

表現学習と深層学習を用いた タンパク質のフォールド認識と残基接触予測

Representation Learning and Deep Architectures for Protein Fold Recognition and Residue Contact Prediction

椿 真史^{1*} 新保 仁¹ 松本 裕治¹

¹ 奈良先端科学技術大学院大学 情報科学研究科 自然言語処理学研究室

Abstract: Predicting the 3D structure of a protein from its amino acid sequence is an important challenge in bioinformatics. Protein fold recognition is a task to classify a protein into one of the “folds”, which are pre-defined structural labels in protein databases. On the other hand, residue contact prediction is a task to determine whether any two residues in a sequence of protein are close to each other in the 3D structure. In this paper, we propose new approaches for these protein structure prediction tasks using (i) learning feature representations from a large protein database and (ii) learning deep neural architectures based on LSTM. Experimental results show that an application of representation learning and deep architectures have a potential to solve the problem of protein structure prediction.

1 はじめに

バイオインフォマティクスにおいて重要な課題の一つに、タンパク質の立体構造予測がある。立体構造予測のタスク設定は様々にあるが、特に重要なのが、おおまかな構造クラスを予測するフォールド認識と、より詳細な3次元立体構造を予測する残基接触予測である。フォールド認識では、与えられたアミノ酸配列を、SCOPE¹ や CATH² などのデータベースで定義された構造クラス(これをフォールドと呼ぶ)に分類することが目標となる。また残基接触予測では、与えられたアミノ酸配列中の残基ペアについて、それらの3次元空間における距離を予測することが目標となる。これらの問題をバイオインフォマティクスでは、生物学の知識と機械学習の技術を用いて解くのが一般的である。

本稿で我々は、表現学習と深層学習を用いたタンパク質のフォールド認識と残基接触予測のモデルを提案する。提案法では、アミノ酸の特徴量を低次元ベクトルで表現するために、近年の自然言語処理で成功した表現学習 [1, 2] を用いる。そして、アミノ酸配列をベクトル系列として表現し、これを入力とした long short-term memory (LSTM) [3] を用いて、タンパク質のフォールド認識と残基接触予測の深層学習モデルを構築する。特に我々の残基接触予測モデルは、既存法が持つ幾つかの制約と問題を解決することができる。我々の知る限

り提案法は、バイオインフォマティクスにおいて初めての、表現学習と深層学習を用いたタンパク質構造予測へのアプローチである。

2 表現学習と深層学習

2.1 単語ベクトル表現学習

2.1.1 Word2vec における skip-gram モデル

Word2vec [1] における skip-gram モデルでは、与えられた単語からその周辺単語を予測するような、低次元の単語ベクトルを学習する。具体的には、単語列 w_1, w_2, \dots, w_L が与えられた時、最小化する損失関数 J は以下のように定義される。

$$J = -\frac{1}{L} \sum_{t=1}^L \sum_{-\ell \leq i \leq \ell} \log p(c_{t+i}|w_t). \quad (1)$$

ここで、 ℓ は窓幅、 w_t は窓幅における中心単語、 c_i は窓幅に含まれる i 番目の周辺単語である。Word2vec の実装では、式 (1) の確率 $p(c_{t+i}|w_t)$ の計算に、負例サンプリングを用いている。負例サンプリングでは、コーパス中で共起する単語ペア (w, c) についてはそれらのベクトルの内積を最大化し、逆にコーパス中で共起しない単語ペア (w, c') についてはそれを最小化すること

*masashi-t@is.naist.jp

¹<http://scop.berkeley.edu/>

²<http://www.cathdb.info/>

によって、確率を以下のように近似して計算する。

$$p(c_{t+i}|w_t) \approx \sigma(\mathbf{c}_{t+i}^\top \mathbf{w}_t) \prod_{k=1}^K \sigma(-\mathbf{c}'_k{}^\top \mathbf{w}_t).$$

