Briefings in Functional Genomics, 00(00), 2021, 1–15

doi: 10.1093/bfgp/elab025
Review Paper

Pretraining model for biological sequence data

Bosheng Song, Zimeng Li, Xuan Lin, Jianmin Wang, Tian Wang and Xiangzheng Fu

Corresponding author: Xiangzheng Fu, College of Information Science and Engineering, Hunan University, Changsha, Hunan, China. Tel: 86-0731-88821907; E-mail: fxz326@hnu.edu.cn

Abstract

With the development of high-throughput sequencing technology, biological sequence data reflecting life information becomes increasingly accessible. Particularly on the background of the COVID-19 pandemic, biological sequence data play an important role in detecting diseases, analyzing the mechanism and discovering specific drugs. In recent years, pretraining models that have emerged in natural language processing have attracted widespread attention in many research fields not only to decrease training cost but also to improve performance on downstream tasks. Pretraining models are used for embedding biological sequence and extracting feature from large biological sequence corpus to comprehensively understand the biological sequence data. In this survey, we provide a broad review on pretraining models for biological sequence data. Moreover, we first introduce biological sequences and corresponding datasets, including brief description and accessible link. Subsequently, we systematically summarize popular pretraining models for biological sequences based on four categories: CNN, word2vec, LSTM and Transformer. Then, we present some applications with proposed pretraining models on downstream tasks to explain the role of pretraining models. Next, we provide a novel pretraining scheme for protein sequences and a multitask benchmark for protein pretraining models. Finally, we discuss the challenges and future directions in pretraining models for biological sequences.

Key words: biological sequence; pretraining model; deep learning

Introduction

Biological sequence data composed of protein, DNA, and RNA sequences are an important field in life science. Based on scientific research, biological sequence data imply life rules and offer an excellent window to explore biochemical roles [1]. Learning biological sequences by deep learning methods, researchers can not only infer the biological properties of unseen sequences but also predict interactions without understanding the underlying physical or biological mechanisms [2]. In particular, during the coronavirus disease 2019 (COVID-19) pandemic, many researchers explore related issues based on biological sequences [3–8]. However, considering that biological sequences are long and nonnumeric, finding a suitable way to convert biological sequences into processable representation is difficult. Moreover, the lack of labeled biological sequences affects their performance on corresponding tasks.

Recently, unsupervised learning [9, 10] on biological sequences has attracted many researchers, among which pretraining model for biological sequence data is a field in great demand. The pretraining model is a saved model that has been trained in advance. In general, pretraining models are first trained on large datasets to be fitted and generalized. Subsequently, the trained parameters and weights of pretraining models will be saved. Finally, saved pretraining models are applied in other tasks directly or after fine-tuning on other datasets. During the development of the pretraining model for biological sequences, many different models have made contributions to this field. Convolutional neural network (CNN) [8, 11–13] models extract

Bosheng Song is a professor at Hunan University. His research interests include biocomputing and bioinformatics.
Zimeng Li is a master at Hunan University. His research interest is pretraining model in bioinformatics.
Xuan Lin is a doctor at Hunan University. His research interests include GNN and drug discover.
Jianmin Wang is a doctor at Hunan University. His research interest is molecular generation.
Tian Wang is a professor at Beijing Normal University. His research interests include internet of things and artificial intelligence.
Xiangzheng Fu is a doctor at Hunan University. His research interest is classification of proteins in bioinformatics.

