Universiteit Leiden
The Netherlands

Opleiding Informatica

Using DLNNs to predict nucleosome position based on sequence and methylation data

Iwan Steenhoff & Wessel ten Hoeve

Supervisors:
F.J. Verbeek & S.J.T van Noort

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)
www.liacs.leidenuniv.nl

08/07/2025

Abstract

Nucleosome positioning is highly important for chromatin formation, epigenetics and DNA accessibility. Various studies have shown the importance of sequence on nucleosome positioning. This study developed the MS_RBL model, which is a DLNN that uses methylation input along with DNA sequence. The model is trained and evaluated on the *Saccharomyces cerevisiae* yeast genome and uses chemical cleavage data as a regression target. The study reports a Pearson's correlation coefficient of ~ 0.73 and an F1 score of ~ 0.57. The 10 bp periodicity known to influence nucleosome positioning has also been found in the model's prediction. Furthermore, the combination of methylation and sequence as input has been shown to outperform both inputs individually. Improvements to the model are yet to be made through hyperparameter tuning and possibly implementing transformer structures.

Contents

1 Introduction

In eukaryotic cell nuclei, DNA is wrapped around histone octamers to form structures called nucleosomes. Each nucleosome contains 147 base pairs (bp) and is bound together by non-nucleosomal DNA called linker DNA in order to stabilize the nucleosome structure [Lug97]. One of the main functions of this structure is to compactly store DNA inside the nucleus, as the combined length of DNA fibers can be meters long [RPR+02]. Furthermore, nucleosomes play an important role in gene regulation. Nucleosomal DNA is bound to histones and is therefore not able to bind to other proteins. This includes proteins that play a role in DNA transcription. Thus, genes located on nucleosomes are effectively turned off. Therefore, knowing the locations of all nucleosomes on any given DNA sequence can give us insights into how those specific genes are regulated. This information could then be used in the development of treatments for certain diseases, as abnormal positioning of nucleosomes can lead to, for example, developmental issues or cancer [JP09b].

Methylation can be used as an indicator for nucleosome position through methylation-seq. A method which uses a methylation enzyme, m6A-methyltransferase, to methylate all available adenines in a given DNA sample. The enzyme can only methylate an adenine nucleotide when it is not bound to a different protein. This results in a string of high and low methylated areas where low methylation indicates the presence of proteins that are bound to the DNA, for example, nucleosomes [Bij25].
A deep learning neural network (DLNN) model will be created that can predict the position of nucleosomes using DNA sequence and additional nucleotide methylation information. DLNNs have been used to automatically extract relevant features from DNA sequences in the past [GSH22]. These studies usually only train their model using DNA sequence information. However, incorporating additional information could lead to an increase in prediction accuracy. Therefore, in this study, methylation information for each nucleotide in the DNA sequence is also included to train the model.

1.1 Research Question

The main research question for this study is as follows: How well does a DLNN predict nucleosome positions based on DNA sequence and methylation data? This question will be answered by building a DLNN that takes sequence and methylation information as input and outputs a dyad probability signal.

1.2 Thesis overview

This section contains the introduction of this thesis; Section 2 introduces and explains relevant biological background concepts; Section 3 discusses previous research regarding nucleosome models; Section 4 gives an overview of the used methods; Section 5 reports on the results of the used methods; Section 6 will discuss the results and finally conclude. This bachelor thesis is supervised by LIACS and, in particular, by F.J. Verbeek.

1.3 Work Division

During this thesis we have been closely working together on most of the content. We have collaborated on developing the model, the data processing, the implementation and most of the performance validation. Moreover at the end of the research we diverged to different subtopics. Iwan Steenhoff has individually worked on the subsections 5.4, 5.5 and 5.7. Wessel ten Hoeve performed the additional research for subsections 5.3 and 5.6. The rest of the thesis has been performed and written in a joined effort.

2 Background

The nucleosome serves as the basic structural component in the organization of DNA into chromatin fibers. Nucleosomes consist of a histone octamer and 147 base pairs (bp) of double-stranded DNA wrapped around it. The histone octamer contains two copies each of the core histones H2A, H2B, H3 and H4. Nucleosomes form on a DNA strand at intervals, with the regions of DNA between them referred to as linker DNA. The nucleosomes interact with each other in a process called stacking, in which higher-order structures are formed. However, the particular structures that arise and their stability have yet to be fully resolved. The interactions between the nucleosomes are largely facilitated by positively charged histone tails that bridge adjacent nucleosomes by neutralizing negative charges of DNA and histone surfaces [YL11]. The length of linker DNA also influences nucleosome stacking, as it determines the relative orientation of nucleosomes. Certain spacing between nucleosomes has been found to provide more thermodynamically stable chromatin structures and decrease nucleosome mobility [LC25]. The length of the linker DNA is, of course, determined by the positioning of nucleosomes along the DNA.

2.1 Nucleosome behavior

Nucleosome positioning refers to the positions of nucleosomes with respect to the DNA sequence. Nucleosome positioning maps are generally created by sequencing small reads cut by specific enzymes and inferring the position from those reads. However, nucleosomes show dynamic behavior and so there are no absolute positions for nucleosomes along the DNA sequence. So nucleosome positioning is generally referred to as averaged over time and cells. The position of a nucleosome with respect to the DNA sequence is indicated by the dyad position. This is the nucleotide exactly in the middle of the 147 bp strand that is wrapped around the histone octamer. The first and last nucleotides wrapped around the histone octamer are not reliable indicators of nucleosome positioning due to nucleosome breathing.
Li et al. detected this behavior in vitro with molecule spectroscopy [LW04]. Nucleosome breathing is one of the dynamic properties of nucleosomes that make it difficult to accurately map nucleosomes. It refers to the DNA strand partially unwrapping from the histone octamer. The unwrapping can occur on both sides of the nucleosome and at the same time. The degree of unwrapping can vary but is typically around 15-25 bp. This partial unwrapping of the DNA is affected by interactions with the histone core and tail [NK25]. However, the exact dynamics that allow for nucleosome breathing are not well understood yet. When nucleosomes are breathing, transcription factors also get access to the DNA that is currently unwrapped. This makes the behavior very relevant for better understanding of gene regulation.

Nucleosomes also exhibit a behavior called repositioning. Nucleosome repositioning happens both spontaneously and through nucleosome remodeling enzymes. A study by Rudnizky et al. found that nucleosomes undergo spontaneous confined diffusion along the DNA strand [RKM+19]. This diffusion happens in small increments of 1-2 bp. These small diffusion are speculated to be the result of twist defects in the helical turn of the DNA. The authors also hypothesize that this behavior is part of a gene regulation mechanism. Nucleosomes are also found to be moved by a protein in the SWI/SNF family of remodelers, specifically RSC, which uses energy from ATP hydrolysis for

repositioning of the nucleosomes [HHD+16]. The RSC remodeler moves the DNA wrapped around the nucleosomes without displacing the nucleosomes entirely. The translocation of this DNA was found to be in a step-size manner where the distributions also exhibit a peak at approximately 1-2 bp.

So despite the dynamic behavior of nucleosomes, positioning maps of nucleosomes in a genome are still created. In certain regions nucleosome positioning is much more stable, and in other regions much more dynamic. Nucleosome positioning maps account for this by using metrics such as occupancy to refer to nucleosome positioning. Occupancy indicates the probability of a certain DNA region being wrapped into a histone octamer.

2.2 Nucleosome positioning dependence on sequence

As mentioned before, nucleosome positioning along the DNA is important for the organization of nucleosomes into higher-order structures. Various factors contribute to the positioning of nucleosomes and their relative importance is still under debate. The main contributors are found to be the DNA sequence, nucleosome remodeling enzymes and transcription factors. Those contributors also influence each other, leading to significant differences in nucleosome positioning between genes and chromosomes [SS13]. This study will focus mainly on the influence of sequence on nucleosome positioning.

Several studies have shown the importance of sequence for nucleosome positioning. Firstly, it has been shown that nucleosomes have a high preference for G + C base pairs. Tillo and Hughes found that G + C presence explained about 50% of the variation in nucleosome occupancy in vitro [DT13]. A more recent study by Trotta shows that G + C content could predict more than 60% of the nucleosome map in vitro [Tro22]. The study was performed on *Saccharomyces cerevisiae*, which is the same species of yeast used in this study. High G + C content likely correlates with nucleosome positioning because it exhibits structural properties that favor the formation and stability of nucleosomes. More specifically, high G + C content is attributed to increased bendability, which thus makes it easier for DNA to wrap around nucleosomes.
There is also a link between nucleosome positioning and the presence of 10 bp periodic patterns of AA, TA and TT dinucleotides running counterphase to GC dinucleotides [TNM08]. This 10 bp periodicity can be explained by the double helix structure of DNA. The A & T dinucleotides have a higher affinity for binding and so they occur in higher quantities when the minor groove is facing the nucleosome. The minor groove faces the nucleosome in a periodic pattern of about 10 bp. So, the minor groove also faces outward from the nucleosome in a 10 bp periodicity, which is where CG dinucleotides are more prevalent.
Long poly(dA:dT) tracts (sequences of exclusively adenine and thymine) are also shown to be nucleosome disfavoring. A study by Field et al. found nucleosome depletion along these Poly(dA:dT) tracts and their flanking regions [FKFM+08]. This is also correlated to the depletion of C + G nucleotides in these regions [Tro22]. Poly(dA:dT) tracts are much more rigid than DNA with G and C nucleotides and so do not bend easily around nucleosomes. Field et al. also suggested that these tracts act as nucleosome positioning signals to regulate gene expression, as nucleosome-depleted regions allow transcription factors to bind to the DNA.
So in vitro nucleosome positioning is highly dependent on sequence. In vivo, different proteins such

as remodeling enzymes and transcription factors also play a large role [SS13]. However, there is a much higher correlation between nucleosome positioning and sequence than originally thought. This is also due to the behavior of remodeling enzymes and transcription factors being partially explained by sequence.

2.3 Neural networks

As mentioned in Section 1, this study will use a DLNN to predict nucleosome positions from sequence and methylation data. DLNNs are characterized by their many hidden non-linear layers. This allows the DLNN to learn features hierarchically with respect to its layers [IWK17]. Meaning, the early layers will learn basic local features and the deeper layers will learn more abstract global features. This has made DLNNs successful in a wide range of applications, including natural language processing, which is very similar to nucleosome positioning from sequence data.
The DLNN used in this study combines several different techniques within neural computing. Namely, convolutional feature extraction, residual learning and a bidirectional LSTM (BiLSTM) layer are all used in the MS_RBL model.

Convolutional layers are designed to preserve spatial relations in data. In contrast to fully connected layers, each neuron is only connected to a subset of neurons in the previous layer. These subsets are called kernels and their associated weights are learned through backpropagation. The kernels are tuned to recognize local features and in later layers, more abstract and global features (due to spatial downsampling and pooling layers).
Convolutional layers are often used in image classification, where spatial information is highly relevant. In the methylation and sequence data, spatial relations are also highly relevant. The kernels could be tuned to recognize the 10 bp periodicity between dense A & T dinucleotide regions along with the counterphase dense C & G dinucleotide regions. In contrast to images, the sequence and methylation data are one-dimensional and so the kernels will also be one-dimensional.
The residual blocks are also a core component in the DLNN. They allow for the deep nature of DLNNs through the residual connections between the residual blocks. The residual connections combat the vanishing gradient problem by making the gradient flow backwards more easily during backpropagation [BK23]. Instead of learning the true full transformation from input to output, the residual blocks learn the residual function. So during backpropagation, the full chain of derivatives does not have to be multiplied with each other, preventing the gradients from vanishing. This makes deep architectures possible without the first few layers becoming obsolete.
LSTMs are a type of RNN that are designed to remember longer connections without suffering from the vanishing or exploding gradient problems. The LSTM combats this by introducing the memory cell, which controls the flow of information through different gates [RK20]. The gates allow the network to remember important relations over longer distances and discard irrelevant relations.
BiLSTMs are simply a combination of two LSTMs where one processes the data forward and one processes the data backward. This is useful for data where both the context before and after the timestamp matters. For example, in sequence data, both the bases before and after a certain base are significant context.

3 Related Work

3.1 Statistical model on sequence data

Many studies have highlighted the importance of sequence for nucleosome positioning. A study by van der Heijden et al. created a model that predicted single nucleosome positioning and genome-wide nucleosome occupancy on sequence only [vdHvVLvN12]. With single nucleosome positioning, the study refers to predicting the dyads on short sequences where the dyads are well defined. The model relied on just three parameters: the probability amplitude (B), dinucleotide periodicity (p) and the length of the periodic probability function (N). The probability amplitude refers to the sequence preference for specific dinucleotides. This was chosen following the research that linked nucleosome positioning to higher G + C content, as mentioned in 2.1. The dinucleotide periodicity was chosen to convey the 10 bp periodicity of increased A & T dinucleotide content in the minor groove facing the nucleosome along with the counter-phase increased G & C content in the minor groove facing away. This periodicity originates from the double helical structure of DNA, as mentioned in 2.1. Finally, the length of the periodic probability function is used to convey the size of the nucleosome in bp to assess the likelihood of nucleosome binding.
Then, by applying Percus's equation, the model was extended to predict nucleosome occupancy genome-wide. The Percus equation was used to model the thermodynamic equilibrium density of nucleosomes along the DNA. With that, the stability of the higher-order structures of nucleosomes was captured to predict occupancy on a genome-wide scale.
The study shows that the DNA sequence can be used to predict nucleosome positioning and therefore, in the DLNN of this thesis, the genome from *Saccharomyces cerevisiae* will also be used as part of the input. The DLNN will hopefully capture the same dependencies found by Van der Heijden et al. and use them to increase the predictive power.

3.2 Machine learning models on sequence data

Earlier studies have also used deep learning to infer nucleosome positioning from DNA sequence. A study by Han et al. used three different deep learning techniques that achieved accuracy scores above 80%. The models were tested on three different species, one of those being H. sapiens [GSH22]. Instead of one-hot encoding the DNA sequence, the study used DNA sequence embedding based on word2vec. This means the model will represent the sequence data as vector embeddings called k-mers (with k being the length of the word). These vector embeddings are created by first collecting all unique k-mers in the dataset. Those are trained on the sequences in the dataset to create contextual embeddings represented as vectors. This allows the model to learn the hidden association information in the sequence.
The DNA sequences represented as word vectors were used to train a CNN, a BiGRU + BiLSTM and a hybrid model. The CNN model was based on a TextCNN with three different-sized filters. Only it uses max pooling with stride 2 instead of global max pooling to further extract salient features.
The BiGRU + BiLSTM model was used to extract relations between word vectors at the ends of nucleosomes. As these models are able to capture relations much further apart compared to the CNN.

The hybrid model combined both the CNN and the BiGRU + BiLSTM models. This model could then capture both close and distant relations in the sequence data. Also, combined models have been shown to perform better at feature extraction. The hybrid model performed best out of the three models.
These models were trained on 404,565 sequences of 147 bp that were either labeled as a nucleosome or as linker DNA. The models were then used to predict nucleosome occupancy on a genome-wide scale by sliding a window of 147 bp along the sequences. A downside to this approach is that the model is not able to learn relations between nucleosomes. It cannot capture the importance of chromatin formation for nucleosome positioning. Another potential downside is the input size of 147 bp, while linker DNA is usually only between 20 and 60 bp. This could cause the model to misinterpret the difference in sizes between nucleosomes and linker DNA.