ここで、 $\mathbf{w} \in \mathbb{R}^d$ は学習する単語 w のベクトル、 $\mathbf{c} \in \mathbb{R}^d$ は学習する周辺単語 c のベクトル、 \mathbf{c}' は負例の周辺単語のベクトル、 K はサンプルする負例の単語数、 σ はシグモイド関数である。

2.1.2 Global vectors

Global vectors (GloVe) [2] は、対数双線形回帰モデルに基づく単語ベクトル表現学習法であり、word2vec における窓幅のような局所的な情報と、LSA におけるコーパス中の共起頻度のような大域的な情報の両方を用いている。具体的には、GloVe において最小化する損失関数 J は、以下のように定義される。

$$J = \sum_{i,j=1}^V f(\#(w_i, c_j)) (\mathbf{w}_i^\top \mathbf{c}_j + b_i + b_j - \log(\#(w_i, c_j)))^2.$$

ここで、 $\#(w, c)$ は単語 w と周辺単語 c の共起頻度、 $\mathbf{w} \in \mathbb{R}^d$ は学習する単語 w のベクトル、 $\mathbf{c} \in \mathbb{R}^d$ は学習する周辺単語 c のベクトル、 b_i と b_j は学習する単語と周辺単語のバイアス項、 $f(x)$ は重み関数である。Pennington ら [2] はこの重み関数を

$$f(x) = \begin{cases} (x/x_{\max})^\alpha & x < x_{\max} \\ 1 & \text{otherwise} \end{cases}$$

と定義しており、実験では $x_{\max} = 100$ 、 $\alpha = 3/4$ と設定している。このように GloVe は、共起頻度に対して重みやバイアス項を考慮した上で、単語ベクトルと周辺単語ベクトルの内積と共起頻度の対数との二乗誤差最小化を行っている。

2.2 Long Short-Term Memory

再帰的ニューラルネットワーク (RNN) は、任意長の系列データに対して広く用いられるニューラルネットワークであるが、勾配消失問題のため長い系列を学習することができない。この問題を解決するために、Hochreiter と Schmidhuber [3] は、long short-term memory (LSTM) を提案した。LSTM は、RNN の隠れ層にメモリーセルを持たせることで勾配消失問題を解決し、長い系列の情報を適切に保持かつ学習することができる。

具体的に、LSTM のメモリーセルは、情報の入出力の流れをコントロールする 3 種類のゲート：入力ゲート $\mathbf{i}_t \in \mathbb{R}^d$ 、忘却ゲート $\mathbf{f}_t \in \mathbb{R}^d$ 、出力ゲート $\mathbf{o}_t \in \mathbb{R}^d$

から構成される。入力ベクトル $\mathbf{x}_t \in \mathbb{R}^d$ 、直前の隠れ層 $\mathbf{h}_{t-1} \in \mathbb{R}^d$ 、直前のセル $\mathbf{c}_{t-1} \in \mathbb{R}^d$ が与えられた時、LSTM は次のセル \mathbf{c}_t と隠れ層 \mathbf{h}_t を以下のように計算する。

$$\mathbf{H} = \begin{bmatrix} \mathbf{x}_t \\ \mathbf{h}_{t-1} \end{bmatrix}, \quad (2)$$

$$\mathbf{i}_t = \sigma(\mathbf{W}_i \mathbf{H} + \mathbf{b}_i), \quad (3)$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_f \mathbf{H} + \mathbf{b}_f), \quad (4)$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_o \mathbf{H} + \mathbf{b}_o), \quad (5)$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tanh(\mathbf{W}_c \mathbf{H} + \mathbf{b}_c), \quad (6)$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t). \quad (7)$$

