rawMSA: proper Deep Learning makes protein sequence profiles and feature extraction obsolete

Claudio Mirabello[1], Björn Wallner[*1]

1 IFM Bioinformatics, Linköping University, Linköping, Sweden

* bjorn.wallner@liu.se

Abstract

In the last decades huge efforts have been made in the bioinformatics community to develop Machine Learning-based methods for the prediction of structural features of proteins with the hope of answering fundamental questions about the way proteins function and their involvement in several illnesses. The recent advent of Deep Learning has renewed the interest on neural networks, with dozens of methods being developed in the hope of taking advantage of these new architectures. On the other hand, most methods are still based on heavy preprocessing of the input data, as well as the extraction and integration of multiple hand-picked, manually designed features. Since Multiple Sequence Alignments (MSA) are almost always the main source of information in *de novo* prediction methods, it should be possible to develop Deep Networks to automatically refine the data and extract useful features from it. In this work we propose a new paradigm for the prediction of protein structural features called rawMSA. The core idea behind rawMSA is borrowed from the field of natural language processing to map amino acid sequences into an adaptively learned continuous space. This allows to input the whole MSA to a Deep Network, thus rendering sequence profiles and other pre-calculated features obsolete. We developed rawMSA in three different flavours to predict secondary structure, relative solvent accessibility and inter-residue contact maps. We have rigorously trained and benchmarked rawMSA on a large set of proteins and determined that it outperforms classical methods based on PSSM when predicting secondary structure and solvent accessibility, while performing on par with the top ranked CASP12 methods in the inter-residue contact map prediction category. We believe that rawMSA represents a promising, more powerful approach to protein structure prediction that could replace older methods based on protein profiles in the coming years.

1 Introduction

Predicting the 3D-structure of a protein from its amino acid sequence has been one of the main objectives in bioinformatics for decades now [1], yet a definite solution has not been found yet.

The most reliable approaches currently involve *homology modelling*, which allows to assign a known protein structure to an unknown protein provided that there is a detectable sequence similarity between the two. When homology modelling is not viable, *de novo* techniques are needed, either based on physical-based potentials [2] or knowledge-based potentials [3–6].

In the first case, an energy function is used to approximate the amount free energy in a given protein conformation along with a search function that tries different structural conformations in order to minimise such energy function [7]. Unfortunately proteins are very large molecules, and the huge amount of available conformations, even for relatively small proteins, makes it prohibitive to fold them even on customized computer hardware [8].

Knowledge-based potentials, on the other hand, can be learned from statistics or machine learning methods to infer useful information from known examples of protein structures. This information can be used to constrain the problem, thus greatly reducing the amount of samples that need to be evaluated when dealing exclusively with physics-based potentials.

In the last couple of decades, a variety of machine learning methods have been developed to predict a number of structural properties of proteins: secondary structure (SS) [9–14], relative solvent accessibility (RSA) [14–17], backbone dihedrals [18], disorder [14, 19, 20], disorder-to-order transition [21, 22],contact maps [23–26], and model quality [27–30].

The most important information used by the most (if not all) methods above is a multiple sequence alignments (MSA) for homologous sequences to the target protein. The MSA consists of aligned sequences and to allow for comparisons and analysis of MSAs, they are compressed into position-specific scoring matrices (PSSM), also called sequence profiles, using the fraction of occurrences of different amino acids in the alignment for each position in the sequence. The sequence profile describes the available evolutionary information of the target protein and is better than a single sequence representation, often allowing for a significant increase in the prediction accuracy [31, 32].

An obvious limitation of compressing an MSA into a PSSM is the loss of information, which could potentially be useful to obtain better predictions. Another potential issue is that whenever the MSA contains few sequences the statistics encoded in the PSSM will not be as reliable and the prediction system may not be able to distinguish between a reliable and an unreliable PSSM.

SS, RSA and similar properties are sometimes used as intermediate steps to constrain and help the prediction of more complex properties in a number of methods [33–35]. An example of this comes from the methods for the prediction of inter-residue contact maps, where evolutionary profiles are integrated with predicted SS and RSA to improve performance [16, 36, 37].

More recently, contact map prediction methods have been at the center of renewed interest after the development of a number of techniques to analyse MSAs in search of direct evolutionary couplings, which have brought along a big leap in the state of the art [38–40]. However, their impressive performance is correlated with the amount of sequences in the MSA, and is not as reliable when few sequences are related to the target. This means that evolutionary couplings methods have not completely replaced older machine learning-based systems, but have been integrated, usually in the form of extra inputs, along with the previously mentioned sequence profiles, SS and RSA, into even more complex machine learning systems. At the same time, the Deep Learning revolution has proved to be a useful tool for better integrating the growing amount and complexity of input features [41–43].

However, one might argue that this kind of integrative approach, combining individually derived features, ignores a key aspect of deep learning, i.e. that features should be automatically extracted by the network rather than being provided to the network as inputs [44]. If we wanted to take advantage of deep learning in the same way it is being used for tasks such as image classification, one idea could be to provide a raw MSA input, which is the basic most low level input most methods is using, and let the deep network take care of the feature extraction avoiding the loss of potentially useful information through the compression into profiles. On the other hand, a MSA is not an image or an audio track, and there is no native way of feeding such a large block of strings as input to a deep network.

In this work we try to overcome this hurdle and introduce a new system for the *de novo* prediction of structural properties of proteins called rawMSA. The core idea behind rawMSA is to borrow from the field natural language processing a technique called *embedding* [45], which we use to convert each residue character in the MSA into a floating point vector of variable size. This way of representing residues is adaptively learned by the network based on *context*, i.e. the structural property that we are trying to predict. To showcase the idea, we designed and tested a number of deep neural networks based on this concept to predict Secondary Structure (SS), Relative Solvent Accessibility (RSA) and Residue-Residue Contact Maps (CMAP). Our test results show that rawMSA matches or outperforms the state of the art in all three applications.

