Data Science & Proteomics

Akash Kadel[1], Brenton Arnaboldi[1], Daniel Amaranto[1] and Richard Bonneau[2]

[1]MS in Data Science, New York University

[2]Director, Center for Data Science, New York University

November 9, 2017

https://github.com/NYU-CDS-Capstone-Project/Beyond-Google

Original Project Description

Our project was not among the original proposals and therefore there is no description to be copied here. Professor Bonneau proposed that we work on a project based on research of protein function prediction that is ongoing at the Flatiron Institute.

1 Research Question

Protein sequences are being identified at a very fast pace, while deeper understanding of the identified sequences is acquired much more slowly. A large number of characteristics of a given protein are of interest to scientists, such as structure, protein-protein interaction, and determining various functions. But confirming those characteristics requires empirical validation that takes longer than merely identifying a sequence. We can expect that the pace of sequence identification will continue to be faster than that of any of the other protein-related discoveries. Therefore one interesting question in microbiology today would be whether strictly using sequence information has the potential to make plausible predictions about the functional characteristics of a protein. In this project we will explore methods that train an algorithm to make accurate predictions of function from sequence information alone.

In order to coordinate efforts to search for an ideal protein function prediction algorithm, a timed challenge among a large number of research institutions has been devised that allows a comparison of methods. The Critical Assessment of Functional Annotations, which has been conducted twice so far, has involved computational techniques that consider a wide variety of features for functional prediction, including functional domains, protein interactions, mass spectrometry,

gene interactions, clinical data, natural language processing, and many more. [5] Successful methods generally take many different protein characteristics into account on top of sequence information. Our bare-bones approach will require novel approaches to extracting deeper meaning from sequence data. If successful, leaner methods could have a big impact on the field of protein prediction research.

2 Data

Currently we have 12475 human protein sequences divided into train, validation, and test sets of sizes 9751, 3871, and 1647, respectively. We also have yeast protein sequences divided into train, validation, and test sets of sizes 3447, 963, and 206, respectively. The human protein sequences collectively predict 147 different GO molecular function terms (GO terms are described in Section 2.2) and the yeast protein sequences collectively predict 26 GO molecular function terms. What follows is a brief explanation of the inputs and outputs.

2.1 Inputs: Protein Sequence

The protein sequences on which are training are chains of amino acids. There are 20 amino acids in human proteins and each one conveys a vast array of chemical versatility. The chemical properties of the amino acids of proteins determine the biological activity of the protein. The body builds proteins one amino acid at a time. When the chain is complete, it twists and folds into a specialized shape. The chemical structure of each amino acid controls the final shape, and the shape determines the function of the protein. In this way the sequence of a protein does in fact dictate everything about it; however, the particular sequence folding and the more macroscopic physical properties of a protein can't be easily predicted from just a sequence.

2.2 Outputs: GO Terms

With respect to labeling, a taxonomy of protein functions has been developed by the Gene Ontology Consortium. Of the three major groups, we will only consider molecular function predictions.

Molecular function is defined as the biochemical activity (including specific binding to ligands or structures) of a gene product. This definition also applies to the capability that a gene product (or gene product complex) carries as a potential. It describes only what is done without specifying where or when the event actually occurs. Examples of broad functional terms are 'enzyme', 'transporter' or 'ligand'. Examples of narrower functional terms are 'adenylate cyclase' or 'Toll receptor ligand'.

2.3 GO as a graph

The structure of GO can be described in terms of a graph [1], where each GO term is a node, and the relationships between the terms are edges between the nodes. GO is loosely hierarchical, with 'child' terms being more specialized than their 'parent' terms, but unlike a strict hierarchy, a term may have more than one parent term (note that the parent/child model does not hold true for all types of relation). For example, the biological process term hexose biosynthetic process has two parents, hexose metabolic process and monosaccharide biosynthetic process. This is because biosynthetic process is a subtype of metabolic process and a hexose is a subtype of monosaccharide. Consider the below figure:

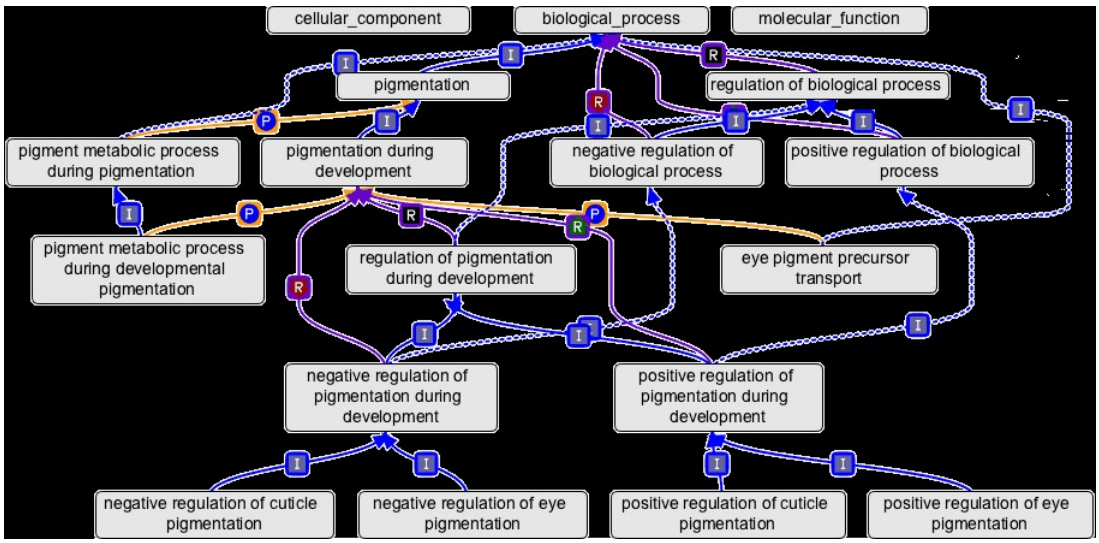

Figure 1: GO ontology example

In the above diagram, relations between the terms are represented by the colored arrows; the letter in the box midway along each arrow is the relationship type. Note that the terms get more specialized going down the graph, with the most general terms — the root nodes, cellular component, biological process and molecular function — at the top of the graph. Terms may have more than one parent, and they may be connected to parent terms via different relations. While we will not attempt to predict hierarchical relationships with our model, understanding their general structure is still important.

2.4 Train, Validation, Test Set Split

The selection of train, validation, and testing data is temporally determined and aligns with the timeline by which discoveries of functions are made empirically. At time t_0, a set of protein sequences is identified and their respective annotations (if present) are recorded. At a later time t_1 the annotations of those same protein sequences are captured and used to set the train, validation,

and test sets as follows: proteins that have had no changes to their annotations become the train set. Proteins that originally had some annotations and then gained more annotations between t_0 and t_1 become the validation set. Proteins that had no annotations at all at t_0 and gained some annotations in t_1 become the test set. Finally, the entire set of annotations is summed and the labels are limited to functions that have between 10 and 1000 total instances. This step eliminates extremely common and extremely uncommon functions. Because of this final filtration, there are some proteins in the set that have no functional predictions at all.

3 Methodology

3.1 Model experimentation

There are a variety of methods that could be promising for the task of using sequence data alone to predicted GO functions. As seen in section 2.1, the amino acid sequence is different for every protein function. This means that sequences have different lengths, i.e. every training example has a different size. This directly converts this problem into a hard classification problem if we were to use usual Machine Learning techniques. Hence, we are going ahead with use of Neural Nets as it deals with data of different lengths very effectively. Some of the concepts/models we would be using are:

- Baseline method - FastText
- Recurrent Neural Networks (RNN)
    1. Vanilla RNN
    2. LSTM
    3. Gated Recurrent Units
- Convolutional Neural Networks (CNN)

We will apply the same hyperparameter tuning algorithm to each of our models. We plan to use random search over different ranges of respective hyperparameter values. A brief description on some of the Models we tried / under construction

3.1.1 FastText

FastText is a simple and very fast algorithm that takes variable length sequences for text classification tasks. In some tasks its performance is similar to embeddings trained with deep learning methods; however, as with many text based models, it requires a considerable amount of training data in order to be effective.

An essential component of FastText is developing word embeddings. For our project so far, we have built embeddings for individual characters (each amino acid is represented by a letter). In the future, we may try embeddings for chunks or n-grams of characters.

