Title: Semantic Segmentation of 3-D Protein Architectures in 2-D Contact Maps
Author: Raphael Eguchi
Category: Computer Vision, CNN, Semantic Segmentation

Abstract

Parsing and identifying protein architectures takes an experienced eye and knowledge of many different folds. Because of this, many well-established protein classification laboratories still use manual evaluation to annotate structural features in proteins. Towards the automation of this process, I present a fully convolutional neural network that performs residue-wise semantic segmentation of multi-domain protein chains, given a pairwise distance matrix or "contact map" as an input. The model classifies each residue into 38 architecture classes and performs with 90.3% position-wise accuracy, 95.2% average within-class accuracy, and 87.0% average within-structure accuracy.

1. Introduction

Proteins are biomolecules that regulate nearly all chemical processes in living organisms. Structurally, proteins are a linked sequence of amino acids ("**residues**"), of which there are 20 different types. It is known that a given string of amino acids will fold into a unique 3-D structure with distinct biochemical functions. Most proteins are comprised of multiple **domains** (Fig. 1), which are structural units that can be assigned to different architecture classes.

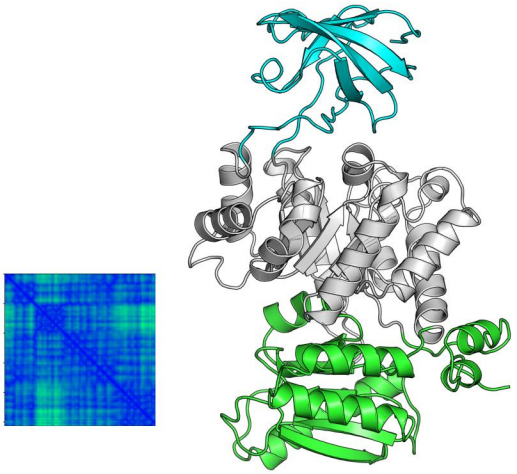


Figure 1. Example of a Multi-Domain Protein
A cartoon-style rendering of PDB 3gqyA (right) and corresponding contact map (left). The protein is single-chain and comprised of three domains, each with a different architecture: β-barrel (blue), αβ-barrel (white), 3-layer-αβα-sandwich (green).

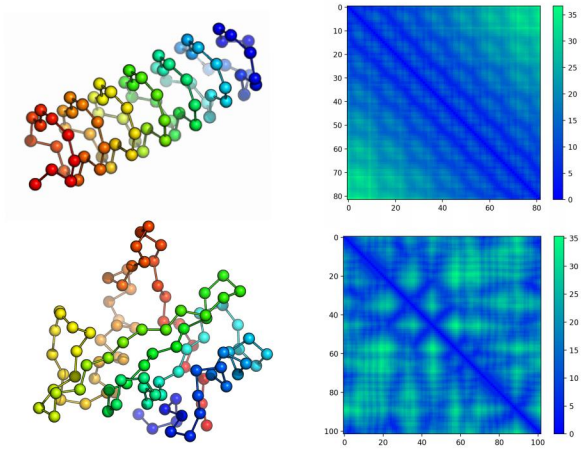


Figure 2. Protein Contact Maps
Two examples of single-domain proteins colored as chainbows (left), with α-carbons shown as spheres. Red indicates the beginning, and blue indicates the end of the chain. The corresponding contact maps are shown to the right of each protein with units in Å. Axes correspond to amino acid index.

When studying protein structure it is often difficult to parse and identify domain architectures without an experienced eye and knowledge of many different folds. As a result, manual evaluation is still widely used to annotate structural features.

Previously, I reported a *single-domain classifier* which returns an architecture class label given the *contact map* of an isolated protein domain. A contact map is a pairwise distance matrix between all amino acids in the protein (α-carbon positions), and is thus a rotationally and translationally invariant representation of a 3-D protein structure in 2-D (Fig. 2). Since my previous report, further training allowed this initial model to achieve a new test accuracy of 84%, and an average within-class accuracy of 87%. Further details of this preliminary model are summarized in my mid-quarter milestone and are excluded here. The new model described below extends the upon the previous architecture but performs the tasks of domain segmentation and classification in one pass. The input of the model is the contact map of a full multi-domain protein chain, and the output is a vector of class predictions for each amino acid.