2 Methods

2.1 Inputs

Compared to the classical machine learning methods for the prediction of protein features, rawMSA does not compress the Multiple Sequence Alignment into a profile but, rather, uses the raw aligned sequences as input and devolves the task of extracting useful feature to the deep network. The input to the deep network is a flat FASTA alignment file. Before it is passed to the input layer of the neural network, each letter in the input file (including gaps) is mapped to an integer ranging from 1 to 25 (21 standard residues plus the non-standard residues B, Z, X and the gap placeholder, $-$). If the alignment file of a protein of length L contains N sequences (including the target, or "master" sequence), it is translated to an array of $L \times \; N$ integers. The master sequence occupies the first row of the array, while the following rows contain all the aligned sequences, in the order of output determined by the alignment software. Since the MSAs can become quite "deep" when aligning larger protein families (up to tens of thousands of sequences), a threshold is set so that no more than Y sequences are used, while the rest (usually alignments at higher E-values) are discarded. For details on the depth threshold, see the "Architecture" paragraph.

When training on or predicting SS or RSA, a sliding window of width 31 is applied to the MSA so that L separate windows of size $31 \times \; Y$, one for each residue in the master sequence, are passed to the network. The central column in the window is occupied by the residue in the master sequence for which a prediction is being made and the corresponding aligned residues from the other sequences. Zero-padding at the N- or C- terminals of the master sequence, is applied, as well as if the number N of aligned sequences is such that $N < Y$ or if the master sequence's length L is such that $L < X$) so that the window is of the correct shape. Note that the dictionary applied to the residue alphabet ranges from 1 to 25 to avoid confusion whenever zero-padding is present.

2.2 Architecture

We developed two different architectures for three different applications. For the SS and RSA prediction, we develop the a network called "SS-RSA", for the contact map prediction, we develop a network called "CMAP".

In Fig. 1 we show an example of how the two networks might look. The networks trained in this work can have different numbers of convolutional, fully connected or BRNN layers, as well as slightly different parameters, but they all share this same basic structure.

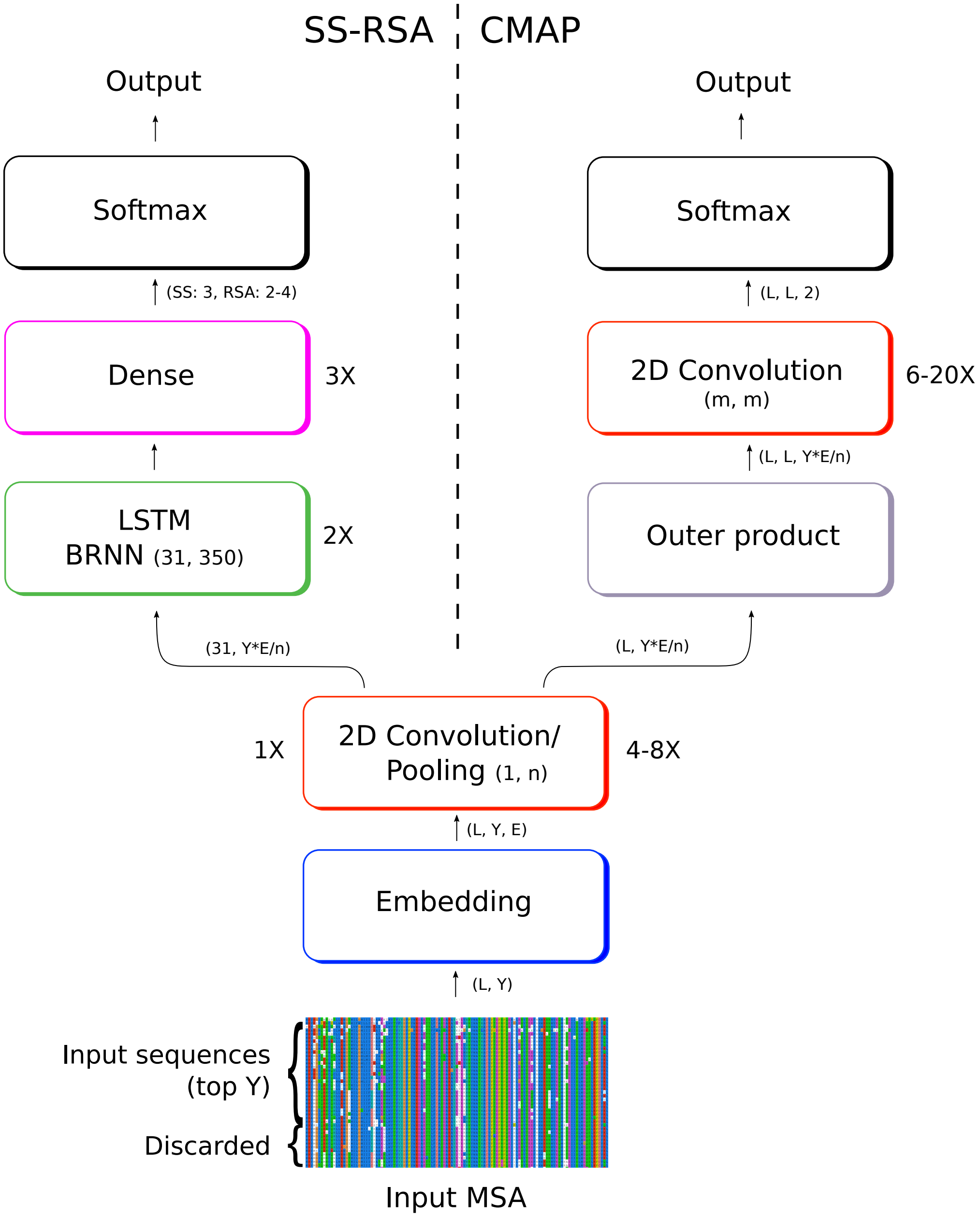


Fig 1. Schematics of two networks developed on rawMSA. On the left, the "SS-RSA" network predicts the secondary structure and relative solvent accessibility of each amino acid, on the right the "CMAP" network predicts the full contact map of the protein. The first layers are in common between the architecture, although with slightly different settings, and provide the basis for the rawMSA approach.

