PSSM-Distil: Protein Secondary Structure Prediction (PSSP) on Low-Quality PSSM by Knowledge Distillation with Contrastive Learning

Qin Wang[1,†], Boyuan Wang [1,2,†], Zhenlei Xu [2], Jiaxiang Wu [2], Peilin Zhao [2], Zhen Li [1*], Sheng Wang [2], Junzhou Huang [2], Shuguang Cui [1]

[1] Shenzhen Research Institute of Big Data, The Chinese University of Hong Kong(Shenzhen), [2] Tencent AI Lab
{qinwang1@link., boyuanwang@link., lizhen@}cuhk.edu.cn, shengwwang@tencent.com

Abstract

Protein secondary structure prediction (PSSP) is an essential task in computational biology. To achieve the accurate PSSP, the general and vital feature engineering is to use multiple sequence alignment (MSA) for Position-Specific Scoring Matrix (PSSM) extraction. However, when only low-quality PSSM can be obtained due to poor sequence homology, previous PSSP accuracy (merely around 65%) is far from practical usage for subsequent tasks. In this paper, we propose a novel **PSSM-Distil** framework for PSSP on low-quality PSSM, which not only enhances the PSSM feature at a lower level but also aligns the feature distribution at a higher level. In practice, the PSSM-Distil first exploits the proteins with high-quality PSSM to achieve a teacher network for PSSP in a full-supervised way. Under the guidance of the teacher network, the low-quality PSSM and corresponding student network with low discriminating capacity are effectively resolved by feature enhancement through EnhanceNet and distribution alignment through knowledge distillation with contrastive learning. Further, our PSSM-Distil supports the input from a pre-trained protein sequence language BERT model to provide auxiliary information, which is designed to address the extremely low-quality PSSM cases, i.e., no homologous sequence. Extensive experiments demonstrate the proposed PSSM-Distil outperforms state-of-the-art models on PSSP by **6%** on average and nearly **8%** in extremely low-quality cases on public benchmarks, BC40 and CB513.

Introduction

Protein structure analysis, especially protein tertiary (3D) structure, plays a critical role for practical protein applications, such as the understanding of the protein functions and the design of drugs (Noble, Endicott, and Johnson 2004). Currently, there are three mainstream methods for protein tertiary structure (3D) prediction, i.e., X-ray crystallography and nuclear magnetic resonance (NMR) (Wuthrich 1989), cryo-EM based methods (Wang et al. 2015) and computer-aided ab initio prediction (Mandell and Kortemme 2009). Given the extremely time-consuming drawback of X-ray crystallography, the sequence length limitation of nuclear magnetic resonance (NMR) and the expensive equipment requirement for cryo-EM, computer-assisted protein structure prediction attracts broad attention due to its convenience and superior performance. For ab initio tertiary structure prediction, protein property, such as protein secondary structure, provides crucial information as it represents the local patterns of protein structure. Therefore, enhancing the accuracy of protein secondary structure prediction (PSSP) is fundamental for subsequent protein structure prediction.

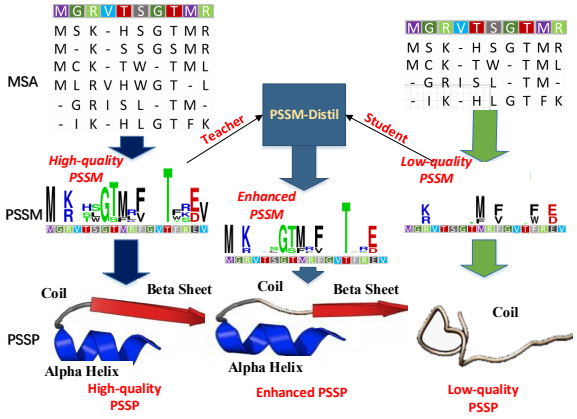

Figure 1: Proposed PSSM-Distil for protein secondary structure prediciton (PSSP) on low-quality PSSM. PSSM-Distil uses a teacher-student network to conduct the knowledge distillation (KD) and contrastive learning (CL) from high-quality PSSM, thus leading to the final improved PSSP.

PSSP is to classify every amino acid on a protein sequence with a secondary structure label (coil, alpha helix, beta-sheet for 3-state secondary structure) indicating the local structure, which is very similar to sequence labeling in natural language processing (NLP). Existing methods usually use the homologs searched from the protein database, which is called multiple sequence alignment (MSA), to generate the Positional-Specific Scoring Matrix (PSSM) for protein sequence. Various sophisticated deep learning models (Li and Yu 2016; Wang et al. 2016; Zhou and Troyanskaya 2014) achieved satisfactory PSSP performance (around 85% Q3 accuracy) when taking high-quality PSSM along with one-hot amino acid sequence as evolutionary information.

*Corresponding author. † Equal first authorship.

This work is done when Boyuan Wang works as intern at Tencent AI Lab

Specifically, Zhou and Troyanskaya (2014) used a deep convolutional network to model the relation between PSSM features and labels. Wang et al. (2016) proposed an improved model by adding a conditional random field after CNN to better model the sequential relation. Sønderby and Winther (2014) tackled the problem with a two layers LSTM, while Li and Yu (2016) added GRU units after convolutional layers to further boost the representation power of the model. Guo et al. (2020) set the CNN and LSTM networks in parallel to capture both local and long-range information. However, even benefiting from the powerful deep discriminating models, the PSSP performance of protein with low sequence homology and low-quality PSSM is still far from being satisfactory, achieving usually around 65% Q3 accuracy. Such low PSSP would directly affect the subsequent protein folding and tertiary structure prediction. Recent work Guo et al. (2020) exploited the "Bagging" mechanism to obtain the enhanced PSSM for protein with low-quality PSSM through a fixed ratio MSA down-sampling in an unsupervised manner. However, such "Bagging" method merely conducts PSSM feature enhancement, while ignoring the joint optimization of the enhanced feature and final PSSP on high-level semantic space, leading to inferior robustness and PSSP performance.