2. Dataset

All data used in this study was obtained from the CATH protein database. A list of unique protein chains was kindly provided by Ian Sillitoe who works with the CATH group at University College London.

2.1. Data Selection

The current CATH database (v4.2) is comprised of 132380 non-redundant fully annotated protein chains (42 classes, 202506 domains, 3.3×10^7 residues). From this set, chains shorter than 520 residues that did not contain domains belonging to classes with fewer than 10 members were selected for use. Structures larger than 512 were center-cropped, and smaller structures were zero-padded to 512. The resulting dataset contains 126069 chains (38 classes, 181753 domains, 2.9×10^7 residues) spanning 95% of all non-redundant chain data. Selection did not drastically alter the overall structure of the data (Fig. 3).

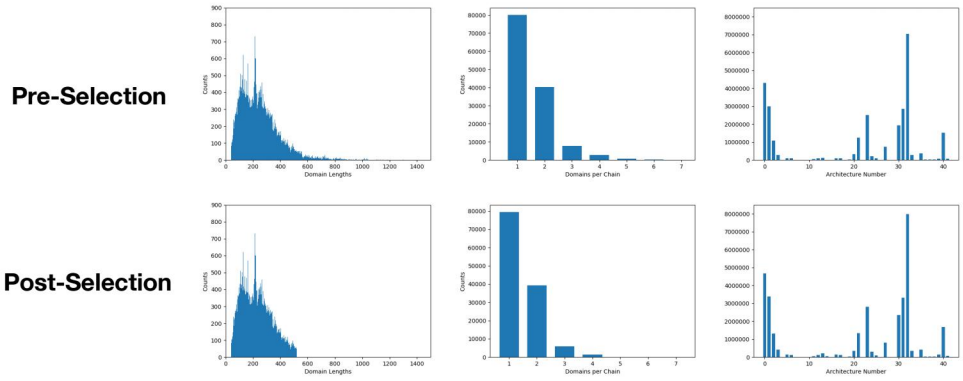


Figure 3. Distribution of Data before and after Selection.
Top: *Distributions of non-redundant protein chains in the CATH database.* ***Bottom:*** *Distributions of the selected dataset. The left-most column shows the distribution of chain lengths, the middle the number of domains per chain, and the right the number of residues per architecture. The selection process does not greatly alter the structure of the data.*

2.2. Data Splitting

Of the selected chains, 8000 were reserved for each the test and development sets and the remaining 110069 were used in training. It should be noted that the dataset is very imbalanced with the largest class containing nearly 7 million residues and the smallest containing less than 3500. To address this, the split was performed in a stratified manner, but stochastically adjusted so that all sets had at least 650 residues from each class present (Fig. 4). During training, examples were weighted to ensure that each class had equal influence.