3.1.2 RNN

We are also going ahead on experimenting with Recurrent Neural networks as it yields promising results with variable sized input and are capable of learning long-range dependencies [4]. We tested an RNN because we believe that the ordering of amino acids might be significant, and LSTMs do a particularly good job of retaining information from earlier in a sequence. One very pressing issue with using Vanilla RNN's is the vanishing gradient problem, due to which we are also training on LSTM's and GRU's as well.

Both of them combat the vanishing gradient problem and share a lot of properties amongst themselves. According to empirical evaluations [6], there isn't a clear winner among both of them, which motivates us to try to experiment with both LSTM's and GRU's.

3.1.3 Additional Models Under Construction

- **One vs All Approach:** Since this a multi-label classification, we are also planning on training an individual model for every GO term in sequence (One vs All approach). So basically, as we have 147 GO terms for the human data, we will have 147 different models (weight values).
- **Human and Yeast Species:** The data is separated into human and yeast sets. There are 147 functions predicted by the human set, and 26 functions determined by the yeast set. 20 functions are common to both sets. We will continue to experiment with these sets separately and will also try combining them, either by analyzing the 20 functions commonly assessed, or by including everything together.

3.2 Evaluation

After consulting with the Flatiron team, we are planning to use the following evaluation methods on each of our models:

- AUC score : AUC score is defined as Area under ROC and it calculates the tradeoff between True Positive Rate (TPR) and False positive Rate (FPR)
- Precision-Recall AUC : For highly skewed classes, AUPR (Area under Precision & Recall) gives a better intuition than AUC [3]. AUPR plots Precision vs recall. ie, the x-axis is same as AUC curve, but the y-axis is now recall (different from False Positive Ratio).

- F-score : F score accounts for the tradeoff between Precision and Recall. Precision aims to minimize FPR, whereas Recall minimizes False Negative Rate (FNR). F-score computes the weighted average of the two. CAFA based F-score varies slightly:

We will assess the metrics (and adjust the formulas accordingly) on both the multilabel classification problem in which all functional terms are predicted at once, and also for the binary prediction task of predicting individual GO terms.

4 Results

4.1 Rubric for success

In our project, we aim to show whether using sequence data alone can compare with existing methods. Based on the Critical Assessment of Functional Annotations (CAFA1) and CAFA2 [5], the top performing models have the following best F-score:

- Molecular Function (highest performing models have F-Score of around .60)

4.2 Performance

So far, we have not been able to achieve much success with either the FastText or LSTM models. Our poor F-score performance is driven almost entirely by low precision (as one would expect in a target set with high sparsity). High sparsity, less data & higher labels (147) makes it extremely extremely hard for the model to learn. The reported F-scores are between 0.1 and 0.2, mainly due to very less training data size.

To overcome the above issue, we will be going ahead in brain-storming to come up with different architectural issues of the input data. Hyper-parameter tuning is another area which we are currently focusing on to tune the model effectively. Also, some papers show that CNN performs comparatively better, which is our next focus for modeling.

5 Discussion

Since our first discussions about this project began in September, we have made considerable progress in some key areas. While it remains to be seen how successful our methods will ultimately be, we feel confident that we can present our work in a rigorous way that demonstrates appropriate methodology. We have conducted a thorough review of dozens of papers that have reported results on the same task. We will use that review both to contextualize our research in our report and also to pick up ideas about different approaches to solving the problem. We have created some

baseline models that perform the multi-label prediction. We have defined and implemented some evaluation metrics that will allow us to compare our work to other approaches that have been published.

An especially important set of details that have changed is the final form of our data sets, which was an open question. The proposed scope of this project originally involved using a diverse set of data types for a prediction of a diverse set of function labels in 3 different species. The data types we were considering were strings of protein sequences, contact maps, and functional domains. The labels were going to encompass molecular, biological, and cellular component functions. The protein species would be human, mouse, and yeast. We have since confirmed that the only available data will include protein sequences, the only available predictions will be in the molecular function space, and the species we will have access to are human and yeast.