Therefore, in this paper, we will address the practical PSSP problem for protein sequence with low sequence homology (i.e., low-quality PSSM) in a large database. We propose a novel framework called **PSSM-Distil**, as illustrated in Fig. 1. The proposed model automatically enhances the low-quality PSSM by aligning its distribution to the high-quality ones. That is to say, PSSM-Distil first exploits proteins with high-quality PSSM to obtain a classifier (any previous general model like BLSTM) for PSSP as a teacher network in a full-supervised way. Under the guidance of teacher network, low-quality PSSM through an EnhanceNet and corresponding student network with low discriminating capacity is effectively resolved by feature enhancement and distribution alignment through knowledge distillation with contrastive learning, which is the core contribution of our proposed PSSM-Distil model. Additionally, our PSSM-Distil model supports the input from the pre-trained BERT (Rao et al. 2019) model on UniRef90 to provide auxiliary information, which is designed to address the extremely low-quality PSSM cases, i.e., a protein with no homologous sequence. Also, extensive experiments demonstrate the proposed PSSM-Distil outperforms state-of-the-art models on PSSP by a large margin on the validation set of CullPDB, public benchmark CB513 and newly proposed large dataset BC40 (release date is 2020-07-28).

Our contributions are summarized as follows: 1) We propose a new framework called PSSM-Distil for protein secondary structure prediction (PSSP) on low-quality PSSM, which exploits a teacher-student network to distill knowledge from high-quality PSSM with contrastive learning. 2) Our PSSM-Distil could not only obtain enhanced PSSM in a self-supervised manner through prior knowledge-based down-sampling, but also align the enhanced PSSM distribution with the high-quality one for final PSSP, leading to a largely improved prediction accuracy, i.e., average **6%** for protein with low-quality PSSM, and over **8%** improvement in extremely low-quality cases. 3) We further release a large scale up-to-date test dataset BC40 (release date is 2020-07-28) to verify the effectiveness of PSSM-Distil. Unlike Rao et al. (2019) who directly utilized BERT's embedding to facilitate PSSP, we are the first paper to sampling MSAs from pre-trained BERT's output to construct BERT Pseudo PSSM which will input to PSSP as auxiliary information and significantly improve the PSSP performance of protein with no homology.

Related Works

Multiple Sequence Alignment (MSA) MSA is a sequence alignment of multiple homologous protein sequences for a target protein (Wang and Jiang 1994). It is a key technique for modeling sequence relationships in computational biology. Given a protein database and a protein sequence, MSA is searched by performing pairwise comparisons (Altschul et al. 1990), Hidden Markov Model-like probabilistic models (Eddy 1998; Johnson, Eddy, and Portugaly 2010; Remmert et al. 2012), or a combination of both (Altschul et al. 1997) to align the sequence against the given database. Once MSA is conducted, it is usually transferred to the Position-Specific Scoring Matrix (PSSM) for subsequent tasks.

Low-quality PSSM Enhancement. Since MSA and PSSM are critical for protein property prediction, "Bagging" (Guo et al. 2020) is the first attempt to enhance the low-quality PSSM. By minimizing the MSE loss between the reconstructed and original PSSM, "Bagging" reconstructs high-quality MSA from down-sampled MSA with low-quality PSSM via an unsupervised method. Even though "Bagging" is the first work to achieve a relatively satisfactory performance, there are still some limitations. First, it exploits a fixed ratio for MSA down-sampling to obtain the low-quality PSSM, which makes the "Bagging" model less robust, especially for sequences with extremely low homology. Second, "Bagging" only conducts PSSM enhancement while ignoring the joint optimization of PSSM and the final PSSP.

Knowledge Distillation. Knowledge distillation transfers the knowledge from a pre-trained teacher network to a student network through training on the soft targets provided by the teacher network, which is originally proposed by Buciluǎ, Caruana, and Niculescu-Mizil (2006) and later improved by Hinton, Vinyals, and Dean (2015). Over the past years, knowledge distillation has numerous applications (Chen et al. 2017; Yim et al. 2017; Yu et al. 2017; Schmitt et al. 2018). Inspired by these works, we propose the first method that exploits the knowledge distillation for PSSP. Same as the motivation for Mirzadeh et al. (2019), our approach aims to close the gap between teacher network and student network. However, instead of directly passing the knowledge from teacher network to student network, our enhancement module enhances low-quality PSSM to a high-quality one for student network learning via contrastive learning.

Contrastive Learning. Contrastive loss was introduced by Hadsell, Chopra, and LeCun (2006) to learn representation by contrasting positive pairs against negative pairs. Recent work in computer vision (Oord, Li, and Vinyals 2018; He et al. 2020; Misra and Maaten 2020; Tian, Krishnan, and Isola 2019; Zhuang, Zhai, and Yamins 2019; Chen et al. 2020) presents promising results on unsupervised visual representation learning using approaches related to the contrastive loss. Inspired by the intuition and the results of contrastive learning, we are the first to import contrastive learning into PSSM enhancement. By contrasting high-quality PSSM to low-quality and the corresponding enhanced one, our model learns to generate enhanced PSSM closer to the high-quality PSSM distribution. We notice that Tian, Krishnan, and Isola (2019) also combines the contrastive learning with knowledge distillation, however, our motivations are quite different. While they try to bridge the gap between student and teacher network with contrastive learning, our method instead takes advantage of both methods to improve our EnhanceNet.

Protein sequence pre-training. Self-supervised learning is a powerful tool for extracting information from unlabeled sequences (Devlin et al. 2018; Peters et al. 2018; Radford et al. 2019; Yang et al. 2019). Like language, large unlabeled datasets of protein sequences are expected to contain significant biological information. Recent work in protein sequence pre-training has shown positive results on various downstream tasks including secondary structure prediction (Alley et al. 2019; Bepler and Berger 2019; Heinzinger et al. 2019; Rao et al. 2019; Rives et al. 2019). TAPE (Rao et al. 2019) is the first work proposing systematical evaluation of the protein sequence pre-training model. They assessed the performance of pre-training on three common types of representation models, which are recurrent, convolutional, attention-based models. They also proposed a benchmark dataset for five downstream tasks including secondary structure prediction. We chose the attention-based BERT model based on its downstream performance reported from the TAPE paper. But, we trained our BERT on a larger database-UniRef90 (Suzek et al. 2015), since it is the common database choice of MSA search for PSSP. After the pre-training process, the model can then provide auxiliary information as PSSM for protein with no homology.

