



MUST-CNN: A MULTilayer Shift-and-sTitch Deep Convolutional Architecture for Sequence-based Protein Structure Prediction

Siavash Pourfallah
Amirkabir University of Technology
Winter 2024

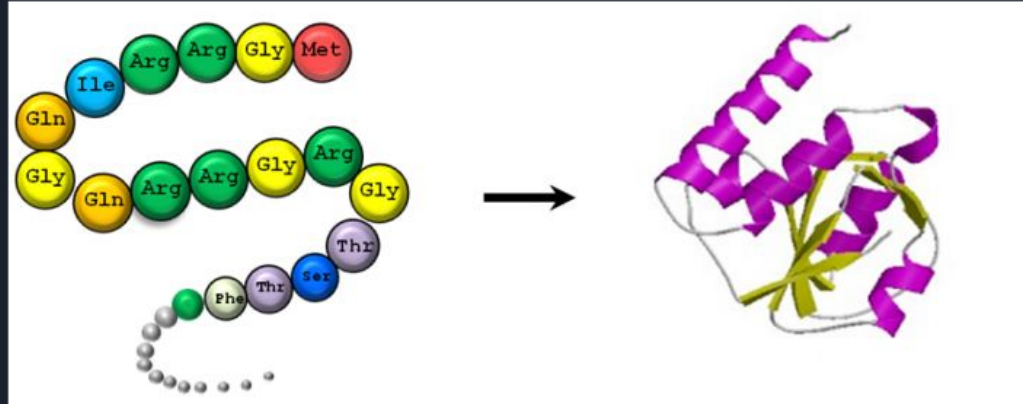



Outline

- Introduction
- Methods
- Experimental Design
- Results

Protein Structure and Function

- **Function:** involved in almost everything
- Function depends on structure
 - 3D structure
- It's very time-consuming and expensive to measure protein structures experimentally





Task: A Sequence to Sequence per-position classification task

- Input X: Primary sequence (a string of amino acids -AA)
- Output Y: Structure properties which means there are multiple output targets.
 - Such as:
 - Secondary structure
 - Solvent accessibility



Drawbacks from Previous Techniques

- Older approaches
 - PSIPred, JPred → Create sliding “windows”, predict per-position output one at a time with MLP networks
- Takes a long time to train MLP (MultiLayer Perceptron) due to millions of labeled AA positions
- Cannot model long-range structured dependencies



Proposed Method

- Beats state-of-the-art performance
- Fast training and testing, simple and scalable algorithm
- Generic enough to be applicable on other per-position labeling problems



Outline

- Introduction
- **Methods**
- Experimental Design
- Results

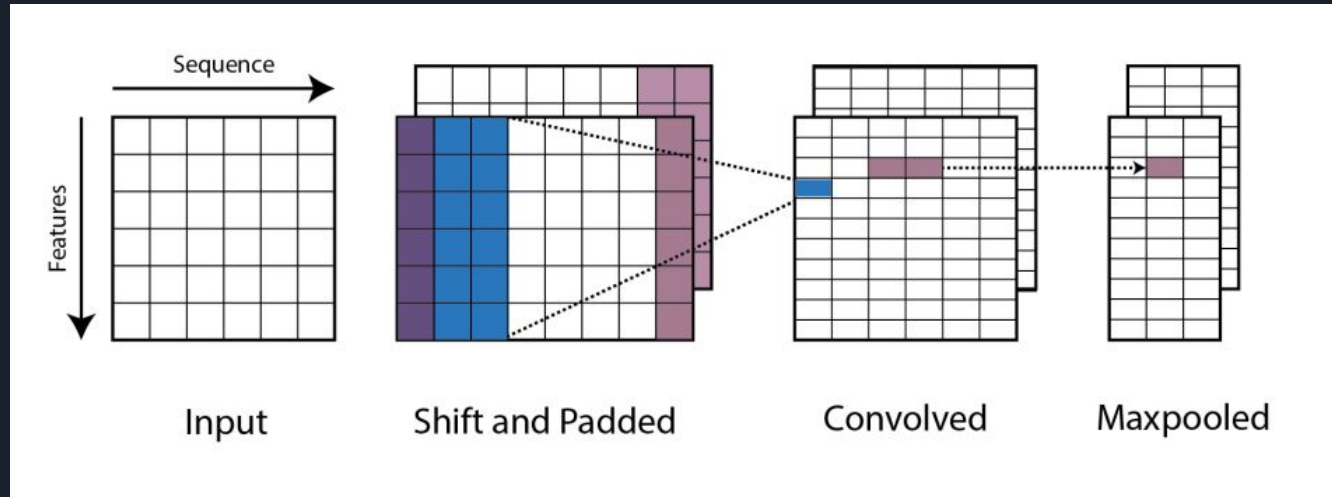


Proposed Solution: MUST-CNN

- How does it remove previous drawbacks?
 - Long time to train → convolutional architecture + GPU
 - Cannot model long-range structured dependencies → deeper models

