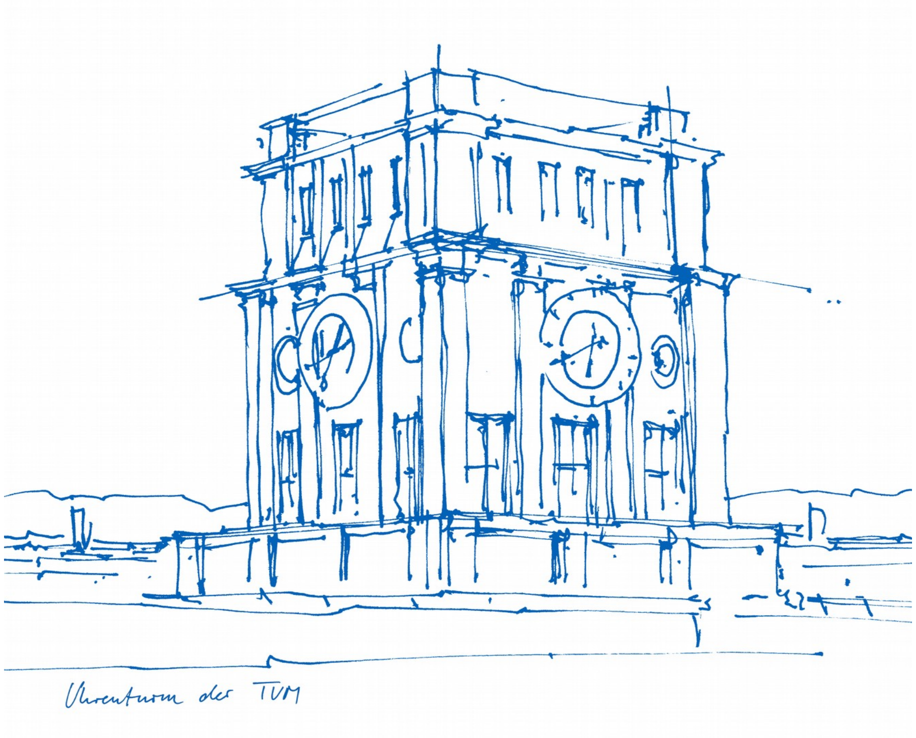
Combining Word Embeddings of Protein Sequences with Evolutionary Information for Secondary Structure Prediction Using Deep Neural Networks



Anna Reithmeir

Introduction to Bachelor‘s thesis

Advisor: Michael Heinzinger

28th November 2018

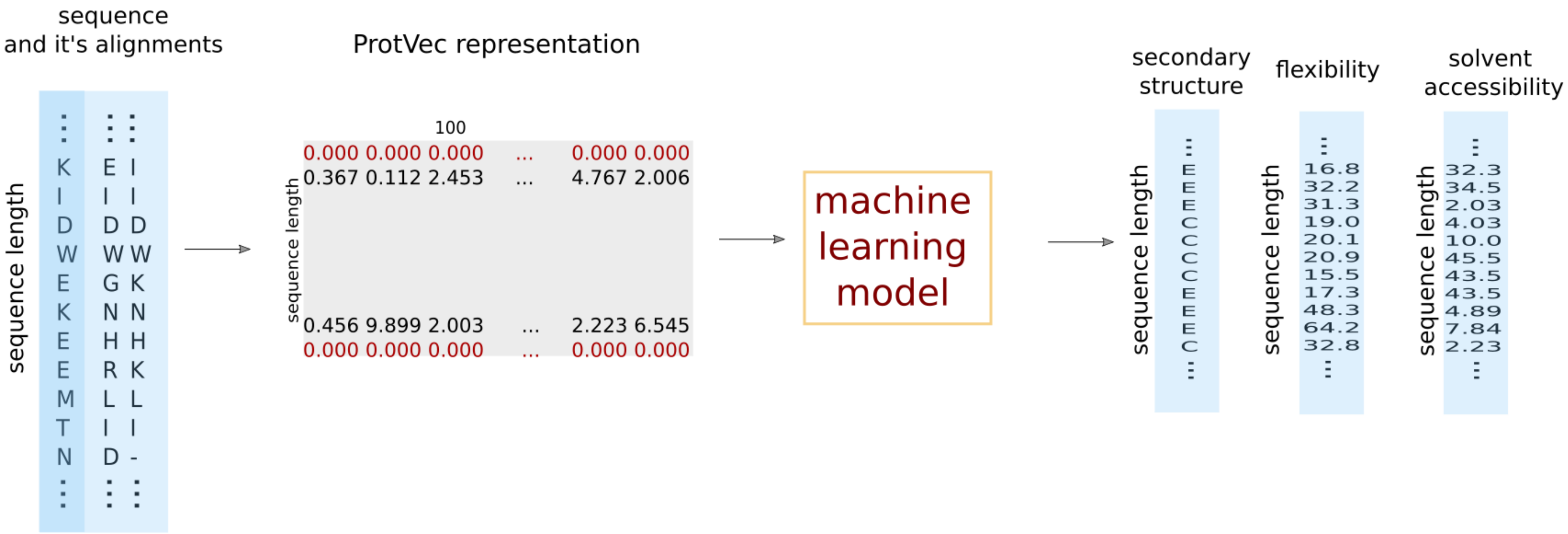
Overview

- Project introduction
- Dataset
- Data preprocessing
- Convolutional neural networks
- Results so far
- Outlook

Project Introduction

- Secondary structure prediction with neural networks
- Integrate evolutionary information in input
- Multi target prediction
  - Secondary structure
  - Solvent accessibility
  - Flexibility
- ProtVec representation on residue basis

