SomeNA

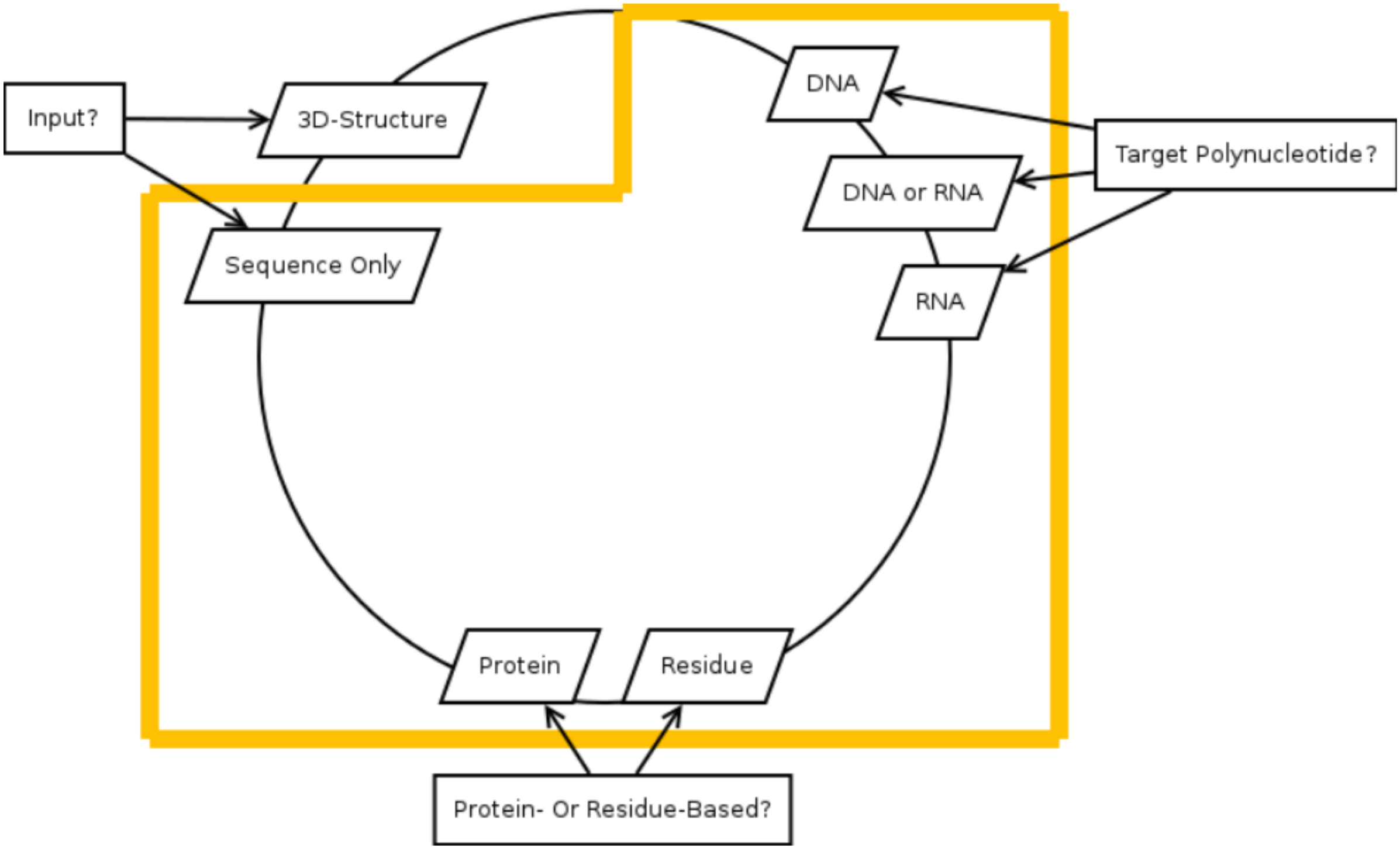
Polynucleotide-Protein Binding

- DNA & RNA molecules interact with proteins
  - transcription factors
  - polymerases
  - nucleases
  - histones
  - splicing

Novelty of the Method

- predicting from sequence alone
- predicting for DNA & RNA and non-binding
- creating protein-based predictions out of residue-based ones

Novelty of the Method



Datasets

1. proteins, which are shown experimentally to bind DNA molecules
2. proteins, which are shown experimentally to bind RNA molecules
3. a negative set, i.e. proteins, which are shown not to bind polynucleotides