Method

Protein Secondary Structure Prediction (PSSP)

There are 20 common amino acids that function as the building blocks of a protein sequence. PSSP is a sequence-to-sequence task where each amino acid x_i in a protein sequence is mapped to a label $y_i \in \{$alpha-helix (H), beta-strand (E), Coil (C)$\}$ for 3-state PSSP.

For PSSP, we adopt the most common choice called Position-Specific Scoring Matrix (PSSM). The PSSM indicates the substitution log-likelihood of all the 20 amino-acid types at each position, based on homologous sequences. PSSM of a protein sequence, denoted by $\mathbf{X}$, is defined as $\mathbf{X}_{k,j} = \log(\frac{\mathbf{P}_{k,j}}{\mathbf{B}_k})$, where $\mathbf{P}$ is the position probability matrix and $\mathbf{B}$ is the background frequency matrix. k is one kind of amino acids and $j \in (1, .., L)$ with L denoting the length of the protein sequence. $\mathbf{P}$ is defined as $\mathbf{P}_{k,j} = \frac{\mathbf{C}_{k,j}+p}{N+20\times p}$, where p is a scaler called pseudo-count to avoid zero-occurrence issue of some amino-acid types which we set as 1 in practice and N is the number of homologous sequences in the MSA. $\mathbf{B}$ is defined as $\mathbf{B}_k = \frac{\sum_{j=1}^{L}\mathbf{C}_{k,j}}{\sum_k\sum_{j=1}^{L}\mathbf{C}_{k,j}}$, which is the frequency of each amino acid occurs in the entire protein MSA. The above $\mathbf{C}_{k,j}$ is the occurrence count of amino acid k in position j of an MSA $\mathbf{M}$, which is defined as $\mathbf{C}_{k,j} = \sum_{i=1}^{N} I(\mathbf{M}_{i,j} = k)$, where I is an indicator function taking value 1 if $\mathbf{M}_{i,j} = k$ and 0 otherwise.

High-quality PSSM is critical for PSSP, thus the enhancement of low-quality PSSM is the key for high accuracy PSSP. To tackle this issue, as shown in Fig. 2, we propose a novel teacher-student framework PSSM-Distil with knowledge distillation (KD) and contrastive learning (CL). Details of each component will be disclosed as follows.

Knowledge Distillation for PSSM Enhancement

A teacher-student framework is exploited to achieve knowledge distillation (KD) on the PSSP task. Specifically, as shown in Fig. 2, we firstly train a teacher classifier F_t with high-quality PSSM $\mathbf{X}_h$ on PSSP task. Then we down-sample the high-quality PSSM $\mathbf{X}_h$ to obtain the low-quality PSSM $\mathbf{X}_l$. Note that here we conduct the down-sampling operation based on prior statistics instead of the fixed down-sample ratio used in "Bagging" (Guo et al. 2020) which details will be given in the experiment section. Based on the low-quality PSSM $\mathbf{X}_l$ and one-hot encoding of protein sequence S, an EnhanceNet F_e is trained to obtain the enhanced PSSM $\mathbf{X}_e$. Furthermore, the auxiliary information $\mathbf{X}_b$ provided by the pre-trained BERT model can also flow into the EnhanceNet as additional input for a better performance, i.e., $\mathbf{X}_e = F_e(S, \mathbf{X}_b, \mathbf{X}_l)$. Successively, the enhanced PSSM $\mathbf{X}_e$ is fed to a student network F_s with its protein sequence S to obtain its classification logits $F_s(S, \mathbf{X}_e)$. Similarly, we obtain the classification logits $F_t(S, \mathbf{X}_h)$ through the pre-trained teacher network for high-quality PSSM. Finally, we define the KD loss $\mathcal{L}_d$ as Eq.1.

$$\mathcal{L}_d = \sigma\mathcal{L}_{ce} + (1-\sigma)\,\mathcal{L}_{kl} \tag{1}$$

where σ is a hyper-parameter weighting the two losses which we set as 0.1 in practice. $\mathcal{L}_{ce}$ is the cross entropy loss between the student prediction distribution $F_s(S, \mathbf{X}_e)$ and the PSSP label Y, i.e., $\mathcal{L}_{ce} = CE\left(F_s\left(S, \mathbf{X}_e\right), Y\right)$. $\mathcal{L}_{kl}$ is the Kullback–Leibler divergence between teacher's and student's prediction distribution as shown in Eq. 2.

$$\mathcal{L}_{kl} = KL_{marginal}\left(F_s\left(S, \mathbf{X}_e\right), F_t\left(S, \mathbf{X}_h\right)\right) \tag{2}$$

By minimizing $\mathcal{L}_d$, both the parameters of student network F_s and EnhanceNet F_e will be updated. Hence, we achieve better PSSM enhancement and adaptation at the same time.

Contrastive Learning on PSSM Distribution

Inspired by the achievement of contrastive learning (CL) on self-supervised vision tasks, we exploit CL loss as additional