Project Introduction



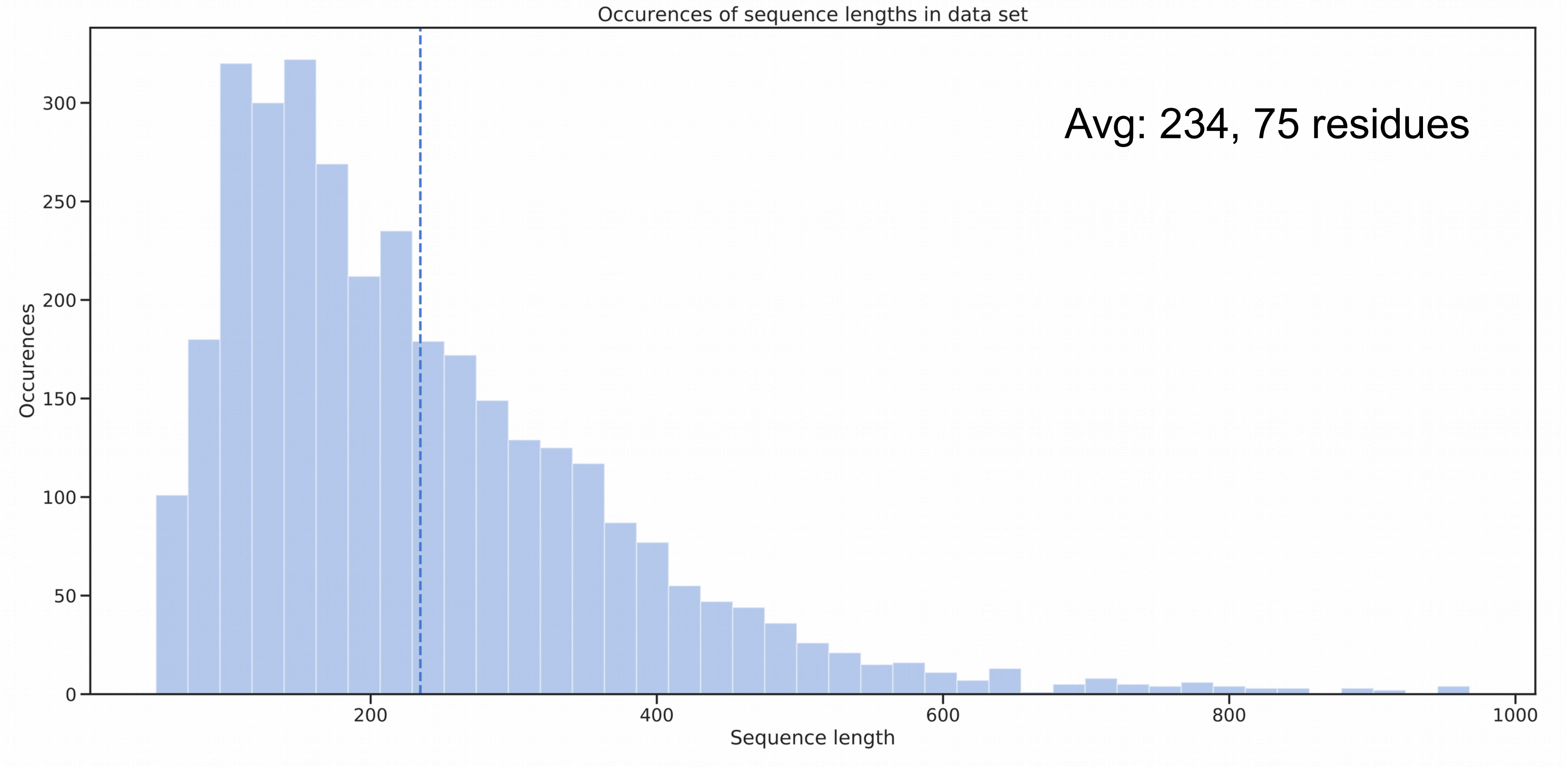
Dataset

- 3313 protein sequences from PDB

- According structure information, solvent accessibility and flexibility from DSSP algorithm

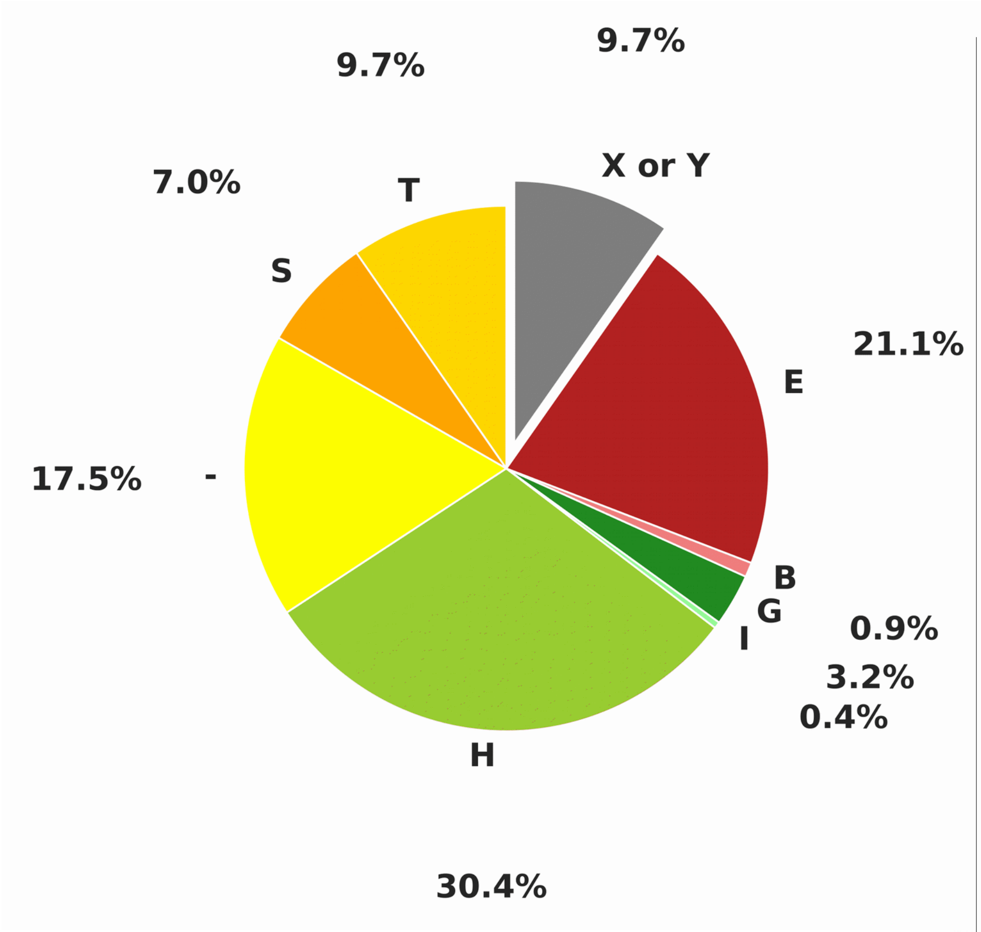
- Aligned sequences for each protein sequence