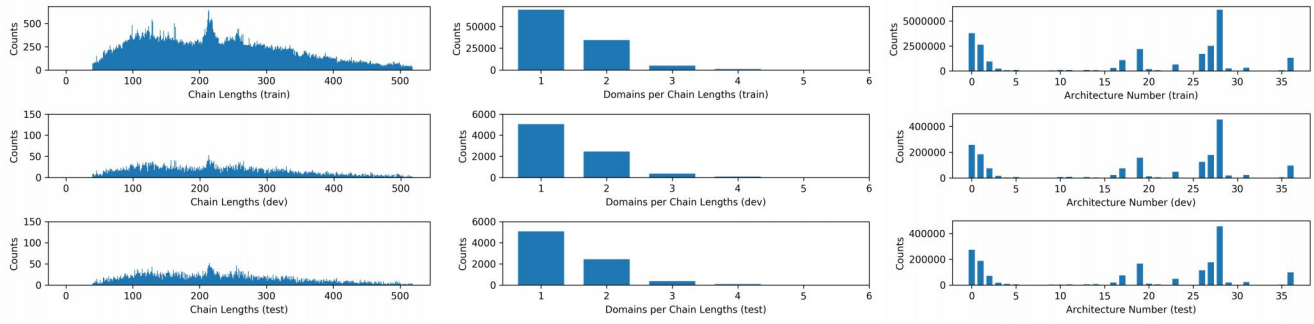


Figure 4. Distribution of the Split Data.
From left to right, each column shows the distribution of chain lengths, the number of domains per chain, and the number of residues per architecture. The split was performed in a stratified manner so that each class is represented by at least 650 residues in each set.

3. Model

3.1. Architecture

The architecture of the model is shown in Figure 5. The model is comprised of six convolutional layers (encoding), followed by a "pixel shuffle" step for upsampling to a 512x512 feature map (decoding). Rectangular convolutions are then used to reshape the map into a 1x512@38 tensor that is passed to a softmax function. Each convolutional layer in the encoding phase is followed by a BatchNorm and a LeakyReLu. The 4x1 convolution is followed only by a LeakyReLu, and the final layer passes directly into a softmax function.

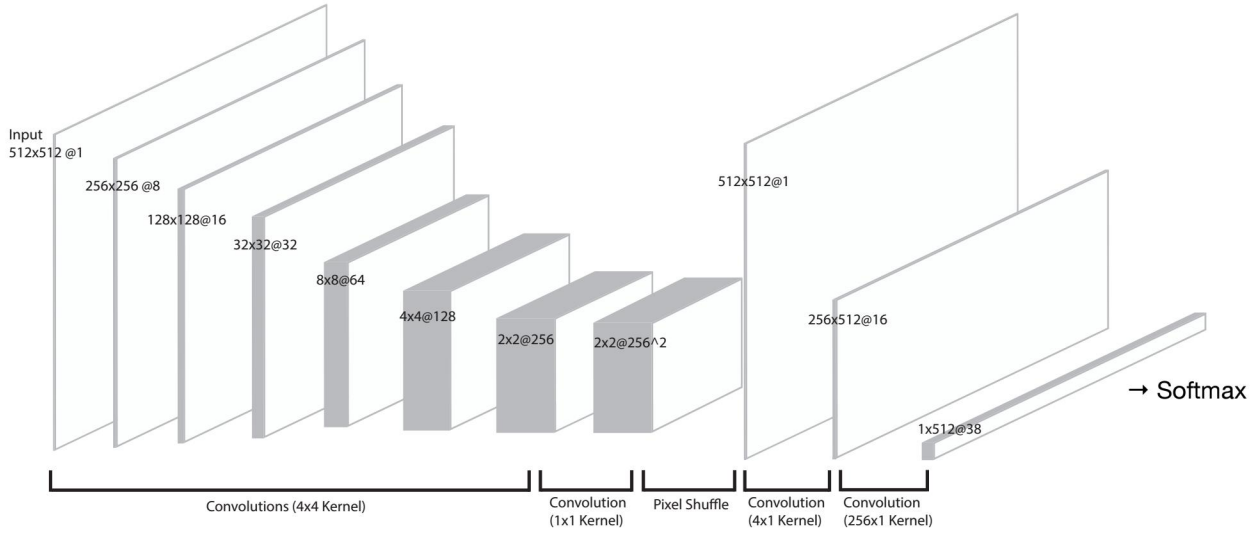
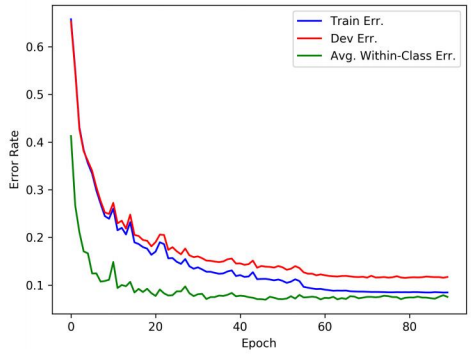


Figure 5. Model Architecture

The model is comprised of six downsampling convolutional layers (encoding), followed by a "pixel shuffle" step for upsampling to a 512x512 map (decoding). Rectangular convolutions are used to reshape the feature map into a 1x512@38 tensor that is passed into a softmax function.