A study by Amato et al. also used DLNNs to classify nucleosome positioning [ABR20]. Similarly to the study by Han et al., the authors treated the problem as a binary classification task where the model trained on sequences of 147 bp that are labeled either as nucleosome or linker DNA. For this study, however, the sequence data was one-hot encoded instead of using embedded word vectors. This means each nucleotide was represented as a 4-dimensional vector where the position of the "1" indicates the type of nucleotide.
The study used a CNN in combination with a recurrent LSTM termed the CORENup model. The model consists of an input convolutional layer and then two processing paths: a convolutional and a recurrent. The combination of these paths is used to create the output. This model was tested against the formerly best-performing model on nucleosome positioning with one-hot encoded sequence data. The LeNup model was produced by Zhang et al. [ZPW18]. The CORENup model was found to perform just as well or better compared to the LeNup model with faster training time. The study treated nucleosome positioning as a binary classification task. The metrics used were accuracy, recall, precision and MCC. A downside to treating nucleosome positioning as a binary problem is that it ignores the dynamic nature of nucleosomes. The model cannot differentiate between regions where nucleosome positions are well defined and poorly defined.

3.3 Models on methylation data

In a study by Kelly et al., a method (NOMe-seq) was built that could map high-resolution nucleosome footprints genome-wide using GpC-methyltransferase [KLL+12]. GpC-methyltransferase is an enzyme that methylates cytosine in GpC dinucleotides, but only when the DNA is accessible (so not wrapped in nucleosomes or tightly bound to proteins). The NOMe-seq method retained the endogenous methylation information on the DNA strand. These are the CpG methylation sites that occur naturally in the cell. So the NOMe-seq method can map both nucleosome positions and retain naturally occurring methylation on a single strand. This allows researchers to track chromatin and methylation patterns at the same time, which can be used to monitor disease progression.
For this paper it is important to note that the NOMe-seq method allows researchers to create nucleosome positioning maps using artificial methylation. The authors compared the NOMe-seq method to MNase-seq, another high-resolution experimental technique for mapping nucleosomes, and found their performance to be similar. However, the methods were only compared in CTCF regions and transcription start sites (TSS). The paper also does not mention the effect of nucleosome

breathing on their methylation footprints.

A study by Martin Bijl used the methylation sequencing data for *Saccharomyces cerevisiae* used in this study in relation to nucleosomes [Bij25]. The study aimed to identify nucleosome breathing in the methylation sequencing data. Instead of a DLNN, the study used a Duration Hidden Markov model (dHMM). The model was tuned to find nucleosome positions by having the transition matrix be of size 148 by 148 (one non-nucleosomal state and 147 nucleosomal states). The model uses an emission matrix that contains the chances of finding a methylation inside a nucleosome and outside a nucleosome. Then the Viterbi algorithm calculates the most likely sequence of hidden states. From this, the dHMM could predict nucleosome positions from the methylation data.
The predictions from the dHMM are used along with a chemical cleavage dataset from Chereji et al. to infer nucleosome breathing. This is possible due to the methylation dataset showing methylations on DNA that is unwrapped from the histone octamer. So smaller footprints in this dataset indicate nucleosomal breathing.

4 Methods

4.1 Approach

In order to answer the research question presented in this study, the MS_RBL (Methylation Sequence Residual BiLSTM) model was created. It is a hybrid DLNN that combines multi-scale convolutional feature extraction, residual learning and bidirectional LSTM layers. The DLNN is trained on sequence and methylation data to predict nucleosome positioning.
The methylation data is obtained from a study done by Leiden University, which is yet to be published [KL24]. The study used a method that uses m6A-methyltransferase, which adds methylations to accessible adenines. If the enzyme is blocked by a protein such as a nucleosome, the adenines attached to it will not be methylated. The result is a series of long DNA strands methylated at accessible parts. A similar method was used in a study by Stergachis et all [SDH+20]
The sequence data is obtained from the sacCer3 strain [Nat23]. Lastly, the performance of the DLNN will be compared to a dataset containing the likelihood of each nucleotide being a dyad, obtained through chemical cleavage [CRBH18]. This dataset was obtained by a chemical cleavage method that releases single-nucleosome dyad-containing fragments, which allowed the authors to map both single nucleosomes and linker DNA with high precision. Their experiment was repeated three times, yielding three different dyad positioning maps: y1, y2 and y3 on the sacCer3 genome. The final model was trained on the average of these three experiments.

Model performance is first evaluated on the different augmentations of the chemical cleavage data, as it contains many outliers, which hurt model performance. The logarithmic transformation of the normalized data is selected as most ideal. From here a 5-fold cross-validation experiment was done to validate the reliability of the model's performance. Then the model was tested on the holdout test set along with the three single-experiment data: y1, y2 and y3. The model is evaluated on the metrics MSE, R^2, Pearson, Spearman, F1 score, precision, and recall. For replication purposes, all training and validation was done using random seed 42. The chemical cleavage data is smoothed with $\sigma = 2$ and normalized. For the final training of the model a logarithmic transformation of the chemical cleavage data will also be applied to the chemical cleavage data. The methylation data is also normalized.
After this general performance evaluation, more specific aspects of the model's performance are analyzed. Firstly, the effect of window size on model performance was explored further. Earlier studies using a DLNN have mostly treated nucleosome positioning as a binary problem, so the influence of window size for regression is largely unexplored. Window sizes from 25 bp up to 15000 bp were compared by training the model for 10 epochs. The stride was kept the same as the window size, $s = w$.
Another important biological feature that is absent from binary nucleosome positioning data is the 10 bp dinucleotide periodicity. So, the 10 bp dinucleotide periodicity in the models predictions is substantiated by analyzing the distance between peaks. A function is made that selects main peaks in the data above a threshold of 0.25. For each main peak, all flanking peaks within a 41 bp flanking region are selected. Flanking peaks whose distances to the main peak are congruent to zero modulo 10, within a 1 bp tolerance, are classified as phased flanks. The percentage of phased flanks is used to infer 10 bp periodicity. For comparison, the same method is applied to the target chemical cleavage data.

To further validate the performance of the model it was compared to a different dataset obtained through a meta-analysis of six studies [JP09a]. This data is binary; therefore, to accurately compare it to the model it was made into a continuous target. Each binary peak was transformed into a normal distribution with the fuzziness and occupancy features reported for that peak. This continuous data was compared to the raw and smoothed $\sigma = 20$ signal of the model on chromosome XVI.
It was also tested whether the 10 bp periodicity of the model aligns with the positions of the binary peaks in the Jiang data. By finding the closest peak predicted by the model to each binary peak, the degree of correlation in 10 bp periodicity could be quantified. Again, the same method was applied on the chemical cleavage data.

Most literature on nucleosome positioning with DLNNs only uses sequence data. So, the individual influence of the sequence and the methylation data was studied by training two extra models. One model receives only sequence data as input and one model receives only methylation data as input. For both models, 5-fold cross-validation was performed and the results were compared to the model receiving both inputs.

Lastly, a new model was trained on individual methylation reads. This is a first step towards a model predicting nucleosome positions for specific DNA strands. The chemical cleavage data is still used as a target for the model however.
For training and validation, roughly 500 reads were collected in every chromosome with at least a length of 7000 bp. The reads were spaced out so that the largest part of every chromosome is covered by them. As done previously, the single-read model is tested on the two chromosomes XV and XVI in the test set.
Validating the predictions of the single-read model for an individual read using chemical cleavage data is not entirely appropriate. Therefore, another comparison was made between the average predicted signal over 50 reads and the chemical cleavage data on chromosomes XV and XVI.

4.2 Data processing

In order to make a DLNN learn nucleosome positioning from sequence and methylation data, the data first needs to be processed. The data consists of three parts: the methylation and sequence data that make up the input data and the dyad positions that make up the output data. The full data processing pipeline is given in Figure 1. The pipeline in the green container shows the pipeline for one chromosome. So this process is repeated for all 16 chromosomes in the *Saccharomyces cerevisiae* genome. The green boxes show the processing of the methylation data, the purple boxes show the sequence data and the red boxes show the truth values. The yellow box shows the joined methylation and sequence data that form the input data. The white boxes show the joined input and output data.
Each chromosome is cut into windows of window size w with stride s. The DLNN will be trained on these windows. An entire chromosome would be too large to feed into a DLNN and would result in very few training samples.

The methylation data consists of many short reads that are sequenced to show the methylation profile. Figure 2 shows the methylation profile for 430 reads on chromosome II. The reference genome is displayed on the bottom, with several gal genes shown in blue. It can be seen that the transcription start sites are heavily methylated on many reads. This indicates that nucleosomes are not present in these regions. Looking closely, periodic white patches can be observed, which are explained by the presence of nucleosomes.
For every nucleotide, the methylation profiles of all reads that contain the nucleotide need to be averaged. This was done by using the *pacbio processing* package [WP]. This package provided the *get_modification_mean()* function, which averaged the methylation values for every nucleotide. So, each nucleotide will then be accompanied by the probability of it being methylated. The averaged methylation values are put into a vector, which is divided into smaller windows.

The sequence data is obtained from the sacCer3 strain in FASTA format [Nat23]. The sequence is cut into small windows for each chromosome and is then one-hot encoded. Meaning each nucleotide becomes a four-dimensional vector where the "1" indicates the type of nucleotide.
Then a fifth dimension is added, which is the averaged methylation value for that nucleotide, as shown in Figure 3.

As mentioned in Section 4 the truth values are obtained through a chemical cleavage study [CRBH18]. The study produced dyad profiles for three experiments, which are averaged into one dyad signal. These target values are combined with the input values in the data loaders. An example of the models input and target can be seen in Figure 3.

To eventually validate the models performance on the Jiang data it had to be altered to a continuous target. The continuous data is created by flattening the single dyad positions with a normal distribution. To better reflect the uncertainty in the data, the occupancy, peak height and fuzziness values from the dyad positions are used to form the normal distribution. The peak height and occupancy are used to give the height of the normal distribution in two different datasets. They are a representation of the strength of a dyad signal for a given position.

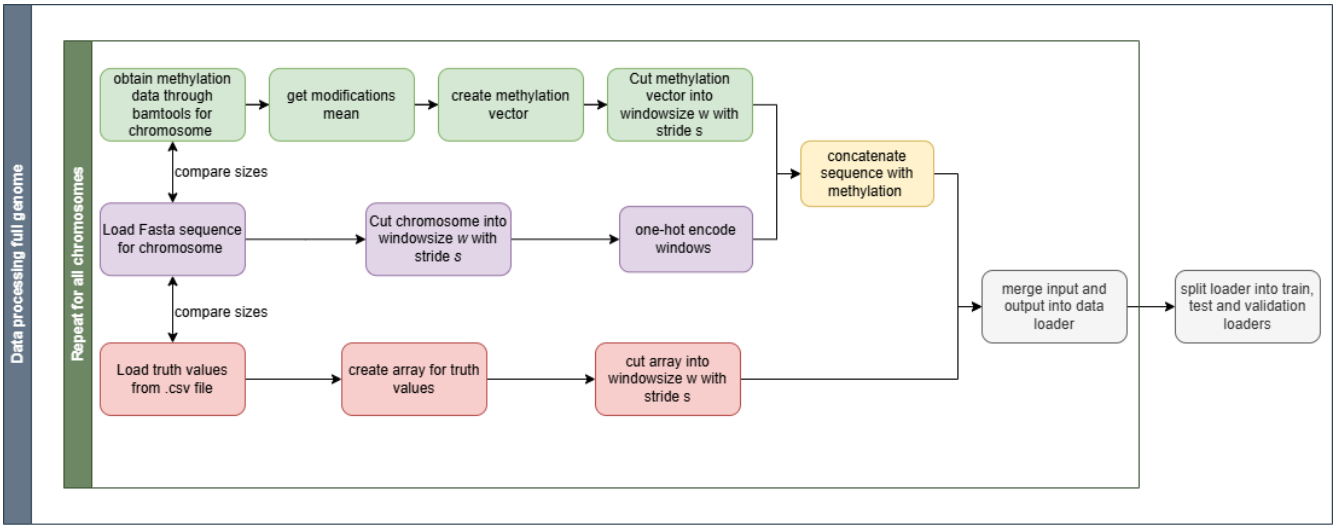


Figure 1: The data processing before feeding it into the DLNN. The pipeline in the green container is repeated for every chromosome in the genome. It shows how the vectorized methylation data is merged with the one-hot encoded sequence data to form the input. The input is loaded alongside the truth values in the data loader. All chromosomes are divided into smaller windows in order for the DLNN to handle the input.

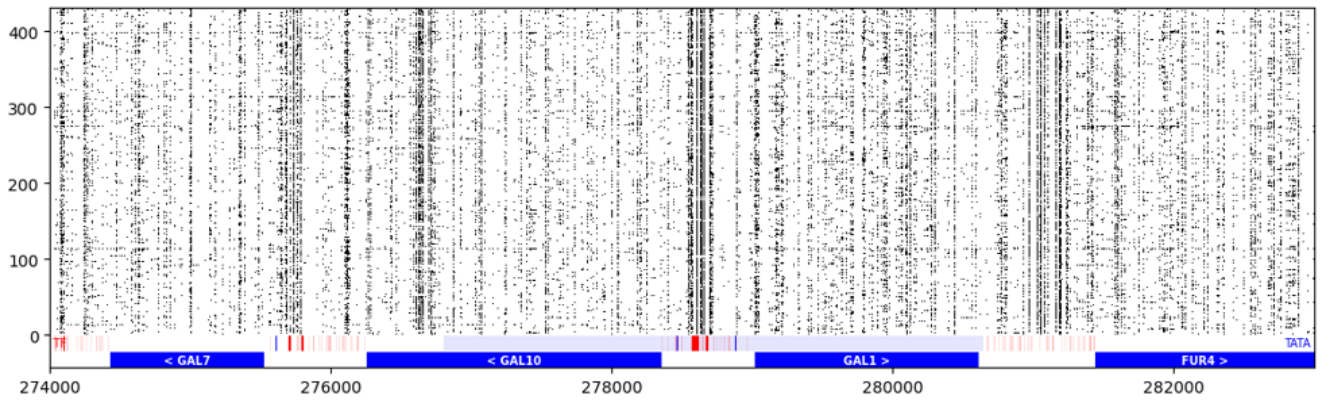


Figure 2: The graph shows the methylation profiles for 430 reads from position 274000 until 283000 on chromosome II. On the bottom, the reference genome is shown with several genes displayed by blue boxes. The graph clearly shows high methylation levels on transcription sites, which can be explained by the absence of nucleosomes. The graph also shows periodic regions with few methylations, which are the nucleosome footprints.