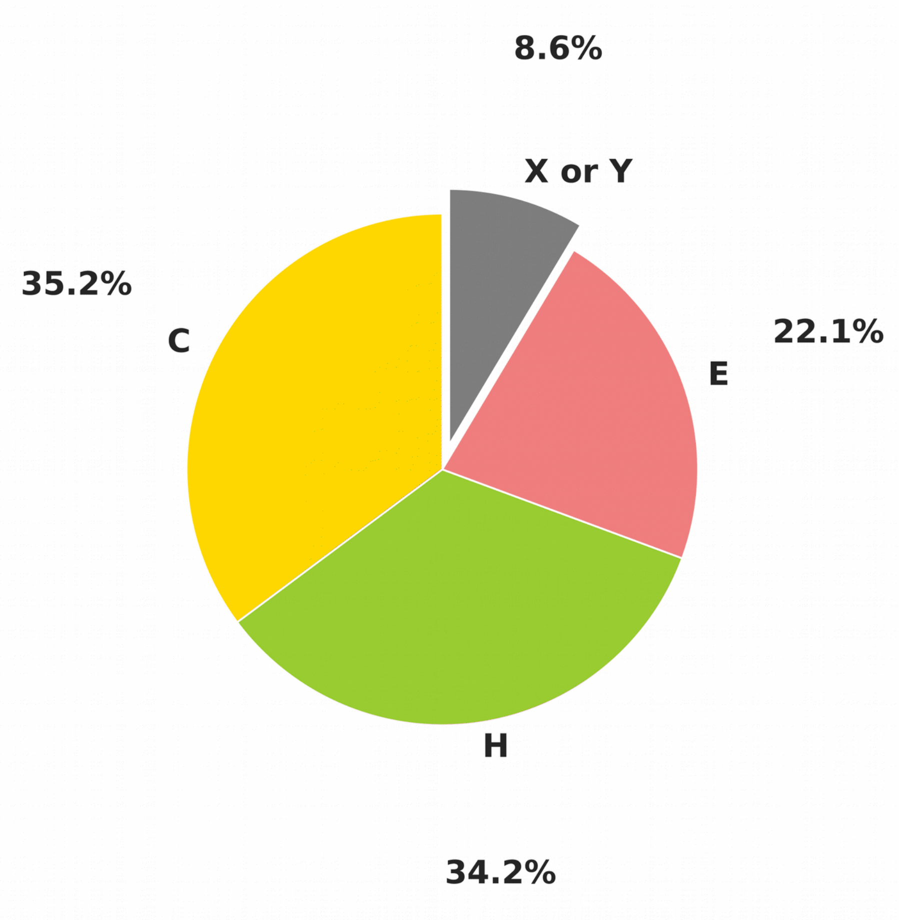
Dataset – Sequence Lengths



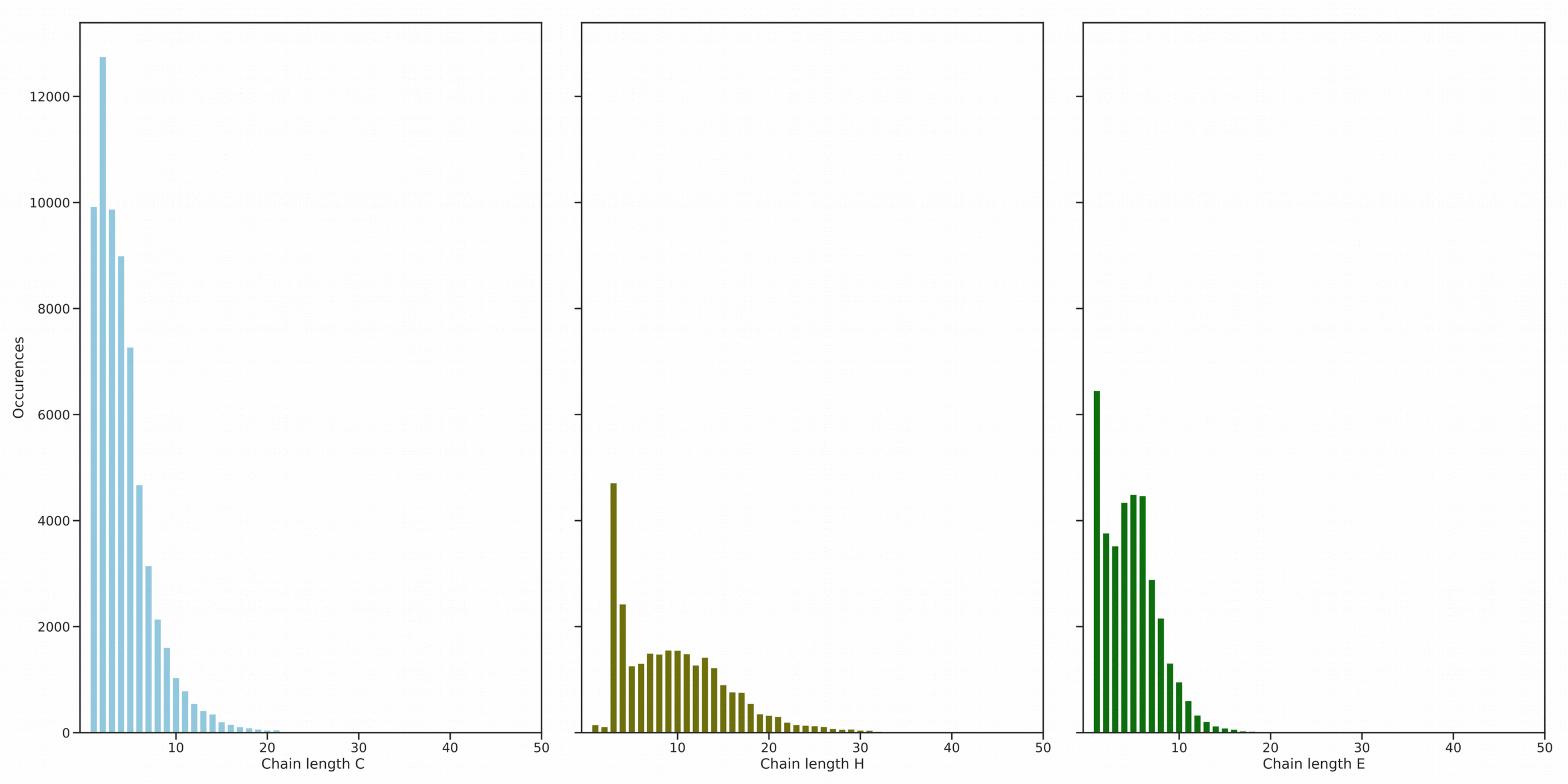
Avg: 234, 75 residues

Dataset – DSSP States





Dataset – DSSP-3 States Chains



Avg: 4,2

Avg: 9,8

Avg: 4,7

Dataset – Training, Validation, Test Set

- 20% of the samples → test set (649 samples)
- Remaining 80% → train and validation set (2092 and 529 samples)
- Random selection, assured equal distribution