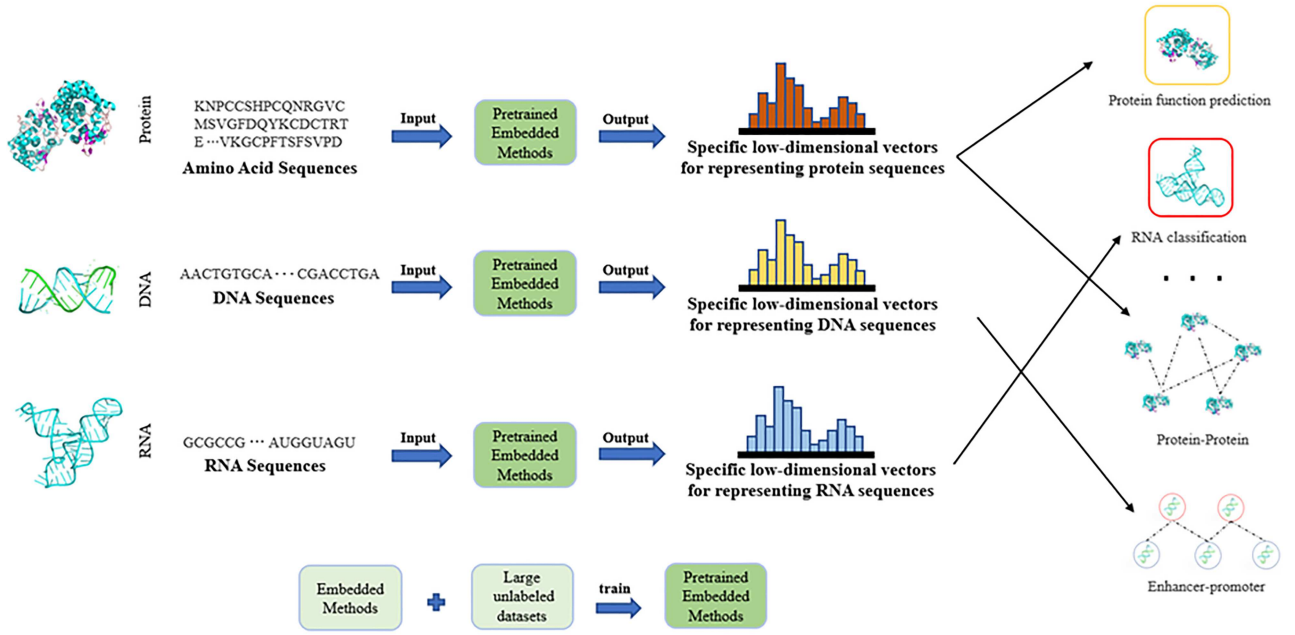

Figure 1. Scheme of using pretraining models for biological sequences. It illustrates the process of using pretraining models for biological sequences and consists of input biological sequences, output representing vectors through pretraining models and downstream applications. Biological macromolecules are generated in Pymol [30].

features from biological sequences efficiently and optimize deep learning models in transfer learning. Word2vec [14] models generate reliable embedding vectors for representing biological sequences and transfer learned knowledge to downstream tasks. Long short-term memory (LSTM) [15] and ELMO [16] language models process long biological sequences while providing context information among biological sequences for downstream tasks. Transformer [17] and Bert [18] models learn biological sequences on the basis of attention mechanism, which improve performance on downstream tasks after pretraining on large-scale sequence datasets.

Biological sequence can be regarded as a special life language, similar to human natural language. Thanks to the advancement of natural language processing (NLP) [19], pretraining models can effectively extract the characteristics of biological sequences and encode biological sequences after training in a large unlabeled corpus. As shown in Figure 1, pretraining language models and word embedding methods in NLP can embed biological sequence with processable low-dimensional representation, which improve the performance when applied in downstream tasks [20, 21]. Consequently, pretraining models using NLP methods have captured syntactic and semantic information in biological sequences.

As shown in many recent works, pretraining models have achieved many biological sequence tasks. Pretraining models for biological sequences have learnt context-sensitive representation from various unlabeled biological sequences, which implicitly reflect general knowledge in biological sequences. Using pretraining models, the knowledge learned from open field can be transferred to downstream tasks such as drug–target interaction (DTI) [22–24], enhancer–promoter interaction (EPI) [25] and protein classification [26–29], making most of the methods perform better with less cost. In addition, features extracted from several unlabeled sequences are beneficial to tasks without enough label data. Moreover, pretraining models can be used for transfer learning. Such models are initially pretrained on different datasets and then fine-tuned on the target datasets for specific tasks, which effectively optimize the network, improve performance and save time with good scalability.

In this survey, we review literature on pretraining models for biological sequence data to make readers understand the area approximately and enlighten other researchers in this area. This survey is important to novices and researchers who are looking for an alternative for improving this area, when it comes to using pretraining models to learn biological sequence data. The key contributions of this survey are as follows: (1) we summarize and briefly introduce some important biological sequence datasets appearing in these works. (2) We conduct a systematic review on pretraining models for biological sequences and organize the current methods by different basic methods. (3) We have introduced some application and methods for downstream tasks with proposed pretraining models. (4) We provide an important scheme and benchmark, which are proposed by previous studies, for protein pretraining models. (5) We discuss the challenges and future directions of pretraining models for biological sequence data, which may provide new ideas for researchers and promote development in the field.

