Predicting Bacterial Property

- A simple linear regression model(considering the value to predict is continuous) will be very fast to train. Moreover, since the number of features is less, the cost function can be minimized very quickly using Newton' Method. The accuracy will mostly not be too bad but could be improved further.
- Random Forest Regressors will do a great job in modeling this data. Does not overfit and does not require much hyperparameter tuning. Do a good job compared to even neural networks[1].
- Internal estimates can also be used to monitor feature importance[2].
- While neural networks provide higher accuracy, they are computationally expensive to train and are prone to overfitting.
- The increased accuracy of neural networks may or may not compensate for the added hassles in training. Moreover, it is difficult to effectively interpret their results[3].

Predicting number of people on the beach

- Since the beach owner is collecting the data, I expect that the dataset will not have too many samples. As is already indicated, the number of features is also less.
- As a result, preventing overfitting will be the primary challenge here.
- We could use a neural network with less number of hidden layers.
- Usage of techniques like L1 or L2 regularization and early stopping to prevent overfitting is vital here. Training is much slower but testing is faster[3].
- Random Forest Regressor could also be used without much hassle[2]. Max depth can be reduced to prevent overfitting.
- Higher accuracy would be obtained using neural networks. However whether the increased accuracy compensates for the increased difficulty in training will depend on the dataset itself. Can also experiment with even simpler models like generalized linear models.
- We will also need to eliminate outliers using outlier detection techniques since outliers can have a significant effect here.

Matrix Multiplication

- We can use the standard matrix multiplication technique with possibly using the Strassen optimization if accuracy is the concern. Parallel computing algorithms can be used to speed up matrix computations.
- We can use a neural network with 18 inputs(to take in the values of the 2 matrices) and 9 outputs(for each position of the output matrix). It will learn the non linear relationships between the elements to get the desired output. Since the relationships are very straightforward, a single hidden layer could do the task. This could end up being faster than the traditional multiplication with an allowed degree of variation in the final output.

- If sufficient power is available, can utilize AlphaTensor to discover new algorithms for multiplying 3*3 matrices. However, only an incremental upgrade is expected[4].

Image to Caption Generator

- We will use a CNN trained on the MS-COCO 2014 images to extract the relevant features. These features will be fed into the LSTM, which will generate a textual description of the image.
- The last hidden layer of the CNN is connected to the LSTM[5]. The function that we want to optimize is[5]:

$$\log p(S|I) = \sum_{t=0}^{N} \log p(S_t|I, S_0, \ldots, S_{t-1})$$

- Where S_t denotes a word of the sentence. We are basically maximizing the log probability that the model generates the correct description given the image. A sentence can be composed of multiple words, hence RNNs serve as the ideal choice here. LSTMs, like always, are used to avoid the vanishing/exploding gradient problem[5].

- Each word S_t is described as a one-hot vector with its size being the number of words in the dictionary.
- The loss function is[5]:

$$L(I, S) = -\sum_{t=1}^{N} \log p_t(S_t) \ .$$

- This loss function is used to optimize all the parameters of the LSTM, the word embeddings and the top layer of the CNN[5].

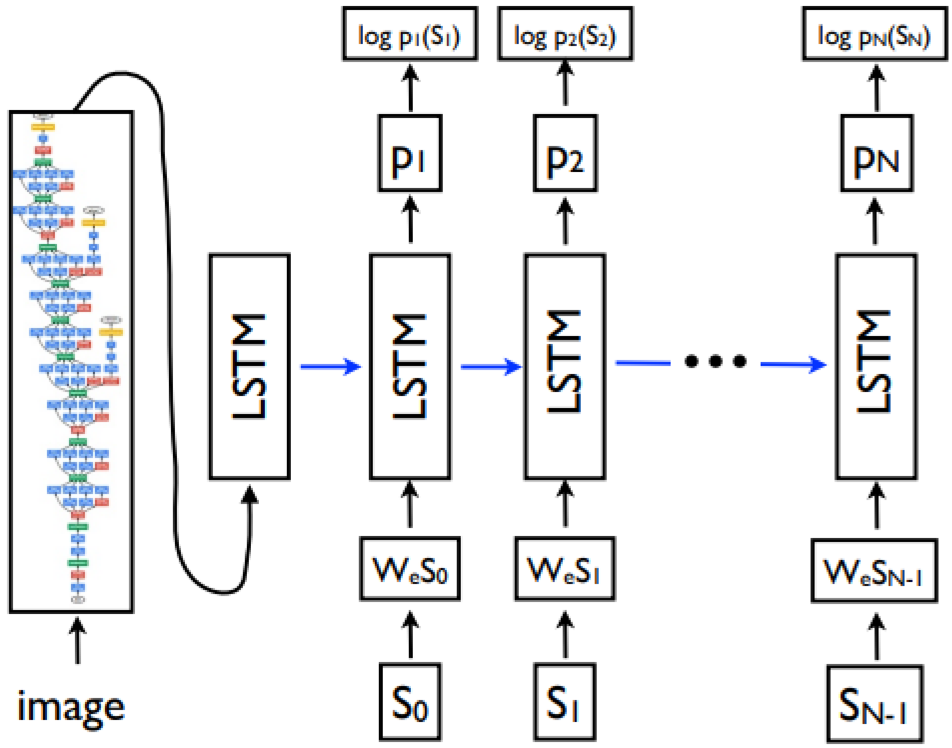

Image Credits: https://arxiv.org/pdf/1411.4555.pdf

References

[1] Liaw, A., & Wiener, M. (2002). Classification and Regression by Random Forest. R News, 2(3), 18-22. Available at: https://cran.r-project.org/doc/Rnews/Rnews_2002-3.pdf

[2] Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32. DOI: 10.1023/A:1010933404324

[3] Sarker, I.H. Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. *SN COMPUT. SCI.* 2, 420 (2021). https://doi.org/10.1007/s42979-021-00815-1

[4] Fawzi, A., Balog, M., Huang, A. *et al.* Discovering faster matrix multiplication algorithms with reinforcement learning. *Nature* 610, 47–53 (2022). https://doi.org/10.1038/s41586-022-05172-4

[5] https://arxiv.org/abs/1411.4555