Implementation So Far

- Training and testing pipeline implemented

- Simple CNN used
  - 3 layers
  - Kernels: 1, 7, 15
  - NLL loss + log_softmax

- Secondary structure prediction without evolutionary information

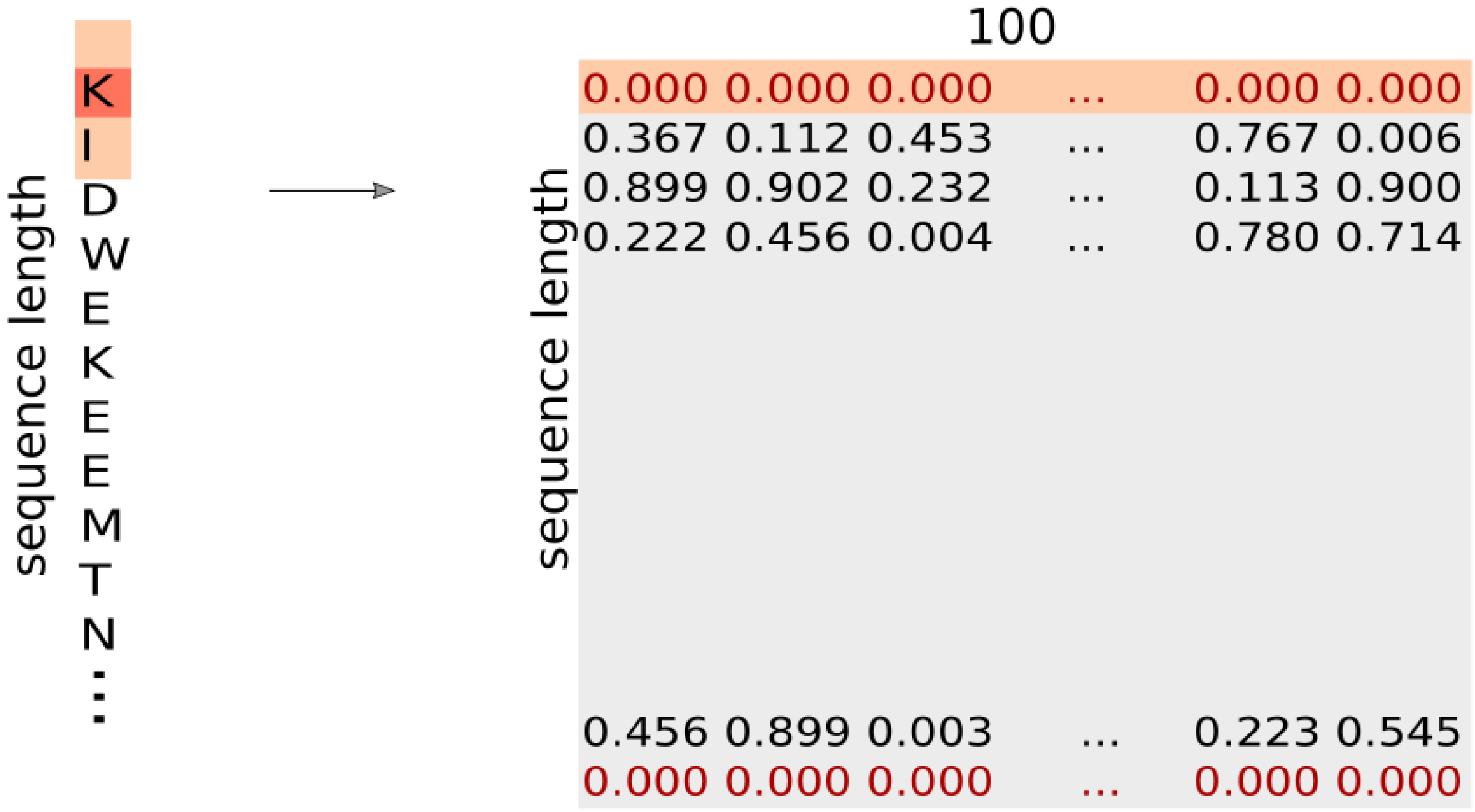
Data Preprocessing

- Protein sequences represented through word embeddings

- ProtVec
  - Vector embeddings learned through Word2Vec
  - Vector for each possible n-gram, fixed length