Datasets – DNA-Binding

- extracted from the Protein-DNA Interface Database (PDIDB)
- processed to determine which protein residues bind DNA and which do not
- minimum length of 45 proteins
- Remove overrepresented families of protein classes
- 144 protein chains which are binding DNA

Datasets – RNA-Binding

- extracted from the Protein-RNA Interface Database (PRIDB)
- processed in an analogous manner
- 102 protein chain in the RNA-binding dataset

Datasets – Non-Binding

- entries in SwissProt are annotated using the Gene Ontology
- direct search and search via regular expressions
- 464 protein chains in the dataset

Features

- sequence-based features & extracted from other sequence-based prediction tools

- sequence-based are local (residue charge, mass) & global (amino acid composition, length)

- extracted are secondary structure, solvent accessibility, disordered regions, etc.

Features:

Mass, volume, hydrophobicity, charge, polarity, helix-breaker, c-beta, distance from the N- and the C-terminus, amino acid composition, length, secondary structure, etc.

Features Selection

Greedy Forward Selection

- First round, all features are tested for their performance individually and the feature used by the best performing model is selected

- Next round, this feature is used along with all remaining features individually again

- this routine is continued as long as the performance raises, which implies that the added feature provides additional signals to the machine- learning algorithm

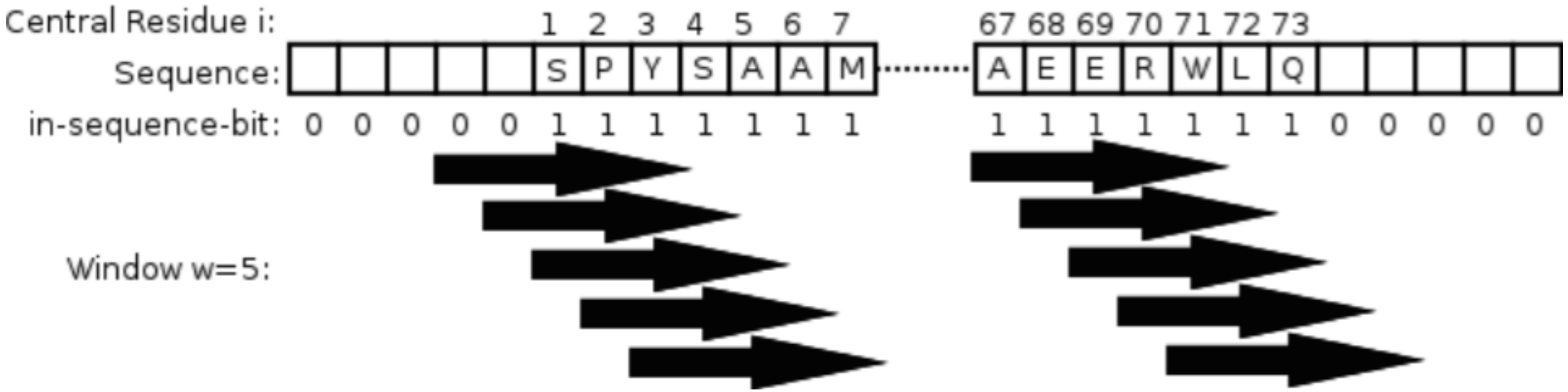
- during the global forward selection, all features described in the methods section were selected except the disorder prediction, and the coiled-coils feature