3.2. Training

Training was performed for a total of 90 epochs with a mini-batch size of 64 using the Adam optimization algorithm. A learning rate of 0.001 was used for the first 55 epochs, 0.0001 for the next 15 epochs, and 0.00001 for the final 20 epochs. The loss function, shown below, is a cross entropy loss averaged across every residue in the input chain. Importantly, the loss was weighted (w_c) so that training examples from under-represented classes had increased influence on training. Weights were chosen so that each class, in total, had equal influence.



$$Loss(z,c) = -\frac{1}{r}\sum_{r} w_c \; log\frac{e^{z_{c,r}}}{\sum_j e^{z_{j,r}}}$$

Figure 6. Error Rates During Training

After 90 epochs of training, the model achieves a peak accuracy of 90% and an average within-class accuracy of 95% on the development set. (Note: the plot shows slightly higher error due to a mistake in forgetting to turn off dropout when computing error during training.

Dropout regularization was used throughout the convolutional layers in the encoding phase with a zeroing-probability of 0.1. All weights were initialized using Xavier Initialization. The model reaches an accuracy of 90% and a within-class accuracy of 95% on both the training and development sets. Model inputs were scaled by -100 and not normalized.

3.3. Performance

The trained model generalizes to the test set and performs with a position-wise accuracy of 90.3%, an average within-class error of 95.2%, and an average within-structure accuracy of 87.0%. Because the data are highly imbalanced and the class distributions of each of the three datasets are very similar, one concern is that the model may simply be overfitting to highly represented classes. The plot below (Fig. 6) suggests otherwise, as there is no significant correlation between the frequency of a class in the training set and class accuracy. The majority of class accuracies are above 90% even for the rarest classes.

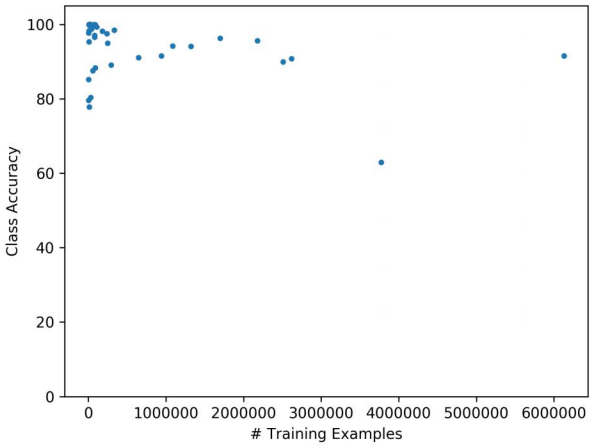


Figure 6. Class Accuracy vs Training Counts
Each point represents a single class. No obvious correlation between the number of examples in the training set and class accuracy on the test set is observed.