ここで、 $\mathbf{W}_i, \mathbf{W}_f, \mathbf{W}_o, \mathbf{W}_c \in \mathbb{R}^{2d \times d}$ は学習する重み行列、 $\mathbf{b}_i, \mathbf{b}_f, \mathbf{b}_o, \mathbf{b}_c \in \mathbb{R}^d$ は学習するバイアスベクトル、 σ はシグモイド関数、 \tanh は hyperbolic tangent 関数、 \odot はアダマール積である。本稿では、上記の式 (2) から (7) の一連の計算を以下のように表現し、これを LSTM 関数とする。

$$\langle \mathbf{c}_t, \mathbf{h}_t \rangle = lstm(\mathbf{x}_t, \mathbf{h}_{t-1}, \mathbf{c}_{t-1}). \quad (8)$$

最終的に、出力ベクトル $\mathbf{y}_t \in \mathbb{R}^k$ は

$$\mathbf{y}_t = \mathbf{W}_y \mathbf{h}_t + \mathbf{b}_y$$

と計算される。ここで、 $\mathbf{W}_y \in \mathbb{R}^{d \times k}$ は学習する重み行列、 $\mathbf{b}_y \in \mathbb{R}^k$ は学習するバイアスベクトル、 k は出力の次元である。

3 タンパク質の構造予測

3.1 フォールド認識

タンパク質のフォールド認識とは、与えられたタンパク質を、SCOPE や CATH などのデータベースで定義された構造クラス (フォールド) に分類するタスクである。問題設定としては、入力が任意長の配列で出力がラベルであるため、機械学習の設定としては比較的単純な分類問題である。分類器には主に SVM などを用いる一方で、多くの研究は、任意長の配列であるタンパク質をどのような特徴ベクトルとして表現するかに焦点が当てられている [4, 5, 6, 7, 8]。

3.2 残基接触予測

タンパク質の立体構造情報は、PDB ファイルで提供されている³。PDB ファイルには、タンパク質中の全

³<http://www.wwpdb.org/>

アミノ酸中の全原子について、それらの3次元空間における座標(単位はÅ)が記述されている。我々は、この座標情報から原子間の距離などを計算することで、タンパク質の立体構造を正確に捉えることができる。

タンパク質の立体構造予測には様々な評価方法があり、またそれはフォールド認識と比べて複雑である。本稿では、タンパク質構造予測の世界的なコンペティションである CASP⁴ の評価方法の一つである、残基接触予測タスクを行う。残基接触予測とは、アミノ酸配列中で24残基⁵以上離れておりかつ3次元空間における距離が8Å以下の残基ペアを予測することである。このような残基ペアを、長距離接触あるいは長距離コンタクトと呼ぶ。残基接触予測の評価には、全残基ペア(配列長が L の時、ペアの総数は $L(L-1)/2$)中の上位 $L/5$ に対して、予測器がコンタクトと予測したものが実際にコンタクトである割合(つまり precision)を用い、これを予測精度とする。

Precision を用いる理由は、一つのタンパク質中では、長距離コンタクトしている残基ペアの数が非常に少ないためである(つまり訓練データはアンバランスデータとなる)。また、 $L/5$ のという少数のデータのみしか評価に用いない点も、同様の理由である。しかしたとえ少数であったとしても、長距離コンタクトを正確に予測できさえすれば、それはタンパク質の構造に関する非常に有益な情報となる。このような事実を踏まえ、残基接触予測はこのような評価方法となっている。

4 提案法

4.1 LSTMを用いたフォールド認識モデル

タンパク質に単語ベクトル表現学習法を適用するため、我々は3-gramアミノ酸をタンパク質における単語と定義する。 w_i を i 番目の単語とすると、タンパク質 P は単語の系列として $P = w_1, w_2, \dots, w_L$ と表現できる。このように表現された大量のタンパク質を学習コーパスとして、我々はskip-gramとGloVeを適用する。