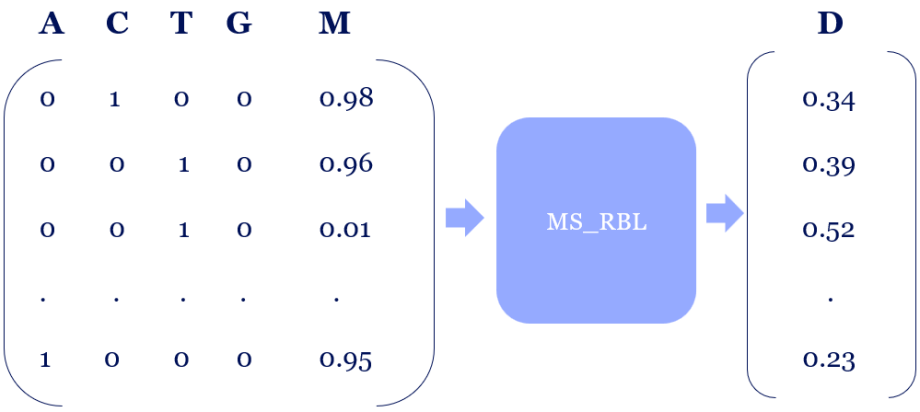


Figure 3: Example of the input and output dimensions of the DLNN. The length of the input and output is equal to the window size w. The input consists of 4 dimensions to encode the sequence and one dimension for the averaged methylation. The output is a one-dimensional binary representation of the dyad positions.

4.3 The model

The model developed in this study was created using Python 3.12.3 and the machine learning library PyTorch 2.7.0. The architecture of the model starts off with a single convolutional stem layer that increases the channel dimension of the input data before forwarding it to the deeper layers. This is done in order for the next residual layers to receive a larger feature space, allowing the network to capture more complex features. The main use of the residual layers is to extract hierarchical features from the input data. Which in this case is genomic data that has a sequential structure containing different motifs, such as the 10 bp dinucleotide periodicity linked to nucleosome positioning [TNM08]. Residual layers have the added benefit over standard convolutional layers of skip connections that allow connections to bypass one or more intermediate layers. As a result, the risk of vanishing or exploding gradients is decreased, as well as reducing the saturation and degradation of the model accuracy, therefore increasing the learning capabilities of deeper neural networks [HZRS16].
The model used for this study implements residual blocks. In these blocks, multiple convolutional layers with different kernel sizes are applied in parallel to capture features and motifs at different scales. The different layers are concatenated and then normalized using batch normalization in order to stabilize training and improve the generalization ability of the model [IS15]. Next, the channel dimension is reduced back to its original size using a 1x1 projection convolution. This is followed by a skip connection that adds the input of the residual block to the projected output, allowing for residual learning. The output of the residual blocks is then finalized using the ReLU activation function to set negative input values to 0.

The residual layers are great at capturing local features and motifs at different scales. However, residual and convolutional architectures might miss additional long-range dependencies and patterns when the sequential data is long. In this study, large genomic data is used and therefore BiLSTM layers are incorporated in the DLNN. LSTM layers are recurrent layers that are able to remember important information over long ranges and can thus capture long-range dependencies. Instead of using normal LSTMs, BiLSTMs can be leveraged to analyze the sequence in both ways instead of just a single one. Nucleosome positioning is influenced by both upstream and downstream

context. Thus, processing information from both ways can improve model performance [SP97].

After the BiLSTM layers, a dropout layer is added that sets input values to 0 according to a fixed probability. This prevents the model from relying too heavily on specific learned features when making the final predictions. As a result, the model becomes more robust and can generalize better, which reduces overfitting [SHK+14].
Finally, the output of the model is produced by a fully connected layer that takes the output of the preceding dropout layer and maps a single value to each position of the input sequence. This results in a single nucleosome signal prediction for each nucleotide position of the DNA sequence/methylation value input.

The specific settings used for each layer of the model used in this study are as follows. The convolutional stem has 5 input channels, uses a kernel size of 5, a padding size of 2 and outputs 32 channels. This is followed by 5 sequential residual blocks; each block contains parallel residual layers with kernel sizes k 5, 9, 17, 29, 45 and 65. The padding size is equal to $k/2$. Then the BiLSTM layers receive an input of size 32 and have a hidden size of 64, meaning the output contains 128 neurons. Furthermore, there are 2 sequential BiLSTM layers. They are followed by the dropout layer with $p = 0.5$. Finally, the fully connected layer receives an input of size 128 and outputs a single value.
During training, the MSE loss function was used along with the Adam optimizer and a learning rate of $1e^{-3}$. Moreover, a batch size of 16 was used and the training was run for 10-15 epochs.

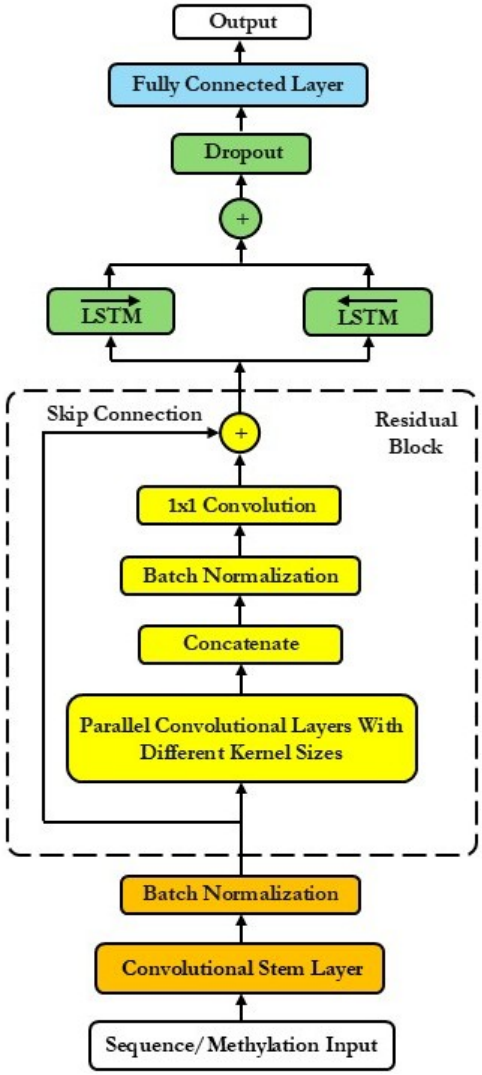


Figure 4: Proposed DLNN model architecture for nucleosome positioning prediction. Sequence and methylation information is used as input for the model. This is inputted to a convolutional stem layer, after which batch normalization is applied. Next, a residual block containing parallel convolutional layers with different kernel sizes is concatenated, after which batch normalization is applied again. A 1x1 convolution is applied, after which the original residual block input can be added to the residual block output through a skip connection. n residual blocks can be used sequentially, after which k sequential BiLSTM layers are applied. This is followed by a dropout layer, which gives the final output of the model in the form of a nucleosome occupancy value for each base pair position in the input.

5 Results

5.1 Performance evaluation metrics

To accurately measure the performance of the model, the metrics used for evaluation need to be carefully chosen. The metrics must be easily interpretable and hold weight biologically. Different metrics explain different aspects of the performance of the model.

First, let's consider the metrics that can be used to train the model. As the chemical cleavage data is continuous, the nucleosome positioning problem can be treated as a regression problem. Standard loss functions for regression are the mean squared error (MSE) and the mean absolute error (MAE). MSE computes the loss as the square of the error and MAE computes the loss directly from the error. Choosing between these two loss functions depends on which output is wanted from the model. MSE penalizes larger errors more heavily in proportion to smaller errors, which forces the model to predict the height of the peaks more accurately. This is useful when one wants the model to predict the dyad positions with clear differences in certainty attached. MAE is more useful when one just wants the approximate positions of the nucleosomes with less emphasis on the certainty of those positions.
MSE is also more susceptible to outliers, which the chemical cleavage data contains many of, as those will be heavily penalized. However, in Section 5.2 a logarithmic transformation will be applied to the data to reduce the effect of outliers. This resulted in very similar performance between the two metrics and so the more common MSE metric was chosen.

MSE can also be used as an evaluation metric. However, this metric alone cannot validate the model's performance, as its score is linked to the scale of the data. Therefore, the metrics coefficient of determination (R^2), Pearson correlation coefficient (r) and Spearman's rank correlation (ρ) will also be given. These metrics are dimensionless and provide a direct comparison with a dumb baseline.
The R^2 score gives a normalized measure of how well the variance in the data is explained by the predicted values. This is calculated by dividing the residual sum of squares by the total sum of squares. An R^2 value of one means perfect predictions and zero means as good as predicting the mean. Pearson measures the strength of the linear relationship between the true and predicted values. Where one is a perfect linear relationship and zero is a no linear relationship. Spearman measures the monotonic relation between the true and predicted values. So, whether high truth values tend to correspond with higher predicted values irrespective of the relationship being linear. In relation to nucleosome positioning, all three metrics care about the positions of the peaks and valleys. However, R^2 also checks whether the absolute height of the peaks is similar. Pearson also checks whether the relative height between the peaks is similar in true and predicted values. Spearman only checks whether the peaks are in the same positions between true and predicted values.

The metrics described above all work on the entire predicted signal from the model. It would furthermore be interesting to purely look at the positions of the predicted peaks without taking the amplitude of the peaks into account. To get the true and predicted peaks, a peak-finding function that identifies peaks higher than a certain threshold height is used. Each position of the predicted signal can be classified as peak or no peak using this function. This information can then be used to

calculate TP, FP, FN and TN. To identify the peaks, an error margin was also used when comparing the true and predicted peaks. This tolerance margin was set at 1.0, which aligns with the error in the chemical cleavage data, meaning predicted peaks 1 bp left or right from the true peaks are also seen as correct. With this function, the model can be evaluated on the binary metrics: precision, recall and F1 score. With precision being the exactness of the model when predicting true values, recall being the proportion of true values that the model predicts successfully and the F1 score being the harmonic mean of precision and recall as can be seen in equations 1. The value of the used peak threshold during evaluation is of great importance. A threshold that is too low will result in misclassifications of noise, reducing precision while increasing recall. Although setting the threshold too high will result in missed true peaks during evaluation and thus a reduction in recall while precision increases. Therefore, finding a threshold value that can find a balance between precision and recall is necessary to yield the maximal F1 score. In Figure 5, this is shown. Threshold values between 0.07 and 0.11 seem to find this balance and achieve the highest F1 scores. A threshold of 0.1 maximizes F1 and is the value used for the evaluation of the model.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \qquad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \qquad F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{1}$$

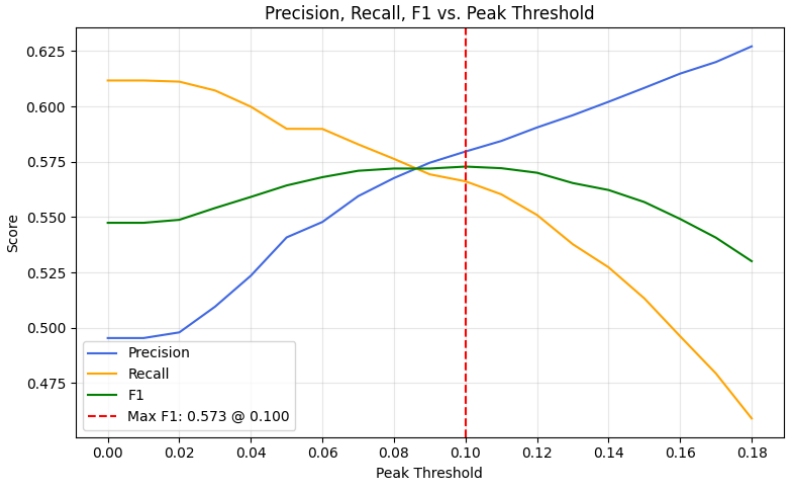


Figure 5: Precision, recall and F1 scores plotted for different peak threshold values used in the peak-finding function. To find the threshold that maximizes the F1 score, there needs to be a balance between precision and recall. A threshold too low results in noise being classified as peaks, thus increasing recall while decreasing precision. A threshold too high results in too few peaks being predicted and therefore true peaks being missed. This leads to a decrease in recall and an increase in precision. A threshold of 0.1 yields the highest F1 score as it balances precision and recall.

5.2 General model performance

Before fully evaluating the model, it needed to be determined whether changing the scale of the chemical cleavage data would improve performance. The chemical cleavage data contains many outliers, which hurt the model's performance. Applying a logarithmic transformation or clipping the top 2% are common solutions for outliers. The results of the model's performance can be seen in Table 1. The table shows fairly similar performance between the clipped data and the logarithmic transformation of the data, with the logarithmic transformation of the data achieving a better R^2.

The raw data shows only similar performance on the F1 score. The larger MSE is due to the data not being normalized, so the error is on a different scale. However, R^2, Pearson and Spearman are not dependent on scale and show significantly worse performance. This is explained by the large errors created by outliers, which hinder models performance.
In the rest of the study, the logarithmic transformation of the normalized chemical cleavage data will be used to train the model as it preserves the relative ordering and continuous nature of the data.

Table 1: A table comparing model performance on different augmentations of the chemical cleavage data. These results are for the average over the three chemical cleavage experiments: y1, y2 and y3. For each augmentation, the metrics are averaged over the model's performance, retrained three times. The logarithmic transformation of the data and the clipped data perform very similarly, but the raw data gets much lower R^2, Pearson and Spearman scores.

	MSE	R^2	**Pearson**	**Spearman**	**F1 score**
Raw data	4.5648	0.2694	0.5308	0.4538	0.5751
Data normalized and highest 2% clipped	0.0086	0.4543	0.6863	0.6832	0.5711
Log of normalized data	0.0056	0.4638	0.6950	0.6810	0.5792

Using the logarithmic transformation of the normalized chemical cleavage data, 5-fold cross-validation was performed. Cross-validation validates the performance of the final model by showing that different train/validation splits yield the same results. The splits are made between chromosomes, which makes sure the model can generalize between different chromosomes. In each fold, 11 chromosomes are used for training and three for validation, except for one where 12 are used for training and two for validation. The results of cross-validation can be seen in Table 2. It shows very similar performance over all folds, meaning the model generalizes well and there is not a single split on which performance is significantly better or worse.
Before cross-validation, two chromosomes are put into a holdout test set on which the final model is evaluated. The holdout test set provides an unbiased evaluation of the model as no hyperparameters are tuned on it. While no active hyperparameter tuning process was applied in this study, hyperparameters were optimized slightly by hand to achieve relatively optimal performance. So the holdout set proves the model still generalizes to unseen data and the hyperparameters were not overfitting on the validation sets from cross-validation.

Along with the holdout test set, the model is also tested on the individual experiments, y1, y2 and y3, which contributed to the chemical cleavage data the model trained on. This testing is slightly biased because the training data was comprised of these individual experiments. However, testing on these individual experiments still gives an indication of the model's performance on different datasets and thus speaks to its ability to generalize. The model will also be evaluated on a completely different dataset in Section **??**. The results of the model's performance on the holdout set along with y1, y2 and y3 are shown in Table 3.
The table shows very similar performance on the holdout set compared to the average performance obtained through cross-validation. This shows the hyperparameters of the model where not overfitting on the validation sets used in cross-validation. y1, y2 and y3 do show a drop in performance compared to the holdout set, especially y3. However, the model is still able to show somewhat similar performance, meaning the model is not overfitting on the combined chemical cleavage data and generalizes to different chemical cleavage data.

