# *SpeciesMLP: Sequence based Multi-layer Perceptron For Amplicon Read Classification Using Real-time Data Augmentation*

**Ali Kishk, Mohamed El-Hadidi***

Bioinformatics Research Group, Center for Informatics Science (CIS), Nile University, Giza, Egypt

* Corresponding Author. Email: melhadidi@nu.edu.eg

## *Abstract:*

Taxonomic assignment is the core of targeted metagenomics approaches that aims to assign sequencing reads to their corresponding taxonomy. Sequence similarity searching and machine learning (ML) are two commonly used approaches for taxonomic assignment based on the 16S rRNA. Similarity based approaches require high computation resources, while ML approaches don't need these resources in prediction. The majority of these ML approaches depend on k-mer frequency rather than direct sequence, which leads to low accuracy on short reads as k-mer frequency doesn't consider k-mer position. Moreover training ML taxonomic classifiers depend on a specific read length which may reduce the prediction performance by decreasing read length. In this study, we built a neural network classifier for 16S rRNA reads based on SILVA database (version 132). Modeling was performed on direct sequences using Convolutional neural network (CNN) and other neural network architectures such as Multi-layer Perceptron and Recurrent Neural Network. In order to reduce modeling time of the direct sequences, In-silico PCR was applied on SILVA database. Total number of 14 subset databases were generated by universal primers for each single or paired high variable region (HVR). Moreover, in this study, we illustrate the results for the V2 database model on 8443 classes on the species level and 1552 on the genus level. In order to simulate sequencing fragmentation, we trained variable length subsequences from 50 bases till the full length of the HVR that are randomly changing in each training iteration. Simple MLP model with global max pooling gives 0.71 & 0.93 test accuracy for the species and genus levels respectively (for reads of 100 base sub-sequences) and 0.75 & 0.96 accuracy for the species and genus levels respectively (on the full length V2 HVR). In this study, we present a novel method (SpeciesMLP https://github.com/ali-kishk/SpeciesMLP) to model the direct amplicon sequence using MLP over a sequence of k-mers faster 20 times than CNN in training and 10 times in prediction.

***Keywords—Taxonomic Classification, 16S rRNA, Metagenomics, Multi-Layer Perceptron***

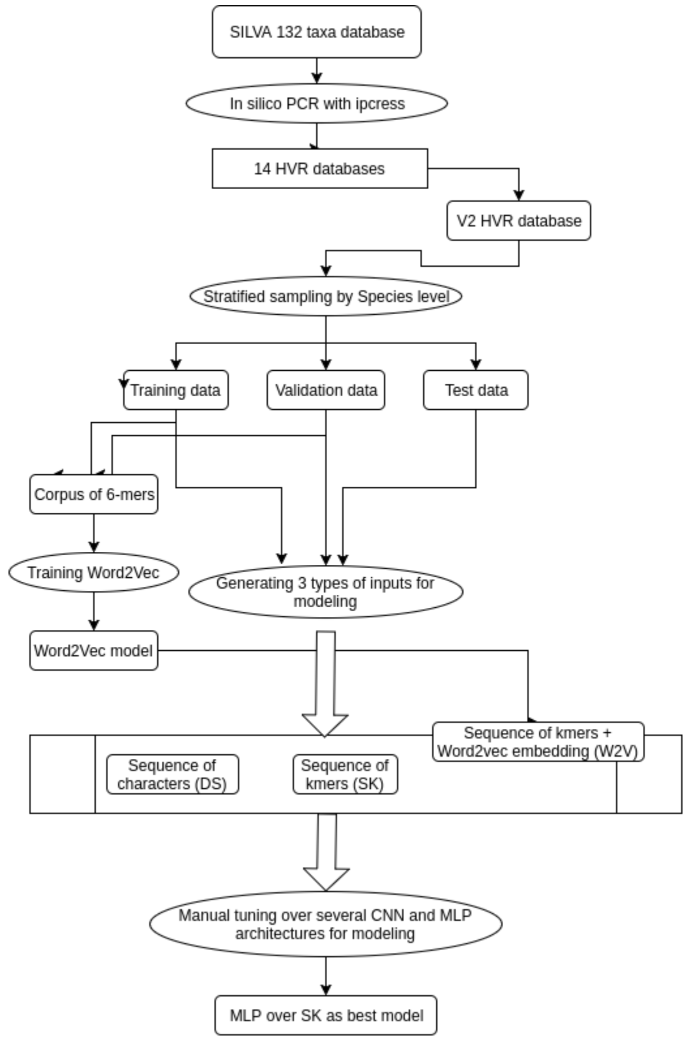
SILVA 132 taxa database
In silico PCR with ipcress
14 HVR databases
V2 HVR database
Stratified sampling by Species level
Training data
Validation data
Test data
Corpus of 6-mers
Training Word2Vec
Generating 3 types of inputs for modeling
Word2Vec model
Sequence of characters (DS)
Sequence of kmers (SK)
Sequence of kmers + Word2vec embedding (W2V)
Manual tuning over several CNN and MLP architectures for modeling
MLP over SK as best model