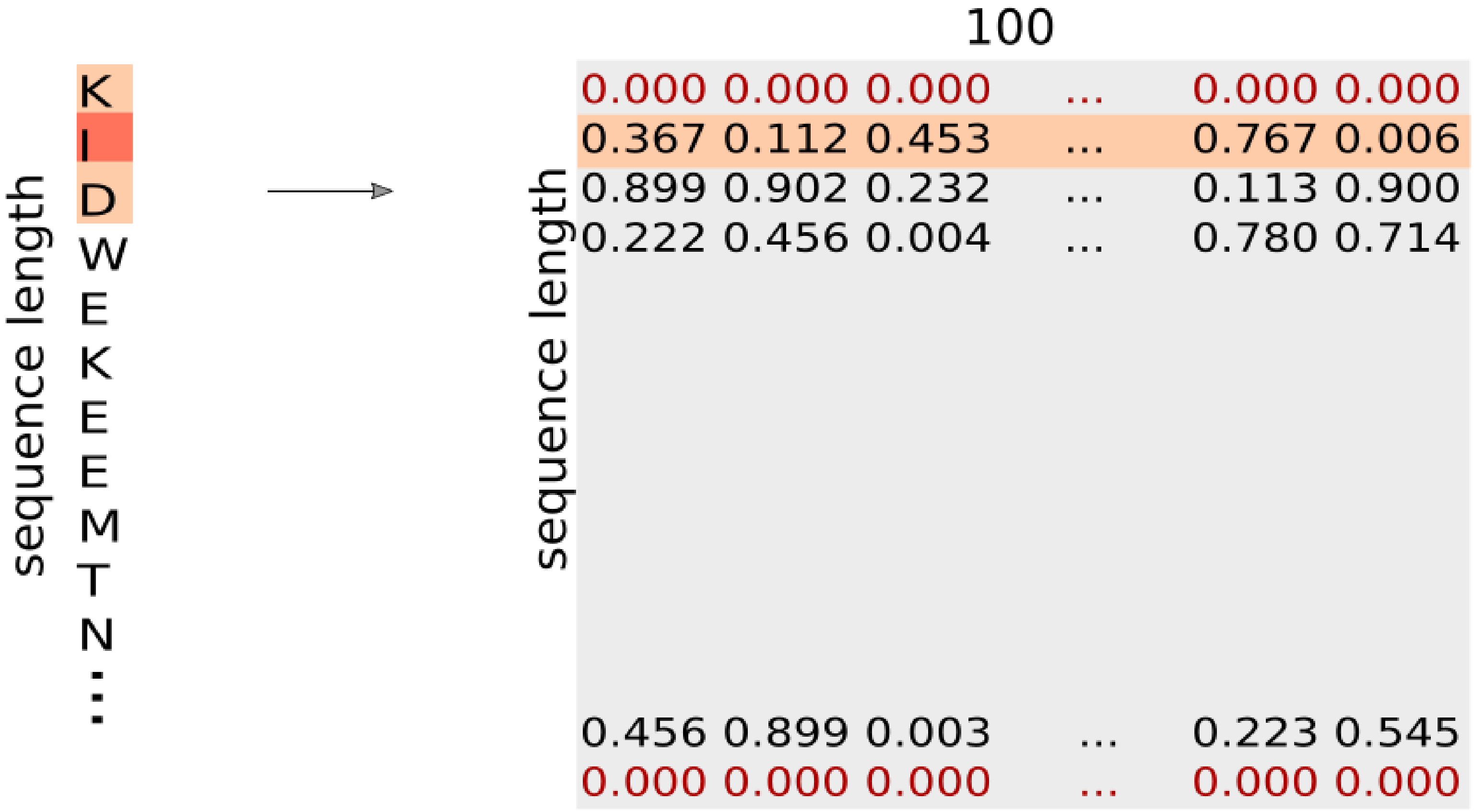
- 3-grams, length 100

Data Preprocessing



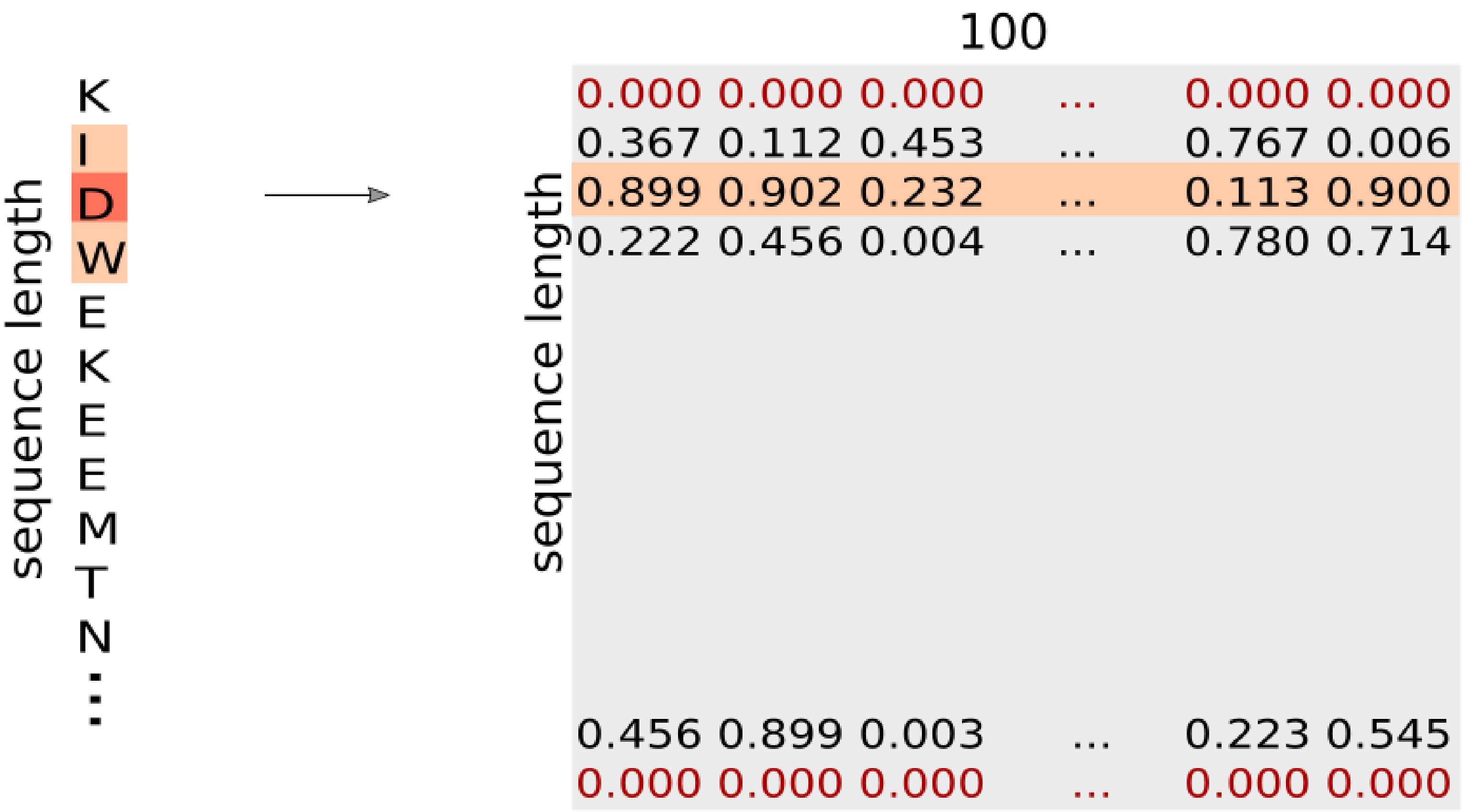
0.000	0.000	0.000	...	0.000	0.000
0.367	0.112	0.453	...	0.767	0.006
0.899	0.902	0.232	...	0.113	0.900
0.222	0.456	0.004	...	0.780	0.714
0.456	0.899	0.003	...	0.223	0.545