Sliding Window

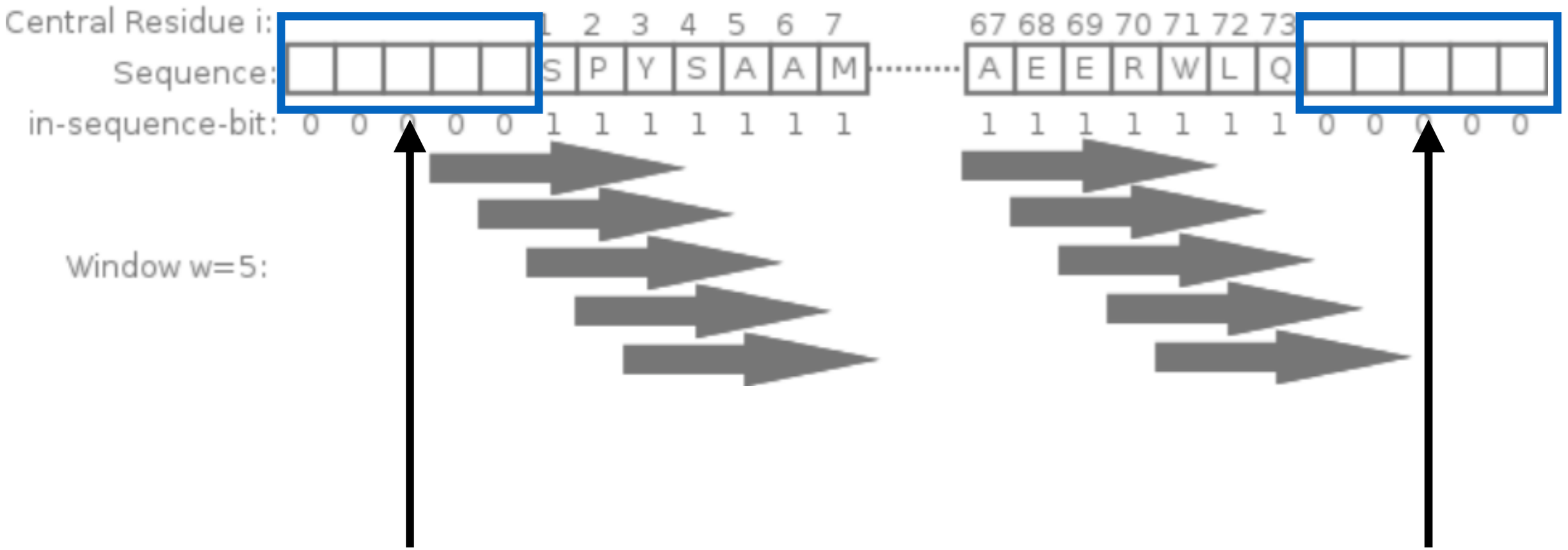
- early sequence-based prediction methods took only the residue itself into account, so none of the information about the local surrounding of the predicted residue is taken as input to improve prediction quality