*Figure 1: Workflow diagram.*

The diagram begins with Insilico PCR to SILVA taxa. The rest pipeline is applied only to the V2 HVR database only from splitting to modeling.

## I. BACKGROUND

Metagenomics is the study of a microbial communities' genetic fingerprint. Recently it was shown that these communities contribute to many biological functions such as antibiotic resistance [1], metabolism [2] and immunity regulation [3]. Regarding its role in the environment, they have important roles in climate change [4], the space environment [5], water biofilm [6] and nuclear waste detoxification [7]. As metagenomics is changing our insight on health and the environment, new sequence analysis methods are becoming more important.

Metagenomics data are generated from two main sequencing techniques, targeted sequencing [8], and whole genome shotgun sequencing [8]. Targeted or amplicon sequencing depends on PCR amplification of the rRNA genes especially 16S rRNA for bacteria and archaea, while shotgun sequencing uses the sequence of the whole genome. Amplicon sequencing allows cheap and rapid identifications of the microbial species whereas shotgun can define the new microbial content and allow another dimension for functional classification [8].

Machine learning is the ability of computers to learn from the data without any stored instructions. ML algorithms are mainly two types: supervised and unsupervised. Supervised ML is modeling data with known outputs feature for each sample where unsupervised ML don't. Supervised ML tries to map an output to a set of inputs, where regression is a supervised ML in case the input is numerical, whereas classification in case of categorical input. Deep learning (DL) is a subtype of ML in both supervised and unsupervised, where it utilizes the connectivity and architecture of the human nervous system for modeling. The revival of DL allowed modeling of complex natural data such images, audio, and text, where it was hard to model using classical ML algorithms such as Random Forest [9] and Support Vector Machine [10].

## II. Introduction

### A. Metagenomics analysis steps

Both whole genome shotgun and amplicon sequencing methods share some bioinformatics algorithms in analysis such as quality processing and taxonomic assignment. Taxonomic assignment [11] is usually the final step of metagenomic analysis and probably the most computationally demanding. It involves assigning each read in each sample to a corresponding taxonomy on many levels from phylum to species or even strain level.

### B. Taxonomic assignment

Taxonomic assignment uses alignment or machine learning for classification. Alignment based approaches such as BLAST [12], MALT [13], and SINA [14] need a reference 16S rRNA database for sequence comparison. ML-based approaches such as RDP [15], 16S Classifier [16] need first to be trained on the reference database to produce a classifier. Once trained, these classifier models can be used for taxonomic assignment. Alignment based approaches are generally more accurate than ML-based but at the expense of the computational resources [17].

### C. Deep Learning in Bioinformatics

DL application covers many bioinformatics fields including genomic [18], transcriptomics [19]. metagenomics [20] and other OMICs fields. Data used in these applications can be classified into a feature-based or direct sequence based. Phenotype classification such as classifying samples to healthy and diseased is the most widespread application of feature-based DL. Phenotype classification can be found in genomics from mutation data [18], transcriptomics from differentially expressed genes [19] and metagenomics from Operating Taxonomy Unit (OTU) tables [21] or k-mer frequency [20]. The ability of DL to model thousands to millions of features allowed direct modeling without feature selection. Multi-layer perceptron (MLP) [22] is the DL architecture commonly used in phenotype classification [20].

### D. Deep Learning in Metagenomics

Phenotype classification and gene ontology prediction [23] are among commonly used DL applications in metagenomics. All phenotype classification methods are feature-based such as MicroPheno depends on k-mer frequency with MLP. Others depend on OTUs tables instead of k-mer frequency such as [21] used MLP and recursive neural network for phenotype classification and learning the hierarchy between the samples. Also, [24] used CNN [25] to learn the hierarchy between the OTUs and phenotype classification. Moreover, [26] used CNN on images generated by phylogenetic sorting on the OTUs for each sample for phenotype classification. DeepARG [23] is an example of gene ontology application using MLP for detecting antibiotic resistant genes.