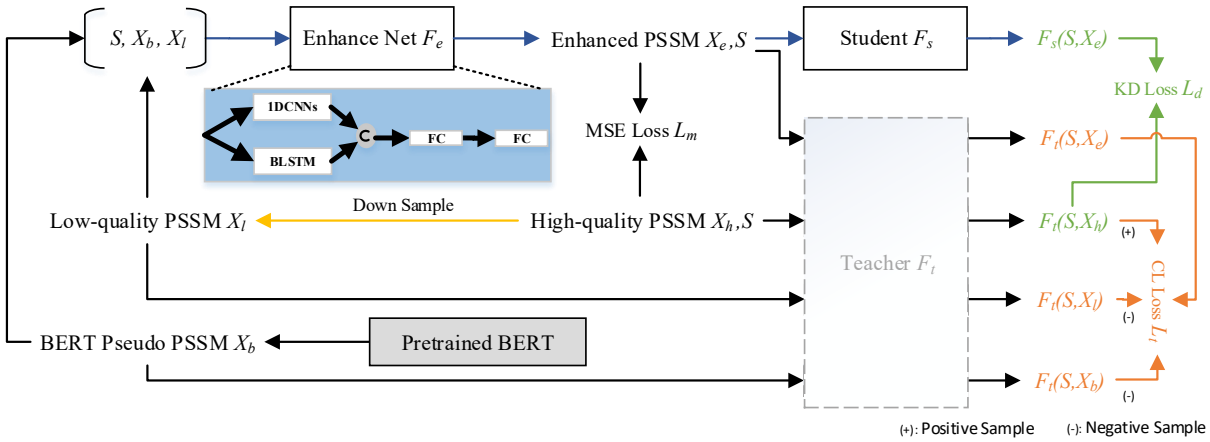

Figure 2: The overall pipeline of our framework: a teacher-student model for knowledge distillation and contrastive learning with an EnhanceNet F_e for low-quality PSSM enhancement. First, the pure sequence S is concatenated with the pre-trained BERT pseudo-PSSM $\mathbf{X}_b$ and the low-quality PSSM $\mathbf{X}_l$ as the input to the EnhanceNet F_e to predict an enhanced PSSM $\mathbf{X}_e$. Then, the MSE loss L_m is used to minimize the difference between the enhanced PSSM $\mathbf{X}_e$ and the high-quality PSSM $\mathbf{X}_h$. Moreover, the high-quality PSSM $\mathbf{X}_h$ which is the ground truth of the $\mathbf{X}_e$ is sent into the teacher network F_t to extract the classification logits $F_t(S, \mathbf{X}_h)$ and $\mathbf{X}_e$ is input to the student network F_s to get $F_s(S, \mathbf{X}_e)$. Then, knowledge distillation is applied between the fixed weights teacher network and the student network by KD loss presented in Eq.1. The high-quality PSSM $\mathbf{X}_h$, low-quality PSSM $\mathbf{X}_l$ and the BERT pseudo-PSSM $\mathbf{X}_b$ are also fed into the teacher network F_t to obtain $F_t(S, \mathbf{X}_h)$, $F_t(S, \mathbf{X}_l)$ and $F_t(S, \mathbf{X}_b)$ respectively. We regard $\mathbf{X}_l$ and $\mathbf{X}_b$ as the negative sample and the $\mathbf{X}_h$ as the positive sample to optimize $\mathbf{X}_e$ and applied a contrastive loss in Eq. 7 as the additional supervision for networks F_e, F_s. Finally, a joint loss in Eq. 9 is exploited to train our model in the end-to-end manner. In inference, only a blue arrow path is used. The Algorithm.1 illustrates more details of the above training process.

supervision to optimize the EnhanceNet by further enhancing the low-quality PSSM in a high-level semantic space. Concretely, as shown in Fig. 2, once the teacher network F_t is well-trained, we can use it as a feature extractor, which transforms the original input feature to a high-level semantic space. For CL loss, we regard the high-quality PSSMs $\mathbf{X}_h$ as positive samples, the low-quality PSSMs $\mathbf{X}_l$ and BERT Pseudo-PSSM $\mathbf{X}_b$ as negative samples and the enhanced PSSM $\mathbf{X}_e$ as enhanced samples. Then we use a conventional contrastive learning inequality defined as Eq. 3 to characterize the shifting of the mapping $\mathcal{F}_X$ in semantic space for the enhanced sample $\mathbf{X}_e$. In particular, the $\mathcal{F}_X$ will move closer to the mapping $\mathcal{F}_P$ of positive sample $\mathbf{X}_h$ and away from the mapping $\mathcal{F}_N$ of negative samples $\mathbf{X}_l, \mathbf{X}_b$.

$$KL_{marginal}\left(\mathcal{F}_X, \mathcal{F}_P\right) \leq KL_{marginal}\left(\mathcal{F}_X, \mathcal{F}_N\right) \tag{3}$$

More specifically, $\mathcal{F}_X$ is the mapping in semantic space of the enhanced PSSM $\mathbf{X}_e$ defined as Eq. 4.

$$\begin{aligned} \mathbf{X}_e &= F_e(S, \mathbf{X}_b, \mathbf{X}_l) \\ \mathcal{F}_X &= F_t(S, \mathbf{X}_e) \end{aligned} \tag{4}$$

Similarly, the $\mathcal{F}_P$ and $\mathcal{F}_N$ are the mappings in semantic space from positive and negative samples respectively. Here, we choose $\mathcal{F}_P$, denoted by Eq. 5, as the classification logits for high-quality PSSM $\mathbf{X}_h$ extracted from the pre-trained teacher network F_t.

$$\mathcal{F}_P = F_t(S, \mathbf{X}_h) \tag{5}$$

We denote $\mathcal{F}_N$ as the combination of the classification logits from the teacher network for low-quality PSSM $F_t(S, \mathbf{X}_l)$ and BERT Pseudo PSSM $F_t(S, \mathbf{X}_b)$. We defined $\mathcal{F}_N$ as Eq. 6

$$\mathcal{F}_N = \frac{1}{2}\left(F_t(S, \mathbf{X}_l) + F_t(S, \mathbf{X}_b)\right) \tag{6}$$

Finally, we adopt a triplet loss to model the inequality in Eq.3 which can be illustrated as Eq. 7, where η is a hyper-parameter and we set it equal to 0.6 in practice.

$$\begin{aligned} \mathcal{L}_t &= \max\left(0, KL(\mathcal{F}_X, \mathcal{F}_P) - KL(\mathcal{F}_X, \mathcal{F}_N) + \eta\right) \\ &= \max\left(0, KL\left(F_t\left(S, \mathbf{X}_e\right), F_t\left(S, \mathbf{X}_h\right)\right)\right. \\ &\left. - KL\left(F_t\left(S, \mathbf{X}_e\right), \mathcal{F}_N\right) + \eta\right) \end{aligned} \tag{7}$$

Loss Function

Additionally, same as in previous work (Guo et al. 2020), we use the mean square error (MSE) loss to directly minimize the difference between the enhanced PSSM $\mathbf{X}_e$ and the high-quality PSSM $\mathbf{X}_h$ referred in Eq. 8.

$$\mathcal{L}_m = \|\mathbf{X}_e - \mathbf{X}_h\|_2 \tag{8}$$

By combining with the aforementioned KD and CL loss, the overall loss function of our framework is shown in Eq. 9 where α, β, γ are the weighting hyper-parameters. In practice, we set $\alpha = 0.16, \beta = 0.016$, and $\gamma = 0.82$.

$$\mathcal{L} = \alpha\mathcal{L}_d + \beta\mathcal{L}_m + \gamma\mathcal{L}_t \tag{9}$$

BERT Pseudo PSSM Generation

To supply auxiliary information for better enhancement of low-quality PSSM, as shown in Fig. 2, BERT Pseudo PSSM

Dataset	CullPDB	CullPDB	CB513	BC40
Type	Training	Validation	Testing	Testing
Size	5600	525	514	36976

Table 1: Details of dataset used in our experiments, including dataset names, types and number of proteins sequence.