Looking more closely at the results of the holdout set, different characteristics of the model can be interpreted. Compared to the Pearson and Spearman metrics the model performs much worse on R^2. As mentioned in Section 5.1, R^2 also evaluates whether the model accurately predicts the amplitude of the dyad signal in the chemical cleavage data. The model performs worse here because it predicts a more smoothed signal compared to the sharp peaks present in the chemical cleavage data. Predicting sharp peaks and valleys is difficult for a neural network because it is heavily punished when it predicts a sharp peak just beside the real peak. The loss function landscape of this regression problem is very sharp, so the model gets pulled to smoother predictions by gradient descent. However, the model does not necessarily have to predict the absolute amplitude of the signal as long as it predicts the relative amplitude between the peaks. Nucleosome position certainty can then still be inferred from the data, only the data is at a different scale.
This is measured by Pearson and the model performs much better on it. So the linear correlation between the model's predictions and the chemical cleavage data is much better than the absolute difference. The Spearman metric gives a similar result as Pearson. This could mean the linear correlation between the model's signal and the chemical cleavage data is similar to the monotonic relationship between the two signals. In other words, the model predicts the relative amplitude between the peaks similar to whether there is a peak or valley in the signal at all.

The density dot plot from Figure 6a visualizes the predicted nucleosome occupancy for each base pair at position i (n(i)) versus the true n(i) from the chemical cleavage data. The plot shows the agreement between the true and model-predicted values. There is a clear diagonal relation, which indicates that the model can predict the overall trend present in the true data well. However, for higher true occupancy values, there is a slight shift of the scatter towards the true n(i) relative to the red dashed line indicating perfect agreement. This shift displays the tendency of the model to underpredict the true n(i) values corresponding to peaks. In addition, the model underestimates lower true n(i) values, which correspond to the valleys in the occupancy signal. These errors in the model predictions can be seen more clearly in Figure 6b, which shows the difference between the predicted and true value for each n(i). Again, high true values are predicted too low, while relatively low true values show increasingly greater prediction errors.
On a different note, horizontal stripes can be seen in Figure 6a and vertical stripes in Figure 6b. These stripes correspond to artifacts from the chemical cleavage data; the main stripe at n(i)$= -7.64$ is equivalent to the pre-logged value 0, as all n(i) values are clipped to a minimal value of $5e^{-3}$ to avoid highly negative values. Many base pairs in the chemical cleavage data have an occupancy value of 0, illustrated by the peak at n(i) $= -7.64$ in the histogram of Figure 6c. The model is unable to predict these areas where no nucleosome occupancy has been measured.
However, in Figure 6c, it can be seen that most of the true values fall between -3 and -2, which is a range where the model predictions are most accurate according to 6b.

These results are also reflected in Figure 7, which shows the model's prediction on a 1000 bp window on chromosome XVI. The figure shows the model is not able to match the amplitude of the sharp peaks and valleys present in the chemical cleavage data. The model does distinguish between the relative height differences of the chemical cleavage peaks. However, the model sometimes misplaces a peak or misses a peak in the chemical cleavage data.

One important note here is that the window in the figure is 1000 bp, while the model is trained on

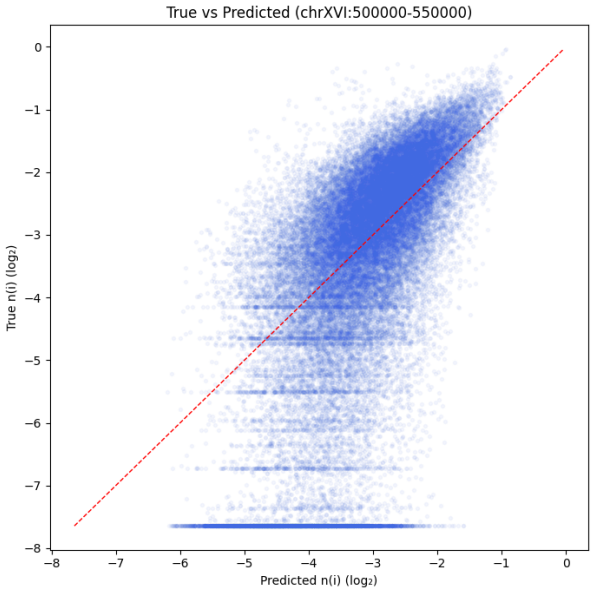
7000 bp windows. For the creation of this figure, the model was fed a 7000 bp window, of which this figure shows 1000 bp. While the model is able to process smaller windows, this greatly reduces the variance in the model's predictions. The models weights are not aligned for window sizes smaller than 7000. Feeding the model larger windows will also generally lead to poorer performance. So, when the model should predict much larger windows, they should be split up into 7000 bp windows and stitched together after model prediction. Making the 7000 bp windows overlap and taking the average would lead to a more coherent signal.

The F1 score presented for the holdout set in Table 3 would generally be considered moderate performance. Many studies boast much higher F1 scores. For example, one of the best-performing models, DeepNup, created by Zhou et al., achieved an F1 score around 0.9 for different species [ZWJ+22]. However, these models are trained on 147 bp windows as a binary classification task. Those models do not have to account for nucleosome interactions or provide a certainty level for dyad positions, as is the case for the regression model presented in this study. So, those results lose some biological relevance from treating nucleosome positioning as a binary problem.

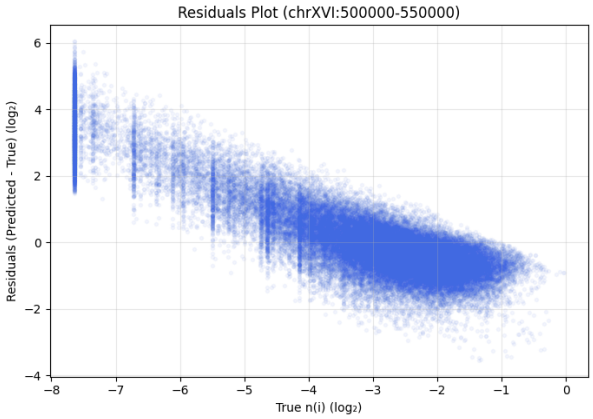
Another important consideration when interpreting the F1 score is the tolerance used to determine the precision and recall scores. As mentioned in Section 5.1, the tolerance is set to 1.0, meaning peaks 1 bp left or right of true peaks will be counted as correct. The 10 bp periodicity in nucleosome positioning roughly results in a 10 bp periodicity in the peaks of the chemical cleavage data. Considering the tolerance of 1.0, this would mean an uninformed model predicting peaks randomly would achieve a precision score of roughly 0.30. As a peak present in roughly 3 out of 10 bp is flagged as a correct prediction. A similar baseline can be made for the recall score. An uninformed model just predicting peaks with a roughly 10 bp periodicity would also achieve a recall score of about 0.30. Both precision and recall scores are significantly higher than this 0.30 baseline, indicating that the model is performing much better than acting randomly.
An important side note is that the precision baseline is in reality lower, as the chemical cleavage data does not exhibit a true 10 bp periodicity. There are many regions longer than 10 bp where no signal is given in the data. This will be discussed further in Section 5.4.

Table 2: The results of 5-fold cross-validation, where four folds contain 2 validation sets and one contains three. The target data is the logarithmic transformation of the normalized chemical cleavage with smoothing $\sigma = 2$. The table shows very similar performance across all folds, indicating no overfitting on specific chromosomes.

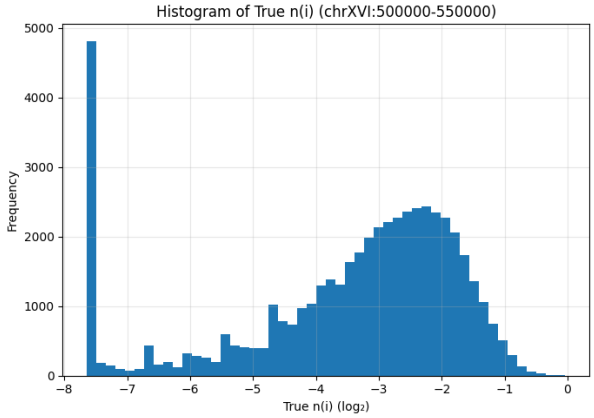
	MSE	R^2	**Pearson**	**Spearman**	**F1 score**
Fold 1	0.0059	0.4783	0.6912	0.6754	0.5851
Fold 2	0.0068	0.5050	0.7181	0.6974	0.5812
Fold 3	0.0057	0.4921	0.7044	0.6822	0.5762
Fold 4	0.0059	0.5035	0.7208	0.7006	0.5817
Fold 5	0.0057	0.4465	0.6739	0.6720	0.5720
Average	0.0060	0.4842	0.7017	0.6855	0.5794



(a) Density dot plot comparing true nucleosome occupancy from the chemical cleavage data to the predicted nucleosome occupancy from the model.



(b) Residual plot illustrating the difference between the predicted and true occupancy values for different true occupancy values.



(c) Histogram showing the frequency of different values of the true nucleosome occupancy.

Figure 6: There is a very clear agreement between the true chemical cleavage data and the model predictions, especially in the range with the most frequent true occupancy values. However, for relatively high occupancy values, the model tends to underpredict the values. Whereas, low occupancy values are more severely overpredicted by the model. The model used for these plots was trained using w=s=3500.

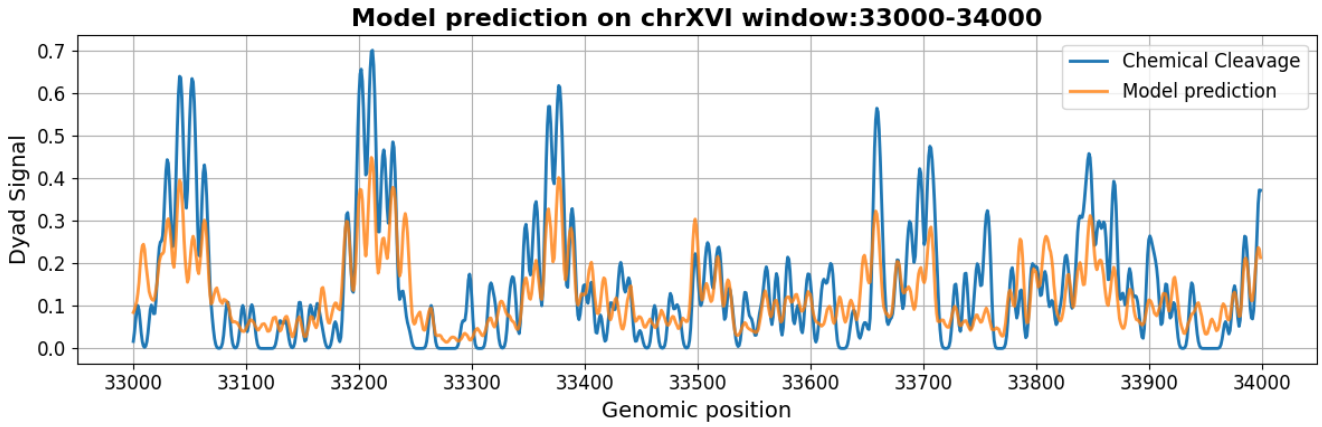


Figure 7: The augmented chemical cleavage data and the models predictions for a 1000 bp window on chromosome XVI from the test set. The model's signal roughly matches the relative height and positions of the peaks, but not the absolute amplitude.

Table 3: The performance of the model on the holdout set and the three individual chemical cleavage experiments. The model's performance on the holdout set matches the averaged performance of the cross-validation, ensuring the absence of hyperparameter-induced overfitting. The model performs worse on the individual experiments due to their sharper signals.

	MSE	R^2	Pearson	Spearman	F1 score	Precision	Recall
holdout set	0.0077	0.4734	0.7297	0.7084	0.5738	0.5993	0.5504
y1	0.0117	0.3941	0.6459	0.6249	0.4948	0.5038	0.4862
y2	0.0103	0.3948	0.6420	0.6017	0.5041	0.5312	0.4797
y3	0.0098	0.1900	0.5373	0.5264	0.4189	0.3869	0.4568

5.3 Impact of input window size

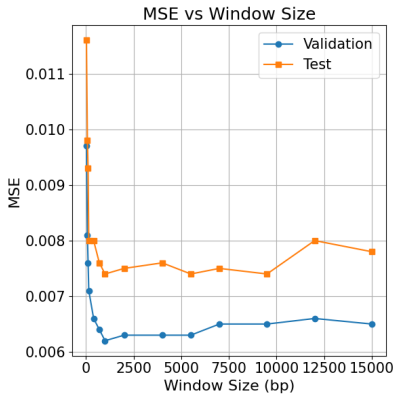
As explained in Section 4.2, the model input consists of the DNA sequence and methylation values that are divided into windows of length w. The chosen value of w has great influence on the performance and efficiency of the model. How large this influence actually is can be tested by training the model using different values for w. During this experiment, the stride s was kept at a constant value $s{=}w$ to keep the comparison fair. This means that there was no overlap between different windows and no input values were used multiple times during training. Each version of the model is run for 10 training epochs, during which chromosomes I-XII and the corresponding methylation values of the sacCer3 yeast are used as training data, chromosomes XIII-XIV are used as validation data and chromosomes XV-XVI are used as test data. During the training of each model, the model parameters of the epoch that corresponds to the lowest MSE loss are saved. The performance metrics of these saved models are plotted against their w in Figure 8.

Figure 8a shows the MSE for different values of w evaluated on the validation and test sets. If a model is trained using a w smaller than ∼400 bp, there is a significant increase in MSE. Models using these small windows lack the context to accurately follow the curve of the chemical cleavage data. The placement of nucleosomes is, of course, also influenced by the position of neighboring nucleosomes, as they cannot overlap. Thus, windows that are too small cannot capture this pattern because there are simply not enough nucleosomes in these windows. However, intermediate-sized

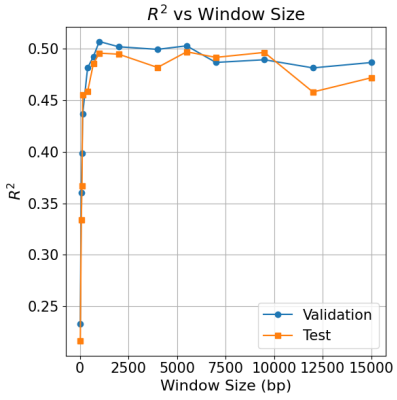
windows of ∼1000-7000 bp can capture more nucleosomes and linkers inside a single window, making it easier to generalize their relative positions. Models using windows greater than ∼7000 bp seem to provide diminishing or no further benefit.

For Figures 8b-d, the same trend can be observed from Figure 8a. The model performance plateaus when trained on intermediately sized windows, while using increasingly smaller windows exponentially hurts performance. Models trained with a w between ∼1000-7000 bp show the maximum performance that can be achieved with the given sequence and methylation input data. These values for w are optimal for explaining the variance in in vivo occupancy data for the given input data. Increasing the amount of context in the input further does not increase performance, likely due to biological limitations of the current input data of the model.