### E. Word2Vec in Bioinformatics

Some DL techniques allow faster sequence classification such as Word2Vec [27]; a simple MLP hidden layer where the inputs are the context k-mers and the output is a target k-mer. Word2Vec is a self-supervised learning method that allows close target k-mers from the concept of replaceability in a specific context of k-mers to have close vectors. Empirically Word2Vec enhances any supervised learning such as classification [28].

## III. Related Work

RDP [15] is the first ML taxonomic classifier using Naive Bayes classification on k-mer frequency. RDP was trained on 300K sequences from Bergey's Taxonomic Outline [29] with a k-mer size of 8 from phylum to genus level. 16S Classifier [16] applied Random Forest for classification on Greengenes [30], where In-silico PCR was used to generate HVR subset databases to reduce the search space. All possible k-mers with size ranges from 2 to 6 were used, then feature selection was applied to reduce them. 16S Classifier produced specific models for each HVR database.

As new DL architectures develop rapidly such as CNN and Recurrent Neural Network (RNN) [31], direct sequence based DL approaches are dominating over feature based approaches. CNN was developed mainly for visual computing but used later in sequence classification. CNN finds temporal features where RNN searches for consecutive features in a sequence. RNA [32] and protein [33] structure prediction and alternative splicing prediction are among direct sequence based applications. The only available example (by mid-June 2018 besides this study) of sequence-based DL application in metagenomics is [34] for OTUs clustering using CNN and PCA.

Word2Vec was recently applied in alternative splicing prediction in SpliceVec [28], where it replaced complex DL architectures with MLP and to increase the model accuracy.

Additionally, Word2Vec has used in medical text mining [35] and protein structure classification [36].

## IV. MATERIALS & METHODS

### A. Generating HVRs databases

In order to decrease the search space by the classifier model, In silico PCR was applied on SILVA database [37] (version 132, SILVA_taxa, accessed in April 2018) to generate a subset for each HVR (eg: V2) or pairs of HVRs (eg: V3-V4). Pairs of primers for each HVR subset were the same in the 16S Classifier [16], all ambiguity characters in any primer sequences were expanded using an in-house developed script. The expended primer database was used in the In silico PCR by ipcress (version 2.2.0 ) [38] against each HVR database. All next steps were applied separately for each HVR dataset.

### B. Taxonomy parsing

Taxonomy ranking from phylum to species was parsed for each sequence. We discard any sequence with at least one missing rank (eg: order is missing in the hierarchy). In some cases, there are reads with unknown genus or species among different families, in order to avoid confusion, their previous ranks were concatenated to their genus and species classes.

### C. Ambiguity character handling

As SILVA database contains ambiguity characters rather than A, C, T, and G, any sequence with ambiguity characters was expanded to only one example. Expanding ambiguity sequences by more than one example resulted later in class in-balance and fast over-fitting. Biopython (version 1.72) [39] was used in taxonomy parsing and Ambiguity character handling.

### D. Stratified sampling

Training, test and validation datasets were split in a stratified manner using the species level as the output. Further preprocessing was done to ensure the same species classes exist in the 3 datasets.

### E. Word2Vec model training

As Word2Vec model deals with words rather than characters, each sequence of nucleotide characters was converted to a sequence of k-mers with k-mer size=6. A corpus of k-mers was generated from the validation and training data. Word2Vec was trained for 5 epochs using the skip-gram algorithm from the locally saved corpus. Gensim package (Version 3.4) [40] was used in training the Word2Vec model.

### F. Classifier training

*a) Architectures:* Several DL architectures in text classification were tested such as CNN, RNNs & MLP. Among tested CNN architectures are Residual network (ResNet) [41] & Inception [42]. The tested CNN architectures use 1D convolution instead of 2D convolution. RNNs are the least explored models such as Gated Recurrent Unit [43] and Long Short-Term Memory [44]. All models end with 6 fully connected layers, each corresponding to each rank from phylum to species. Species-level output has the higher number of classes. Weighted Adam optimization [45] was applied with early stopping if the model didn't coverage within 5 epochs. Manual architecture tuning was done on V2 HVR only as an example for the rest HVR databases. All the networks were built using Keras (Version 2.2.0) [46].