$\mathbf{X}_b$ derived from pre-trained BERT is concatenated with low-quality PSSM $\mathbf{X}_l$ and fed to EnhanceNet F_e for enhancement. For the generation of the BERT Pseudo PSSM $\mathbf{X}_b$, we take advantage of BERT training objective. Since BERT is a masked language model and uses neighboring contexts to recover the masked token, we mask each position of a protein sequence one at a time to obtain the predicted probability vector of 20 amino acids for the masked position. By repeating the above procedure, we could obtain the BERT sequence probability map for a specific protein sequence. Then we sample 2000 pseudo protein sequences as MSA from the probability map to generate BERT Pseudo PSSM $\mathbf{X}_b$.

Algorithm 1: PSSM-Distil for PSSP

Input: Protein Sequence S; High-quality PSSM $\mathbf{X}_h$; BERT Pseudo PSSM $\mathbf{X}_b$; Low-quality PSSM $\mathbf{X}_l$; Student F_s; Teacher F_t; EnhanceNet F_e; Label Y;

// Training Phase
$F_t \leftarrow Pretrain\ Teacher\ Network\ F_t\ by\ \mathbf{X}_h$;
$\mathbf{X}_l \leftarrow Downsample\ \mathbf{X}_h$;
$\mathbf{X}_e \leftarrow F_e(S, \mathbf{X}_l, \mathbf{X}_b)$;
// Mappings of Enhanced, Positive and Negative Samples
$\mathcal{F}_X \leftarrow F_t(S, \mathbf{X}_e)$;
$\mathcal{F}_P \leftarrow F_t(S, \mathbf{X}_h)$;
$\mathcal{F}_N \leftarrow \frac{1}{2}\left(F_t(S, \mathbf{X}_l) + F_t(S, \mathbf{X}_b)\right)$;
// Using $\mathcal{L}_d$, $\mathcal{L}_m$ and $\mathcal{L}_t$ to Optimize F_s, F_e
$F_s, F_e \leftarrow Minimize\ \alpha\mathcal{L}_d + \beta\mathcal{L}_m + \gamma\mathcal{L}_t$;
// Inference Phase
$\mathrm{X}_e \leftarrow F_e(S, \mathbf{X}_l, \mathbf{X}_b)$;
PSSP $\leftarrow Argmax(F_s(S, \mathbf{X}_e))$;

Output: Parameters of F_t, F_s, F_e

Experiment

Dataset

We train the PSSM-Distil framework on the training set of CullPDB (Wang and Dunbrack Jr 2003). CullPDB validation set, CB513 (Kryshtafovych et al. 2014) and a new dataset BC40 constructed by ourselves are used to evaluate the performance of our method and conduct comparisons with previous methods. For the CullPDB dataset, any two proteins share less than 25% sequence identity. Following the same procedure as in (Zhou and Troyanskaya 2014), we divide the CullPDB dataset into a training set and validation set with no more than 25% of the training set shared with CB513 and BC40. We conduct an MSA search for all training, validation and test protein sequences from the Uniref90 database (Suzek et al. 2015). The protein sequence labels for all training, validation and test proteins are generated by DSSP (Kabsch and Sander 1983). The dataset details are shown in Table 1.

BC40 Dataset[1] To further validate our proposed approach on real-world PSSP applications, we construct the BC40 dataset (release date is 2020-07-28) in which each entry is publicly available from PDB. Specifically, PDB will cluster all protein chains by MMseq2 (Steinegger and Söding 2017) at 30%, 40%, ..., 90%, 95%, and 100% sequence identity each week to remove redundancy, and BC40 is the dataset with 40% cutoff such that the proteins share no more than 40% sequence identity. Additionally, we also remove the proteins that share more than 25% sequence identity with our CullPDB dataset.

Network Architecture

Like other sequence labeling models in NLP, our teacher network F_t and student network F_s share a similar design which consists of BiLSTM and linear fully-connected layers, while EnhanceNet F_e has additional 1-dimensional convolution layers. More specifically, an embedding layer with dimension 32 in EnhanceNet F_e is used to map the original protein sequence to a higher dimensional semantic space with dimension $(L \times 32)$. Then, embedding features $(L \times 32)$ is concatenated with the low-quality PSSM $\mathbf{X}_l$ $(L \times 20)$ and BERT Pseudo PSSM $\mathbf{X}_b$ $(L \times 20)$ as an $L \times 72$ dimensional input for the latter part, where L is the sequence length. As shown in the blue part of Fig. 2, the principal part of F_e consists of two branches: BiLSTM and 1D-CNN branches which extract features independently from previous $L \times 72$ input. In the BiLSTM branch, there is a BiLSTM model that contains two hidden layers and each layer has 400 hidden units to extract local features at the token level and global features at the sequence level. In the 1D-CNN branch, three 1D-CNN layers are used to extract local features for each token position along the 72-dimension of input and the hidden number for each CNN is 300. Finally, we concatenate the output of the two branches and use two linear fully-connected layers to regress out the final enhanced PSSM $\mathbf{X}_e$ which has the same shape as the low-quality PSSM $\mathbf{X}_l$ input. The teacher and student networks F_s, F_t are two PSSP classifiers. Each of them consists of a 2-layer BiLSTM to extract features and 2 linear FC layers with a final softmax layer for prediction.

Implementation Details

We use PyTorch to implement our work. Three networks F_e, F_s, F_t are trained in an end-to-end manner. Particularly, as depicted in Algorithm 1, we first train the teacher network F_t by using high-quality PSSM $\mathbf{X}_h$, which learns how to use a good PSSM to predict SS. Then, EnhanceNet F_e and F_t are jointly optimized by loss function $\mathcal{L}$ in Eq. 9. We use Adam optimizer with an initial learning rate that equals 0.01 and conducts learning rate decay every 50 epochs. We employ a drop out layer before the softmax layer in each network with the dropout set to 0.75. Specifically, we train our models on

[1] https://drug.ai.tencent.com/protein/bc40/download.html

one Tesla V100 GPU. Greed search is utilized for hyper-parameter tuning. Source code[2] for our inference phase with pretainined models have been released for demonstration.