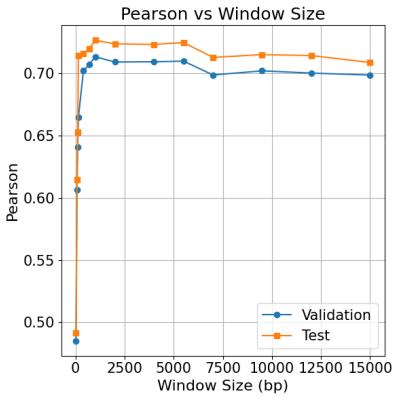
Another factor when choosing the desired w is the training speed. In Figure 8f, the relation between w and the time to train t for 10 epochs is shown. For values of w smaller than 400 bp, there seems to be a linear relationship between w and t. Since the batch size is kept at 16 during this experiment, models trained using small window sizes utilize less data per batch, as a batch contains a fixed amount of windows. Therefore, the GPU is not fully utilized and results in slower learning speed. For values of w larger than 400 bp, the GPU is working at full capacity, causing t to plateau. Increasing w indefinitely to increase the amount of context in a single window is not possible. The amount of data in a single batch is limited by the memory capacity of the used hardware.



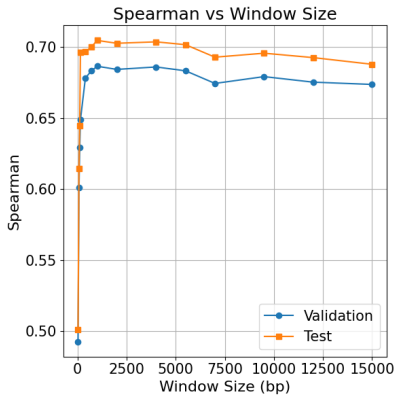
(a) Relation between MSE and window size w



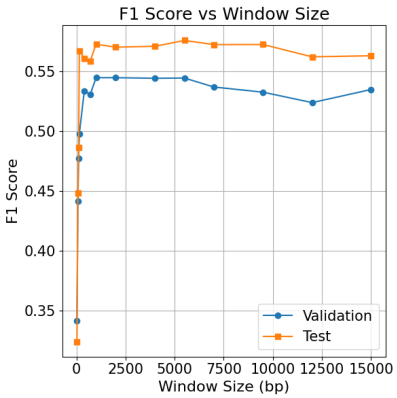
(b) Relation between R^2 and window size w



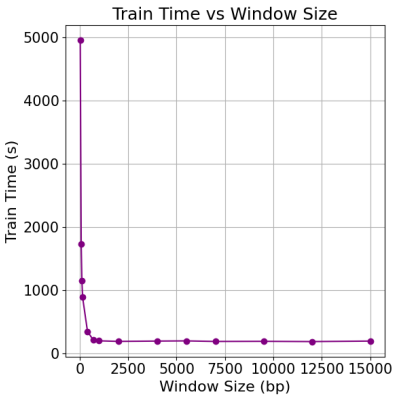
(c) Relation between Pearson and window size w



(d) Relation between Spearman and window size w



(e) Relation between F1 and window size w



(f) Relation between time to train and window size w

Figure 8: The model was trained for 10 epochs using different values for w plotted against different performance metrics. MSE decreases as the context in a single batch increases up to ∼1000 bp, where the metric plateaus. Increasing w further provides no additional benefit. R^2 increases as the context in a single batch increases up to ∼1000 bp. However, the variance in the true chemical cleavage data is not explained better by the predicted values of the model as w increases. For Pearson and Spearman, we again see performance stagnate after increasing w to ∼1000 bp. The correlation is limited by the given DNA and methylation input data. The F1 score also does not improve from an increase of w beyond 1000 bp. The ability of the model to position nucleosomes without taking certainty into account plateaus. Training time decreases linearly for values of w smaller than 400 bp, after which it plateaus to around 200s.

5.4 10bp periodicity in predicted signal

As discussed in Section 2, the 10 bp periodicity of higher A & T content running counterphase to higher C & G content plays a large role in nucleosome positioning. This is also reflected in the chemical cleavage data. The data is the result of combining dyad positions across many different reads inferred from chemical cleavage. This usually results in one or two high peaks for each nucleosome flanked by smaller peaks. All those peaks will be separated by roughly 10 bp. No reads will have dyads out of phase. As the mechanical properties of the DNA strand are out of phase in relation to the histone complex, this does not allow for easy bending around in the histone complex.

It is important to see the model also reflects this 10 bp periodicity in its predictions. To analyze this, a function had to be made that measures the degree of 10 bp periodicity in regions that have high dyad signals. It focuses on these regions with high dyad signals because those are regions with a high likelihood of nucleosome formation and thus a strong 10 bp periodicity. DNA regions that are often linker DNA will not exhibit strong 10 bp periodicity. Those regions will have little and low-amplitude peaks in the chemical cleavage data and are thus not of interest for the function. The function starts by identifying the main peaks that form the center of the regions with a high likelihood of single nucleosome placement. For this, a threshold needs to be set above which peaks are selected as main peaks. From those main peaks, a flanking region is selected to the right and left of the main peak. Any peaks present in those flanking regions will be selected as flanking peaks. If a larger peak is present in the flanking region, then that peak will become the main peak. The function then calculates how many peaks have a distance to the main peak of 0, 1 or 9 modulo 10. So, there is a tolerance of 1 bp to the left or right from the exact 10 bp periodicity.

The function will test the model on chromosome XVI, which was in the holdout set, of the sacCer3 genome. The model's prediction was made with a window of 7000 bp and a stride of 3500 bp. The signal is then averaged over the two overlapping regions.
To use this function, first a biologically justified threshold for identifying main peaks needs to be established. The flanking region of a main peak represents the region with a high likelihood of the positioning of a single nucleosome. So, the threshold should be chosen so that the number of main peaks found resembles the number of nucleosomes expected to be found on chromosome XVI.
The average nucleosome repeat length in yeast is around 167 bp [RM12]. Chromosome XVI contains 948066 bp, which means roughly $948066/167 \approx 5700$ nucleosomes are expected on the chromosome. This is a very rough estimate, as the average nucleosome repeat does not account for nucleosome-depleted regions. So, in reality, the number will be marginally lower. Following this justification, a threshold of 0.25 was selected, which found 5260 main peaks on chromosome XVI.
The flanking region should also be representative of the average width of the high-likelihood nucleosome regions. Smoothing the data with $\sigma = 20$ gives an average peak width of approximately 80 bp. So, a flanking region of 41 bp to the left and right of the main peak is chosen. This allows for the 1 bp tolerance.
The function is visualized in Figure 9. It shows the threshold for selecting the main peaks and which of the found flanking peaks are in phase and which are out of phase with the main peak.
With these settings applied to the function, the results in Table 4 are obtained. They indicate that approximately half of the flanking peaks are in phase with a tolerance of 1 bp. Looking at the total number of flanking peaks found, they are approximately seven times as numerous as the

Table 4: The results of the peak-finding function on the models predictions and the chemical cleavage data on chromosome XVI. The function uses a 1 bp tolerance to identify phased flanks that have a distance modulo 10 from the chosen main peaks. The phased flank percentage indicates a higher 10 bp periodicity in the model's signal compared to the chemical cleavage data.

	Total Main Peaks	Total Flanking Peaks	Total Phased Flanks	Phased Flanks Percentage
Prediction Model $Threshold = 0.25$	5260	34229	17652	51.6%
Chemical Cleavage $Threshold = 0.40$	5340	22689	10707	47.6%

main peaks. Given the 41 bp flanking regions, 8 times as many flanking peaks might be expected. However, some flanking peaks fall below the threshold set for identifying flanking peaks at 0.05, which is used to filter noise at the bottom.
Similar to the precision and recall calculations in Section 5.2, there is also a baseline phased flanks percentage from an uninformed model. There are 12 bp out of the 41, not counting the 1 bp tolerance next to the main peak, where an uninformed model's peak would be flagged as a phased flank by chance. This gives a baseline for the phased flank percentage of approximately 29.26%. The performance of the model is much higher than this baseline, showing a clear 10 bp periodicity in the signal produced by the model.

Figure 10a further illustrates the 10 bp periodicity by showing the offset of all flanks modulo 10 compared to the main peak. It shows most flanking peaks have an offset of zero, meaning they are perfectly in phase with the main peak. After an offset of zero, the flanking peaks are most likely to have an offset of 1 bp or 9 bp. These are the flanks that would fall in the 1 bp tolerance for the phased flanks percentage. The number of flanking peaks then slowly decreases the more out of phase the flanking peaks become compared to the 10 bp periodicity. This symmetry in the graph perfectly illustrates that the signal is centered around a 10 bp periodicity.

The function was also applied to the original chemical cleavage data. The results can be seen in Table 4 and Figure 10b. The threshold for finding main peaks was increased to 0.40, as the individual amplitudes of the main peaks are higher. Table 4 shows many fewer flanking peaks were found in the chemical cleavage data. It could indicate the chemical cleavage experiment was not large enough to pick up all phased peaks where nucleosomes might position themselves. The sparsity observed in the flanking peaks may reflect a strong positional preference of nucleosomes for specific phased locations. In contrast, the model's predictions exhibit a distribution of main and phased peaks that more closely approximates a normal distribution, suggesting a less discrete positioning pattern.
The percentage of phased flanks in the chemical cleavage data is also slightly less than that from the models prediction, indicating the small error present within the chemical cleavage method. This is also reflected in the slightly more uniformly distributed offsets of the chemical cleavage data in Figure 10b. The model appears to have captured the 10 bp periodicity and generated a signal that accentuates this periodic pattern even more prominently than the original data.

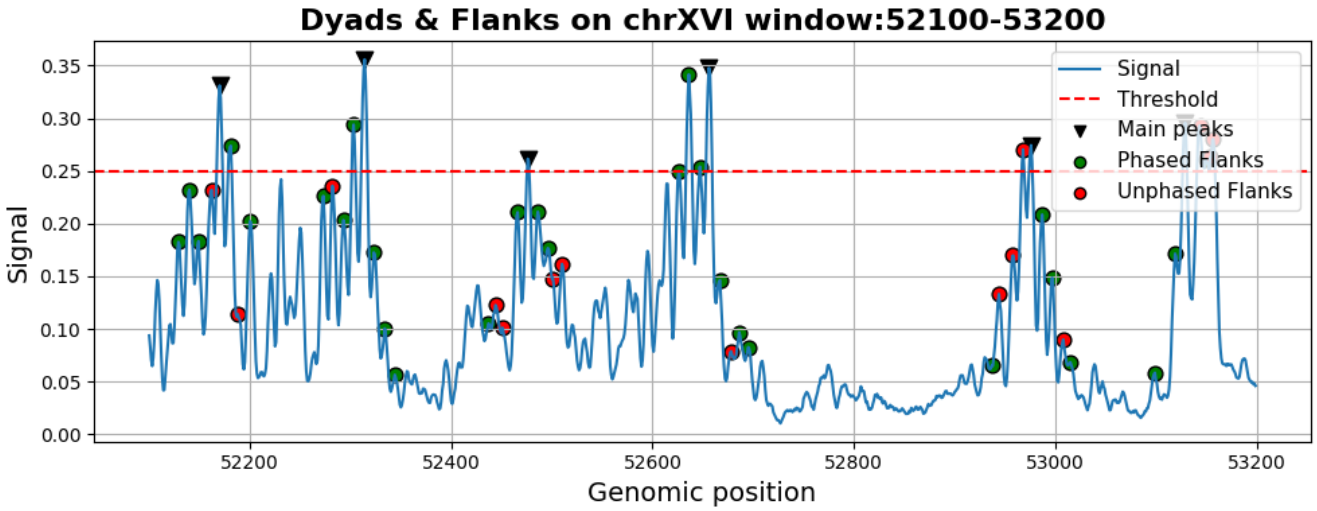
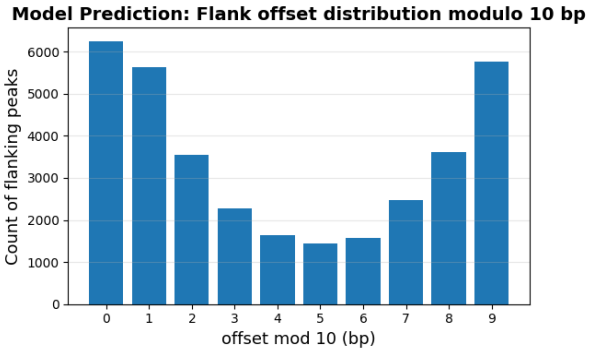
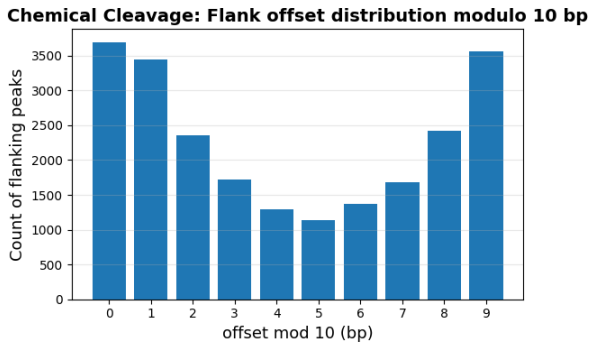


Figure 9: An illustration of the peak-finding function on an 1100 bp window on chromosome XVI. It shows the identified main peaks above the threshold and whether the peaks within the 41 bp flanking regions are in phase with the main peak. It uses a tolerance of 1 bp.



(a) The distribution of flanking peak counts for each offset in the distance to the main peak was calculated modulo 10 on the model's signal. It shows strong 10 bp periodicity.



(b) The distribution of flanking peak counts for each offset in the distance to the main peak was calculated modulo 10 on the chemical cleavage data. It shows strong 10 bp periodicity.

Figure 10: Two histograms displaying the distribution of flanking peak counts for each offset in the distance to the main peak, calculated modulo 10. The model's prediction shows slightly higher 10 bp periodicity.

5.5 Validation on Jiang et al.

To further validate the performance of the model, it was tested on a different genome-wide dataset of the sacCer3 genome from Jiang et al. [JP09a]. As mentioned in Section 4, this dataset is a meta-analysis combining six different studies. The studies used for this analysis utilized different nucleosome positioning methods, such as microarray-based and high-throughput sequence-based nucleosome mapping. Jiang et al. combined these studies into single binary dyad positions. Meaning the dyad is presented as an absolute position and no 10 bp periodicity peaks are present in the data. The study does provide some extra features for each dyad position which are peak height, number of studies that contributed to the dyad, occupancy and fuzziness. Occupancy is the likelihood of a nucleotide being wrapped up into a nucleosome. For dyad positions, it can be used as a confidence measure. Fuzziness describes the standard deviation for the dyad calculated in the analysis.
The model's signal cannot be directly compared to the binary Jiang data so it must be transformed using these features. The occupancy and fuzziness measures can be used to create a normal distribution for each dyad position. The occupancy provides the height of the distribution and the fuzziness provides the standard deviation. These normal distributions will be comparative to the grouped main and flanking peaks resembling a normal distribution in the models predictions.