Since with rawMSA we abandon the use of sequence profiles, which are also useful in order to represent the amino acid information in a computer-friendly format, i.e. a matrix of floating points, we need to come up with a way of representing the input. In this case, where the inputs can be very large (up to hundreds of thousands of amino acids), categorical data cannot be translated with sparse, memory inefficient techniques such as *one-hot encoding*.

To solve this issue, the first layer of rawMSA is a trained, shallow two-layer neural network called an *embedding layer*. Embeddings are a compact way of representing a discrete set of inputs into a continuous space [45]. This technique is widely used in natural language processing where the inputs are made of a sequence of words taken from a dictionary and mapped to a n-dimensional space in a vector of floats of size n. When dealing with word embeddings in natural language, words that represent similar concepts, at least in a certain context, will be in close proximity in the output space. This can also be very useful when dealing with the discrete set of amino acids [46], since some do share context-dependent similarities. For example, when it comes to the context of secondary structure, if we look at the Chou-Fasman amino acid propensities table [47], glutamic acid and methionine are both strongly associated with alpha-helices, so it might be useful that such amino acids are represented by similar vectors when predicting secondary structure.

The embedding layers in rawMSA output a vector of size 10 to 30, depending on the model, for each input residue in the alignment. The embedding layer is used both in the SS-RSA and the CMAP networks.

In the case of SS-RSA, we stacked a 2D convolutional layer on top of the embedding layer, followed by a max pooling layer. The convolutional layer has a number of filters equal to the dimensionality of the embedding space. The convolution filters have the shape of column vectors, rather than square matrices as usually done, so the size of the convolution windows vary between 1×10 to 1×30 depending on the model. This means that convolution is done along each column in the MSA and the information does not move across columns (i.e. across adjacent residues in the input sequence). Pooling is performed selecting the maximum value in a window of the same size. In this way, if the dimension of the input is 31 residues by 500 alignments before the embeddings and $31 \times 500 \times 10$ after embedding, this is reduced to a vector of size $31 \times 50 \times 10$ following the convolution and pooling layers, if the convolution and pooling windows are of size 1×10. The convolutional and pooling layers are followed by a stack of 2 Long Short-Term Memory (LSTM) bidirectional recurrent layers, where each LSTM module contains 350 hidden units. The final three layers are fully connected, with a softmax layers to output the classification prediction for SS (3 classes: Helix, Strand, Coil) or RSA (4 or 2 classes), depending on the model. Dropout is used after each recurrent or dense layer to avoid overfitting, with variable fractions of neurons being dropped depending on the model (0.4 to 0.5). All the convolutional layers have ReLU activations and the outputs are zero-padded to match the two first dimensions of the inputs.

For CMAP, we wanted to predict a whole contact map of size $L \times L$ for a protein of length L. The input, in this case, is not split in windows, but we use the whole width MSA at once, while the depth is cut at the Y top alignments. The network for the first part of CMAP is similar to SS-RSA, with an embedding layer followed by up to 6 (rather than only one) 2D convolutional layers/max pooling layers along each MSA column. In this case, though, we want the output of the network to resemble a contact map with shape $L \times L$, and the preceding layers should represent the interaction between couples of residues. This change of dimensionality is performed with a layer that performs a high-dimensional outer product between the hidden units output by the first bank of convolutional layers with themselves. This operation generates a 4-dimensional hidden layer of shape $L \times L \times F \times S$, where F and S are the last two dimensions of the hidden vector before the outer product. This output is then reshaped to a 3-dimensional vector of shape $L \times L \times F * S$ and passed to a new bank of 6 to 20 (depending on the model) 2D convolutional layers with squared convolutions of varied size (3x3, 5x5, 10x10) and number of filters (10 to 50). The last convolutional layer has shape $L \times L \times 2$ and is followed by a softmax activation layer that outputs the contact prediction with a probability from 0 to 1. Also in this case, all the convolutional layers have ReLU activations and the outputs are zero-padded to match the two first dimensions of the inputs. Batch normalization is performed in the outputs of the convolutional layers in the CMAP network.

2.3 Training

We have written the code for rawMSA in Python using the Keras libraries [48] with a TensorFlow backend [49]. Training and testing were performed on NVIDIA GeForce 1080Ti, TESLA K80, and Quadro P6000 GPUs with CUDA 8.0.

Training was done with one protein per batch block, regardless of the size, and using a RMSprop optimizer with sparse categorical cross-entropy as loss function. In the SS-RSA network, training runs for five epochs, in the CMAP network, training runs for up to 200 epochs. During training, a random 10% of the testing samples are reserved for validation, while the rest are used for the training itself. After training is done, the model with highest accuracy on the validation data is saved for testing.

2.4 Data sets

The data set is composed of protein chains extracted from a 70% redundancy-reduced version of PDB compiled by PISCES [50] in April 2017 with minimum resolution of 3.0 Å and R-factor 1.0. This set contains 29653 protein chains.