Prior Distribution based PSSM Down-sampling

As illustrated by the yellow arrow in Fig. 2, we down-sample the high-quality PSSM $\mathbf{X}_h$ to obtain low-quality $\mathbf{X}_l$, which is one input for the subsequent module. Since EnhanceNet F_e is employed to enhance low-quality PSSMs, in the training phase, the down-sampled low-quality PSSM $\mathbf{X}_l$ should exist no domain gap to the natural low-quality PSSM. Thus, different from Bagging using a fixed down-sampling ratio, we exploit the prior native distribution based PSSM down-sampling strategy. In practice, we first calculate the MSA count distribution $\mathcal{X}$ based on native sequences with low-quality PSSM in the training set, i.e., calculate the frequency of MSA count when the MSA count less than 60 (Guo et al. 2020). Once the prior distribution of native low-quality $\mathcal{X}$ has been achieved, in the training phase, we randomly select a batch of MSAs from original MSAs with a count number which is sampled from distribution $\mathcal{X}$. Benefiting from the low-quality PSSMs $\mathbf{X}_l$ achieved through domain aligned down-sampling, our EnhanceNet can output more realistic high quality $\mathbf{X}_e$, leading to superior and robust performance than previous methods.

Results

We evaluate our PSSM-Distil framework on low-quality PSSM protein sequences from three public datasets: CullPDB, CB513 and BC40. The comparison experiment with the previous state-of-the-art models confirms the supreme priority of our approach. Particularly, our method is effective on all levels of difficulties with **7**$\%$**-15**$\%$ of improvements over the vanilla model denoted as "Real" and surpasses the previous state-of-the-art method "Bagging" (Guo et al. 2020) by 6% on average and nearly 8% in the extreme low-quality cases. The accuracy is computed on a per-protein basis. Moreover, the performance gained from each component of our model is well examined by the ablation study.

Comparison Experiment. We compare the PSSP results of PSSM-Distil, the previous state-of-the-art model "Bagging" and the vanilla PSSP model "Real". "Real" is trained on the protein sequences with low-quality PSSMs from CullPDB without any enhancement. To give a more detailed comparison, we split the protein sequences with low-quality PSSM into several divisions of MSA count and MSA meff according to Guo et al. (2020). Shown in Table 2 and Table 3, our approach achieves the best performance on protein sequences with low-quality PSSMs under regardless of low MSA count score or low MSA meff score settings. Furthermore, for the extreme low-quality cases, i.e., MSA count equals to 0 or the meff score is less than 5, our method still gains relatively satisfactory results against previous approaches. In particular, in the case of MSA Count equals to 0, our method still has 73.7% accuracy with 7.6% improvement over the previous best method "Bagging" on the BC40 test set, which proves our method is robust and effective under the condition that no MSA is available. Note that for CB513, we only report the results with threshold level ≤ 60 for MSA count in Table 2, due to too few sequences existing for lower threshold levels such as $\leq 30, = 0$ to make it representative.

[2]https://github.com/qinwang-ai/PSSM-Distil

MSA Counts	Datasets	Number	Real	Bagging	Our
≤ 60	BC40	1861	0.707	0.736	**0.778**
	CullPDB	30	0.755	0.765	**0.807**
	CB513	18	0.702	0.703	**0.725**
≤ 30	BC40	1231	0.687	0.717	**0.766**
	CullPDB	19	0.739	0.746	**0.784**
≤ 10	BC40	639	0.649	0.689	**0.759**
	CullPDB	9	0.714	0.736	**0.779**
$= 0$	BC40	177	0.594	0.661	**0.737**
	CullPDB	2	0.759	0.773	**0.877**

Table 2: PSSP results on BC40, CullPDB and CB513 test sets for for protein sequence with low-quality PSSM leveled by MSA count score. The "MSA Counts" stands for the number of alignment sequences in the MSA of a protein sequence. The "Number" column stands for the number of the protein sequences in the datasets that their searched MSAs falling in the MSA Counts category. The "Real" column is the baseline result without any enhancement technique. The "Bagging" column is the result of a previous data enhancement method. Our experimental results show large improvement over the baseline method and "Bagging".

Meff	Datasets	Number	Real	Bagging	Our
≤ 35	BC40	2833	0.725	0.749	**0.786**
	CullPDB	56	0.775	0.793	**0.808**
≤ 25	BC40	2338	0.716	0.745	**0.780**
	CullPDB	44	0.739	0.780	**0.804**
≤ 15	BC40	1708	0.698	0.731	**0.772**
	CullPDB	29	0.751	0.762	**0.788**
≤ 5	BC40	886	0.655	0.699	**0.755**
	CullPDB	11	0.732	0.740	**0.766**

Table 3: PSSP results on BC40 and CullPDB for protein sequence with low-quality PSSM leveled by Meff score.

To specify the effect of MSA count and Meff score on PSSP accuracy in more detail, we also conduct the quantitative comparison between our method and "Bagging" on PSSP accuracy improvement against the vanilla "Real" model over different MSA count and Meff score ranges, which is shown in Fig. 3. It is worth noting that our PSSM-Distil is increasingly more effective as PSSM quality decreases, demonstrating that our method successfully targets the lower quality cases and outperforms the previous best method "Bagging" by a large margin in those extreme low-quality PSSMs. Particularly, on bin $[0, 5]$ for MSA count and $[0, 1]$ for MSA Meff, our improvements have 8% more than "Bagging". Moreover, our method achieves improvement on all score ranges for both MSA count and MSA meff, while "Bagging" only improves on low MSA count and low MSA meff bins and even worsen the performance for some high MSA count and high MSA meff ranges, which they pointed