次に我々は、単語列中のすべての単語をベクトルに変換し、 x_1, x_2, \dots, x_L の単語ベクトル系列を得る。ここで、 $x_i \in \mathbb{R}^d$ は i 番目の単語ベクトルである。この単語ベクトル系列を入力とし、LSTMを用いて、隠れ層系列 h_1, h_2, \dots, h_L を得る。ここで、 $h_i \in \mathbb{R}^d$ は i 番目の単語の隠れ層である。特に我々は、双方向かつ階層的な構造を持つLSTMであるstacked bi-directional LSTM [9]を用いて、この隠れ層系列を得る。具体的にはまず、式(8)を用いて、 $lstm_f$ をフォワード(単語

列を左から右に読む)のLSTM関数、 $lstm_b$ をバックワード(単語列を右から左に読む)のLSTM関数とした上で、stacked bi-directional LSTMのフォワードとバックワードの計算を、以下のように記述する。

$$\begin{aligned} \langle \vec{c}_i, \vec{h}_i \rangle &= lstm_f(x_i, \overleftarrow{h_{i-1}}, \overleftarrow{c_{i-1}}), \\ \langle c_i, h_i \rangle &= lstm_b(\overrightarrow{h_i}, \overleftarrow{h_{i+1}}, \overleftarrow{c_{i+1}}). \end{aligned}$$

このように、stacked bi-directional LSTMは、フォワード計算した隠れ層を次のバックワード計算の入力として、再び隠れ層を計算する。そして我々は、出力ベクトル $y \in \mathbb{R}^k$ を、すべての隠れ層の総和を用いて以下のように計算する。

$$y = W_y \sum_{i=1}^L h_i + b_y.$$

ここで、 $W_y \in \mathbb{R}^{d \times k}$ は学習する重み行列、 $b_y \in \mathbb{R}^k$ は学習する重みベクトル、 k は出力の次元(つまり分類するフォールドのクラス数)である。最後に、ソフトマックス関数を用いて

$$p_t = \text{softmax}(y) = \frac{\exp(y_t)}{\sum_{i=1}^k \exp(y_i)}$$

を計算する。ここで、 t はクラスのインデックスである。最小化する損失関数 \mathcal{L} は、以下の正則化付きクロスエントロピーである。

$$\mathcal{L}(\Theta) = - \sum_{i=1}^N \log p_{t_i} + \frac{\lambda}{2} \|\Theta\|_2^2.$$

ここで、 N は訓練データの総数、 Θ は学習パラメータの集合であり、LSTMの学習パラメータと単語ベクトル表現である。このように我々は、フォールド認識モデルであるLSTMのパラメータだけでなく、入力である3gramアミノ酸の特徴ベクトル表現も同時に学習する。図1に、提案法の概略図を示した。

4.2 LSTMを用いた内部座標に基づく残基接触予測モデル

機械学習、特にニューラルネットワークを用いた残基接触予測の研究は古くから存在し[13, 14]、近年では深層学習を用いた手法も提案されている[11, 12]。しかし、これらのアプローチの基本的な部分は以下の点で似通っており、またそれに伴い、主に以下の3つ問題が生じる。