2.4.1 Avoiding homolog contamination

When training our networks, we want to make sure that the testing and training sets are rigorously separated so that no protein in the test set is too similar to any protein that the network has already "seen" during the training phase.

While most secondary structure and solvent accessibility prediction methods have been using 25-30% sequence identity as threshold to separate testing and training sets [51–54], this practice has been discouraged as it has been shown that it is not sufficient to avoid information leakage [55]. This is apparently also valid for raw MSA inputs, as in our tests separating sets at 25% sequence identity yields higher accuracies compared to our final results (data not shown).

In order to correctly split training and testing sets, we used two databases based on a structural classification of the proteins: ECOD [56] and SCOPe [57] databases to assign one or more superfamilies to each of the protein chains in the initial set. Then, we removed any chains that were related to more than one superfamily. The set generated from ECOD contains 16675 proteins (ECOD set), while the one generated from SCOPe contains 9885 proteins (SCOPe set). The SCOPe set contains fewer proteins than the ECOD set since SCOPe has a lower coverage of the PDB. We splitted each set into 5 subsets by making sure that no two proteins from the same superfamily were placed in two separate subsets. This ensures that the respective MSA inputs will not be too similar to each other and is the recommended practice when training neural networks using sequence profiles, which are themselves extracted from MSAs, as inputs [55].

We used the SCOPe subsets to perform 5-fold cross-validation on SS-RSA. We also used one of the ECOD subsets to train and validate CMAP, where the validation set is used to determine when to stop training to avoid overfitting, and to select the models that will be ensembled and tested (i.e. the models with the lowest validation error).

2.4.2 Multiple Sequence Alignments

The MSAs for both SS-RSA and CMAP were obtained with HHblits [58] by searching with the master sequence against the HMM database clustered at 20% sequence identity from February, 2016 for 3 iterations, with 50% minimum coverage, 99% sequence similarity threshold, and 0.001 maximum E-value.

We also obtain a second set of MSAs by running JackHMMER [59], for 3 iterations and $1e-3$ maximum E-value on the UniRef100 database from February, 2016.

The HHblits alignments were used to train and test the SS-RSA networks. The HHblits alignments were used to train the CMAP network, while both the HHblits and the JackHMMER alignments have been used as inputs in the CMAP ensembling step (see Ensembling section below), as it improved the prediction accuracy. This approach was also tried for the SS-RSA network, but no improvement was observed.

2.4.3 Test sets

We labelled the CMAP data sets by assigning a native contact map to each protein. A contact is assigned to a couple of residues in a protein if the euclidean distance between their $C\beta$ atoms ($C\alpha$ for Gly) in the crystal structure from the PDB is lower than 8 Å. Otherwise, the two residues are assigned the non-contact label.

We tested CMAP on the last CASP12 RR (Residue-Residue) benchmark [60], which is composed of 37 protein chains/domains of the Free Modelling class, i.e. protein targets for which no obvious protein homologs could be found at the time of the experiment (May-August 2016). To ensure a fair comparison with the predictors which participated in CASP12, we performed the benchmark in the same conditions to which all the other predictors where subjected to at the time of the CASP experiment. We made sure that all protein structures (from Apr, 2016) in the training set (Apr, 2016) and sequences (from Feb, 2016) in the HHblist HMM and UniRef100 databases were released before CASP12 started.

In order to test SS-RSA, we calculated the secondary structure (SS) and the Relative accessible Surface Area (RSA) with with DSSP 2.0.4 [61]. We reduced the 8 SS classes (G, H, I, E, B, S, T, C) to the more common 3 classes: Coil, Helix, Extended (C, H, E). We used the theoretical Maximum Accessibility Surface Area (Max ASA) defined in [62] to calculate the RSA from the absolute surface areas (ASA) in the DSSP output and we used $[0, 0.04], (0.04, 0.25], (0.25, 0.5], (0.5, 1]$ as thresholds for the 4-class RSA predictions (Buried, Partially Buried, Partially Accessible, Accessible), and $[0, 0.25], (0.25, 1]$ as thresholds for the 2-class RSA predictions (Buried, Accessible). We discarded the proteins for which DSSP could not produce an output as well as those that had irregularities in their PDB formats. The final set contains 9680 protein chains.

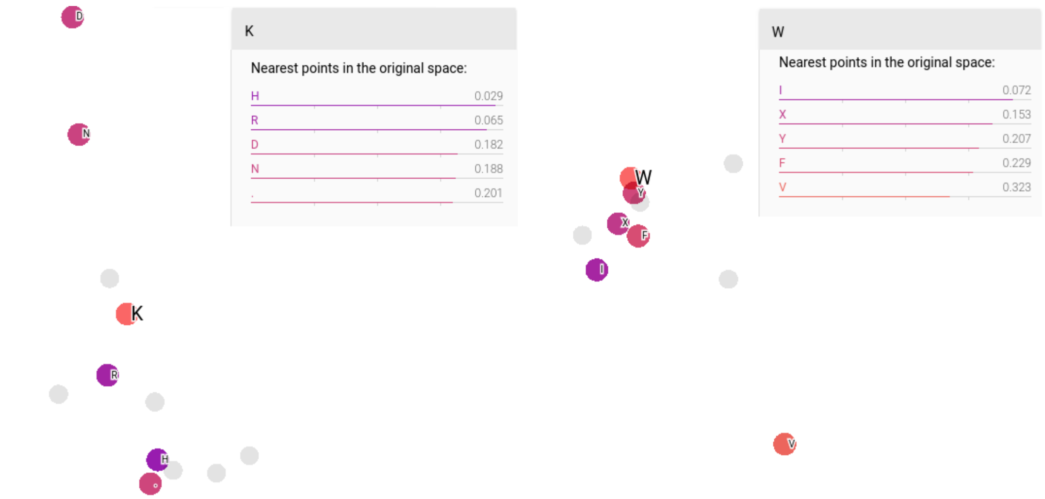
2.4.4 Quality Measures

The measure of the performance of the trained ensemble of SS-RSA networks is the 3-class accuracy (Q3) for SS and the 4-class and 2-class accuracy for RSA, which are calculated by dividing the number of correctly classified residues by the total number of residues in the dataset.