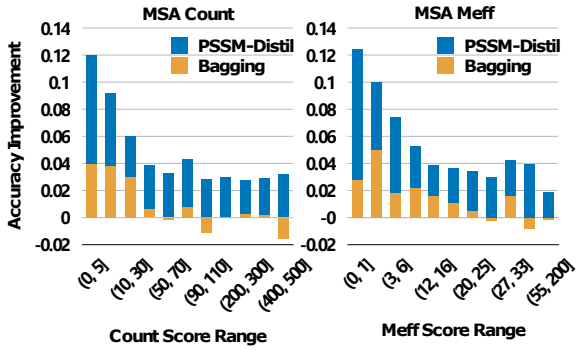

Figure 3: PSSM-Distil PSSP accuracy improvement comparison with "Bagging" over vanilla "Real" model on different count and meff score range. The blue bars are the improvement results of PSSM-Distil, whereas the orange bars are the improvement results of "Bagging". PSSM-Distil significantly improves over "Real" on both low-quality and high-quality cases, while "Bagging" only improves the low-quality cases with the cost of damaging the high-quality prediction accuracy

MSA Counts	Our	w/o BERT	w/o CL	w/o MSE
≤ 60	0.778	0.768	0.772	0.776
≤ 30	0.766	0.754	0.757	0.760
≤ 10	0.759	0.740	0.743	0.748
$= 0$	0.737	0.707	0.714	0.725

Table 4: Ablation study results on the BC40 dataset of our method. The "Our" column is the full pack result of the our method. "w/o BERT" is the result without BERT Pseudo PSSM $\mathbf{X}_b$. "w/o CL" is the result without the triplet loss $\mathcal{L}_t$ from constrastive learning. "w/o MSE" is the result without MSE loss $\mathcal{L}_m$ between $\mathbf{X}_e$ and $\mathbf{X}_h$. The obvious degeneration from ablating each component from our method implies the important role of these components for our method.

out as the side-effect of their method in their paper, which may result from the fixed down-sampling strategy.

Ablation Study. We conduct the ablation study to demonstrate the effectiveness of each designed component for the PSSM-Distil framework and evaluate the performance for low-quality PSSMs on the BC40 dataset with the same four divisions of MSA count. In particular, we remove the auxiliary input $\mathbf{X}_b$ from pre-trained BERT, contrastive learning loss $\mathcal{L}_t$ and MSE loss $\mathcal{L}_m$ respectively to show the ablation results on different MSA count ranges in Table 4.

As shown in Table 4, the results from the "w/o BERT" column entail that the extremely low-quality case suffers the most with 3% of degeneration, which demonstrates the effectiveness of the pre-trained BERT model on these extreme cases. The "w/o MSE" column breaks the direct link of our model to the previous state-of-the-art "Bagging" models, which shows the smallest decreases among the ablation of all the components. These results demonstrated the superior framework choice of PSSM-Distil.

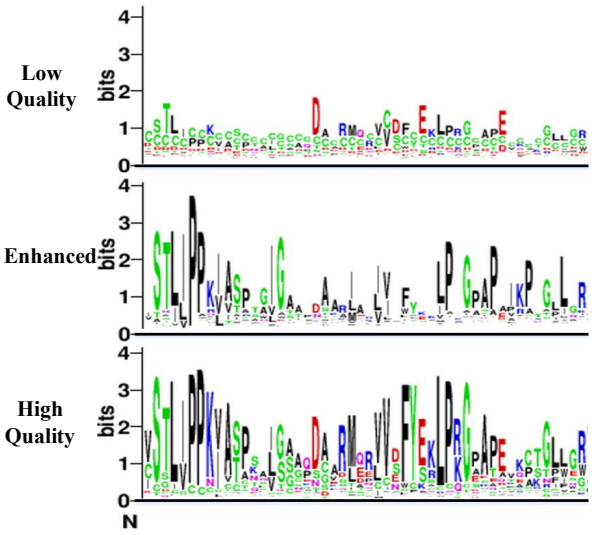

Figure 4: The visualization of the MSA quality by sampling it from low-quality (top row), enhanced (middle row), and high-quality (bottom row) PSSM. Our enhanced PSSM demonstrates higher fidelity reconstruction of the high-quality PSSM with more complex MSA patterns, while low-quality PSSM shows only simple and repetitive patterns.

PSSM Visualization. The visualization results of enhanced PSSMs are given in Fig. 4, which illustrates the capacity of different amino acids on each residue. The graphs are generated using WebLogo[3]. We can easily observe that the original low-quality PSSM (top row) is very sparse and contains less information comparing with high-quality PSSM (bottom row). However, our enhanced PSSM (middle row) is informative and shares similar patterns with high-quality PSSM, which exactly proves the effectiveness of PSSM-Distil on PSSM enhancement.

Conclusion

We present PSSM-Distil, a PSSM enhancement method with knowledge distillation and contrastive learning to tackle the low homologous PSSP issue. We first achieve a teacher network for PSSP by using high-quality PSSMs. Next, we jointly train the EnhanceNet and a student network for PSSM enhancement and PSSP by using the low-quality PSSMs which are down-sampled from high-quality PSSMs. Regardless of the sophisticated architecture design, the novel loss function is elaborated with knowledge distillation, contrastive loss and MSE loss to jointly optimize the EnhanceNet and the student network for the generation of high-quality PSSMs and accurate PSSP. Additionally, we explicitly utilize the BERT pseudo PSSM for extreme low-quality cases' enhancement, i.e., protein sequences with no homology at all. Our work opens up the possibility for research on newly discovered or unknown protein structure prediction with low-quality PSSMs.

[3]https://weblogo.berkeley.edu

Acknowledgments

The work was supported in part by NSFC-Youth 61902335, by the Key Area RD Program of Guangdong Province with grant No. 2018B030338001, by the National Key RD Program of China with grant No. 2018YFB1800800, by Guangdong Regional Joint Fund-Key Projects 2019B1515120039, by Shenzhen Outstanding Talents Training Fund, by Guangdong Research Project No. 2017ZT07X152 and by CCF-Tencent Open Fund.

References

Alley, E. C.; Khimulya, G.; Biswas, S.; AlQuraishi, M.; and Church, G. M. 2019. Unified rational protein engineering with sequence-based deep representation learning. *Nature methods* 16(12): 1315–1322.

Altschul, S. F.; Gish, W.; Miller, W.; Myers, E. W.; and Lipman, D. J. 1990. Basic local alignment search tool. *Journal of molecular biology* 215(3): 403–410.