- to handle this issue, a sliding window approach is used

Sliding Window

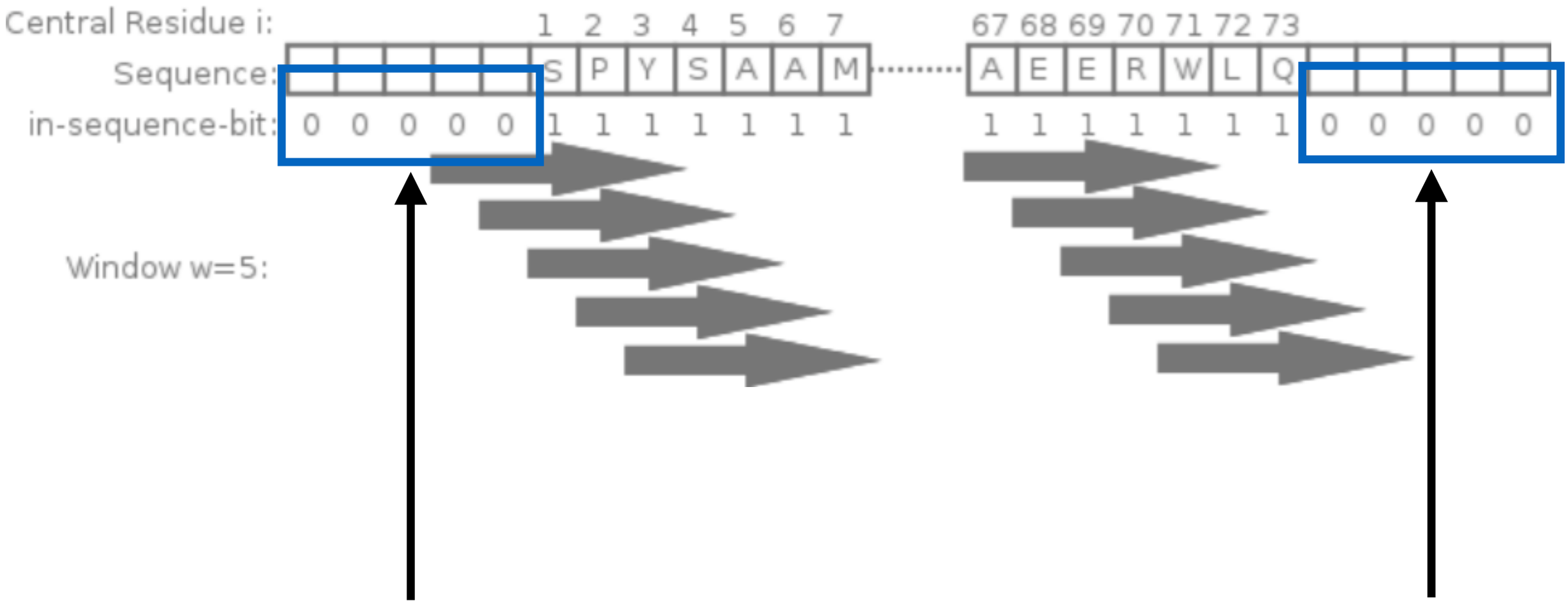


Sliding Window