*b) Real-time variable length modeling:* In order to simulate variability in read lengths generated by NGS, a custom training function generated variable length reads was applied. This function generated sub-sequences of the original ones that are changing in each training epoch. The maximum length was 320 as it's the maximum read length in V2 HVR dataset. The minimum length was set to 50. To compare the effect of variable length against fixed length data augmentation, we applied the same hyper-parameters of our best model for training a fixed length reads of length 100 bases.

*c) Input representation & Model Evaluation*: For the Word2Vec, 3 types of input were tested: the direct sequence of characters (DC), the sequence of k-mers with Word2Vec embedding (W2V) and the sequence of k-mers without Word2Vec (SK). A range of read lengths (25, 50, 75, 100, 125, full_length HVR) was used to simulate fixed-length reads data on the test data for each length. Training time and the maximum test accuracy are the parameters in choosing of the best model.

## V. RESULTS

| | *Training time in mins* | *Prediction time in secs* | *Species* | *Genus* | *Family* | *Order* | *Class* | *Phylum* |
|---|---|---|---|---|---|---|---|---|
| ***Variable_len_MLP_W2V*** | 172 | 18 | 0.7471 | 0.9626 | 0.9919 | 0.9947 | 0.9978 | 0.9984 |
| ***Variable_len_MLP_SK*** | **82** | **18** | **0.7412** | **0.9618** | **0.9911** | **0.9944** | **0.9976** | **0.9978** |
| ***Variable_len_MLP_DC*** | 28 | 18 | 0.1220 | 0.1356 | 0.1369 | 0.2032 | 0.3020 | 0.4002 |
| ***Variable_len_ResNet_W2V*** | 1660 | 174 | 0.7526 | 0.9641 | 0.9921 | 0.9953 | 0.9982 | 0.9987 |
| ***Fixed_len_MLP_SK*** | 26 | 18 | 0.0660 | 0.1356 | 0.0666 | 0.0776 | 0.3020 | 0.3412 |
| ***Fixed_len_ResNet_W2V*** | 1590 | 174 | 0.7426 | 0.9605 | 0.9920 | 0.9953 | 0.9980 | 0.9986 |

*Table 1: Test accuracy on the size ranks and different simulated read lengths.*

*Variable_len: Used variable length data augmentation, Fixed_len: Used fixed length data augmentation, MLP: Multilayer Perceptron, ResNet: Residual Network as an example of CNN, DC: direct sequence of characters, W2V: sequence of k-mers with Word2Vec embedding, SK: sequence of k-mers without Word2Vec.*

Recent CNN architectures such as ResNet and Inception achieved high accuracy at all levels till the genus level on the direct sequence of characters. On the other hand, they achieved ~75% test accuracy on the species level for full-length HVR. ResNet was chosen as an example of CNN models to compare with MLP. This accuracy was the maximum result for all of our models. These models required at least 27 hrs of training to converge.

On the other hand, MLP achieved the maximum accuracy on W2V and SK with only ~1.5 & 3 hrs respectively of training. Also, MLP models prediction time on Tesla K16 GPU was 10 times faster than ResNet. It's worth mentioning that Word2Vec didn't increase the accuracy nor the training time in all CNN networks. Despite Word2Vec allowed probable convergence to the maximum accuracy, whereas this didn't happen in SK. Fixed length data augmentation didn't allow

MLP convergence but allowed for ResNet near the maximum accuracy.

| | *25* | *50* | *75* | *100* | *125* | *Full_HVR* |
|---|---|---|---|---|---|---|
| ***Phylum*** | 0.897 | 0.983 | 0.994 | 0.996 | 0.997 | 0.998 |
| ***Class*** | 0.862 | 0.976 | 0.991 | 0.995 | 0.997 | 0.998 |
| ***Order*** | 0.774 | 0.947 | 0.979 | 0.987 | 0.993 | 0.994 |
| ***Family*** | 0.726 | 0.922 | 0.966 | 0.981 | 0.988 | 0.991 |
| ***Genus*** | 0.588 | 0.818 | 0.894 | 0.930 | 0.950 | 0.962 |
| ***Species*** | 0.453 | 0.617 | 0.678 | 0.710 | 0.728 | 0.741 |