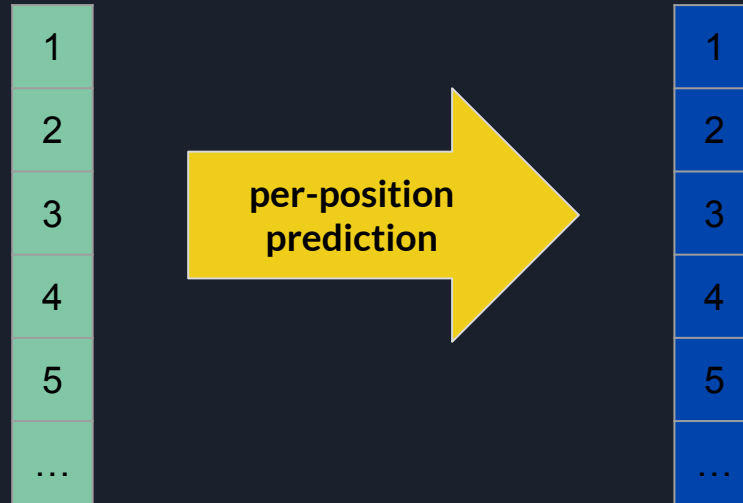
Convolutional Neural Networks (CNNs) for Sequences

- Drawback → CNN's pooling step leads to a decreased output resolution.



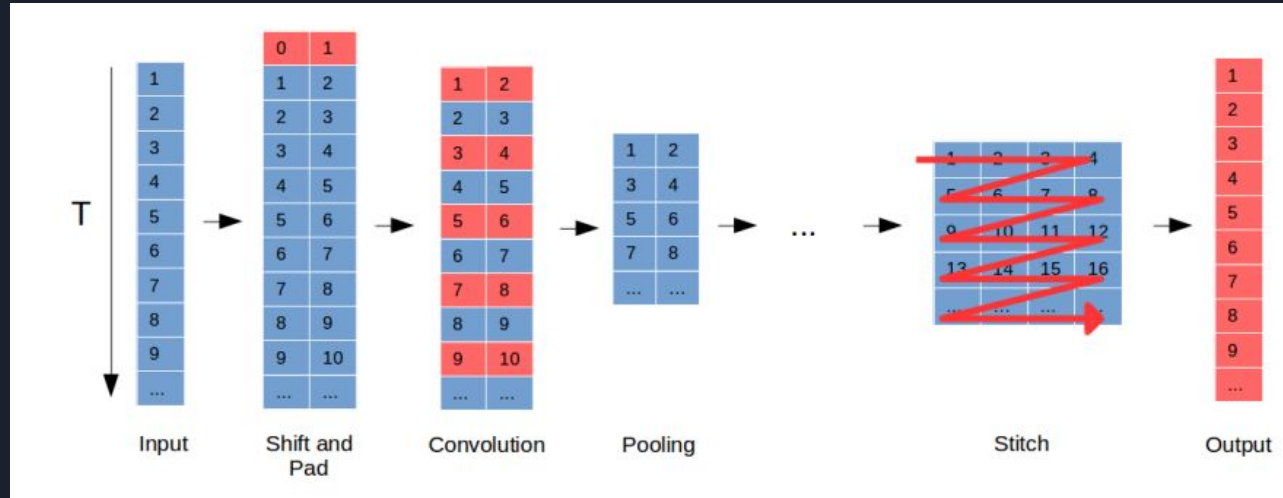
CNN for Sequence Input and Output

- First Issue: Boundary Positions
- Second Issue: Per-position Sequence to Sequence Classification



MUltilayer Shift-and-sTitch (MUST)

- MUST allows us to tag every element of an input with a multilayer CNN all at once