CMAP predictions for the CASP12 RR benchmark set were evaluated in accordance to the CASP criteria by calculating the accuracy of the top $L/5$ predicted long range contacts, where L is the length of the protein and the long range contacts are contacts between residues with sequence separation distance over 23.

2.4.5 Ensembling

Ensembling models usually yields a consensus model that performs better than any of the networks included in the ensemble [63]. A number of networks both for CMAP and SS-RSA have been trained with different parameters (see “Results” section). Even though some models have worse performances on average, they are still saved. All the



K

Nearest points in the original space:

H	0.029
R	0.065
D	0.182
N	0.188
.	0.201

W

Nearest points in the original space:

I	0.072
X	0.153
Y	0.207
F	0.229
V	0.323

(a) **(b)**

Fig 2. 2D PCA of the space of the embedded vectors representing the single residues. The original space has dimensionality of four. The residues that are closest (lower cosine between the 4D vectors) to (a) Lysine and (b) Tryptophan are colored (the closer the residue, the darker the hue).

saved models that have been trained on the same set are used at testing time. The outputs from each model are ensembled to determine the final output. This is done by averaging all outputs from the softmax units and selecting the final class by picking the class with the highest average probability.

In the CMAP case, each model in the ensemble is used to make two predictions for each target using either HHblits or JackHMMER alignment. Although the CMAP network is trained only on HHblits alignments, using the JackHMMER alignments in the ensemble improved the overall accuracy of the predictor.

3 Results and discussion

3.1 Embeddings

Tensorboard in Tensorflow can be used to visualize the output of the embeddings layer to see how each residue type is mapped on the output space. In Fig. 2 we show the embeddings outputs for a early version of the SS-RSA network where each amino acid type is mapped on a 4D space. The 4D vectors are projected onto a 2D space by principal component analysis on Tensorboard. In the figure, the amino acids that are the closest (lowest cosine between the 4D vectors) to Lysine (Fig. 2-a) and Tryptophan (Fig. 2-b) are highlighted. The amino acids closest to Lysine (K) are Histidine (H) and Arginine (R), which makes sense, since they can all be positively charged. Similarly, the residues closest to the hydrophic Tryptophan (W) are also hydrophobic. Indicating that the embeddings can discriminate between different kinds of amino acids and map them onto a space that makes sense from a chemical point of view.

3.2 SS and RSA predictions

We have used the 5-fold cross-validation results to determine the testing accuracy for SS-RSA. To compare the rawMSA approach against a classic profile-based method, we

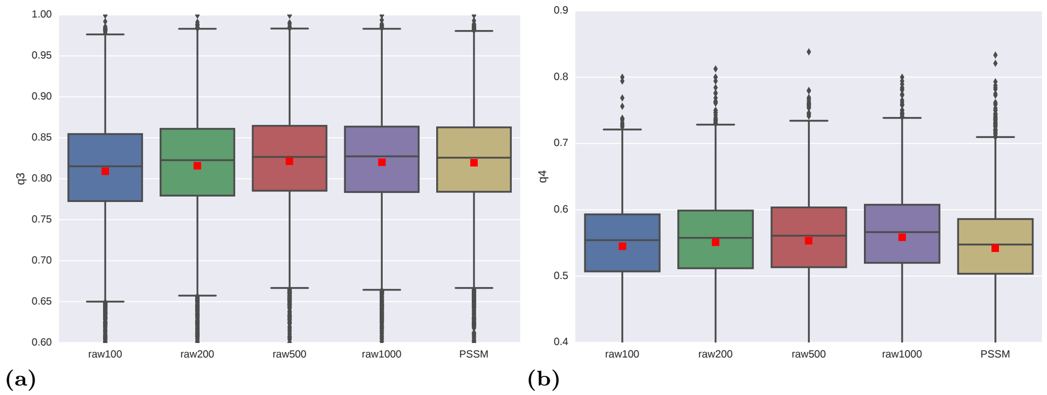


Fig 3. Boxplots showing the accuracy distribution (target-wise) for the predictions by a number of rawMSA networks against a classical PSSM network trained and tested on the same dataset. Here are shown the results both for secondary structure (a) and 4-class solvent accessibility (b) predictions. Four different rawMSA networks are tested at variable MSA depths (number of top alignments considered from the HHblits MSAs as input to the network). Here, the top 100, 200, 500 or 1000 alignments from the MSAs are given as input to the SS-RSA network, respectively. The average accuracies are also shown (red squares).

trained a separate network by removing the bottom layers from the SS-RSA network (embedding and first 2D convolution/Pooling layer) and we trained it by using the PSSMs calculated from the HHblits alignments as inputs. We tested and trained the PSSM network in the same way we trained the other networks both for SS and RSA. In Fig. 3 we compare the performance of the PSSM network against a number of rawMSA networks trained on different numbers of input sequences (100 to 1000 MSA sequences). The boxplot shows how the rawMSA networks with more input sequences perform generally better, with the rawMSA500 and rawMSA1000 networks performing slightly better than the classic PSSM network in predicting secondary structure, and all the rawMSA networks outperforming the PSSM network in predicting solvent accessibility.