*Table 2: Best model test accuracies over different lengths and ranks.*

*Full_HVR: The full length of the High variable region*

## VI. Discussion

Simple DL models such as MLP over SK allow 20X faster in training and 10X in prediction in comparison to CNN models. Moreover, the sequence of k-mers takes into account the k-mers position. We train each model to predict the 6 ranks in the same time because a multi-output model can learn better than a separate model for each output [48]. By comparing MLP models with SK and W2V, both converged near the maximum accuracy but SK was faster in training. On the other hand, Word2Vec prevented complex CNN models from over-fitting with a sequence of k-mers. Even this advantage didn’t achieve higher accuracy than DC but W2V with CNN might be useful in one-shot learning [49].

We model SILVA taxa, as it has more examples per class than SILVA nr, moreover, Greengenes latest update was on 2013. k-mer size of 6 was chosen empirically to provide unique k-mers but won’t generate a high number of k-mers that will increase the classifier complexity. In addition, Word2Vec corpus was saved locally to save memory during training the Word2Vec model. RNNs were the least explored architectures as they are vulnerable to over-fitting in case of the low number of samples per class.

Real-time data augmentation eliminate the need to generate all possible sub-sequences of a specific length, which will lead to increasing data size by at least 100 folds. Also, it allows generating new inputs for the model in each epoch to prevent the overfitting. Using HVR specific models decrease the search space during training, also it is usable in the application as any amplicon study should have known targeted HVR. ResNet model complexity may be the reason why fixed length data augmentation allowed ResNet convergence which is not the case  in MLP.

## VII. Conclusion & Future Work

Most taxonomic classifiers depend on features rather than the direct sequence which can be modeled by DL. These features ignore their positions, resulting in low accuracy on short reads. Reducing the DL model complexity from CNN to MLP with global max pooling converged to species level classification with 74% test accuracy. MLP model decreased the training and prediction time by 20 and 10 folds respectively in comparison to CNN over a sequence of k-mers. Variable length data augmentation was the best data preprocessing for MLP besides its role in preventing overfitting and saving memory. Nevertheless we modeled only V2 HVR database, Future work will include modeling the rest HVRs databases. moreover, one-shot learning will be tested as many species class has very few samples.

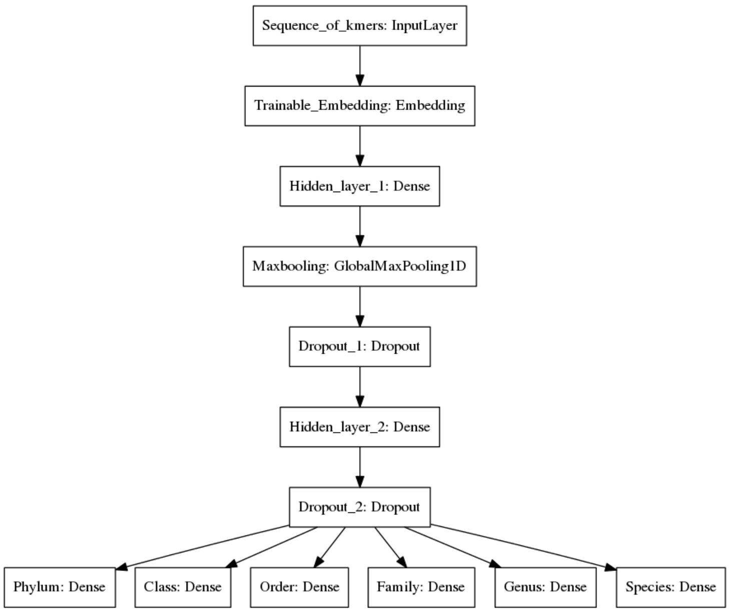

*Figure 2: Best model architecture.*

The model begins with input layer to a sequence of k-mers, followed by trainable embedding layer, followed with 2 hidden fully connected layer with Global max pooling between them. Dropouts were added to prevent the overfitting. The model ends with 6 fully connected layers as outputs for each rank.

## Acknowledgment

Thanks for Karim Amer from the Ubiquitous & Visual Computing Group for his advices on architecture engineering. Most of the training was done on Google Colab GPUs.

