_gtex (AUC = 0.83)
ssnet_pangolin (AUC = 0.90)
ssnet_gtex_pangolin (AUC = 0.88)
ssnet_pangolin_gtex (AUC = 0.83)
Random Classifier
ROC Curve: all label
False Positive Rate
SpliceAI (AUC = 0.74)
pangolin (AUC = 0.81)
ssnet_gtex (AUC = 0.81)
ssnet_pangolin (AUC = 0.88)
ssnet_gtex_pangolin (AUC = 0.86)
ssnet_pangolin_gtex (AUC = 0.81)
Random Classifier
PR Curve: expert label
Precision
Recall
SpliceAI (AUC = 0.84)
pangolin (AUC = 0.89)
ssnet_gtex (AUC = 0.87)
ssnet_pangolin (AUC = 0.93)
ssnet_gtex_pangolin (AUC = 0.92)
ssnet_pangolin_gtex (AUC = 0.87)
PR Curve: all label
Precision
Recall
SpliceAI (AUC = 0.79)
pangolin (AUC = 0.81)
ssnet_gtex (AUC = 0.80)
ssnet_pangolin (AUC = 0.87)
ssnet_gtex_pangolin (AUC = 0.85)
ssnet_pangolin_gtex (AUC = 0.80)


BRCA1 Transcript and Exons
4.120
4.121
4.122
4.123
4.124
4.125
4.126
4.127
1e7
SNP Positions and Predictions
SSNet_pangolin Prediction Score
0.0
0.2
0.4
0.6
0.8
Benign in SpliceAI
Pathogenic in SpliceAI
Position


BRCA1 Transcript and Exons
exon8
exon9
exon10
exon11
Predicted Values of Ref Seq and Alt Seq
donor
acceptor
Ref
Alt
Pos
28000
29000
30000
31000
32000
33000
34000
35000
0.0
0.2
0.4
0.6
0.8
variant
Block Number of Local Attention


a
Local Attention of Block 208
Ref
Alt
b
Local Attention of Block 209
Ref
Alt
variant
c


SpliceAI
Pangolin
SSNet_gtex
SSNet_pangolin
SSNet_gtex_pangolin
SSNet_pangolin_gtex
1
0.75
0.5
0.25
0
auPRC
1-2
3-10
11-40
41-200
201-500
501-1000
-1001
Distance of variants from the nearest splice site (bp)