Deep Learning Approach to Recognize Genome Functional Elements Using Diverse Genomic Data

Nazar Beknazarov[a], Seungmin Jin[a] and Maria Poptsova[a]

[a] *Laboratory of Bioinformatics, Faculty of Computer Science, National Research University Higher School of Economics, 11 Pokrovsky boulvar, Moscow, Russia 101000*

Abstract

As a result of the revolution in genome sequencing a lot of -omics data were generated. After obtaining a primary genomic sequence the next major task is to study genomic regulatory code. Epigenetic data sets provide a hint of how regulatory patterns are distributed in different tissues. Other layer of genome regulatory code comprises DNA secondary structures, which can work as regulators of various genomic processes. Having Big Data from next-generation sequencing experiments, machine learning approaches were chosen to solve the task of recognizing genomic functional elements. The earlier attempts to solve the problems of genome annotation with different classes of functional ele-ments, i.e. nucleosomic DNA, exon-intron boundaries, enhancers used machine learning algorithms that required manual collection of different features needed to characterize genomic regions. Lately deep learning approaches including convolution neural networks and recurrent neural networks become successful in recognizing genomic functional elements based on sequence information on-ly and/or with additional information on epigenetics and known regulatory ele-ments. Here we discuss a deep learning approach and provide an example of building a deep learning model for the task of recognition of DNA secondary structures.

Keywords

DNA secondary structures, histone code, histone marks, epigenetics, machine learning, deep learning, convolutional neural networks, recurrent neural networks

1. Introduction

Deep learning is becoming popular and easy to apply in solving various tasks. Among them, CNN (Convolutional Neural Network) and RNN (Recurrent Neural Network) are the most popular deep learning architectures, which may show the state-of-the-art performance in the majority of applications [1]. This is achieved by the combination of the top performance in spatial and temporal dimen-sions. CNN may capture the hierarchical information in space. The mechanism of CNN is essentially in exploring a region of the input, one at a time, and mapping it to a specific feature space. By generating a series of convolutions at each region the network may learn the space features hierarchically [2]. For instance, for the task of face recognition, CNN starts to gather convolutions from lines or cir-cles in face images, and then it filters these features for building up the feature maps of nose, eyes, and ears, and finally it recognizes the face [3].

RNN can learn temporal order using its context, and additionally, being turing-complete, it may learn, theoretically, any kind of function [4]. Essentially RNN model keeps passing the context vector, which compresses the in-formation at a certain time step to predict outcome in the future time steps. It means RNN may handle arbitrary length of input [5]. This feature makes RNN useful in many sequential tasks, such as machine learning translation, time series prediction, speech recognition, and signal processing. However, in practice RNN does n ot work well alone, especially for the feature

Modeling and Analysis of Complex Systems and Processes - MACSPro'2020, October 22–24, 2020, Venice, Italy & Moscow, Russia
EMAIL: nazar.s.beknazarov@gmail.com (A. 1); mpoptsova@hse.ru (A. 3)
ORCID: 0000-0002-7198-8234 (A. 3);

extraction and long term prediction tasks [4, 5]. This is why modulating CNN and RNN is a common practice and shows the best results in deep learning tasks [6-8].

In Bioinformatics, research in deep learning has been rapidly increasing since early 2000s and CNN and RNN are widely applied to various tasks [6]. For example, CNN applied to predict gene expression from epigenomic data, anomaly classification in biomedical imaging, brain decoding in biomedical signal processing [6]. RNN also was applied to protein structure classification, and anomaly classification in biomedical signal processing. Although combining two models in practice shows good performance, there is a tendency to use them separately in bioinformatics tasks [6]. One of the pioneering example of hybrid CNN and RNN model to predict function of the DNA sequence was implemented and tested in DanQ [7]. Another hybrid CNN-RNN model was applied for a task of predicting enhancers based on histone modification marks [8]. In this research, we continue testing deep learning approach combining two models to recognize genome functional elements using diverse genomic data.