The augmented data from Jiang et al. was compared to the models predictions on chromosome XVI. Table 5 shows the result of the different metrics comparing these two signals. The Jiang data is also compared to a $\sigma = 20$ smoothed signal from the model. This is done to remove the 10 bp periodicity from the model's signal. The comparison with the raw model's signal produces unrepresentative metrics, as the 10 bp periodicity is absent from the Jiang data and thus greatly diminishes correlation. Table 5 also shows much higher correlation between the two signals on R^2, Pearson and Spearman for the smoothed signal.
The R^2 score for the smoothed data is worse than the correlation between the Jiang data and its mean. This is due to the different scales of the signals, and so the Pearson and Spearman metrics are more representative of the correlation. Both metrics show significantly worse performance compared to the results achieved on the chemical cleavage data. This is largely to be expected due to the different nature of the Jiang data. Still, the metrics show there is a clear correlation between the two signals. The Spearman correlation is slightly higher, indicating the linear relationship is slightly smaller than the monotonic relationship. This is also to be expected, as the Jiang data shows little variance in peak height.
Another interesting result shown in Figure 11 is the correlation between uncertainty in the models signal and the Jiang data. Sub-figures 11a & 11c show regions with high fuzziness in the Jiang data. Consequently, the model's signal is also noisy and lacks clear peak structures. Similarly, in Subfigures 11b & 11d, the Jiang data shows low fuzziness and the models signal is much less diffuse, showing clear peak structures.

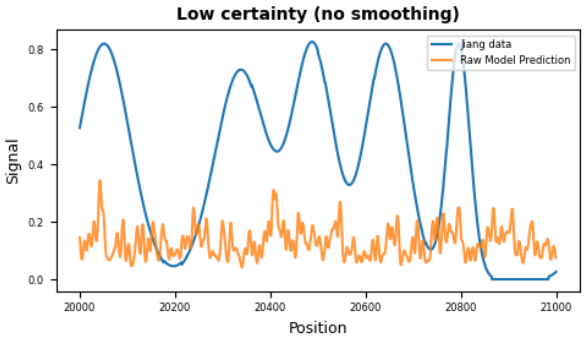
As done previously, the peaks in the two signals were also converted to binary positions to compare nucleosome positioning more thoroughly. These scores reflect the similarity between the most probable nucleosome positions, rather than exact dyad locations, as was the case in the previous sections. The thresholds for identifying peaks were again chosen to roughly approximate the number of expected peaks on chromosome XVI. As can be seen in Table 6, this yielded an average distance of 32.32 bp. It shows that the most probable nucleosome positions between the

Table 5: A comparison between the models predicted signal and the Jiang data on chromosome XVI. Pearson and Spearman show a significant correlation between the two signals. Removing the 10 bp periodicity through smoothing with $\sigma = 20$ considerably increases correlation.

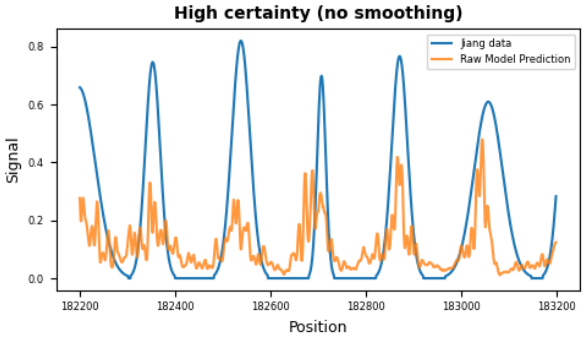
	MSE	R^2	**Pearson**	**Spearman**
no smoothing ChemClev	0.0725	-0.0377	0.4188	0.4386
smoothing ChemClev $\sigma = 20$	0.0734	-0.0503	0.4916	0.5119

models signal and the Jiang data vary quite significantly. However, the general normal distributions do show much overlap.
When the 30 bp distance is applied as tolerance in calculating precision and recall, the resulting scores do exceed 0.5. This is likely due to a few outliers in peak distance, which greatly increase the average, while the majority of distances fall below 32.32 bp.

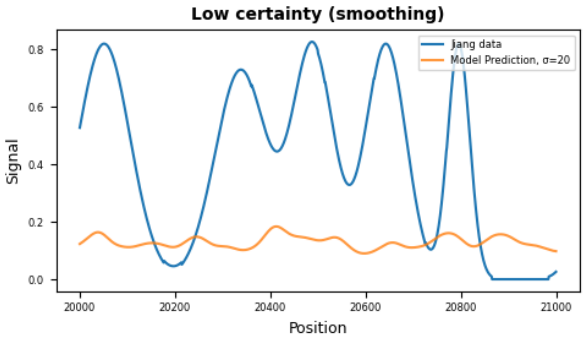
The 10 bp periodicity in the models signal can also be compared to the Jiang data. As each binary peak in the data should fall significantly close to a peak in the model's signal. Figure 12 shows a comparison between the binary data and the model's signal, with the green triangles indicating binary peaks that fall within a 1 bp tolerance of the closest model peak. Table 7 shows the results across chromosome XVI. It shows that 26% of binary peaks are in phase with the model's signal on chromosome XVI. This is a significantly lower result than is needed to show a correlation higher than the random baseline. The average distance between the binary peaks and the model's signal, with outliers clipped, suggests there is no correlation at all between the phase of the binary peaks and the model's peaks. Considering the 10 bp periodicity in the models signal, the range of distances between binary peaks and models peaks is one to five. So, an average distance of three is slightly below the no correlation baseline of 2.5.
However, when the binary data is compared to the original chemical cleavage data it shows the same statistics. This suggests there is no 10 bp periodicity present in the binary data. During the averaging over the different studies of the meta-analysis performed in Jiang, the 10 bp periodicity might not have been taken into account.



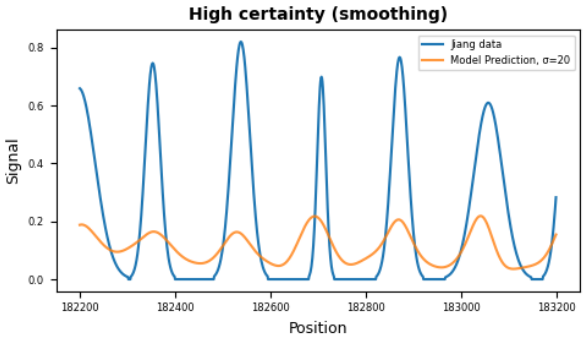
(a) A region of high certainty in the Jiang dataset corresponds to a noisy, unsmoothed signal from the model.



(b) A region of high certainty in the Jiang dataset corresponds to a more distinct, peak-like structure in the model's unsmoothed predictions.



(c) A region of high certainty in the Jiang dataset corresponds to a noisy smoothed signal from the model.



(d) A region of high certainty in the Jiang dataset corresponds to a more distinct, peak-like structure in the model's smoothed predictions.

Figure 11: A comparison between the Jiang data and the models predictions for low- and high-certainty regions on a 1000 bp window on chromosome XVI. The figures show a relation between uncertainty in the Jiang data and the models predictions.

Table 6: The first table shows the average distance between peaks in the model's signal $\sigma = 20$ and the Jiang data. The second table shows the binary peak comparison with the tolerance of 30 bp. The tables show a rough correlation between the highest probability dyad positions.

	Avg Peak Distance	Total Model Peaks	Total Reference Peaks
Peak Finding Function	32.32	5139	5114

	F1	Precision	Recall
Peak Comparison Tolerance=30bp	0.7001	0.6983	0.7018

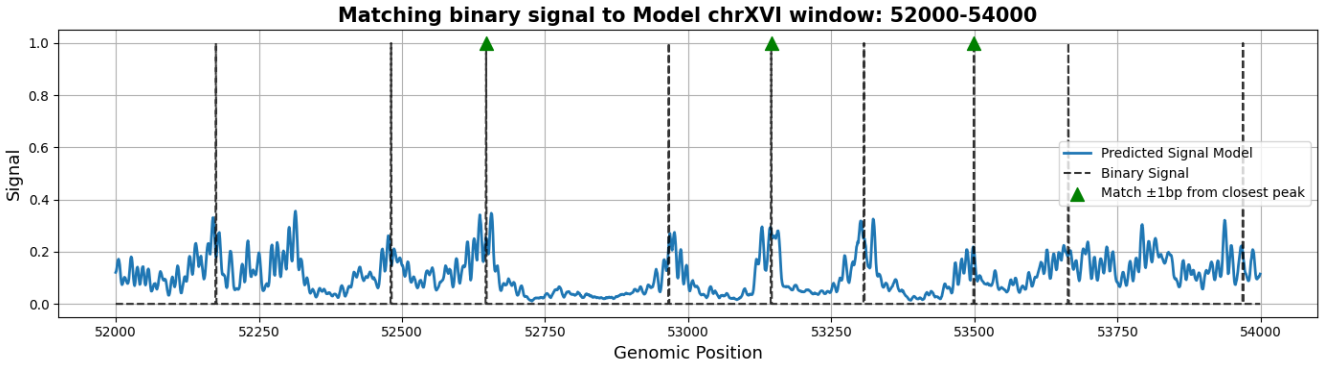


Figure 12: A comparison between the originally binary Jiang data and the models predictions on a 2000 bp window on chromosome XVI. The green triangles show binary peaks within a 1 bp tolerance of a model peak. The figure shows no significant 10 bp periodic relation between the two signals.

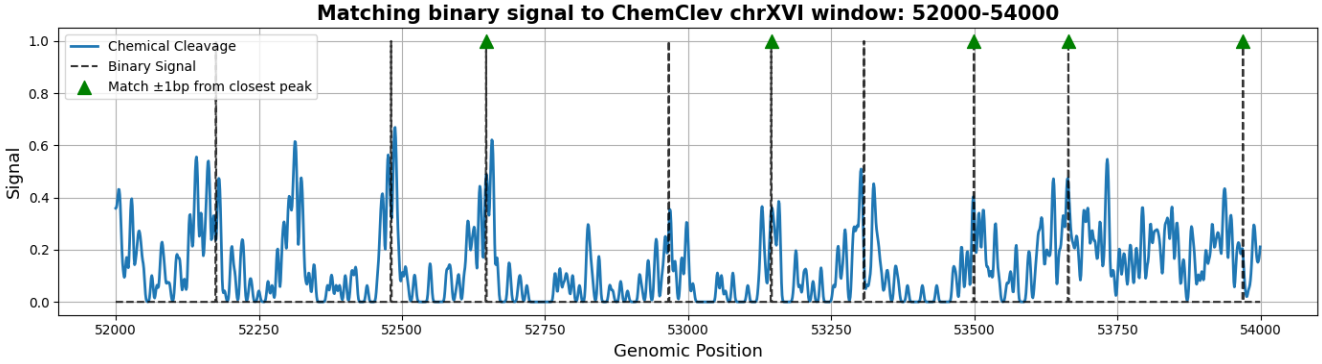


Figure 13: A comparison between the originally binary Jiang data and the chemical cleavage data on a 2000 bp window on chromosome XVI. The green triangles show binary peaks within a 1 bp tolerance of a model peak. The figure shows no significant 10 bp periodic relation between the two signals.

Table 7: A comparison between the distance of a binary peak to the closest model peak and chemical cleavage peak with a tolerance of 1 bp. The percentage and the mean distance indicate a slightly lower correlation compared to a random baseline for both the models predictions and the chemical cleavage data.

	Total Binary Peaks	Correct Binary Peaks	Percentage	Mean Distance
Model	4762	1238	26.00	3.030
Chemical Cleavage	4762	1144	24.02	3.218

5.6 Influence of sequence and methylation on model performance

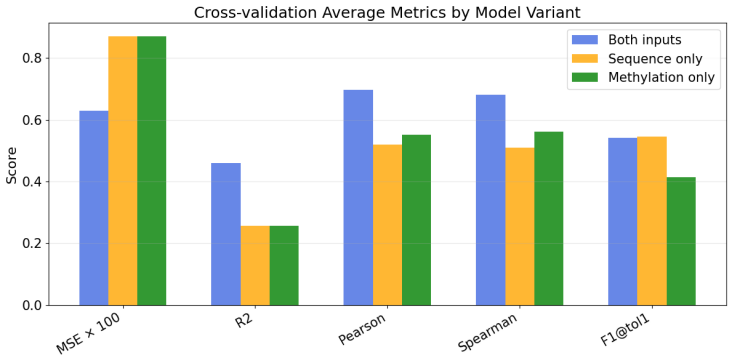
To identify the contributions of the different model input data types, three different variations of the model are created and compared to each other. The first variant only uses DNA sequence as input, the second variant only uses the methylation values of each nucleotide position as input and the third uses both as input. The metrics for each model variant from Figure 14a have been obtained through 5-fold cross-validation in the same way as described in Section 5.2. In Figure 14b-f, the different folds are compared for each metric and model variant.

The values for Pearson, Spearman and F1 are very consistent across different folds for all variants; however, MSE and R^2 are more variable. These two metrics are more affected by the scale of the true and predicted values as well as outliers in the true values. This can explain the relatively high differences between different folds for MSE and R^2. However, Figures 14b & 14c also show that all different variants have relatively similar performance for each fold. If, for example, the model using only sequence as input performs relatively badly on these metrics because of an unfavorable split of the train and validation chromosomes, the other two model variants also show decreased performance and vice versa. On another note, the other metrics from Figure 14d-f are much more stable across different folds for each model variant. Pearson, Spearman and F1 are much less influenced by the scale of the occupancy signal and are less prone to outliers. Furthermore, these three metrics represent the goal of our application better than MSE and R^2. The model needs to identify areas in the occupancy signal that are relatively high or low, which can be quantified through Pearson and Spearman correlation. The actual detection of peaks in the signal where nucleosomes are present is measured through F1. These abilities of the model are more important than following the exact absolute occupancy signal of the true values, which can contain errors or biological inaccuracies.

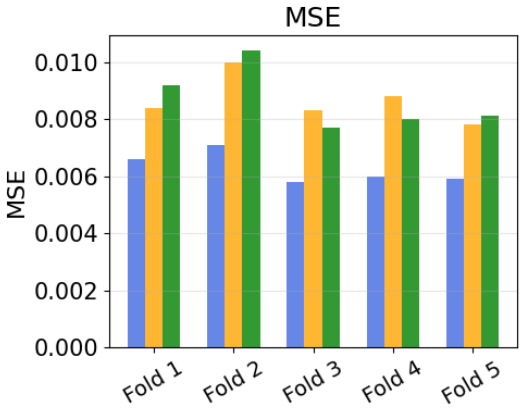
As for the average cross-validation results in Figure 14a, the standard model variant using both sequence and methylation as input shows increased or similar performance for each metric compared to the other model variants using only a single type of input. Thus, it is beneficial to use both DNA sequence and methylation data if the goal is to accurately and robustly predict nucleosome occupancy. As it seems, both input types deliver complementary information that, when combined, results in better performance compared to either model type alone. Comparing both of the single-type model variants, it seems both perform very similarly for MSE and R^2. However, the methylation-only model outperforms the sequence-only model for the correlation metrics. Indicating that the methylation data provides a better signal for capturing the nucleosome occupancy patterns compared to sequence alone.
Nevertheless, the sequence-only model achieves very similar F1 scores as the model using both types of input. Thus, identifying peaks using only DNA sequence as input seems to be more informative compared to using only methylation as input. Moreover, including methylation as input for the model does not translate to an increased ability to predict precise nucleosome positioning.

