

Improved Architecture of Artificial Neural Network for Secondary Structure Analysis*

Semyon Grigorev, Polina Lunina

Saint Petersburg State University

7/9 Universitetskaya nab., St. Petersburg, 199034, Russia

JetBrains Research

Primorskiy prospekt 68-70, Building 1, St. Petersburg, 197374, Russia

semyon.grigorev@jetbrains.com, lunina.polina@mail.ru

Algorithms that can efficiently perform sequences classification and sub-sequences detection have recently become a focus in bioinformatics and many of them utilize the idea about considering these sequences secondary structure [1, 2, 3, 4]. One of the classical ways of describing secondary structure is formal grammars.

An approach for biological sequences processing by combination of formal grammars and neural networks is proposed in the work [5]. While classical way is to model secondary structure of the full sequence by using grammar, the proposed approach utilizes it only for primitive secondary structure features description. These features can be extracted by parsing algorithm and processed by using artificial neural network. It is shown that this approach is applicable for real-world data processing and some questions are formulated for future research. In this work we provide answers to some of them.

The first question is whether it is possible to use convolutional neural networks for parsing result processing. The result of matrix-based parsing algorithm for an input string and fixed nonterminal is an upper triangular boolean matrix. Presently, we came up with two possible ways of these

*The research was supported by the Russian Science Foundation grant 18-11-00100 and a grant from JetBrains Research.

matrices representation. The first one is to drop out the bottom left triangle and vectorize the rest of matrix row by row. It requires the equal length of the input sequences, therefore we propose to either cut sequences or add some special symbol till the definite length. The second way is to represent this matrix as an image: the false bits of matrix as white pixels and the true bits as black ones. This approach makes it possible to process sequences with different length since the images may be easily transformed to the same size. To handle these images we use network with a small number of convolutional layers, linearization and then the same structure as for vectorized data (dense and dropout layers with batch normalization).

The second question is whether it is possible to move parsing to network training step. This question is important because parsing is the most time-consuming operation of the proposed solution. We solve this problem by using two-staged learning. At the first step, we prepare a neural network (vector- or image-based) for our task which takes parsed data as an input. After that we extend trained network with a number of input layers that should convert original nucleotide sequence into parsing result. This way we create a network which can handle sequences, not parsing result. So, parsing is required only for training the first network.

We use the proposed improvements to create neural networks for tRNA sequences analysis problems: classification of tRNA into 2 classes: eukaryotes and prokaryotes (EP) and 4 classes: archaea, bacteria, plants and fungi (ABFP). We train networks on sequences from databases [6, 7]. Results for both image- and vector-based classifiers are presented in the table 1, where base model means network which handles parsing result and extended model handles sequences and is based on the corresponding base model.

Classifier	EP		ABFP	
Approach	Vector-based	Image-based	Vector-based	Image-based
Base model accuracy	94.1%	96.2%	86.7%	93.3%
Extended model accuracy	97.5%	97.8%	96.2%	95.7%
Total samples (train:valid:test)	20000:5000:10000		8000:1000:3000	

Table 1: Test results

References

- [1] Rivas E, Eddy S.R. *The language of RNA: a formal grammar that includes pseudoknots* // Bioinformatics. — 2000.
- [2] Knudsen Bjarne, Hein Jotun. *RNA secondary structure prediction using stochastic context-free grammars and evolutionary history.* //Bioinformatics (Oxford, England).— 1999.— Vol. 15, no. 6.— P. 446–454.
- [3] Yuan C. et al. *Reconstructing 16S rRNA genes in metagenomic data* //Bioinformatics. — 2015. — №. 12. — P. 135-143.
- [4] Dowell R. D., Eddy S. R. *Evaluation of several lightweight stochastic context-free grammars for RNA secondary structure prediction* //BMC bioinformatics.— 2004.— №. 1.— P. 71.
- [5] Grigorev S., Lunina P. *The Composition of Dense Neural Networks and Formal Grammars for Secondary Structure Analysis*
- [6] Genomic tRNA Database. — URL: <http://gtrnadb2009.ucsc.edu/> (online; accessed: 17.05.2019).
- [7] tRNADB-CE. — URL: <http://trna.ie.niigata-u.ac.jp/cgi-bin/trnadb/index.cgi> (online; accessed: 17.05.2019).