As a genomic functional element we chose Z-DNA belonging to DNA secondary structures. The role of DNA secondary structures in the regulation of genomic processes was confirmed experimentally for quadruplexes, cruciform structures, triplexes, and Z-DNA. Experiments on whole-genome detection of Z-DNA regions are under development, and currently several experimental datasets are available [9, 10]. Building and testing machine learning models that would aggregate information from experimental data is an urgent task, since there is a need for computer methods of genome annotation with functional elements. Here we tested several machine learning approaches including deep learning to detect Z-DNA regions. We showed that deep learning, and specifically hybrid CNN plus RNN models achieved the best performance in the task of Z-DNA recognition.

2. Material and Methods

2.1. Data on Z-DNA, epigenetics, RNA polymerase, and transcription factor binding sites

The positions of Z-DNA are taken from the dataset of the Chip-Seq experiment on identification of binding sites of the Zaa protein, which binds to the left-twisted form of DNA [10]. To improve the prediction quality of the sequence we added information on epigenetic and regulatory code. Histone marker positions and DNase hypersensitivity sites, which mark regions of an open chromatin, are taken from the international consortium project Roadmap Epigenomics [11]. Information on the binding sites of RNA polymerase and transcription factors are taken from the Encyclopedia of DNA elements (ENCODE) project [12]. Totally, 1065 features are selected.

DNA subsequence with Z-DNA regions is considered as an output vector. A binary value is assigned to every nucleotide depending on its location inside the Z-DNA region. We considered subsequences of 5000 bp, thus, every output vector has a length of 5000.

2.2. Construction of train and test datasets

We encoded human DNA sequence using one hot encoding method where a sequence is transformed to a binary matrix of 4xL where L is the length of the sequence and 4 rows correspond to the 4 nucleotides, TCAG. This matrix is filled with zeros and has only one value at the corresponding nucleotide cell in each position. Epigenomic data and RNA polymerase and transcription factors binding sites were added to the encoded DNA sequence. Finally, we create a set of matrices for every chromosome, which has the same length of DNA sequence. The shape of input matrix is 1069xL, where 1064 comes from additional features and 4 from one-hot encoded DNA, and L is the length of the sequence. In order to avoid any dependencies between Z-DNA sites and borders of DNA subsequences, DNA is uniformly divided into subsequences of length 5000. Then we split

subsequence into train and test sets in a ratio of 4 to 1 respectively preserving the proportion of subsequences with Z-DNA in each set.

2.3. Machine learning models

2.3.1. Baseline model

In order to show the level of performance of deep learning models, we prepared a boosting classifier as a baseline. The term 'boosting' here means that it converts weak learners to strong learners. Basically, boosting is an ensemble method for improving the model predictions of any given learning algorithm. This method consists of sequential training of simple models, where each subsequent model corrects the errors of the previous one. Boosting is a well-known method in the bioinformatics domain and generally shows good results in many classification tasks [13-15].

2.3.2. Deep learning models

DNA has patterns in the form of one-dimensional sequence motifs, which CNN may capture very well, and, from the other hand, DNA is a text, so RNN may learn the context from it. Therefore, we expect the best result when we combine two models, CNN and RNN. For the proper comparison, we also trained independent CNN along with CNN + RNN.

2.3.3. CNN

We experimented with several hyperparameters for CNN models. We considered different sizes of the kernels and strides because it may influence the result. The number of output kernels was set to 1 and we use a softmax layer at the end. Thus, these models have a vector of outcome with length of input, each nucleotide corresponds to a probability value from 0 to 1. For each nucleotide, there are C boolean values, where C is kernel size. Every boolean value depicts the presence of Z-DNA in this very point. Averaging on these C values was used as a target for the outcome cell. Since the padding is absent, the number of outcomes of the models equals the number of averaged values. That means each model will predict the average number of nucleotides that occurred in a given segment, and assign this number to the middle of the segment. Increasing layer number or kernel size make worse its complexity but may have better results. Next set of models has more convolutional layers with ReLU activation. In this case, the target variable is calculated in a slightly different way. Averaging is performed by the size of the last layer. The size and number of kernels on the first and second layers were selected from a predefined set of values.

2.3.4. CNN+RNN

This type of hybrid model was successfully implemented in the DanQ [7]. CNN extracts important motifs and simultaneously RNN can learn complex regulatory grammar between the motifs. It is assumed that the motifs that were detected by the CNN layer also have recurrent dependencies. In theory, such a network is able to recognize a succession of motifs on which Z-DNA configuration depends. The model architecture used for Z-DNA detection is shown in Fig. 1.