1. アミノ酸の特徴量としてその生物学的な情報が用いられるが、これらは主にバイナリであり、構造学習の中で再学習できない。

⁴<http://www.predictioncenter.org/>

⁵より正確には、各アミノ酸中の C_β 原子(ただしグリシンには C_β 原子がないので C_α 原子)である。

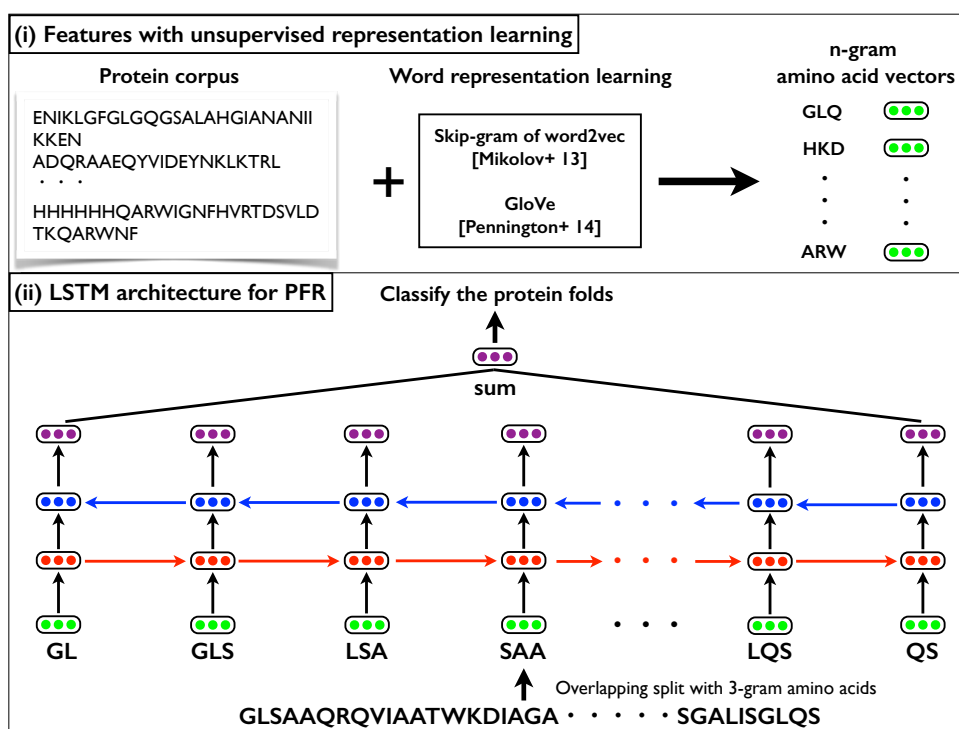


図 1: タンパク質コーパスからの単語ベクトル表現学習と、stacked bi-directional LSTM を用いたフォールド認識モデルの概略図。フォールド認識は、各々の単語が出力する隠れ層の総和を用いた分類モデルである。一方で残基接触予測は、出力が各々の単語の内部座標であり、3次元実数値ベクトル系列を教師とした回帰モデルとなる。

2. 配列中のアミノ酸の特徴ベクトルのペアを入力として、それらがコンタクトか否かを判定する分類器を学習する。しかしこれでは、配列の局所的な情報を個別に与えて学習することになってしまう。
3. 1つのタンパク質中では長距離コンタクトの割合は非常に少ないため、訓練データがアンバランスになり、学習の際に負例データのほとんどを捨てなければならない。

以上の問題を我々は、以下に述べる提案法で解決する。

提案する残基接触予測モデルは、フォールド認識モデルと同様に、単語ベクトル系列を入力とした stacked bi-directional LSTM を用いる点では共通である。しかしその出力は、各アミノ酸の内部座標である。内部座標とは、PDF ファイルの相対座標から計算される、直前の2つの残基によって張られる2次元平面に基づく3次元空間の座標表現である。つまり我々は、入力をタンパク質の単語ベクトル系列、出力を内部座標の3次元実数値ベクトル系列として、分類問題ではなく回帰問題として残基接触予測を解く。

具体的には、まず stacked bi-directional LSTM によって計算された隠れ層 h_i を用いて、出力 $y_i = W_y h_i + b_y$ を計算する。そして、最小化する損失関数 \mathcal{L} を以下

の二乗誤差とする。

$$\mathcal{L}(\theta) = \frac{1}{2} \sum_{P \in \mathcal{D}} \sum_{i=1}^{|P|} \|t_i - y_i\|_2^2 + \frac{\lambda}{2} \|\theta\|_2^2. \quad (9)$$

ここで、 \mathcal{D} はタンパク質の訓練データ集合、 P はタンパク質、 $|P|$ はそのタンパク質の長さ、 $t_i \in \mathbb{R}^3$ は内部座標、 θ は学習パラメータの集合である。この θ は、フォールド認識モデルと同様、LSTM のパラメータと単語ベクトルである。