Non-existent positions at the start and at the end

Sliding Window

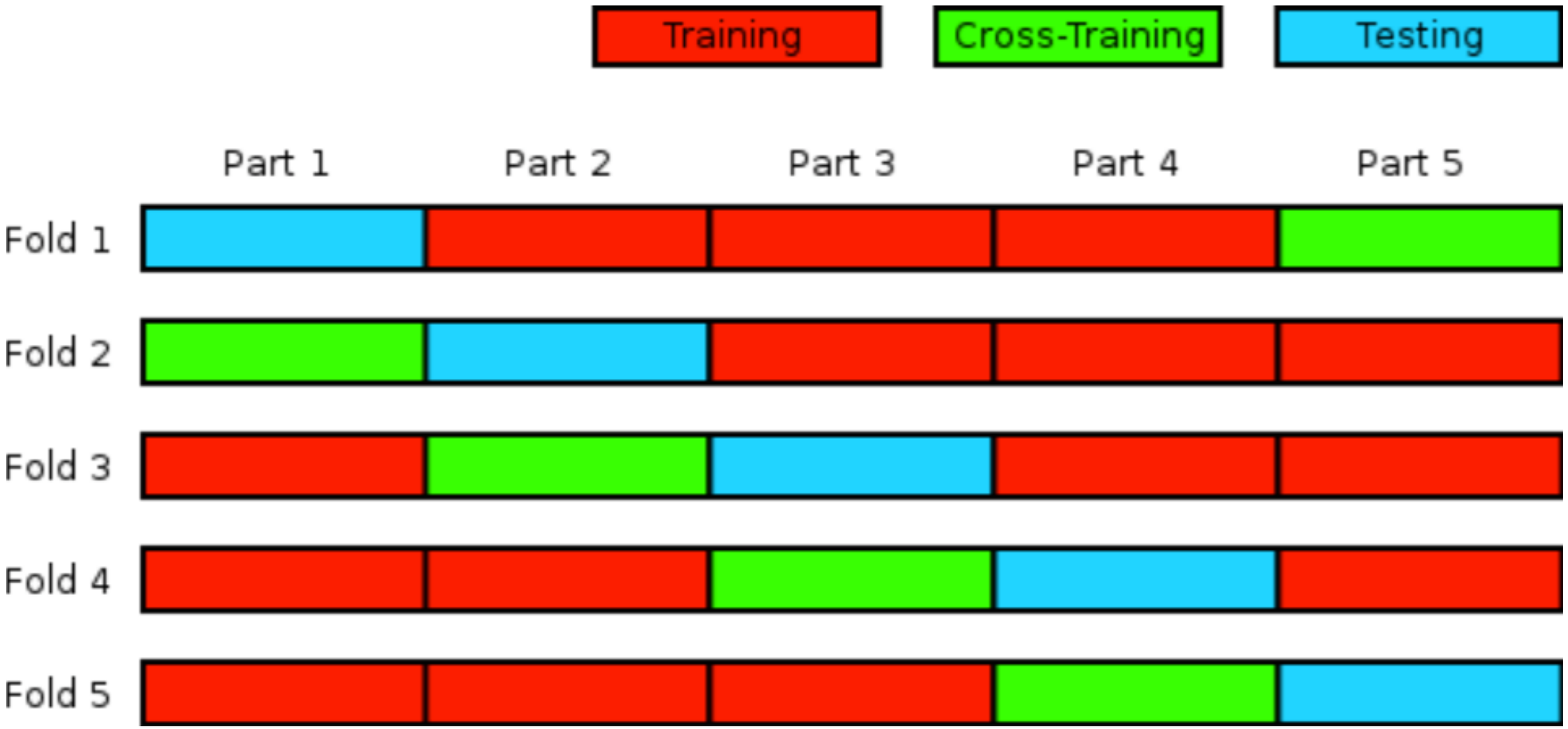


"In-sequence-bit"