## References

[1] *Torres-Cortés, Gloria, et al. "Characterization of novel antibiotic resistance genes identified by functional metagenomics on soil samples." Environmental microbiology 13.4 (2011): 1101-1114.*

[2] *Gianoulis, Tara A., et al. "Quantifying environmental adaptation of metabolic pathways in metagenomics." Proceedings of the National Academy of Sciences (2009): pnas-0808022106.*

[3] *Levy, Maayan, Christoph A. Thaiss, and Eran Elinav. "Metagenomic cross-talk: the regulatory interplay between immunogenomics and the microbiome." Genome medicine 7.1 (2015): 120.*

[4] *Long, Philip E., et al. "Microbial metagenomics reveals climate-relevant subsurface biogeochemical processes." Trends in microbiology 24.8 (2016): 600-610.*

[5] *Nicholas, A. Be, et al. "Whole metagenome profiles of particulates collected from the International Space Station." Microbiome 5.1 (2017): 81.*

[6] *Schneider, Dominik, et al. "Metagenomic and metatranscriptomic analyses of bacterial communities derived from a calcifying karst water creek biofilm and tufa." Geomicrobiology Journal 32.3-4 (2015): 316-331.*

[7] *Vázquez-Campos, Xabier, et al. "Response of microbial community function to fluctuating geochemical conditions within a legacy radioactive waste trench environment." Applied and environmental microbiology (2017): AEM-00729.*

[8] *Hodkinson, Brendan P., and Elizabeth A. Grice. "Next-generation sequencing: a review of technologies and tools for wound microbiome research." Advances in wound care 4.1 (2015): 50-58.*

[9] *Liaw, Andy, and Matthew Wiener. "Classification and regression by randomForest." R news 2.3 (2002): 18-22.*

[10] *Hearst, Marti A., et al. "Support vector machines." IEEE Intelligent Systems and their applications 13.4 (1998): 18-28.*

[11] *Santamaria, Monica, et al. "Reference databases for taxonomic assignment in metagenomics." Briefings in bioinformatics 13.6 (2012): 682-695.*

[12] *McGinnis, Scott, and Thomas L. Madden. "BLAST: at the core of a powerful and diverse set of sequence analysis tools." Nucleic acids research 32.suppl_2 (2004): W20-W25.*

[13] *Herbig, Alexander, et al. "MALT: fast alignment and analysis of metagenomic DNA sequence data applied to the Tyrolean Iceman." (2017).*

[14] *Pruesse, Elmar, Jörg Peplies, and Frank Oliver Glöckner. "SINA: accurate high-throughput multiple sequence alignment of ribosomal RNA genes." Bioinformatics 28.14 (2012): 1823-1829.*

[15] *Wang, Qiong, et al. "Naive Bayesian classifier for rapid assignment of rRNA sequences into the new bacterial taxonomy." Applied and environmental microbiology 73.16 (2007): 5261-5267.*

[16] *Chaudhary, Nikhil, et al. "16S classifier: a tool for fast and accurate taxonomic classification of 16S rRNA hypervariable regions in metagenomic datasets." PloS one 10.2 (2015): e0116106.*

[17] *Gao, Xiang, et al. "A Bayesian taxonomic classification method for 16S rRNA gene sequences with improved species-level accuracy." BMC bioinformatics 18.1 (2017): 247.*

[18] *Laksshman, Sundaram, et al. "DeepBipolar: Identifying genomic mutations for bipolar disorder via deep learning." Human mutation 38.9 (2017): 1217-1224.*

[19] *Aliper, Alexander, et al. "Deep learning applications for predicting pharmacological properties of drugs and drug repurposing using transcriptomic data." Molecular pharmaceutics 13.7 (2016): 2524-2530.*

[20] *Asgari, Ehsaneddin, et al. "MicroPheno: Predicting environments and host phenotypes from 16S rRNA gene sequencing using a k-mer based representation of shallow sub-samples." bioRxiv (2018): 255018.*

[21] *Ditzler, Gregory, Robi Polikar, and Gail Rosen. "Multi-layer and recursive neural networks for metagenomic classification." IEEE Transactions on NanoBioscience 14.6 (2015): 608-616.*

[22] *Ruck, Dennis W., et al. "The multilayer perceptron as an approximation to a Bayes optimal discriminant function." IEEE Transactions on Neural Networks 1.4 (1990): 296-298.*

[23] *Arango-Argoty, Gustavo, et al. "DeepARG: a deep learning approach for predicting antibiotic resistance genes from metagenomic data." Microbiome 6.1 (2018): 23.*