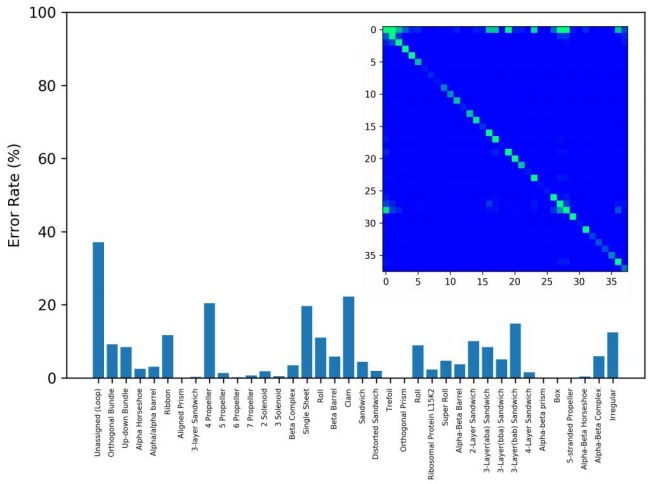


Figure 7. Error Rates
The error rates for each classes. The confusion matrix is shown as an insert and indexed by architecture number. Green indicates a higher count. The matrix is clipped for visibility.

3.4. Output Analysis

Example outputs are shown on the following page in Figure 8. Overall the model performs well, and is able to effectively segment multi-domain structures. In many of the test cases, there are minor deviations in boundary positions. However, the majority of these do not appear to hinder the coherence of the predictions. Importantly, it appears that the model is able to recognize delocalized features in the 2-D contact map that correspond to localized domains in the 3-D protein structure:

(1) *The model is able to recognize domains with discontinuous residue ordering.*
This is demonstrated in the case of the central alpha-beta barrel in 3gqyA, where the residues comprising the barrel are distant in sequence. This is apparent also in the case of the two tre-foil domains of 4i1eA — each domain has significantly different residue-ordering, and yet the model is able to properly identify both.

(2) *The model is able to recognize differences in secondary structure organization.*
Both the orthogonal-bundle in 3sahB and the alpha-horseshoe in 4i1eA are purely alpha-helical in content, and both are comprised of residues in continuous sequence. Despite these similarities, the model is able to distinguish between the secondary-structure organizational differences in these architectures.

4. Remarks and Future Directions.

I plan to implement my model as a plugin for the popular structure-viewing application PyMOL. As a protein researcher myself, such a tool would greatly aid my everyday viewing of protein structures and, I believe would also serve as a useful educational device for students of structural biology. The overall success of this model may also suggest that contact maps are a powerful, compact representation of 3-D protein structures, and should be explored further.