Cross-Validation

- a 5-fold cross- validation is used to evaluate the best parameter and feature combinations
- for the 5-fold cross-validation, the dataset is split into five equal-sized parts, n = 1 to 5
- in the fold n, the parts n to (n+2) modulo 5 are taken for training the model, which means the data points included in those parts are presented to the learning algorithm
- the part (n+3) modulo 5, namely cross-training data, is used to determine when to stop training and to evaluate the performance between the models trained with different parameter/feature combinations to select the best one
- the (n+4) modulo 5 part of the data is for testing the final performance of the beforehand selected model
- the average performances on the testing data of all five folds add up to the final performance of the predictor

Cross-Validation



Training Stop Criterion

- a trained model could memorize the training data and therefore be over-fitted

- to avoid this situation, the training has to be stopped at an earlier point, where the model sufficiently fits the training data, but is also applicable to data describing the same problem

- after each training epoch, which means after all training samples have been presented to the learning algorithm, the model's performance is validated on the cross-training data

Training Stop Criterion

Training Stop Criterion

- at first, the model performs better with each epoch on the training data as well as on the cross-training data, however at some point, the performance on the training data increases further, while the performance on the cross-training data decreases

- at this point, the training is stopped and the model performing best on the cross-training data is chosen to be the final model for this feature/parameter combination

- to avoid local maxima in cross-training data's performance, the training is continued for 10 further epochs looking for other, even higher maxima

Output

- the neural network has two output nodes to represent two states, or classes: binding and non-binding

- these nodes, o1 for binding and o2 for non-binding, can range each from 0 to 1 (o1=1 and o2=0 in the case of a binding residue, and o1=0 and o2=1 otherwise)

- a one-valued score is introduced using score = o1 - o2

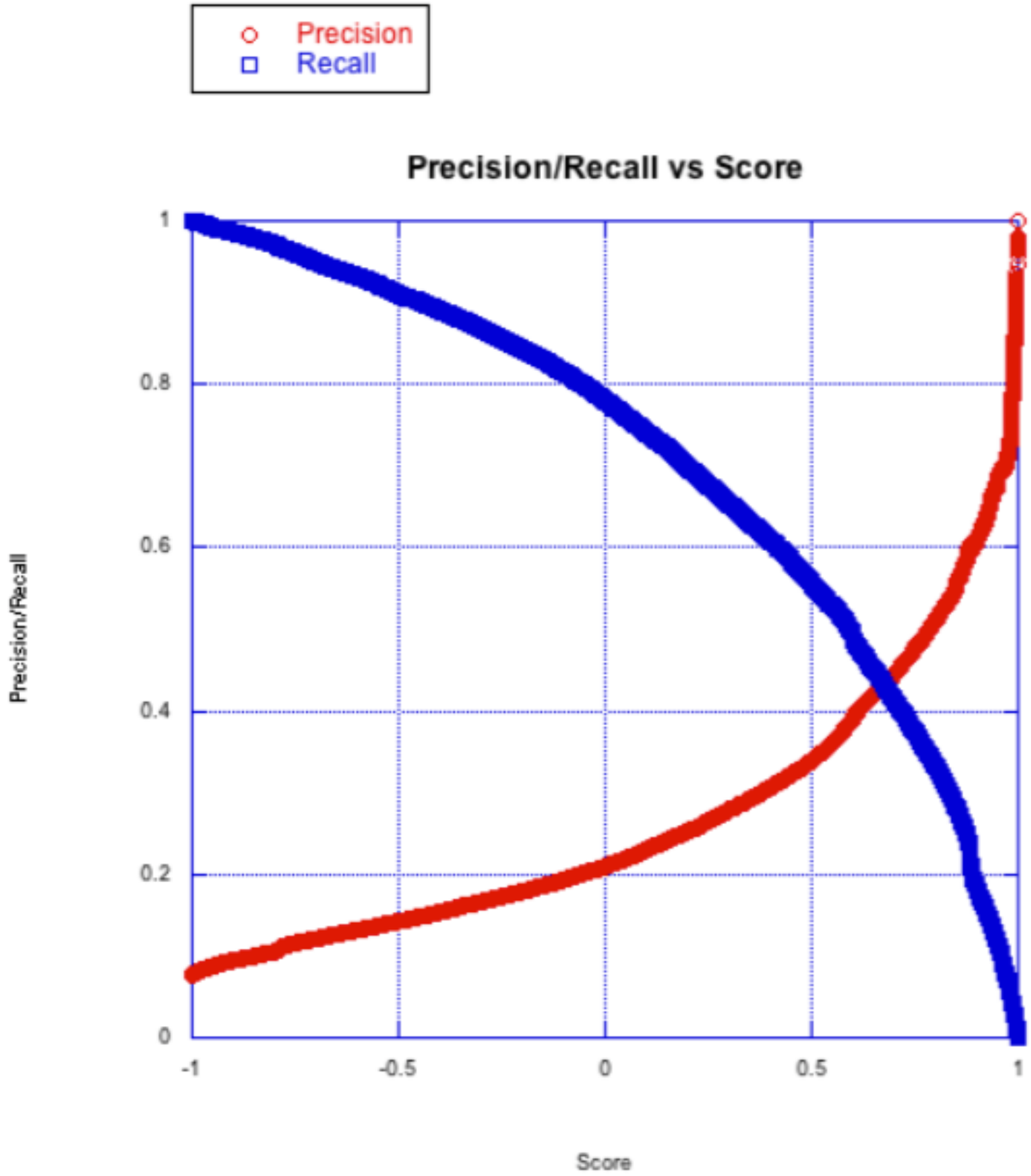
- for a very sure prediction towards the class “binding”, the score will be nearly 1, while the score for the class “non-binding” should approximate -1.