The final SS-RSA network is an ensemble of six networks trained in 5-fold cross-validation on 100 to 3000 input MSA sequences per target.

Predictor		Accuracy (%)
rawMSA SS ensemble		83.4
rawMSA RSA ensemble	(4-class)	57.7
rawMSA RSA ensemble	(2-class)	81.2

Table 1. Accuracy results for the SS-RSA network trained to predict secondary structure and solvent accessibility (2-class and 4-class).

The results for the ensemble network, as well as the previously mentioned rawMSA1000 and PSSM network are shown in Table 1. The final accuracy for the SS predictions is 83.4%, the final accuracy for the RSA predictions is 54.1% (4-class) and 81.2% (2-class). It is difficult to make a direct comparison of rawMSA against other predictors in literature because of inevitable differences in the datasets. One example of this comes from secondary structure prediction systems, that have recently been reported predicting at accuracies (Q3) of up to 84% [64], yet we have not been able to find a recent study where the reported accuracy is supported by a proper splitting of the training and testing sets (see "Avoiding homolog contamination" paragraph). Running local versions of existing software does not solve that problem since it is not clear exactly which sequences were used for training. Also, in many cases a final network is trained using all available sequence making any test bound to be contaminated by homologous information. However, given the very large size of our test set, the rigorousness of our experimental setup, and the fact that rawMSA outperforms our own PSSM-based method, we believe that rawMSA compares favorably against the state of the art.

3.3 CMAP predictions

The final CMAP network is an ensemble of 10 networks trained on 10 to 1000 input sequences and varying numbers of layers (10 to 24 convolutional layers).

The CMAP predictions for each target have been sorted by the contact probability measure output by the ensemble, then the top $L/5$ long range contacts have been evaluated against the native contact map. The final accuracy has been calculated as the average of the accuracies for all targets. In order to make a fair comparison against the other predictors, we have downloaded all of the predictions made in CASP12 and evaluated them with the same system. In Table 2 we compare the top $L/5$ long range accuracy of rawMSA CMAP against the top 5 CASP12 predictors.

Predictor	Domain Count	L/5 LR Accuracy (%)
rawMSA CMAP	**37**	**43.8**
RaptorX-Contact	37	43.0
iFold_1	36	42.3
Deepfold-Contact	37	38.6
MetaPSICOV	37	38.4
MULTICOM-CLUSTER	37	37.9

Table 2. Comparison of rawMSA against the top 5 contact prediction methods in CASP12. All predictions from CASP12 have been re-evaluated to ensure a fair comparison. The accuracy is calculated on the top L/5, long range (LR) contacts.

rawMSA performs on par with the top predictors in CASP12 under the same testing conditions. This is especially impressive since it is the only top predictor not to use any kind of explicit coevolution-based features, or any other inputs other than the MSA. It is important to note that only up to 1000 input MSA sequences could be used in training and testing because of limits in the amount of GPU RAM available (<25GB). Since there is a correlation between the testing accuracy and the number of sequences used as input, it is reasonable to expect that rawMSA will benefit from training on GPUs with larger memory.

4 Conclusion

We have presented a new paradigm for the prediction of structural features of proteins called rawMSA, which consists in using raw multiple sequence alignments (MSA) of proteins as input instead of compressing them into protein sequence profiles, as is common practice today. Furthermore, rawMSA does not need any other manually designed or otherwise handpicked extra feature as input, but rather exploits the capability that deep networks have of automatically extracting any relevant feature from the raw data.

In order to convert MSAs, which could be described as categorical data, to a more machine-friendly format, rawMSA adopts embeddings, a technique from the field of Natural Language Processing to adaptively map discrete inputs from a dictionary of symbols into vectors in a continuous space.

To showcase our novel representation of the MSA, we developed a few different flavours of rawMSA to predict secondary structure, relative solvent accessibility and inter-residue contact maps. All these networks use the same and only kind of input, i.e. the MSA. After rigorous testing, we show how rawMSA SS-RSA sets a new state of the art for these kinds of predictions, and rawMSA CMAP performs on par with the top predictors at the CASP12 experiment even though relatively few alignments (up to 1000 per protein in this experiment) could be used as input, due to limitations in the amount of available GPU RAM.

We believe that rawMSA represents a very promising, and powerful approach to protein structure prediction and, with the ever-increasing amounts of computational power and RAM available on the next-generation GPU architectures, could very well replace older methods based on protein sequence profiles in the coming years.

Acknowledgement

This work was supported by Swedish Research Council grants 2012-5270, 2016-05369, The Swedish e-Science Research Center, and the Foundation Blanceflor Boncompagni Ludovisi, née Bildt. Computations were performed on resources provided by the Swedish National Infrastructure for Computing (SNIC) at the National Supercomputer Centre (NSC) in Linköping and at the High Performance Computing Center North (HPC2N) in Umeå. We would also like to thank Hops (www.hops.io), for providing some extra GPU resources, while others were graciously provided by NVIDIA Corporation through the GPU Grant Program. We also thank Isak Johansson-Åkhe for helpful discussions.