0.000	0.000	0.000	...	0.000	0.000

Data Preprocessing



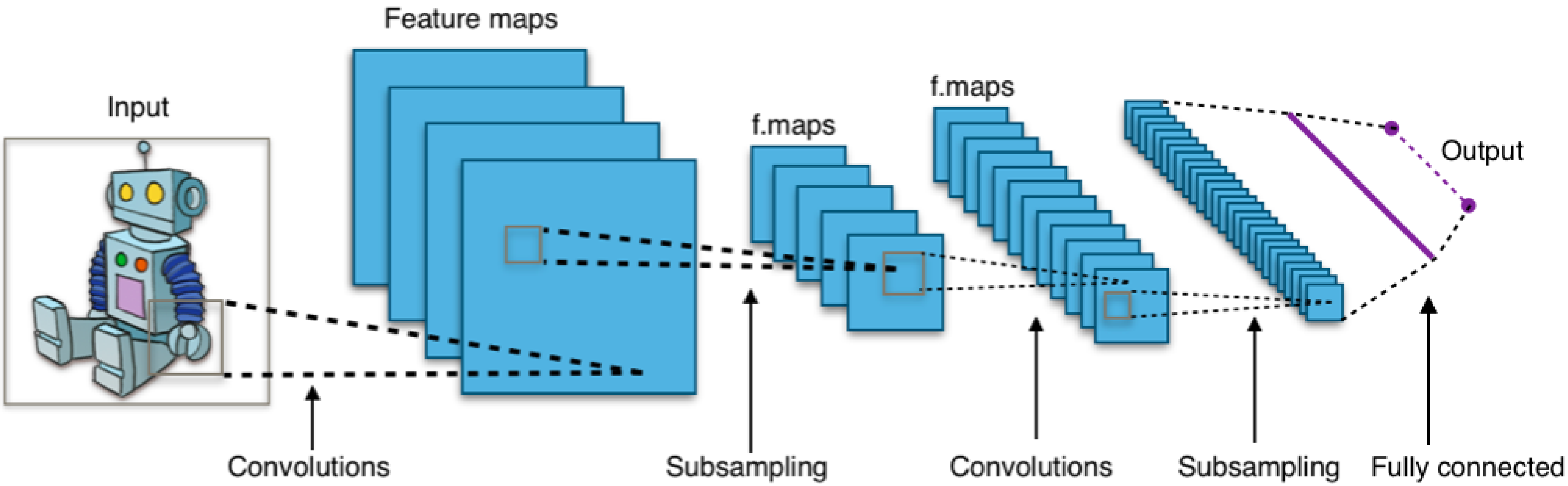
0.000	0.000	0.000	...	0.000	0.000
0.367	0.112	0.453	...	0.767	0.006
0.899	0.902	0.232	...	0.113	0.900
0.222	0.456	0.004	...	0.780	0.714
0.456	0.899	0.003	...	0.223	0.545
0.000	0.000	0.000	...	0.000	0.000