Altschul, S. F.; Madden, T. L.; Schäffer, A. A.; Zhang, J.; Zhang, Z.; Miller, W.; and Lipman, D. J. 1997. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic acids research* 25(17): 3389–3402.

Bepler, T.; and Berger, B. 2019. Learning protein sequence embeddings using information from structure. *arXiv preprint arXiv:1902.08661* .

Buciluǎ, C.; Caruana, R.; and Niculescu-Mizil, A. 2006. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 535–541.

Chen, G.; Choi, W.; Yu, X.; Han, T.; and Chandraker, M. 2017. Learning efficient object detection models with knowledge distillation. In *Advances in Neural Information Processing Systems*, 742–751.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709* .

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* .

Eddy, S. R. 1998. Profile hidden Markov models. *Bioinformatics (Oxford, England)* 14(9): 755–763.

Guo, Y.; Wu, J.; Ma, H.; Wang, S.; and Huang, J. 2020. Bagging MSA Learning: Enhancing Low-Quality PSSM with Deep Learning for Accurate Protein Structure Property Prediction. In *International Conference on Research in Computational Molecular Biology*, 88–103. Springer.

Hadsell, R.; Chopra, S.; and LeCun, Y. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, 1735–1742. IEEE.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9729–9738.

Heinzinger, M.; Elnaggar, A.; Wang, Y.; Dallago, C.; Nechaev, D.; Matthes, F.; and Rost, B. 2019. Modeling the Language of Life-Deep Learning Protein Sequences. *bioRxiv* 614313.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* .

Johnson, L. S.; Eddy, S. R.; and Portugaly, E. 2010. Hidden Markov model speed heuristic and iterative HMM search procedure. *BMC bioinformatics* 11(1): 431.

Kabsch, W.; and Sander, C. 1983. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers: Original Research on Biomolecules* 22(12): 2577–2637.

Kryshtafovych, A.; Barbato, A.; Fidelis, K.; Monastyrskyy, B.; Schwede, T.; and Tramontano, A. 2014. Assessment of the assessment: evaluation of the model quality estimates in CASP10. *Proteins: Structure, Function, and Bioinformatics* 82: 112–126.

Li, Z.; and Yu, Y. 2016. Protein secondary structure prediction using cascaded convolutional and recurrent neural networks. *arXiv preprint arXiv:1604.07176* .

Mandell, D. J.; and Kortemme, T. 2009. Computer-aided design of functional protein interactions. *Nature chemical biology* 5(11): 797–807.

Mirzadeh, S.-I.; Farajtabar, M.; Li, A.; Levine, N.; Matsukawa, A.; and Ghasemzadeh, H. 2019. Improved Knowledge Distillation via Teacher Assistant. *arXiv preprint arXiv:1902.03393* .

Misra, I.; and Maaten, L. v. d. 2020. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6707–6717.

Noble, M. E.; Endicott, J. A.; and Johnson, L. N. 2004. Protein kinase inhibitors: insights into drug design from structure. *Science* 303(5665): 1800–1805.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* .

Peters, M. E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; and Zettlemoyer, L. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365* .

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2019. Language models are unsupervised multitask learners. *OpenAI Blog* 1(8): 9.

Rao, R.; Bhattacharya, N.; Thomas, N.; Duan, Y.; Chen, P.; Canny, J.; Abbeel, P.; and Song, Y. 2019. Evaluating protein transfer learning with TAPE. In *Advances in Neural Information Processing Systems*, 9686–9698.

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

Rives, A.; Goyal, S.; Meier, J.; Guo, D.; Ott, M.; Zitnick, C. L.; Ma, J.; and Fergus, R. 2019. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *bioRxiv* 622803.

Schmitt, S.; Hudson, J. J.; Zidek, A.; Osindero, S.; Doersch, C.; Czarnecki, W. M.; Leibo, J. Z.; Kuttler, H.; Zisserman, A.; Simonyan, K.; et al. 2018. Kickstarting deep reinforcement learning. *arXiv preprint arXiv:1803.03835* .

Sønderby, S. K.; and Winther, O. 2014. Protein secondary structure prediction with long short term memory networks. *arXiv preprint arXiv:1412.7828* .

Steinegger, M.; and Söding, J. 2017. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology* 35(11): 1026–1028.

Suzek, B. E.; Wang, Y.; Huang, H.; McGarvey, P. B.; Wu, C. H.; and Consortium, U. 2015. UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics* 31(6): 926–932.

Tian, Y.; Krishnan, D.; and Isola, P. 2019. Contrastive representation distillation. *arXiv preprint arXiv:1910.10699* .

Wang, G.; and Dunbrack Jr, R. L. 2003. PISCES: a protein sequence culling server. *Bioinformatics* 19(12): 1589–1591.

Wang, L.; and Jiang, T. 1994. On the complexity of multiple sequence alignment. *Journal of computational biology* 1(4): 337–348.

Wang, R. Y.-R.; Kudryashev, M.; Li, X.; Egelman, E. H.; Basler, M.; Cheng, Y.; Baker, D.; and DiMaio, F. 2015. De novo protein structure determination from near-atomic-resolution cryo-EM maps. *Nature methods* 12(4): 335–338.

Wang, S.; Peng, J.; Ma, J.; and Xu, J. 2016. Protein secondary structure prediction using deep convolutional neural fields. *Scientific reports* 6(1): 1–11.

Wuthrich, K. 1989. Protein structure determination in solution by nuclear magnetic resonance spectroscopy. *Science* 243(4887): 45–50.

Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R. R.; and Le, Q. V. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, 5754–5764.

Yim, J.; Joo, D.; Bae, J.; and Kim, J. 2017. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4133–4141.

Yu, R.; Li, A.; Morariu, V. I.; and Davis, L. S. 2017. Visual relationship detection with internal and external linguistic knowledge distillation. In *Proceedings of the IEEE international conference on computer vision*, 1974–1982.

Zhou, J.; and Troyanskaya, O. G. 2014. Deep supervised and convolutional generative stochastic network for protein secondary structure prediction. *arXiv preprint arXiv:1403.1347* .

Zhuang, C.; Zhai, A. L.; and Yamins, D. 2019. Local aggregation for unsupervised learning of visual embeddings. In *Proceedings of the IEEE International Conference on Computer Vision*, 6002–6012.