[24] *Fioravanti, Diego, et al. "Phylogenetic convolutional neural networks in metagenomics." BMC bioinformatics 19.2 (2018): 49.*

[25] *Kim, Yoon. "Convolutional neural networks for sentence classification." arXiv preprint arXiv:1408.5882 (2014).*

[26] *Nguyen, Thanh Hai, et al. "Deep Learning for Metagenomic Data: using 2D Embeddings and Convolutional Neural Networks." arXiv preprint arXiv:1712.00244 (2017).*

[27] *Goldberg, Yoav, and Omer Levy. "word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method." arXiv preprint arXiv:1402.3722 (2014).*

[28] *Dutta, Aparajita, et al. "SpliceVec: distributed feature representations for splice junction prediction." Computational biology and chemistry 74 (2018): 434-441.*

[29] *Garrity, George M., Julia A. Bell, and T. G. Lilburn. "Taxonomic outline of the prokaryotes. Bergey's manual of systematic bacteriology." Springer, New York, Berlin, Heidelberg (2004).*

[30] *DeSantis, Todd Z., et al. "Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB." Applied and environmental microbiology 72.7 (2006): 5069-5072.*

[31] *Funahashi, Ken-ichi, and Yuichi Nakamura. "Approximation of dynamical systems by continuous time recurrent neural networks." Neural networks 6.6 (1993): 801-806.*

[32] *Wong, Pak-Kan, Man-Leung Wong, and Kwong-Sak Leung. "Long-Short Term Memory Network for RNA Structure Profiling Super-Resolution." International Conference on Theory and Practice of Natural Computing. Springer, Cham, 2017.*

[33] *Yang, Yuedong, et al. "Spider2: A package to predict secondary structure, accessible surface area, and main-chain torsional angles by deep neural networks." Prediction of Protein Secondary Structure. Humana Press, New York, NY, 2017. 55-63.*

[34] *Zheng, Hao, et al. "Sequence-based Deep Learning Reveals the Bacterial Community Diversity and Horizontal Gene Transfer."*

[35] *Habibi, Maryam, et al. "Deep learning with word embeddings improves biomedical named entity recognition." Bioinformatics 33.14 (2017): i37-i48.*

[36] *Tsubaki, Masashi, Masashi Shimbo, and Yuji Matsumoto. "Protein Fold Recognition with Representation Learning and Long Short-Term Memory." IPSJ Transactions on Bioinformatics 10 (2017): 2-8.*

[37] *Quast, Christian, et al. "The SILVA ribosomal RNA gene database project: improved data processing and web-based tools." Nucleic acids research 41.D1 (2012): D590-D596.*

[38] *Slater, Guy St C., and Ewan Birney. "Automated generation of heuristics for biological sequence comparison." BMC bioinformatics 6.1 (2005): 31.*

[39] *Cock, Peter JA, et al. "Biopython: freely available Python tools for computational molecular biology and bioinformatics." Bioinformatics 25.11 (2009): 1422-1423.*

[40] *Rehurek, R., and P. Sojka. "Gensim–python framework for vector space modelling." NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic 3.2 (2011).*

[41] *Wu, Zifeng, Chunhua Shen, and Anton van den Hengel. "Wider or deeper: Revisiting the resnet model for visual recognition." arXiv preprint arXiv:1611.10080 (2016).*

[42] *Szegedy, Christian, et al. "Inception-v4, inception-resnet and the impact of residual connections on learning." AAAI. Vol. 4. 2017.*

[43] *Chung, Junyoung, et al. "Gated feedback recurrent neural networks." International Conference on Machine Learning. 2015.*

[44] *Gers, Felix A., Jürgen Schmidhuber, and Fred Cummins. "Learning to forget: Continual prediction with LSTM." (1999): 850-855.*

[45] *Loshchilov, Ilya, and Frank Hutter. "Fixing weight decay regularization in adam." arXiv preprint arXiv:1711.05101 (2017).*

[46] *Chollet, François. "Keras." (2015).*

[47] *Collobert, Ronan, and Jason Weston. "A unified architecture for natural language processing: Deep neural networks with multitask learning." Proceedings of the 25th international conference on Machine learning. ACM, 2008.*

[48] *Fei-Fei, Li, Rob Fergus, and Pietro Perona. "One-shot learning of object categories." IEEE transactions on pattern analysis and machine intelligence 28.4 (2006): 594-611.*