このように、内部座標を教師としたタンパク質の構造学習モデルは、この節の冒頭で述べた3つの問題を以下のように解決することができる。

1. 単語ベクトル表現は、構造学習の中で学習することができ、残基接触予測により適した特徴量となる。
2. アミノ酸の特徴ベクトルのペアを個別に与えて学習するのではなく、配列中のアミノ酸間の長距離依存関係を考慮した上で、タンパク質全体の構造を学習できる。
3. タンパク質の構造を、あくまでアミノ酸の系列データとして学習するため、訓練データは大量の

既存法 (特徴量 & 分類器)	精度 (%)
Taguchi and Gromiha, 2007 [6] (co-occurrence & SVM)	51.0
Ghanty and Pal, 2009 [7] (PF1 & SVM)	50.6
Ghanty and Pal, 2009 [7] (PF2 & SVM)	48.2
Yang et al., 2011 [10] (PF & SVM)	53.4
Sharma et al., 2013 [8] (15 physicochemical attributes & SVM)	40.2
Sharma et al., 2013 [8] (15 physicochemical attributes & NB)	45.3
Sharma et al., 2013 [8] (30 physicochemical attributes & SVM)	43.6
Sharma et al., 2013 [8] (30 physicochemical attributes & NB)	50.9
提案法	精度 (%)
Stacked bi-directional LSTM (skip-gram)	52.4
Stacked bi-directional LSTM (skip-gram, re-training of word rep)	55.9
Stacked bi-directional LSTM (GloVe)	54.0
Stacked bi-directional LSTM (GloVe, re-training of word rep)	56.9

表 1: DD データセットにおける, 既存法と提案法のフォールド認識精度の比較.

既存法	精度 (%)
Eickholt et al., 2012 [11]	29.1
Lena et al., 2012 [12]	31.0
提案法	精度 (%)
Stack bi-directional LSTM (GloVe)	9.7
Stack bi-directional LSTM (Transfer)	15.5

表 2: CASP9 における, 既存法と提案法の残基接触予測精度の比較.

負例が存在するようなアンバランスデータとはならない.

さらに我々は, フォールド認識で学習した単語ベクトル表現を, 残基接触予測の初期値に用いる. このような転移学習によって, フォールド認識モデルで獲得された構造クラスを分類する特徴量が, 残基接触予測においても有効どうかを検証する.

5 実験

5.1 フォールド認識

フォールド認識のデータセットには, ベンチマークの1つである DD データセット [5] を用いた. このデータセットには 695 のタンパク質が含まれており, 各々の配列類似度は 35–40 %, フォールドのクラス数は 27 となっている.

単語ベクトル表現の訓練コーパスには, CATH に登録されている 86670 のタンパク質を用いた. 単語ベクトルの次元は $d = 100$, 窓幅は $\ell = 10$, skip-gram での負例サンプリングの数は $K = 5$ とした. その他のハイパーパラメータは word2vec⁶ と GloVe⁷ のツールキットで設定されている初期値を用いた. これらの単語ベクトルは, LSTM の初期値として用いるが固定ではなく, LSTM の学習の際に同時に学習する.

LSTM は Chainer⁸ で実装し, その学習はほぼ標準的な設定で行った. 最適化は ADAM [15] を用い, バッチサイズは 50 とし, 10 フォールドクロスバリデーションで, ドロップアウト率と L2 正則化項の λ を決定した.

表 1 に, 既存法と提案法のフォールド認識精度を示す. アミノ酸の共起情報 (co-occurrence) やペアワイズの頻度 (pairwise frequency: PF) を用いる手法や, 様々な物理化学的な特徴量 (physicochemical attributes) を用いる手法に比べて, 提案法は高い精度を達成した. 特に, GloVe の単語ベクトル表現を初期値として, それを LSTM の学習の際に再学習することで, より高い精度が達成できることを確認した.