Data Preprocessing



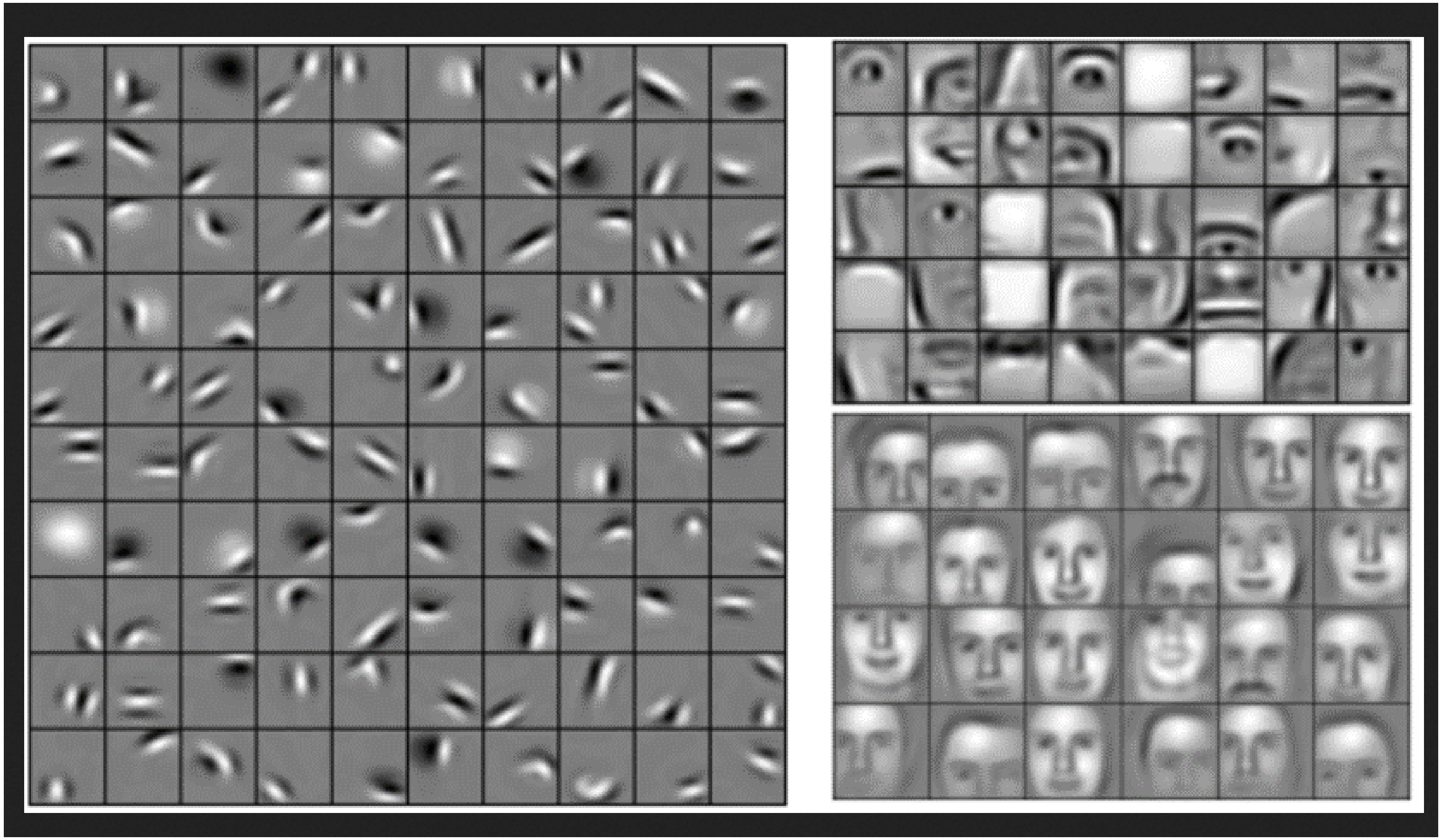
0.000	0.000	0.000	...	0.000	0.000
0.367	0.112	0.453	...	0.767	0.006
0.899	0.902	0.232	...	0.113	0.900
0.222	0.456	0.004	...	0.780	0.714
0.456	0.899	0.003	...	0.223	0.545
0.000	0.000	0.000	...	0.000	0.000