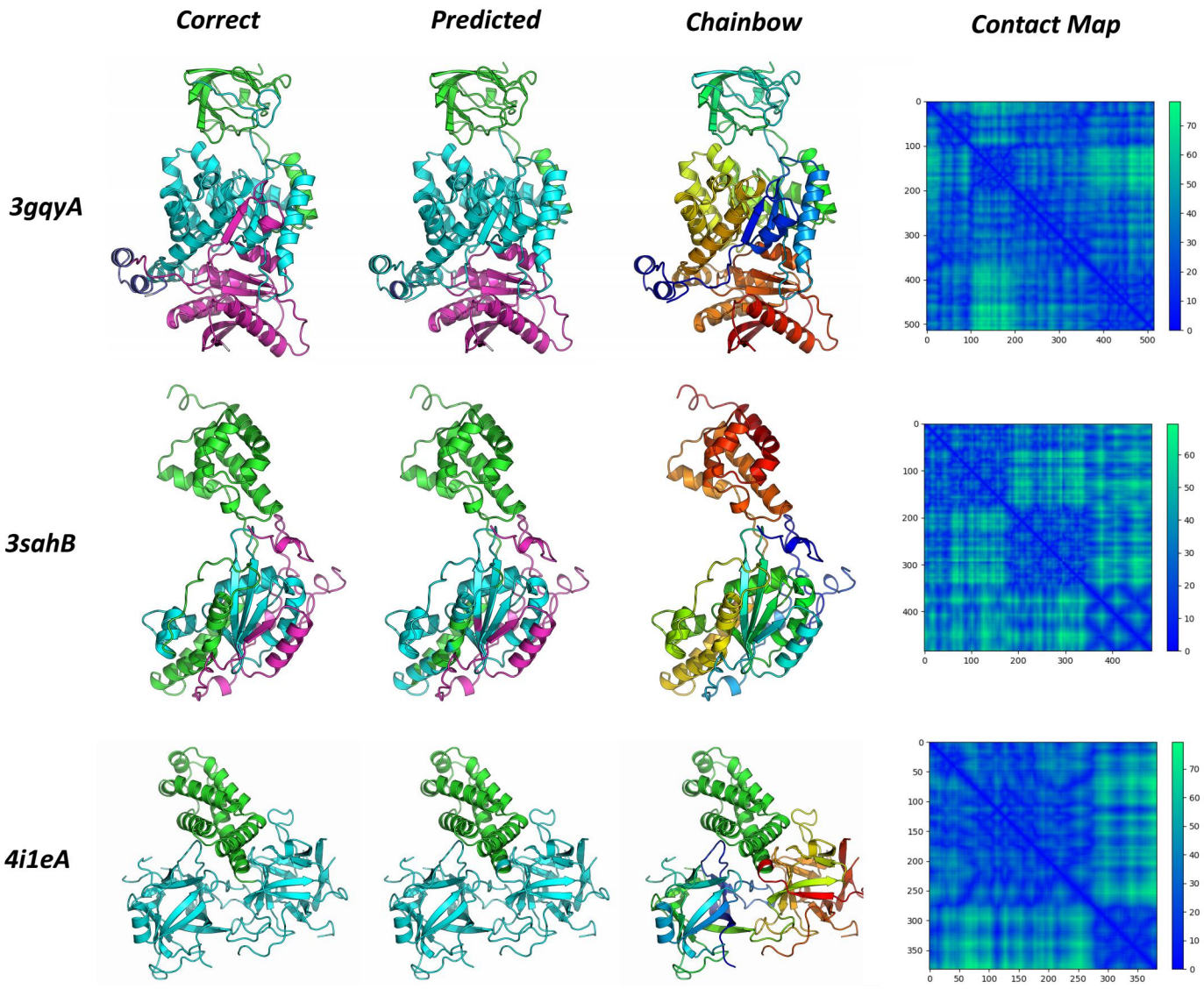


Figure 8. Segmented Examples and Contact Maps.
Segmented examples taken from the test set. From the left: (1) the correctly segmented structure, (2) the model prediction, (3) protein chain colored in residue order (rainbow order), (4) contact map in Å. 4i1eA: 99.2% position-wise accuracy. 3 domains: alpha-horseshoe (green). trefoil (cyan). 3gqyA: 86.9% position-wise accuracy. 3 domains: beta-barrel (green), alpha-beta barrel (cyan), loop (dark blue), 3-layer-αβα(magenta). 3sahB: 89.0% position-wise accuracy. 3 domains: orthogonal-bundle (green), 2-layer sandwich (cyan), loop (magenta).

5. Acknowledgements

Many thanks to Olivier Moindrot for his helpful feedback during office hours and for the idea of using column-wise LSTMs for segmentation — I tried this, and it worked however, the fully convolutional architecture performed better. Namrata Anand provided much technical advice, coding help and shared many of her own tips in handling contact map data — I highly recommend reading her paper about generative modeling for protein structures cited below. Ian Sillitoe kindly provided a list of unique protein chains in the CATH database. All code is available on Github at *egurapha/prot_arch_classifier/semantic_segmentation.*

6. References

A. Garcia-Garcia et.al. “A Review on Deep Learning Techniques Applied to Semantic Segmentation.” *arXiv:1704.06857v1 (2017)*
E. Shelhamer et. al. “Fully Convolutional Neural Networks for Semantic Segmentation.” *arXiv:1605.06211v1 (2016)*
N. Anand, P. Huang. “Generative Modeling for Protein Structures.” *ICLR submission. (2018)*
P. Wang et. al. “Understanding Convolution for Semantic Segmentation.” *arXiv:1702.08502v2 (2017)*
S. Wang et. al. “Accurate De Novo Prediction of Protein Contact Map by UltraDeep Learning Model” PLOS Comp. Bio. (2018)