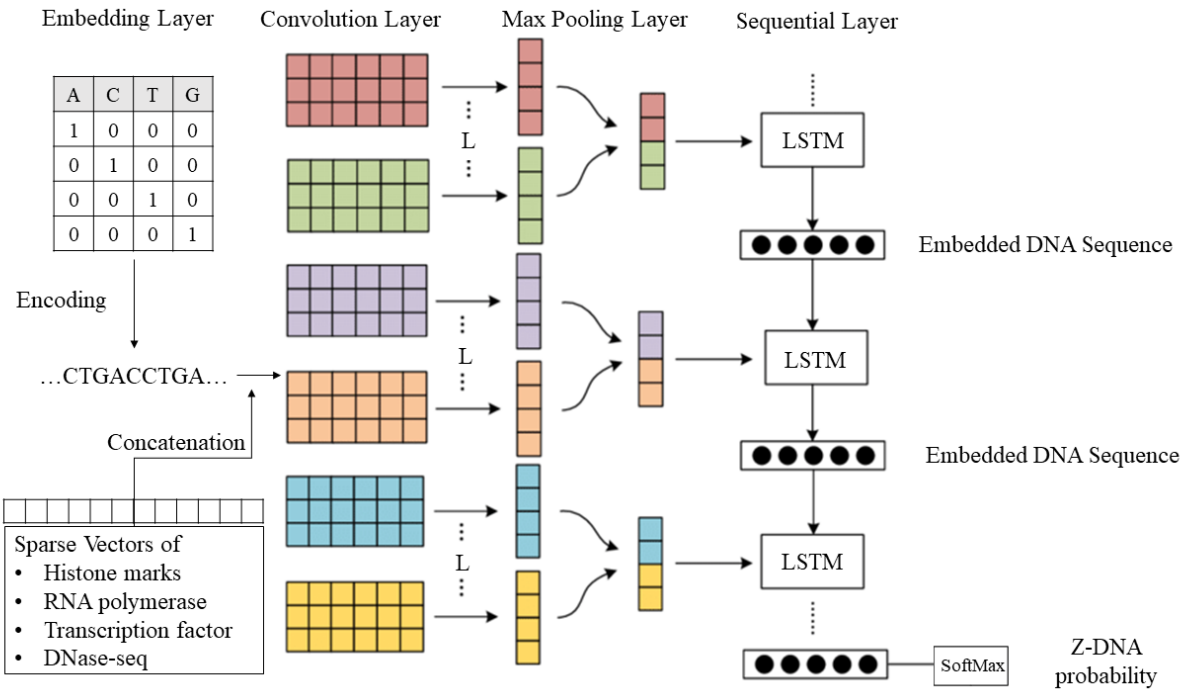

Figure 1: Architecture of a hybrid model, CNN + RNN for Z-DNA prediction. DNA sequence data transformed with one-hot encoding was concatenated with sparse vectors of epigenomic data.