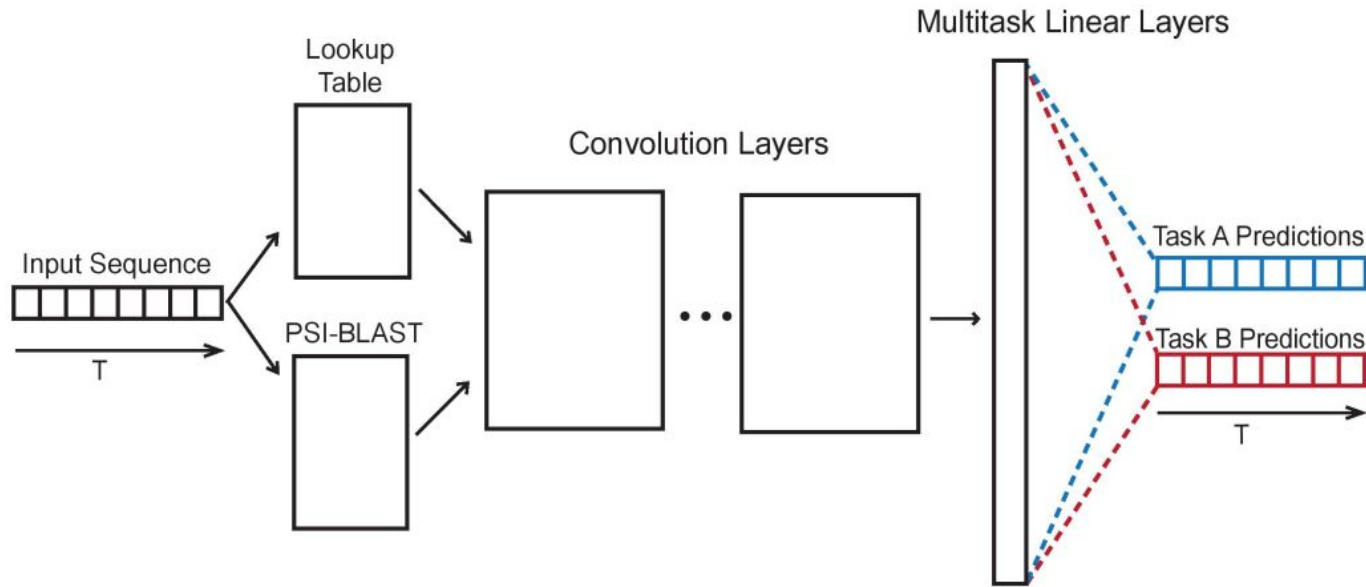




Advantages

- All operations are implemented easily.
 - Shift and pad is just duplication, zero padding and concatenation
 - Stitching is just a vector reshape
- Fast, batched computations
 - Shift-and-stitch allows us to run batches of convolutions.
 - Output predictions for all positions at once

End-to-End Architecture





Connection to Previous Methods

- Conditional Neural Field (CNF) (Wang, 2011)
 - Conditional Random Field with a NN feature extractor
- Generative Stochastic Network (GSN) (Zhou, 2014)
 - Trained similar as Restricted Boltzmann Machine
 - Slower convergence
- OverFeat (Sermanet et al. 2013)
 - Introduced shift-and-stitch, on per-pixel scene labeling
- LSTM
 - Protein sequences have no innate direction



Outline

- Introduction
- Methods
- Experimental Design
- Results



Implementation

- Each convolution layer also includes a nonlinearity
 - tanh
 - ReLU
 - PReLU
- Pooling
 - Maxpooling specifically



Implementation

- Dropout: Randomized masks of outputs
 - Acts as a regularizer and prevents overfitting
- Stochastic gradient descent for optimization
- Hardware
 - All training and testing uses a Tesla C2050 GPU unit
- Feature input of each amino acid
 - (AA embedding + PSIBlast)



Implementation

- Hyperparameter tuning
 - Bayesian optimization
- Multi-task learning
 - Negative log-likelihood summing over all tasks and all elements in the sequence



Two large-scale Datasets

Data Name	Train Size	Validate	Test
4prot	1.50 million	0.51 million	0.51 million
CuIPDB	0.95 million	0.24 million	85,000



Outline

- Introduction
- Methods
- Experimental Design
- Results



Results on 4prot and CullPDB datasets

- 4prot
 - Qc accuracy used for measurement
- CullPDB
 - Model is extremely simpler than CNF, GSN and LSTM
 - Trained on CullPDB, tested on CB513



Conclusion and Discussion

- **Robustness:** Same model does well on two different large datasets
- **Speed:** Predictions for half a million samples under two seconds
- **Extendability:** As long as input and output sequence lengths match up