References

1. Dill KA, Ozkan SB, Weikl TR, Chodera JD, Voelz VA. The protein folding problem: when will it be solved? Current opinion in structural biology. 2007;17(3):342–346.
2. Shell MS, Ozkan SB, Voelz V, Wu GA, Dill KA. Blind test of physics-based prediction of protein structures. Biophysical journal. 2009;96(3):917–924.
3. Sippl MJ. Calculation of conformational ensembles from potentials of mean force. An approach to the knowledge-based prediction of local structures in globular proteins. Journal of molecular biology. 1990;213(4):859–883.
4. Jones DT, Taylor WR, Thornton JM. A new approach to protein fold recognition. Nature. 1992;358(6381):86–89.
5. Sippl MJ. Knowledge-based potentials for proteins. Current opinion in structural biology. 1995;5(2):229–235.
6. Lazaridis T, Karplus M. Effective energy functions for protein structure prediction. Current opinion in structural biology. 2000;10(2):139–145.
7. Simons KT, Kooperberg C, Huang E, Baker D. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions. Journal of molecular biology. 1997;268(1):209–225.
8. Shaw DE, Maragakis P, Lindorff-Larsen K, Piana S, Dror RO, Eastwood MP, et al. Atomic-level characterization of the structural dynamics of proteins. Science (New York, NY). 2010;330(6002):341–346.
9. Jones DT. Protein secondary structure prediction based on position-specific scoring matrices1. Journal of molecular biology. 1999;292(2):195–202.
10. Cuff JA, Clamp ME, Siddiqui AS, Finlay M, Barton GJ. JPred: a consensus secondary structure prediction server. Bioinformatics (Oxford, England). 1998;14(10):892–893.
11. Pollastri G, Przybylski D, Rost B, Baldi P. Improving the prediction of protein secondary structure in three and eight classes using recurrent neural networks and profiles. Proteins: Structure, Function, and Bioinformatics. 2002;47(2):228–235.
12. Pollastri G, Mclysaght A. Porter: a new, accurate server for protein secondary structure prediction. Bioinformatics. 2004;21(8):1719–1720.
13. Drozdetskiy A, Cole C, Procter J, Barton GJ. JPred4: a protein secondary structure prediction server. Nucleic acids research. 2015;43(W1):W389–94.
14. Wang S, Li W, Liu S, Xu J. RaptorX-Property: a web server for protein structure property prediction. Nucleic acids research. 2016;44(W1):W430–5.
15. Rost B, Sander C. Conservation and prediction of solvent accessibility in protein families. Proteins: Structure, Function, and Bioinformatics. 1994;20(3):216–226.
16. Pollastri G, Baldi P, Fariselli P, Casadio R. Prediction of coordination number and relative solvent accessibility in proteins. Proteins: Structure, Function, and Bioinformatics. 2002;47(2):142–153.

17. Adamczak R, Porollo A, Meller J. Accurate prediction of solvent accessibility using neural networks–based regression. Proteins: Structure, Function, and Bioinformatics. 2004;56(4):753–767.

18. Gao Y, Wang S, Deng M, Xu J. RaptorX-Angle: real-value prediction of protein backbone dihedral angles through a hybrid method of clustering and deep learning. BMC bioinformatics. 2018;19(Suppl 4):100.

19. Linding R, Jensen LJ, Diella F, Bork P, Gibson TJ, Russell RB. Protein disorder prediction: implications for structural proteomics. Structure (London, England : 1993). 2003;11(11):1453–1459.

20. Ward JJ, McGuffin LJ, Bryson K, Buxton BF, Jones DT. The DISOPRED server for the prediction of protein disorder. Bioinformatics (Oxford, England). 2004;20(13):2138–2139.

21. Jones DT, Cozzetto D. DISOPRED3: precise disordered region predictions with annotated protein-binding activity. Bioinformatics (Oxford, England). 2015;31(6):857–863.

22. Basu, Sankar, Söderquist, Fredrik, Wallner, Björn. Proteus: a random forest classifier to predict disorder-to-order transitioning binding regions in intrinsically disordered proteins. Journal of computer-aided molecular design. 2017;31(5):453–466.

23. Fariselli P, Casadio R. A neural network based predictor of residue contacts in proteins. Protein engineering. 1999;12(1):15–21.

24. Punta M, Rost B. PROFcon: novel prediction of long-range contacts. Bioinformatics (Oxford, England). 2005;21(13):2960–2968.

25. Kukic P, Mirabello C, Tradigo G, Walsh I, Veltri P, Pollastri G. Toward an accurate prediction of inter-residue distances in proteins using 2D recursive neural networks. BMC bioinformatics. 2014;15:6.

26. Wang S, Sun S, Li Z, Zhang R, Xu J. Accurate De Novo Prediction of Protein Contact Map by Ultra-Deep Learning Model. PLoS computational biology. 2017;13(1):e1005324.

27. Wallner B, Elofsson A. Can correct protein models be identified? Protein science : a publication of the Protein Society. 2003;12(5):1073–1086.

28. Ray A, Lindahl E, Wallner B. Improved model quality assessment using ProQ2. BMC bioinformatics. 2012;13(1):224.

29. Uziela, Karolis, Menendez Hurtado, David, Shu, Nanjiang, Wallner, Björn, Elofsson, Arne. ProQ3D: improved model quality assessments using deep learning. Bioinformatics (Oxford, England). 2017;33(10):1578–1580.

30. Cao R, Adhikari B, Bhattacharya D, Sun M, Hou J, Cheng J. QAcon: single model quality assessment using protein structural and contact information with machine learning techniques. Bioinformatics (Oxford, England). 2017;33(4):586–588.

31. Rost B, Sander C. Prediction of protein secondary structure at better than 70% accuracy. Journal of molecular biology. 1993;232(2):584–599.

32. Cuff JA, Barton GJ. Application of multiple sequence alignment profiles to improve protein secondary structure prediction. Proteins: Structure, Function, and Bioinformatics. 2000;40(3):502–511.

33. Rohl CA, Strauss CE, Misura KM, Baker D. Protein structure prediction using Rosetta. In: Methods in enzymology. vol. 383. Elsevier; 2004. p. 66–93.