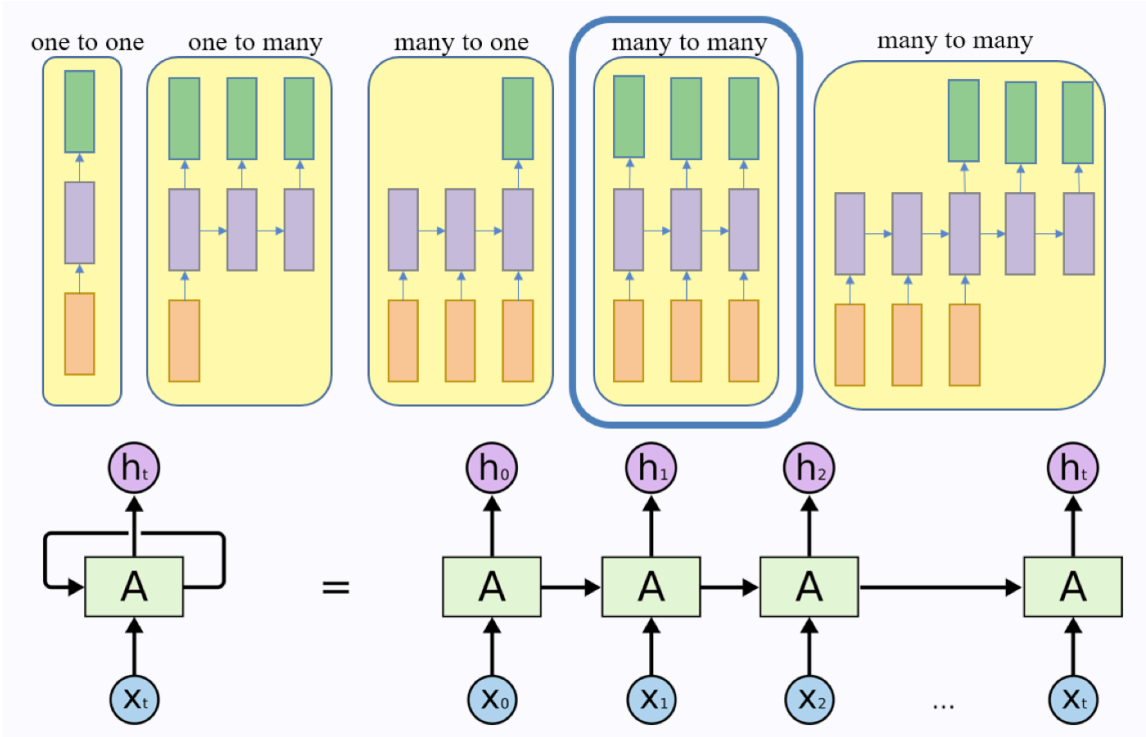

Figure 2: Schematic representation of approaches for the classification using RNN architecture.

There are several ways to use RNN: one-to-one, one-to-many, many-to-one, and many-to-many (Fig. 2). In this paper, we considered two approaches, many-to-many and many-to-one.

2.3.5. Approach many-to-one

In this case, the structure of a model is as follows. The first part of the model is one or several CNN layers, and each column of the received out-put is separately transferred to the RNN network. In our case, a multi-layer bidirectional LSTM is selected for RNN. Next, the number of layers in the CNN and LSTM parts will be selected. The sizes of kernels and hidden layers will be selected. At the end and beginning of the sequence, the RNN layer will output 2 vectors that are associated with long-term LSTM memory cells. Two LSTM context vectors were included since this RNN model is bidirectional. Then the vectors are passed to the fully connected layer, which makes the prediction. The target variable is a boolean value of Z-DNA presence in the region in this sequence.

2.3.6. Approach many-to-many

This architecture completely copies the previous one, except for one element. After the RNN layer, the output of the long-term memory element is ignored and the short-term memory outputs of each direction are aggregated. Next, each unit of the sequence corresponds to two vectors, which are passed to the fully connected layer and then predictions are made for each part of the sequence. The target variable in this case will be calculated exactly as in the case of CNN. That is, each unit of the sequence will be mapped to the average of a certain region of the chain.

3. Results

Quantiles were calculated for the distribution of random AUC using bootstrap sampling (Table 1). You can see that the first model has a rather low quality, indistinguishable from that of a random choice. The best CNN model among all showed 69 AUC on test set. The architecture can be listed as follows. For the best CNN model, the first layer is a convolutional layer with 36 kernels, kernels size 13, stride 2 and padding 6. Second layer is a ReLU. Third layer is a convolutional layer with 2 kernels, kernels size 13, stride 2 and padding 6. Last layer is a Sigmoid. The performance of the hybrid CNN+RNN showed quality higher than CNN model.

Table 1
Experiment result

Model	AUC	Accuracy
Boosting	0.532	0.691
CNN	0.69	0.55
CNN+RNN	0.865	0.75

Best model with a many-to-one approach showed 86.5 AUC. The architecture of the best CNN+RNN model can be listed as follows. The first layer is a convolutional layer with 64 kernels, kernels size 13, stride 4 and padding 6.Second layer is a ReLU. Output of ReLU was sent to bidirectional LSTM layer with hidden size 64 and 2 layers. Hidden state of LSTM goes to the dropout layer with probability 0.7. Last fully connected layer has 2 neurons.