- a benefit of reducing the number of output values from two to one is the possibility of shifting the threshold for the predictions – e.g., if only very precise binding predictions should be accepted, the threshold can be adjusted towards 1 (e.g. 0.7), resulting in an increase of precision for the binding class, at the cost of its recall

Threshold Selection

- after what score should we regard result as binding or non-binding?
- using a threshold of 0 often results in a biased prediction in either the direction of a good recall or a good precision regarding the binding class
- to balance this out, the precision and the recall are plotted against the score to decide at which score both of those measures are similarly high

Threshold Selection



Threshold Selection

- assuming a score of 0 the recall is fairly near 0.8, at the cost of a very low precision
- if the score threshold is adjusted to about 0.7, where the recall and the precision lines meet, the recall decreases, while a significantly higher precision is gained
- using this method results in a more balanced prediction regarding precision and recall, in comparison to setting the threshold arbitrarily by hand

From Residue to Protein-Based Prediction

- transform the residue-based prediction into a protein-based one

- a protein is declared as binding a polynucleotide if at least one residue is an effective interacting residue

- a protein where the method predicts a polynucleotide-binding residue should be a binding protein, too, however, since the residue-based prediction is not perfect, FPs would cause a high number of falsely positive predicted binding proteins

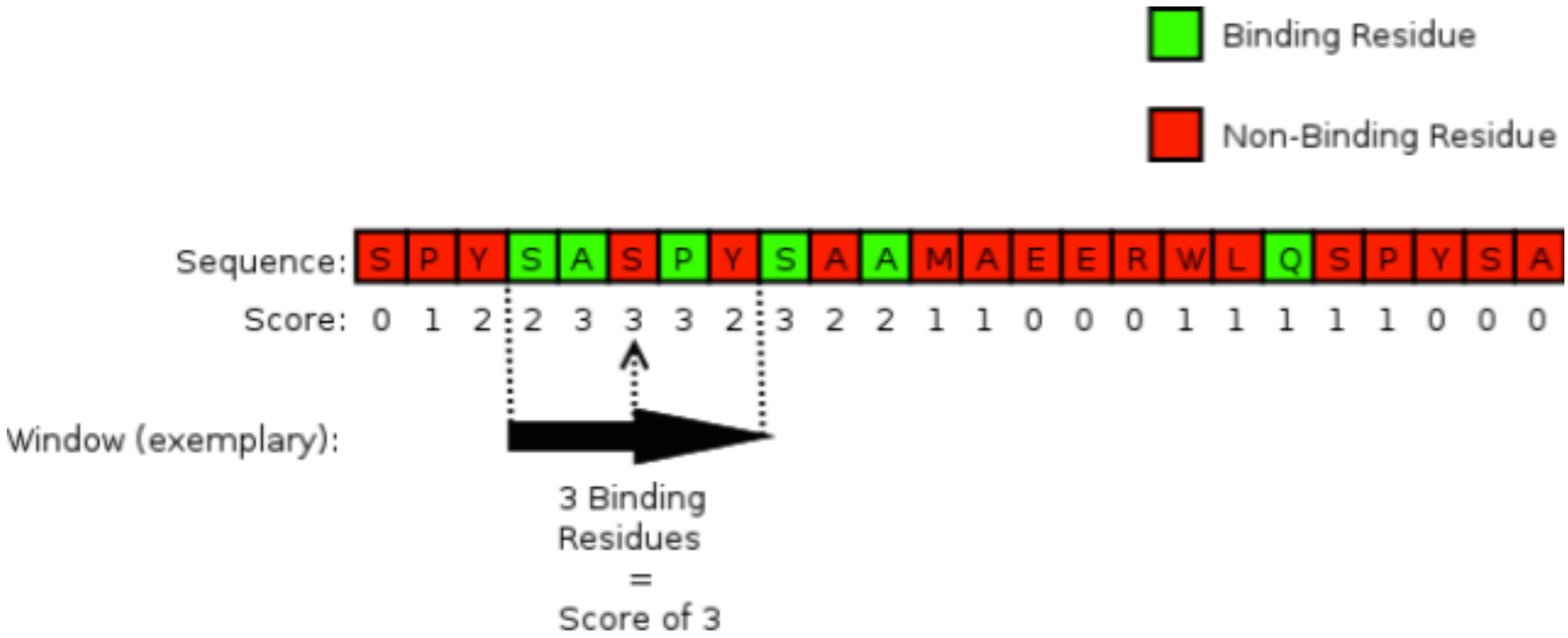
From Residue to Protein-Based Prediction

- further investigations of the distribution of binding residues show, that they in fact are not correlated with the length of the proteins, but they tend to accumulate on the protein's sequence forming binding regions

- to benefit from that circumstance, the proceeding is to score residues predicted as binding better if they have other binding residues near them

- because binding residues, which are far away in sequence can be close in 3D, binding residues appearing separate also contribute to the score, although not with such a high impact

From Residue to Protein-Based Prediction



From Residue to Protein-Based Prediction

- the final solution is a window approach, which incorporates clusters of binding residues for the scoring

- every residue in the protein receives a score that depends in the number of binding residues found in a window of 5 around the current scored residue, including itself

- thus, this value can range from 0, meaning no binding residue has been predicted in this window, to 5, implying all residues in the window are predicted as binding

- after this step, the score from every residue is added up to the final score

Training Workflow

- The training of the method involves the interlocking of several neural network models and datasets, each tailored for its specific task in the whole training pipeline. The first steps consist of data gathering and the extraction of the target classes, in this case binding DNA, RNA or none of them. After this, these three data sets are split into the five parts for the cross-validation. Then, three clean datasets are created, each containing one of those target classes. The following

- stages refer to all folds created by this split, it has to be noted that from this procedure on every fold is a self-contained system never influencing, or getting influenced by one of the other folds.

- Because the machine learning method has to be capable to distinguish between different positive classes, those data sets are merged in different ways to create three further data sets, which, in fact, contain the same proteins, but their residues have different labels according to the current sub-problem, which has to be solved. One of those data sets has only the DNA binding residues labeled as binding. The second data set of those three has only the RNA binding residues labeled as binding, while in the third set the DNA as well as the RNA binding residues are the positive cases. As can be seen in Figure 16 there is also a clean DNA as well as a RNA binding set for a more precise residue-based binding prediction once a protein has been declared as polynucleotide binding.

- After creating the features for each protein, the following step is the most computational intense one: the training of the neural network models. This training step is a loop, which consists of trying out different parameter and feature combinations, and assessing them on the cross-training data as long as no further performance increase can be achieved. Once the best model for each sub-classification part has been found, the prediction performance is balanced out by shifting the threshold for each neural network output so that precision and recall on the cross-training data are at a similar high level.

- The last step to train is the threshold for the protein-based scoring, which is also done on the cross-training data targeting on a balanced prediction result.

- Finally each part of the prediction method, including all models and thresholds, are evaluated on the test data of each fold, resulting in the ultimate prediction performance.