34. McGuffin LJ, Bryson K, Jones DT. The PSIPRED protein structure prediction server. Bioinformatics. 2000;16(4):404–405.

35. Roy A, Kucukural A, Zhang Y. I-TASSER: a unified platform for automated protein structure and function prediction. Nature protocols. 2010;5(4):725.

36. Baú D, Martin AJ, Mooney C, Vullo A, Walsh I, Pollastri G. Distill: a suite of web servers for the prediction of one-, two-and three-dimensional structural features of proteins. BMC bioinformatics. 2006;7(1):402.

37. Tegge AN, Wang Z, Eickholt J, Cheng J. NNcon: improved protein contact map prediction using 2D-recursive neural networks. Nucleic acids research. 2009;37(suppl_2):W515–W518.

38. Morcos, Faruck, Pagnani, Andrea, Lunt, Bryan, Bertolino, Arianna, Marks, Debora S, Sander, Chris, et al. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. Proceedings of the National Academy of Sciences of the United States of America. 2011;108(49):E1293–301.

39. Jones DT, Buchan DW, Cozzetto D, Pontil M. PSICOV: precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. Bioinformatics. 2011;28(2):184–190.

40. Ekeberg M, Hartonen T, Aurell E. Fast pseudolikelihood maximization for direct-coupling analysis of protein structure from many homologous amino-acid sequences. Journal of Computational Physics. 2014;276:341–356.

41. Jones DT, Singh T, Kosciolek T, Tetchner S. MetaPSICOV: combining coevolution methods for accurate prediction of contacts and long range hydrogen bonding in proteins. Bioinformatics. 2014;31(7):999–1006.

42. Wang S, Sun S, Li Z, Zhang R, Xu J. Accurate de novo prediction of protein contact map by ultra-deep learning model. PLoS computational biology. 2017;13(1):e1005324.

43. Adhikari B, Hou J, Cheng J. DNCON2: Improved protein contact prediction using two-level deep convolutional neural networks. Bioinformatics. 2017;.

44. LeCun Y, Bengio Y, Hinton G. Deep learning. nature. 2015;521(7553):436.

45. Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. arXiv preprint arXiv:13013781. 2013;.

46. Asgari E, Mofrad MR. Continuous distributed representation of biological sequences for deep proteomics and genomics. PloS one. 2015;10(11):e0141287.

47. Chou PY, Fasman GD. Conformational parameters for amino acids in helical, β-sheet, and random coil regions calculated from proteins. Biochemistry. 1974;13(2):211–222.

48. Chollet F, et al.. Keras; 2015. https://github.com/fchollet/keras.

49. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, et al.. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems; 2015. Available from: https://www.tensorflow.org/.

50. Wang G, Dunbrack Jr RL. PISCES: a protein sequence culling server. Bioinformatics. 2003;19(12):1589–1591.

51. Fang C, Shang Y, Xu D. MUFOLD-SS: New deep inception-inside-inception networks for protein secondary structure prediction. Proteins: Structure, Function, and Bioinformatics. 2018;86(5):592–598.

52. Torrisi M, Kaleel M, Pollastri G. Porter 5: fast, state-of-the-art ab initio prediction of protein secondary structure in 3 and 8 classes. bioRxiv. 2018; p. 289033.

53. Yang Y, Heffernan R, Paliwal K, Lyons J, Dehzangi A, Sharma A, et al. Spider2: A package to predict secondary structure, accessible surface area, and main-chain torsional angles by deep neural networks. In: Prediction of Protein Secondary Structure. Springer; 2017. p. 55–63.

54. Wang Y, Mao H, Yi Z. Protein secondary structure prediction by using deep learning method. Knowledge-Based Systems. 2017;118:115–123.

55. Söding J, Remmert M. Protein sequence comparison and fold recognition: progress and good-practice benchmarking. Current opinion in structural biology. 2011;21(3):404–411.

56. Cheng H, Schaeffer RD, Liao Y, Kinch LN, Pei J, Shi S, et al. ECOD: an evolutionary classification of protein domains. PLoS computational biology. 2014;10(12):e1003926.

57. Fox NK, Brenner SE, Chandonia JM. SCOPe: Structural Classification of Proteinsextended, integrating SCOP and ASTRAL data and classification of new structures. Nucleic acids research. 2013;42(D1):D304–D309.

58. Remmert M, Biegert A, Hauser A, Söding J. HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. Nature methods. 2012;9(2):173–175.

59. Eddy SR. HMMER: Profile hidden Markov models for biological sequence analysis. 2001;.

60. Schaarschmidt J, Monastyrskyy B, Kryshtafovych A, Bonvin AM. Assessment of contact predictions in CASP12: Co-evolution and deep learning coming of age. Proteins: Structure, Function, and Bioinformatics. 2018;86:51–66.

61. Joosten RP, Te Beek TA, Krieger E, Hekkelman ML, Hooft RW, Schneider R, et al. A series of PDB related databases for everyday needs. Nucleic acids research. 2010;39(suppl_1):D411–D419.

62. Tien MZ, Meyer AG, Sydykova DK, Spielman SJ, Wilke CO. Maximum allowed solvent accessibilites of residues in proteins. PloS one. 2013;8(11):e80635.

63. Naftaly U, Intrator N, Horn D. Optimal ensemble averaging of neural networks. Network: Computation in Neural Systems. 1997;8(3):283–296.

64. Yang Y, Gao J, Wang J, Heffernan R, Hanson J, Paliwal K, et al. Sixty-five years of the long march in protein secondary structure prediction: the final stretch? Briefings in bioinformatics. 2016;19(3):482–494.