The best model with a many-to-many approach showed 80.5 AUC. First layer is a convolutional layer with 36 kernels, kernels size 25, stride 2 and padding 12. Second layer is a ReLU. Third layer is a convolutional layer with 64 kernels, kernels size 25, stride 2 and padding 12.Fourth layer is a ReLU. Output of ReLU was sent to bidirectional LSTM layer with hidden size 64 and 2 layers. Hidden state of LSTM goes to the dropout layer with probability 0.7. Last fully connected layer has 2 neurons.

4. Conclusions and Discussion

The following conclusions can be drawn from the obtained results. Although CNN model shows higher performance than the baseline, it does not handle the sequential nature of DNA sequence. Baseline and CNN models perform much worse than a model that contains an RNN layer. The maximum quality that can be achieved on this dataset with the power of this set of architectures does not exceed 86 % of the AUC, which indicates that the task can be solved using available data.

Here we presented results of a deep learning approach for the Z-DNA prediction, in particular a hybrid model of two famous deep learning network architectures - CNN and RNN. This architecture outperforms both models based only on CNN and classical machine learning models such as gradient boosting. As we expected CNN + RNN shows better results than CNN because RNN may capture the sequential pattern using its context. We assume our approach may be applied to many other bioinformatics tasks, which are required for mapping spatial data to sequential output.

One of the advantages of our approach is scalability, where we can upgrade the system when more epigenetics and regulatory data become available. Thus, the same type of models can be applied to recognition of quadruplexes or triplexes as well as patterns of association of DNA secondary structures and epigenetic code. We expect that inclusion of omics data will improve prediction quality of the model. However there is a drawback in having a large feature space that will increase the time of mod-el training. It would be beneficial first to find a minimal set that would achieve the desired model quality and then train the model with the reduced size of feature space. It will also help to find scientifically important associations between studied functional and epigenetic and/or regulatory elements.

Deep neural networks are capable of processing effectively aggregated information from different levels of genome organization. At the present time, when next-generation sequencing experiments are still too expensive, machine learning models for annotating genomes with functional genomic elements are very important. For some species next-generation sequencing experiments on epigenomic and regulatory code are not available at all. Finding de novo or imputed novel functional elements with computational artificial intelligence systems would help researchers in understanding principles and mechanisms of genome functioning.

5. References

[1] Meireles, M.R.G., Almeida, P.E.M., Simoes, M.G.: A comprehensive review for industrial applicability of artificial neural networks. IEEE Transactions on Industrial Electronics 50, (2003) 585-601.

[2] LeCun, Y., Haffner, P., Bottou, L., Bengio, Y.: Object recognition with gradient-based learning In: Forsyth, D.A., Mundy, J.L., Gesú, V.d., Cipolla, R. (eds.) Shape, contour and grouping in computer vision, pp. 319-345. Springer, Berlin, Heidelberg (1999.)

[3] Hu, G., Yang, Y., Yi, D., Kittler, J., Christmas, W., Li, S.Z., Hospedales, T.: When face recognition meets with deep learning: an evaluation of convolutional neural networks for face recognition. . Proceedings of the IEEE international conference on computer vision workshops. (2015)
142-150.

[4] Goodfellow, I., Bengio, Y., Courville, A.: Deep learning. MIT press (2016).

[5] Hochreiter, S., Schmidhuber, J.: Long short-term memory. . Neural computation 9(8), (1997) 1735-1780.

[6] Fan, Y., Lu, X., Li, D., Liu, Y., : Video-based emotion recognition using CNN-RNN and C3D hybrid networks. Proceedings of the 18th ACM International Conference on Multimodal Interaction (2016) 445-450.

[7] Wang, J., Yang, Y., Mao, J., Huang, Z., Huang, C., Xu, W.: Cnn-rnn: A unified framework for multi-label image classification. Proceedings of the IEEE conference on computer vision and pattern recognition (2016) 2285-2294.

