Authors' Instructions: Preparation of Camera-Ready Contributions to SCITEPRESS Proceedings

First Author Name[1], Second Author Name[1] and Third Author Name[2]

[1]*Institute of Problem Solving, XYZ University, My Street, MyTown, MyCountry*

[2]*Department of Computing, Main University, MySecondTown, MyCountry*

{*f_author, s_author*}*@ips.xyz.edu, t_author@dc.mu.edu*

Keywords: The paper must have at least one keyword. The text must be set to 9-point font size and without the use of bold or italic font style. For more than one keyword, please use a comma as a separator. Keywords must be titlecased.

Abstract: Abstract is very abstract

1 INTRODUCTION

Algorithms that can efficiently and accurately identify and classify bacterial taxonomic hierarchy have become a focus in computational genetics. The idea that secondary structure of genomic sequences is sufficient for solving the detection and classification problems lies at the heart of many tools (Rivas and Eddy, 2000; Knudsen and Hein, 1999; Yuan et al., 2015; Dowell and Eddy, 2004). The secondary structure can be specified in terms of formal grammars. The sequences obtained from the real bacteria usually contain a huge number of mutations and "noise" which renders precise methods impractical. Probabilistic grammars and covariance models (CMs) are a way to take the noise into account (Durbin et al., 1998). For example, CMs are successfully used in the Infernal tool.Neural networks is another way to deal with "noisy" data. The works (Sherman, 2017; Higashi et al., 2009) utilize neural networks for 16s rRNA processing and demonstrate promising results.

2 PROPOSED SOLUTION

We combine neural networks and ordinary context-free grammars to detect genomic sequences. We extract features by using the ordinary (not probabilistic) context-free grammar and use the dense neural network for features processing. Features can be extracted by any parsing algorithm and then presented as a boolean matrix M such that $M[i, j] = 1$ iff $S \Rightarrow_G^* w[i, j]$ where w is the input sequence and G is context-free grammar with the start nonterminal S.

2.1 Grammar

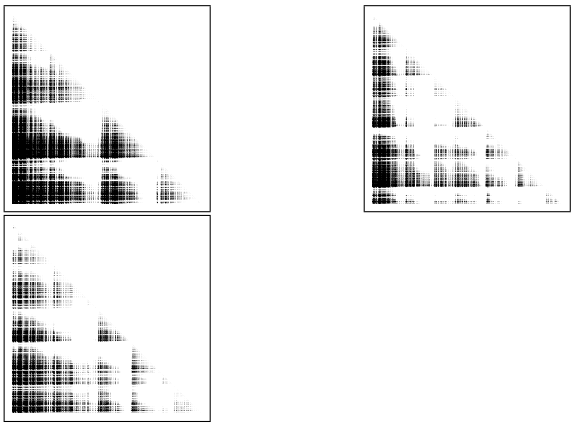
An example of grammar which we use in our experiments is presented in figure 1.

2.2 Matrices

We use line-by-line compressed matrix representation: sequence of 32 cells (bits) is compressed to unsigned integer. Top right triangle of matrix is always empty, so can be ignored. We hope that compression to 16 bit integer or byte may decrease complexity of neural network and improve result quality, but it requires significantly more memory on GPGPU which can be serious technical problem.

```
s1: stem<s0> any

a_0_7 : any*[2..10]

s0: a_0_7 | a_0_7 stem<s0> s0

any: A | U | C | G

stem1<s>: A s U | G s C | U s A | C s G

stem2<s>: stem1< stem1<s> >

stem<s>:
      A stem<s> U
    | U stem<s> A
    | C stem<s> G
    | G stem<s> C
    | stem1< stem2<s> >
 }
```

Figure 1: Grammar for secondary structure features extraction



2.3 Newral Network

We use dense neural network with 14 dense layers. Almost all of them are wrapped with dropout (up to 75%) and batch normalization layers for learning stabilization.

3 EVALUATION

We evaluate the proposed approach for 16s rRNA detection. We specify context-free grammars which detect stems with the hight of more than two pairs and their arbitrary compositions. For network training we use dataset consisting of two parts: random subsequences of 16s rRNA sequences from the Green Genes database form positive examples, while the negative examples are random subsequences of full genes from the NCBI database. All sequences have the length of 512 symbols, totally up to 310000 sequences. After training, current accuracy is 90% for validation set (up to 81000 sequences), thus we conclude that our approach is applicable.

4 FUTURE WORK

The presented is a work in progress. The ongoing experiment is finding all instances of 16s rRNA in full genomes. Also we plan to use the proposed approach for the filtration of chimeric sequences and the classification. Composition of our approach with other methods and tools as well as grammar tuning and detailed performance evaluation may improve the applicability for the real data processing.

5 DISCUSSION

Protenomics
Different letghts
Embedding: string -¿ matrix
Other types of NNs

ACKNOWLEDGEMENTS

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

REFERENCES

Dowell, R. D. and Eddy, S. R. (2004). Evaluation of several lightweight stochastic context-free grammars for rna secondary structure prediction. *BMC bioinformatics*, 5(1):71.

Durbin, R., Eddy, S. R., Krogh, A., and Mitchison, G. (1998). *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge university press.

Higashi, S., Hungria, M., and Brunetto, M. (2009). Bacteria classification based on 16s ribosomal gene using artificial neural networks. In *Proceedings of the 8th WSEAS International Conference on Computational intelligence, man-machine systems and cybernetics*, pages 86–91.

Knudsen, B. and Hein, J. (1999). Rna secondary structure prediction using stochastic context-free grammars and evolutionary history. *Bioinformatics (Oxford, England)*, 15(6):446–454.

Rivas, E. and Eddy, S. R. (2000). The language of rna: a formal grammar that includes pseudoknots. *Bioinformatics*, 16(4):334–340.

Sherman, D. (2017). Humidor: Microbial community classification of the 16s gene by training cigar strings with convolutional neural networks.

Yuan, C., Lei, J., Cole, J., and Sun, Y. (2015). Reconstructing 16s rrna genes in metagenomic data. *Bioinformatics*, 31(12):i35–i43.