5.2 3次元構造予測

残基接触予測データセットとして我々は, 訓練データに DNcon [11], テストデータに CASP9 のデータセットを用いた. 提案法の LSTM は, フォールド認識と同様に Chainer で実装し, その学習も同様の設定で行っ

⁶<https://code.google.com/archive/p/word2vec/>

⁷<http://nlp.stanford.edu/projects/glove/>

⁸<http://chainer.org/>

た。ただし単語ベクトルは、コーパスで学習した GloVe と、フォールド認識で再学習した GloVe の 2 種類を用いて、転移学習の影響を比較検証した。

表 2 に、既存法と提案法の残基接触予測精度を示す。提案法は、既存法の深層学習モデルの 2 つ [11, 12] と比べて低い精度となった。既存法では、アミノ酸の生物学的な特徴量を、大規模なデータベースやバイオインフォマティクスのツールを用いて作成しているため、特徴量設計の時点でまず大きな差があると考えられる。あるいは、提案法は既存法の問題点を解決できた反面、教師が 3 次元実数値ベクトルである回帰問題という、分類問題と比べてより難しい問題になってしまったと考えられる。しかし、フォールド認識で学習した単語ベクトルを用いると、精度の向上が見られた。このことから、より大規模なデータセットでフォールド認識を学習し、それを転移学習させることで、残基接触予測の精度を向上させることができると期待される。

6 結論と今後の課題

本稿では、表現学習と深層学習を用いて、タンパク質のフォールド認識と残基接触予測の新たなモデルを提案した。フォールド認識では提案法の優位性を確認できた一方で、残基接触予測では課題が残った。今後は、モチーフなどの別の単語表現や、echo state network などの別の RNN を用いていきたい。

参考文献

- [1] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
- [2] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Empirical Methods on Natural Language Processing (EMNLP)*, 2014.
- [3] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [4] Inna Dubchak, Ilya B Muchnik, and Sung-Hou Kim. Protein folding class predictor for scop: approach based on global descriptors. In *Ismb*, 1997.
- [5] Chris HQ Ding and Inna Dubchak. Multi-class protein fold recognition using support vector machines and neural networks. *Bioinformatics*, 17(4):349–358, 2001.
- [6] YH Taguchi and M Michael Gromiha. Application of amino acid occurrence for discriminating different folding types of globular proteins. *BMC bioinformatics*, 8(1):1, 2007.
- [7] Pradip Ghanty and Nikhil R Pal. Prediction of protein folds: extraction of new features, dimensionality reduction, and fusion of heterogeneous classifiers. *NanoBioscience, IEEE Transactions on*, 8(1):100–110, 2009.
- [8] Alok Sharma, Kuldeep K Paliwal, Abdollah Dehzangi, James Lyons, Seiya Imoto, and Satoru Miyano. A strategy to select suitable physicochemical attributes of amino acids for protein fold recognition. *BMC bioinformatics*, 14(1):1, 2013.
- [9] Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. How to construct deep recurrent neural networks. *arXiv preprint arXiv:1312.6026*, 2013.
- [10] Tao Yang, Vojislav Kecman, Longbing Cao, Chengqi Zhang, and Joshua Zhexue Huang. Margin-based ensemble classifier for protein fold recognition. *Expert Systems with Applications*, 38(10):12348–12355, 2011.
- [11] Jesse Eickholt and Jianlin Cheng. Predicting protein residue-residue contacts using deep networks and boosting. *Bioinformatics*, 28(23):3066–3072, 2012.
- [12] Pietro D Lena, Ken Nagata, and Pierre F Baldi. Deep architectures for protein contact map prediction. *Bioinformatics*, 28(19):2449–2457, 2012.
- [13] P Fariselli and R Casadio. A neural network based predictor of residue contacts in proteins. *Protein Engineering*, 12(1):15–21, 1999.
- [14] Piero Fariselli, Osvaldo Olmea, Alfonso Valencia, and Rita Casadio. Prediction of contact maps with neural networks and correlated mutations. *Protein engineering*, 14(11):835–843, 2001.
- [15] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.