Interestingly, when smoothing the true chemical cleavage data and the model input methylation data with $\sigma = 20$, the Person and Spearman correlation coefficients become -0.559 and -0.520, respectively. This highly negative correlation was aimed for in the creation of the methylation data. Only adenine nucleotides that were not already bound to a different protein were able to be methylated using m6A-methyltransferase. As a consequence, DNA in nucleosomes shows a relatively

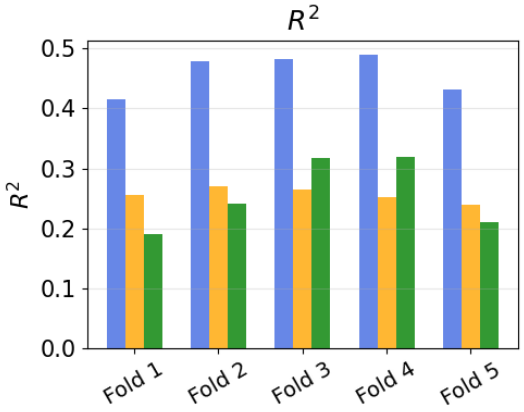
low methylation signal compared to linker DNA, where the methylation signal is relatively high. This relation is shown in Figure 15, the peaks of the methylation signal fall in between the peaks of the true chemical cleavage signal in well-defined occupancy areas.



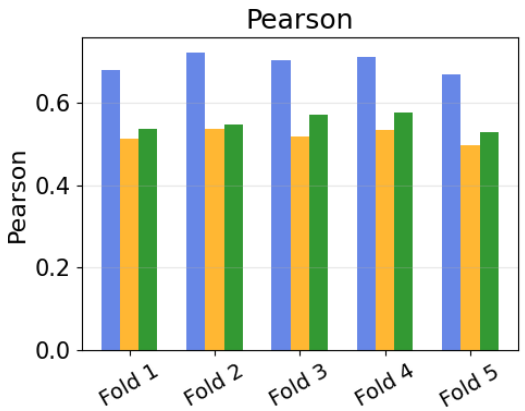
(a) Average 5-fold cross-validation results for each model variant (sequence only, methylation only and both). The different model variants are compared using multiple metrics. The standard model using both DNA sequence and methylation as input outperforms both single-type model variants on most metrics. The values for MSE are multiplied by 100 to increase the visibility.



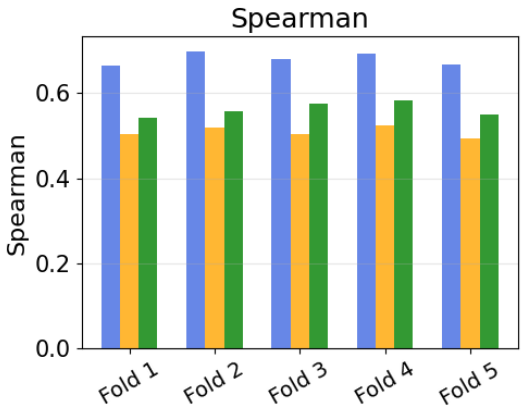
(b) MSE scores for the different model variants across multiple folds.



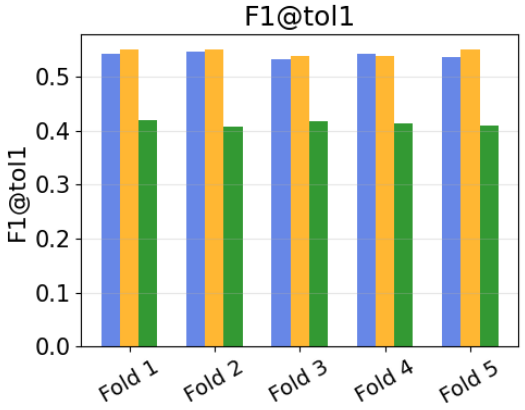
(c) R^2 scores for the different model variants across multiple folds.



(d) Pearson correlation coefficients for the different model variants across multiple folds.



(e) Spearman correlation coefficients for the different model variants across multiple folds.



(f) F1 score with a tolerance of 1 bp and a threshold of 0.1 for the different model variants across multiple folds.

Figure 14: The model using both DNA sequence and methylation as input outperforms the other model variants on every fold for all metrics except F1. Indicating that both input types offer complementary information. However, for F1, the model using both DNA sequence and methylation as input performs similarly to the model using only DNA sequence as input. This indicates that only sequence data is needed to optimally predict nucleosome occupancy peaks. The cross-validation was performed on model variants using $w{=}s{=}3500$.

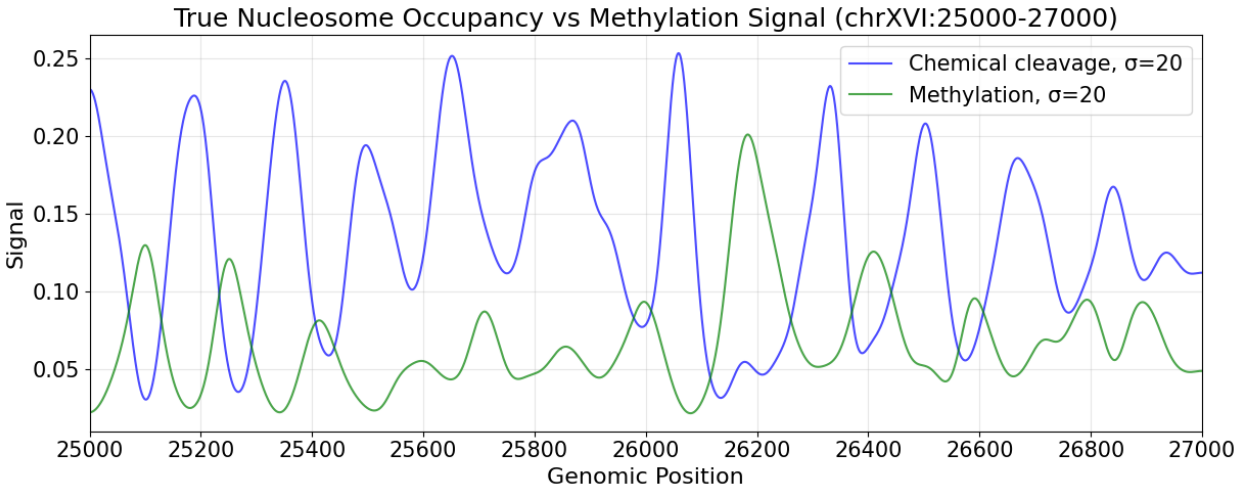


Figure 15: The true chemical cleavage occupancy signal and the methylation signal have a highly negative correlation. This means that a high true signal is accompanied by a low methylation signal at the same genomic position and vice versa. By smoothing the data with $\sigma = 20$, the correlation is clearly visible.

5.7 Single-read methylation model variant

In the previous sections, the model was trained on an averaged methylation profile across the whole sacCer3 genome. This yields a model capable of predicting the averaged nucleosome positioning signal from the averaged methylation profile. However, the model cannot predict nucleosome positioning on individual reads. This is valuable for when one wants to infer the structure of individual strands, which is not possible with an averaged nucleosome positioning signal. So, for this section, the model was trained on individual methylation reads instead of their average. Note that the model's predicted signal is still validated on the chemical cleavage data, which is also an averaged nucleosome positioning signal, as there is no data with exact nucleosome positions for each read. This means the model will also still produce an averaged signal similar to the chemical cleavage data. Therefore, to infer precise nucleosome positions from the model's predictions on individual reads, a dedicated algorithm must be developed to extract exact dyad positions from the predicted signal. For example, by choosing the highest peaks as the true dyad positions.

To train this new model, roughly 500 reads of at least 7000 bp were selected per chromosome. The selected reads were spaced apart so that the largest part of every chromosome was covered. As the model expects a standard window length, only the first 7000 bp were selected from every read. Lastly, the windows were split per chromosome to create the train, validation and test sets.

Figure 16 shows the predicted signal of the single-read model on an individual read compared to the chemical cleavage data. The red marks at the top of the figure indicate the methylation sites present on the read.
Table 8 shows the performance of the single-read model on the chemical cleavage data. The results of both the validation set and test set are similar, indicating no overfitting on hyperparameter tuning. The performance of the model trained on the averaged methylation profile is also shown. It shows significantly poorer performance. This is to be expected, as this model is not tuned to handle binary methylation data of individual reads.

Table 8: The performance of the single-read model on the chemical cleavage data for both the validation and test sets. The performance on the single reads is poorer than on the averaged methylation data validated with chemical cleavage. For comparison, the considerably worse performance of the averaged methylation model on the single-read test set is also shown.

	MSE	R^2	Pearson	Spearman	F1 score
Single-Read Model (Validation Set)	0.0089	0.2887	0.5644	0.5410	0.5339
Single-Read Model (Test Set)	0.0113	0.2361	0.5761	0.5606	0.5347
Average Methylations Model	0.0211	-0.4244	0.3114	0.3525	0.4368

Table 9: The performance of the averaged predictions over 50 reads of the single-read model on chromosomes XV and XVI. The results show the averaged model predictions have poorer correlation with the chemical cleavage data compared to the model's prediction on a single read.

	MSE	R^2	Pearson	Spearman	F1 score
Averaged Predictions Reads=50	0.0119	0.1931	0.5178	0.4983	0.5447

The results of the single-read model compared to the results of the averaged methylations model in Table 3 do show poorer performance. Which is to be expected, as the single-read model must predict an averaged nucleosome positioning signal from an individual read. The single model's predicted signal is likely more similar to the probabilistic nucleosome positioning profile of each individual read.

So, a more accurate way of comparing the single-read model to the chemical cleavage data is to average the model's predictions over many reads. To test this, the model was fed 7000 bp windows of the test chromosomes XV and XVI. For each 7000 bp window, the model made a prediction on 50 different reads. Those predictions were averaged and the metrics were calculated. Table 9 shows the averaged results for the two test chromosomes.
Interestingly, the averaged model predicted signal shows less correlation with the chemical cleavage data compared to the single predictions. So, on average, the prediction on a single read shows more resemblance to the chemical cleavage data than the average prediction over many reads. This might be indicating that the single-read model predicts the averaged nucleosome positioning signal fairly well. Only when those predictions are averaged does the sharpness of the signal decrease, which reduces the correlation with chemical cleavage. This is also reflected by the F1 score staying the same while the other metrics decreased.
It is also important to note that the sequence might play a large role in the single-read model's ability to infer the averaged nucleosome positioning signal fairly well.

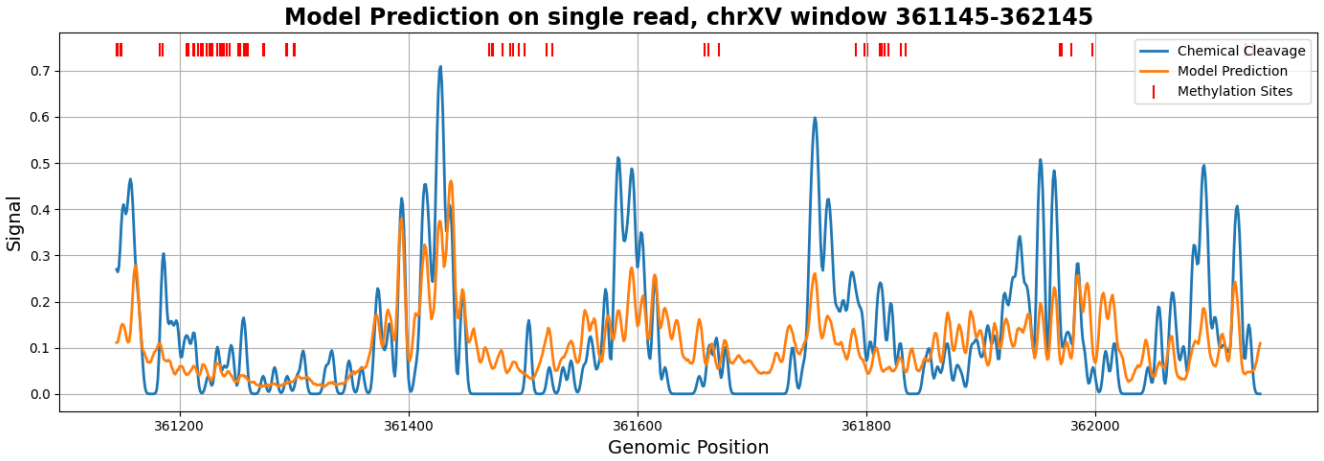


Figure 16: A 1000 bp window of the single-read model's prediction on a read from chromosome XV compared to the chemical cleavage data. The red markings show the methylations from the corresponding read. The single-read model can be seen to still approximate the chemical cleavage data from one methylation read.

6 Conclusion and Discussion

In this study, the performance of a DLNN on nucleosome positioning has been analyzed in many different ways. The analysis has carefully quantified the influence of both the target data and the input data on performance. Furthermore, the models predicted signal has been compared to a different dataset and a model trained on single methylation reads has been studied.
To correctly and completely measure the performance of the model, a small inquiry was made into validation metrics. MSE measures the absolute difference between the two signals and is dependent on the scale. R^2 measures the degree to which the predicted signal explains variance in the ground truth. Pearson measures the linear correlation between the two signals. Spearman measures the monotonic relation between the two signals. The F1 score indicates how many peaks in the two signals align given a tolerance. Pearson, Spearman and F1 score have the most relevance for the nucleosome positioning problem. Pearson can be used as a measure for peak and valley placement along with their relative height. Spearman can be used for peak and valley placement and the F1 score focuses specifically on peak placement.
These metrics give a good indication of how well the model performs compared to a baseline. However, it is difficult to compare the model's results directly to other published literature. All literature found using a DLNN on nucleosome positioning treats it as a binary problem with 147 bp windows.
When examining the continuous output of nucleosome positioning over large genomic windows, this study most closely resembles the work of van der Heijden, which employed a statistical mechanics model [vdHvVLvN12]. The van der Heijden study reported a Pearson's correlation coefficient of 0.66 on a fitting window of 20000 bp in vivo. While a fitting window is different from how a window is used in the context of this study, the results can be compared to some extent. This study reported a slightly higher value for Pearson's correlation coefficient for a window size of 7000 bp. However, it is unclear what could have caused the slight increase due to the large difference in approach.
When interpreting the results, one must keep in mind the data augmentations made and their biological impact. An important augmentation to improve training was applying a logarithmic transformation to the chemical cleavage data. This greatly reduces the effect of outliers on the loss function. However, these outliers do have biological significance. They indicate places with a very high likelihood for nucleosome positioning. Nucleosome placement in these positions might be essential for structural purposes or cell processes. So, possibly important biological information is lost when applying a logarithmic transformation to the chemical cleavage data.