[8] Selvin, S., Vinayakumar, R., Gopalakrishnan, E.A., Menon, V.K., Soman, K.P., : Stock price prediction using LSTM, RNN and CNN-sliding window model. international conference on advances in computing, communications and informatics (icacci), IEEE (2017) pp. 1643-1647.
[9] Min, S., Lee, B., Yoon, S.: Deep learning in bioinformatics. Brief Bioinform 18, (2017) 851-869.
[10] Quang, D., Xie, X.: DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. Nucleic Acids Res (2016) 44, e107.
[11] Lim, A., Lim, S., Kim, S.: Enhancer prediction with histone modification marks using a hybrid neural network model. Methods 166, (2019) 48-56.
[12] Kouzine, F., Wojtowicz, D., Baranello, L., Yamane, A., Nelson, S., Resch, W., Kieffer-Kwon, K.R., Benham, C.J., Casellas, R., Przytycka, T.M., Levens, D.: Permanganate/S1 Nuclease Footprinting Reveals Non-B DNA Structures with Regulatory Potential across a Mammalian Genome. Cell Syst 4, (2017) 344-356 e347.
[13] 13. Shin, S.I., Ham, S., Park, J., Seo, S.H., Lim, C.H., Jeon, H., Huh, J., Roh, T.Y.: Z-DNA-forming sites identified by ChIP-Seq are associated with actively transcribed regions in the human genome. DNA Res (2016).
[14] 14. Roadmap Epigenomics, C., Kundaje, A., Meuleman, W., Ernst, J., Bilenky, M., Yen, A., Heravi-Moussavi, A., Kheradpour, P., Zhang, Z., Wang, J., Ziller, M.J., Amin, V., Whitaker, J.W., Schultz, M.D., Ward, L.D., Sarkar, A., Quon, G., Sandstrom, R.S., Eaton, M.L., Wu, Y.C., Pfenning, A.R., Wang, X., Claussnitzer, M., Liu, Y., Coarfa, C., Harris, R.A., Shoresh, N., Epstein, C.B., Gjoneska, E., Leung, D., Xie, W., Hawkins, R.D., Lister, R., Hong, C., Gascard, P., Mungall, A.J., Moore, R., Chuah, E., Tam, A., Canfield, T.K., Hansen, R.S., Kaul, R., Sabo, P.J., Bansal, M.S., Carles, A., Dixon, J.R., Farh, K.H., Feizi, S., Karlic, R., Kim, A.R., Kulkarni, A., Li, D., Lowdon, R., Elliott, G., Mercer, T.R., Neph, S.J., Onuchic, V., Polak, P., Rajagopal, N., Ray, P., Sallari, R.C., Siebenthall, K.T., Sinnott-Armstrong, N.A., Stevens, M., Thurman, R.E., Wu, J., Zhang, B., Zhou, X., Beaudet, A.E., Boyer, L.A., De Jager, P.L., Farnham, P.J., Fisher, S.J., Haussler, D., Jones, S.J., Li, W., Marra, M.A., McManus, M.T., Sunyaev, S., Thomson, J.A., Tlsty, T.D., Tsai, L.H., Wang, W., Waterland, R.A., Zhang, M.Q., Chadwick, L.H., Bernstein, B.E., Costello, J.F., Ecker, J.R., Hirst, M., Meissner, A., Milosavljevic, A., Ren, B., Stamatoyannopoulos, J.A., Wang, T., Kellis, M.: Integrative analysis of 111 reference human epigenomes. Nature 518, (2015) 317-330.
[15] 15. Davis, C.A., Hitz, B.C., Sloan, C.A., Chan, E.T., Davidson, J.M., Gabdank, I., Hilton, J.A., Jain, K., Baymuradov, U.K., Narayanan, A.K., Onate, K.C., Graham, K., Miyasato, S.R., Dreszer, T.R., Strattan, J.S., Jolanki, O., Tanaka, F.Y., Cherry, J.M.: The Encyclopedia of DNA elements (ENCODE): data portal update. Nucleic Acids Res 46, (2018) D794-D801.
[16] 16. Hothorn, T., Buhlmann, P.: Model-based boosting in high dimensions. Bioinformatics 22, (2006) 2828-2829.
[17] 17. Dettling, M., Buhlmann, P.: Boosting for tumor classification with gene expression data. Bioinformatics 19, (2003) 1061-1069.
[18] 18. Eickholt, J., Cheng, J.: Predicting protein residue-residue contacts using deep networks and boosting. Bioinformatics 28, (2012) 3066-3072.