The biggest problems we will face in the remaining weeks of this project all have to do with building successful models. While it often happens that research does not yield promising results, we would naturally prefer to report a novel method that augments the set of tools used in protein prediction. That said, literature review of protein prediction methods revealed that, with the exception of certain groups that solely relied on modifying BLAST techniques, CAFA contributors relied on a wide variety of source data (contact maps, protein-protein interactions, functional domains, homologs). For example, Cozzetto et al. produced a high-performing method that was reported in the second results of the CAFA competition and they credited the integration of various data sources as a key factor in their model's success. [2] Given that we are only working with sequence data, a comparison of our performance to the heterogeneous models may very reinforce the necessity of using a variety of data sources. We will continue to apply different deep learning methods to our data, and we will attempt to devise a wider variety of features from the sequences. Hopefully these efforts will pay off and will uncover some techniques that contribute to protein prediction success rates. If not, we will simply have to report that our methods did not work, and also propose reasons why that might be.

References

[1] (Gene Ontology Consortium). Go as a graph.

[2] Cozzetto, D., Buchan, D., Bryson, D., and Jones, D. (2013). Protein function prediction by massive integration of evolutionary analyses and multiple data sources. BMC Bioinformatics.

[3] Davis, J. and Goadrich, M. (2006). The relationship between precision-recall and roc curves. pages 233–240.

[4] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. Neural Comput., 9(8):1735–1780.

[5] Jiang, Y., Oron, T. R., Clark, W. T., Bankapur, A. R., D'Andrea, D., Lepore, R., Funk, C. S., Kahanda, I., Verspoor, K. M., Ben-Hur, A., Koo, D. C. E., Penfold-Brown, D., Shasha, D., Youngs, N., Bonneau, R., Lin, A., Sahraeian, S. M. E., Martelli, P. L., Profiti, G., Casadio, R., Cao, R., Zhong, Z., Cheng, J., Altenhoff, A., Skunca, N., Dessimoz, C., Dogan, T., Hakala, K., Kaewphan, S., Mehryary, F., Salakoski, T., Ginter, F., Fang, H., Smithers, B., Oates, M., Gough, J., Törönen, P., Koskinen, P., Holm, L., Chen, C.-T., Hsu, W.-L., Bryson, K., Cozzetto, D., Minneci, F., Jones, D. T., Chapman, S., BKC, D., Khan, I. K., Kihara, D., Ofer, D., Rappoport, N., Stern, A., Cibrian-Uhalte, E., Denny, P., Foulger, R. E., Hieta, R., Legge, D., Lovering, R. C., Magrane, M., Melidoni, A. N., Mutowo-Meullenet, P., Pichler, K., Shypitsyna, A., Li, B., Zakeri, P., ElShal, S., Tranchevent, L.-C., Das, S., Dawson, N. L., Lee, D., Lees, J. G., Sillitoe, I., Bhat, P., Nepusz, T., Romero, A. E., Sasidharan, R., Yang, H., Paccanaro, A., Gillis, J., Sedeño-Cortés, A. E., Pavlidis, P., Feng, S., Cejuela, J. M., Goldberg, T., Hamp, T., Richter, L., Salamov, A., Gabaldon, T., Marcet-Houben, M., Supek, F., Gong, Q., Ning, W., Zhou, Y., Tian, W., Falda, M., Fontana, P., Lavezzo, E., Toppo, S., Ferrari, C., Giollo, M., Piovesan, D., Tosatto, S. C., del Pozo, A., Fernández, J. M., Maietta, P., Valencia, A., Tress, M. L., Benso, A., Di Carlo, S., Politano, G., Savino, A., Rehman, H. U., Re, M., Mesiti, M., Valentini, G., Bargsten, J. W., van Dijk, A. D. J., Gemovic, B., Glisic, S., Perovic, V., Veljkovic, V., Veljkovic, N., Almeida-e Silva, D. C., Vencio, R. Z. N., Sharan, M., Vogel, J., Kansakar, L., Zhang, S., Vucetic, S., Wang, Z., Sternberg, M. J. E., Wass, M. N., Huntley, R. P., Martin, M. J., O'Donovan, C., Robinson, P. N., Moreau, Y., Tramontano, A., Babbitt, P. C., Brenner, S. E., Linial, M., Orengo, C. A., Rost, B., Greene, C. S., Mooney, S. D., Friedberg, I., and Radivojac, P. (2016). An expanded evaluation of protein function prediction methods shows an improvement in accuracy. Genome Biology, 17(1):184.

[6] Jozefowicz, R., Zaremba, W., and Sutskever, I. (2015). An empirical exploration of recurrent network architectures. Journal of Machine Learning Research.