Convolutional Neural Networks



https://de.wikipedia.org/wiki/Convolutional_Neural_Network

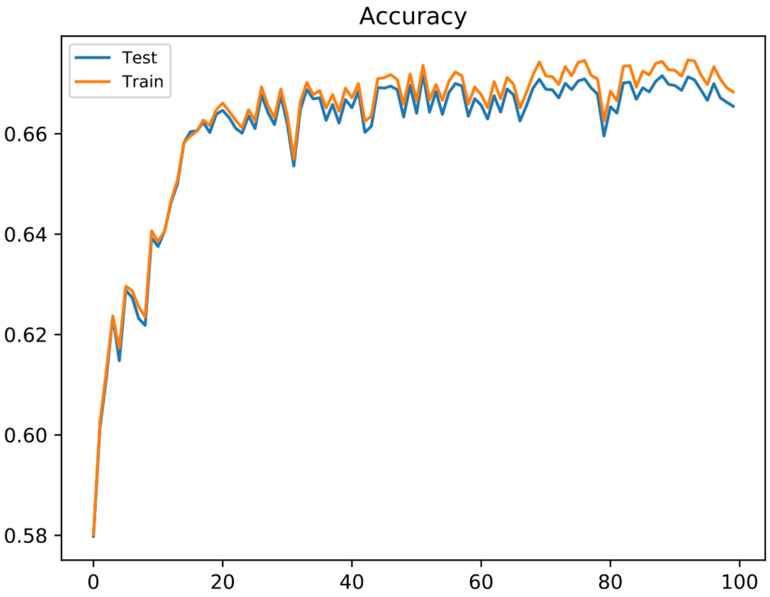
Convolutional Neural Networks



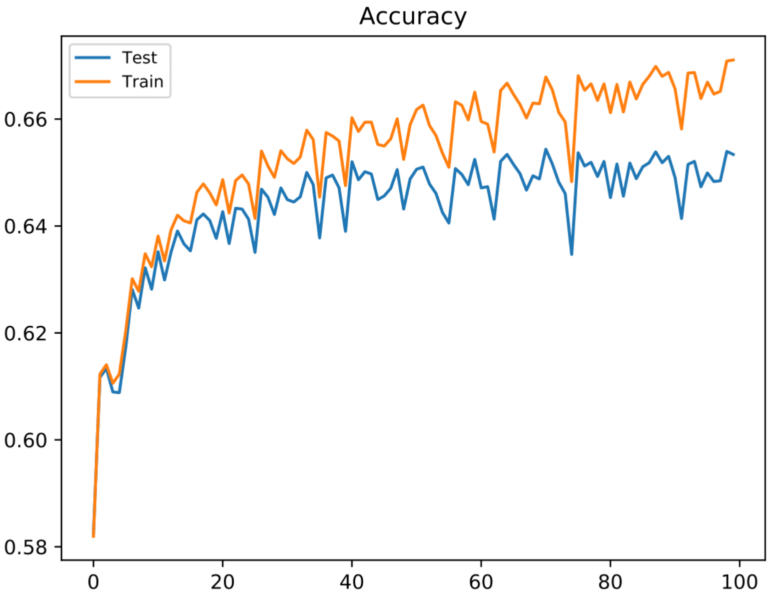
https://brohrer.github.io/images/cnn18.png

Results So Far – 1hot & ProtVec Accuracy

1hot

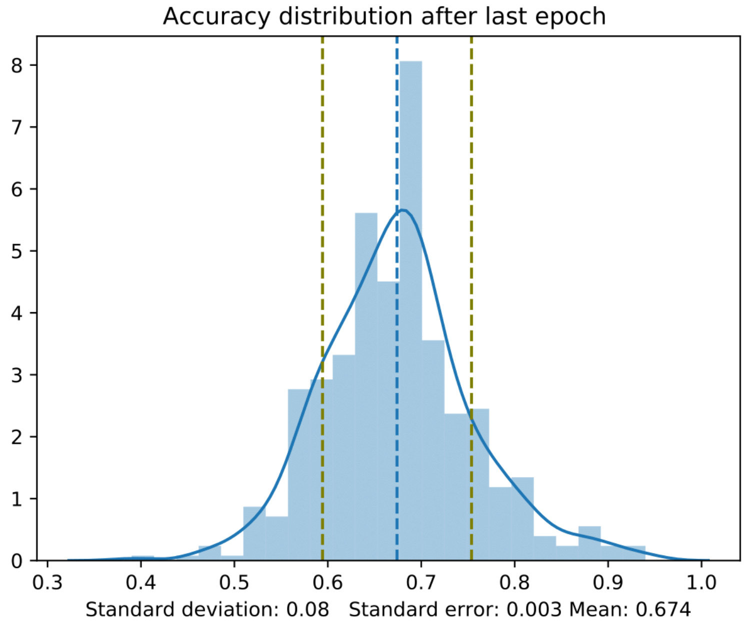


ProtVec

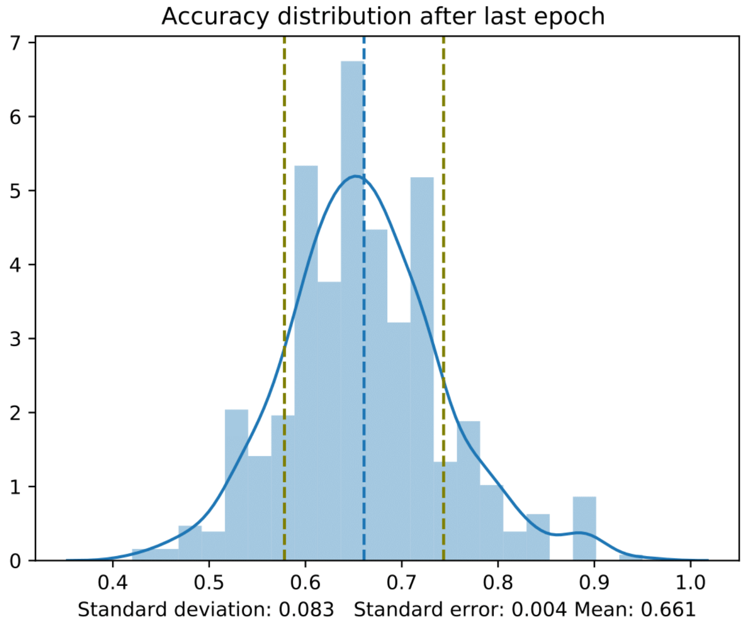


Results So Far – 1hot & ProtVec Accuracy

1hot



ProtVec



Outlook

- Incorporate evolutionary information
- Flexibility and solvent accessibility prediction
- Tune hyperparameters
  - Kernel sizes
  - Regularization
  - Non-linear activation functions e.g. SeLU
- LSTM networks
  - Capable of learning long-term dependencies

Thank you for your attention!