Other notable assumptions were made when analyzing the 10 bp periodicity in the models signal and the chemical cleavage data. To begin with, the estimate of the number of main peaks is based on a rough assumption. It is only used, however, to show the threshold is not identifying noise as nucleosome peaks. Secondly, large parts of the chemical cleavage data and the model's signal show no clear high peaks and smaller flanking peaks. As many regions of the genome display low confidence in nucleosome positioning. The main peak threshold was only used to identify positions from which the 10 bp periodicity should be visible. For, while certain genomic regions exhibit 10 bp periodicity in nucleosome positioning, these regions are not necessarily in phase with one another across the entire chromosome.

The main model's architecture in this study has been designed to receive two types of input,

DNA sequence and nucleotide methylation data for the genomic area of interest and is trained on the true chemical cleavage data. Using residual convolutional layers, the model learns local patterns and trends, such as the 10 bp periodicity for AA, TA, TT and GC dinucleotides present in the DNA sequence. Furthermore, the model makes use of BiLSTM layers to capture long-range dependencies between neighboring nucleosomes, which can be fully utilized using an input window size between 1000 and 7000 bp.

This study has also made a small inquiry into a model trained on single methylation reads. The model showed moderate performance on reconstructing the average chemical cleavage profile across the genome. With additional development and hyperparameter tuning, this model could potentially be further improved. A single-read model achieving performance comparable to that of the averaged methylation model is highly desirable, as it requires significantly less data.
It is, however, still unclear how significant the role of the individual methylation reads was for predicting this averaged signal. As shown in Section 5.6, a model trained only on sequence also shows moderate performance on predicting the chemical cleavage data. So, a direct comparison still has to be made between the single-read model and a sequence-only model.
This research may also act as a stepping stone to models predicting single-molecule nucleosome positioning from single methylation reads. These models would actually learn the heterogeneity in nucleosome positioning instead of the blurred average. For example, the model presented in this study will not be able to recognize regulatory regions that are either fully open or fully closed. The averaging washes out this variance and the model will interpret those regions as partially accessible. Models able to predict nucleosome positioning for individual molecules could give valuable insights into epigenetics and chromatin formation.

Another key part of this study in order to answer the research question was the comparison between different model variants that use a different combination of input data types. In order to identify the added contribution of the methylation data to the commonly used sequence data, their respective and combined contributions to the model performance needed to be isolated. The 5-fold cross-validation experiment for each model variant revealed that the model using both DNA sequence and methylation data outperformed the variants using only a single type of input. Indicating that the methylation data contains additional information about the positioning of nucleosomes that is not fully captured in the DNA sequence and vice versa. The methylation-only model achieved greater Pearson and Spearman correlations compared to the sequence-only model, which shows that the methylation data can more effectively be used to differentiate between high and low nucleosome occupancy signal regions. This finding is also supported by the significant negative correlation between the chemical cleavage data and the methylation data. On the other hand, the sequence-only model achieved a higher F1 score compared to the methylation-only model and even performed similarly to the model using both inputs. From this we can conclude that the methylation data does not provide any additional information useful for identifying precise nucleosome occupancy peaks that is not already present in the sequence data. In other words, the methylation data is more suitable for differentiating nucleosomal DNA from linker DNA, while DNA sequence data is better suited for determining exact locations of nucleosome dyad positions.

Overall, it has become clear that the value of combining different input data types for increased robustness and accuracy is great. The addition of methylation data to the model input allows the

model to learn from both DNA sequence patterns and epigenetic nucleotide modifications, resulting in an increased ability to predict the position of nucleosomes for the *Saccharomyces cerevisiae* yeast genome. In the future, different data types could be added to the input of the model in order to capture additional features or patterns not present in sequence or methylation data. For in vivo occupancy data other factors also play a large role. Transcription factors and remodeling enzymes, for example, are also indicators for nucleosome positioning [SS13]. Thus, expanding the diversity of training data will likely increase the correlation between the predicted and true values and break the performance plateau currently observed.

Moreover, to improve the performance of the model even further, hyperparameter tuning should be performed to find the optimal settings and parameters for our current model. Due to time constraints, there was no time to tune the model; therefore, it is unlikely that the model is performing fully optimally in this state. However, the additional gains from comprehensive tuning of the current model architecture would most likely only result in marginal gains based on the manual tuning experiments carried out. Therefore, it may also be worthwhile to explore alternate architectures, such as transformer-based models [VSP+17]. Transformers have been demonstrated to perform well for sequence modeling tasks. Therefore, they could also contribute to solving the nucleosome positioning problem. Transformers have already been used in a past study [FADA23] where their proposed model outperformed traditional DNA feature extractors [GSH22]. However, in these studies the task is treated as a binary classification instead of genome-wide occupancy prediction as in this study. Furthermore, none of the existing transformer-based models incorporate both DNA sequence and methylation data. Using specific epigenetic methylation data for nucleosome position prediction is a novel area of research, particularly in combination with transformer-based models. Thus, in the future a transformer-based model that can more accurately predict nucleosome occupancy across whole genomes should be developed.

References

[ABR20] Domenico Amato, Giosue' Lo Bosco, and Riccardo Rizzo. Corenup: A combination of convolutional and recurrent deep neural networks for nucleosome positioning identification. *BMC bioinformatics*, 21:1–14, 2020.

[Bij25] Martin Bijl. Using dhmmtoreveal nucleosome breathing in methylation-seq. *Leiden university*, 2025.

[BK23] Lokesh Borawar and Ravinder Kaur. Resnet: Solving vanishing gradient in deep networks. pages 235–247, 2023.

[CRBH18] Razvan V Chereji, Srinivas Ramachandran, Terri D Bryson, and Steven Henikoff. Precise genome-wide mapping of single nucleosomes and linkers in vivo. *Biophysical Journal*, 114(3):564a, 2018.

[DT13] Timothy R Hughes Desiree Tillo. G+c content dominates intrinsic nucleosome occupancy. *nature*, Nature Structural Molecular Biology volume 20:267–273, 2013.

[FADA23] Ahtisham Fazeel, Areeb Agha, Andreas Dengel, and Sheraz Ahmed. Np-bert: A two-staged bert based nucleosome positioning prediction architecture for multiple species. pages 175–187, 2023.

[FKFM+08] Yair Field, Noam Kaplan, Yvonne Fondufe-Mittendorf, Irene K Moore, Eilon Sharon, Yaniv Lubling, Jonathan Widom, and Eran Segal. Distinct modes of regulation by chromatin encoded through nucleosome positioning signals. *PLoS computational biology*, 4(11):e1000216, 2008.

[GSH22] Ying Li Guo-Sheng Han, Qi Li. Nucleosome positioning based on dna sequence embedding and deep learning. *BMC Genomics*, 23:301, 2022.

[HHD+16] Bryan T Harada, William L Hwang, Sebastian Deindl, Blaine Bartholomew, and Xiaowei Zhuang. Stepwise nucleosome translocation by rsc remodeling complexes. *Biophysical Journal*, 110(3):515a, 2016.

[HZRS16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[IS15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015.

[IWK17] Roman Ilin, Thomas Watson, and Robert Kozma. Abstraction hierarchy in deep learning neural networks. pages 768–774, 2017.

[JP09a] Cizhong Jiang and B Franklin Pugh. A compiled and systematic reference map of nucleosome positions across the saccharomyces cerevisiae genomes. *Genome biology*, 10:1–11, 2009.

[JP09b] Cizhong Jiang and B Franklin Pugh. Nucleosome positioning and gene regulation: advances through genomics. *Nature Reviews Genetics*, 10(3):161–172, 2009.

[KL24] Gert-Jan Kuijntjes and Tineke Lenstra. Dataset collected at the netherlands cancer institute (nki), 2024. Unpublished dataset used in this thesis.

[KLL+12] Theresa K Kelly, Yaping Liu, Fides D Lay, Gangning Liang, Benjamin P Berman, and Peter A Jones. Genome-wide mapping of nucleosome positioning and dna methylation within individual dna molecules. *Genome research*, 22(12):2497–2506, 2012.

[LC25] Stephen E Farr Jinyue Luo Bryan A Gibson Lynda K Doolittle Jorge R Espinosa Jan Huertas Sy Redding Rosana Collepardo-Guevara Michael K Rosen Lifeng Chen, M Julia Maristany. Nucleosome spacing can fine-tune higher order chromatin assembly. *Biophysical journal*, 124, 2025.

[Lug97] Mäder A. Richmond R. et al. Luger, K. Crystal structure of the nucleosome core particle at 2.8Å resolution. *Nature*, 389:251–260, 1997.

[LW04] Gu Li and Jonathan Widom. Nucleosomes facilitate their own invasion. *Nature structural & molecular biology*, 11(8):763–769, 2004.

[Nat23] National Center for Biotechnology Information (NCBI). Saccharomyces cerevisiae genome (saccer3) - fasta format. https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSM5861292, 2023. Accessed: 2025-03-01.

[NK25] Sihyeong Nho and Hajin Kim. Dynamics of nucleosomes and chromatin fibers revealed by single-molecule measurements. *BMB reports*, 58(1):24, 2025.

[RK20] Alexander Rehmer and Andreas Kroll. On the vanishing and exploding gradient problem in gated recurrent units. *IFAC-PapersOnLine*, 53(2):1243–1248, 2020.

[RKM+19] Sergei Rudnizky, Hadeel Khamis, Omri Malik, Philippa Melamed, and Ariel Kaplan. The base pair-scale diffusion of nucleosomes modulates binding of transcription factors. *Proceedings of the National Academy of Sciences*, 116(25):12161–12166, 2019.

[RM12] Julien Riposo and Julien Mozziconacci. Nucleosome positioning and nucleosome stacking: two faces of the same coin. *Molecular bioSystems*, 8(4):1172–1178, 2012.

[RPR+02] Christophe Redon, Duane Pilch, Emmy Rogakou, Olga Sedelnikova, Kenneth Newrock, and William Bonner. Histone h2a variants h2ax and h2az. *Current Opinion in Genetics Development*, 12:162–169, 2002.

[SDH+20] Andrew B Stergachis, Brian M Debo, Eric Haugen, L Stirling Churchman, and John A Stamatoyannopoulos. Single-molecule regulatory architectures captured by chromatin fiber sequencing. *Science*, 368(6498):1449–1454, 2020.

[SHK+14] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[SP97] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681, 1997.

[SS13] Kevin Struhl and Eran Segal. Determinants of nucleosome positioning. *Nature structural & molecular biology*, 20(3):267–273, 2013.

[StH25] Iwan Steenhoff and Wessel ten Hoeve. Scriptienucleotidemodels. https://github.com/WtenHoeve/ScriptieNucleotideModels, 2025.

[TNM08] Ilya P. Ioshikhes Xiaoyong Li Bryan J. Venters Sara J. Zanton Lynn P. Tomsho Ji Qi Robert L. Glaser Stephan C. Schuster David S. Gilmour Istvan Albert B. Franklin Pugh Travis N. Mavrich, Cizhong Jiang. Nucleosome organization in the drosophila genome. *nature*, 453:358–362, 2008.

[Tro22] Edoardo Trotta. Gc content strongly influences the role of poly(da) in the intrinsic nucleosome positioning in saccharomyces cerevisiae. *Wiley*, 39:262–271, 2022.

[vdHvVLvN12] Thijn van der Heijden, Joke JFA van Vugt, Colin Logie, and John van Noort. Sequence-based prediction of single nucleosome positioning and genome-wide nucleosome occupancy. *Proceedings of the National Academy of Sciences*, 109(38):E2514–E2522, 2012.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[WP] John van Noort Wim Pomp. Pacbio-processing. https://github.com/JvN2/Pacbio-processing. Accessed: 2025-03-01.

[YL11] Ye Yang Yanping Fan Renliang Yang Chuan-Fa Liu Nikolay Korolev Lars Nordenskiöld Ying Liu, Chenning Lu. Influence of histone tails and h4 tail acetylations on nucleosome-nucleosome interactions. *Elsevier*, 414:749–764, 2011.

[ZPW18] Juhua Zhang, Wenbo Peng, and Lei Wang. Lenup: learning nucleosome positioning from dna sequences with improved convolutional neural networks. *Bioinformatics*, 34(10):1705–1712, 2018.

[ZWJ+22] Yiting Zhou, Tingfang Wu, Yelu Jiang, Yan Li, Kailong Li, Lijun Quan, and Qiang Lyu. Deepnup: prediction of nucleosome positioning from dna sequences using deep neural network. *Genes*, 13(11):1983, 2022.

7 Appendix

Table 10: Exact values for datapoints in Figure 8. Shows the relation between window size w and performance on different metrics. "v" or "t" stands for validation and test, respectively. So, for example, "400v" means validation performance for w=400.

Window Size (bp)	MSE	R^2	Pearson	Spearman	F1 score	Train Time (s)
25v	0.0097	0.2333	0.4852	0.4925	0.3418	4949
25t	0.0116	0.2167	0.4913	0.5007	0.3244	4949
74v	0.0081	0.3603	0.6064	0.6008	0.4414	1734
74t	0.0098	0.3337	0.6147	0.6145	0.4484	1734
110v	0.0076	0.3984	0.6406	0.6293	0.4776	1149
110t	0.0093	0.3669	0.6524	0.6445	0.4867	1149
147v	0.0071	0.4369	0.6647	0.6487	0.4979	886
147t	0.0080	0.4555	0.7140	0.6964	0.5672	886
400v	0.0066	0.4816	0.7019	0.6782	0.5338	344
400t	0.0080	0.4584	0.7158	0.6966	0.5605	344
700v	0.0064	0.4925	0.7070	0.6832	0.5307	211
700t	0.0076	0.4857	0.7196	0.7001	0.5586	211
1000v	0.0062	0.5071	0.7132	0.6865	0.5447	198
1000t	0.0074	0.4958	0.7264	0.7047	0.5726	198
2000v	0.0063	0.5020	0.7090	0.6843	0.5447	189
2000t	0.0075	0.4948	0.7236	0.7027	0.5702	189
4000v	0.0063	0.4995	0.7092	0.6860	0.5442	194
4000t	0.0076	0.4819	0.7231	0.7037	0.5709	194
5500v	0.0063	0.5029	0.7097	0.6833	0.5444	197
5500t	0.0074	0.4971	0.7246	0.7017	0.5758	197
7000v	0.0065	0.4868	0.6986	0.6744	0.5369	188
7000t	0.0075	0.4917	0.7127	0.6929	0.5723	188
9500v	0.0065	0.4894	0.7019	0.6792	0.5326	190
9500t	0.0074	0.4965	0.7149	0.6957	0.5724	190
12000v	0.0066	0.4815	0.7001	0.6753	0.5239	186
12000t	0.0080	0.4581	0.7142	0.6926	0.5621	186
15000v	0.0065	0.4867	0.6985	0.6737	0.5348	194
15000t	0.0078	0.4719	0.7086	0.6879	0.5630	194

All of the code used in this thesis and pretrained models can be found on Github [StH25].