The remaining of this review is organized as follows: we provide introduction to biological sequence data and related popular databases in Overview of biological sequence data. Pretraining models for biological sequences are summarized in Pretraining model. Some applications on downstream tasks with proposed pretraining models for biological sequences are illustrated in Application of pretraining model. Scheme and benchmark offers a novel pretraining model scheme and benchmark for protein sequences. Challenges and future directions discusses the challenges in the current methods and future directions in this field.

Overview of biological sequence data

Biological sequence data contain abundant biometric information, which are stored in the sequence structure [31]. Exploring hidden roles in healthy and diseases with biological

sequences is important. In addition, with the advancement of next-generation sequencing technology, a growing number of sequences can be obtained and invested in various deep learning research tasks. Consequently, in the past few years, many biological sequence datasets are proposed for different tasks in various published papers.

In this section, we introduce some details about biological sequence data and popular historical databases that are used frequently in research. We also summarize the biological sequence databases and datasets that are used by the surveyed papers in Table 1. The brief description and accessible URL are supposed to provide a convenient way for novices and researchers, which help them to obtain the necessary databases and datasets.

Biological sequence data

In general, biological sequence is the long sequence represented by a string of different and fixed alphabets, in which different alphabets usually represent different micromolecules. For example, a DNA sequence is made up of a four-letter alphabet 'A', 'C', 'T' and 'G', which represent different kinds of deoxynucleotide in DNA [52].

DNA sequence, RNA sequence and protein sequence are collectively called biological sequence. Proteins are important fundamental macromolecules in the human body, which have vital functions in life activities. Proteins are always folded into a unique three-dimensional structure by several amino acid chains. Protein functions have been determined by protein structure and sequences for the greater part. Due to the specific and various 3D conformations, proteins with amino acid sequences have specific and wide array of functions, such as transmitting nerve pulses and binding specificity [53]. Consequently, amino acid sequence is often the key research object to explore protein properties and interactions, such as DTI [22], compound-protein interaction (CPI) [54], protein classification [26] and protein function prediction [55].

Similar to protein, DNA is also a bioactive macromolecule essential for the development and normal operation of organisms in biological cells. Given its vital genetic information, DNA not only controls the biological inheritance and activities but also serves as the basis of RNA and protein synthesis. Different from the folding structure of protein, the molecular structure of DNA is double helix formed by two polydeoxynucleotide chains [56]. In bioinformatics, DNA fragment sequences are usually used for scientific research, such as exploring the interactions between promoter and enhancer [25, 44]. As a genetic information carrier, RNA is another common biological macromolecule. Most of RNAs are single stranded, which are made up of a ribonucleotide chain [57]. RNA is closely related to proteins, which can control protein synthesis. Thus, RNA sequences can reflect life information from another point of view in bioinformatics [58].

Databases for biological sequence

Protein Data Bank (PDB) [32] is the important collection of biological macromolecules, which is a 2.5-dimensional structure database, which preserves crucial information on various biological macromolecules, such as atomic coordinates, sequences, references and level 1 and level 2 structural information. The Structural Classification of Proteins (SCOP) [30] and Structural Classification of Proteins—extended (SCOPe) [39] are commonly used protein structural databases that classify known proteins by family, superfamily, common fold and class, thereby providing rich information about proteins, such as structural, sequence data and evolutionary relationships. Pfam [35] is a well-known protein family database that divides proteins into different families by multiple sequence comparisons and hidden Markov models. Because of the protein family characteristics of Pfam, researchers always use Pfam as data sources when proteins with similar functions are needed by tasks.

Universal Protein (UniProt) [36] is the informative protein database that integrates the resources of three major databases: EBI, SIB and PIR. With the continuous increase of protein sequence data, UniProt has more than 120 million protein sequences and annotations. SWISS-PROT [33] is the complete annotated protein sequence database. The advantage of SWISS-PROT is the detailed annotation information and standardized nomenclature of protein sequences, which provide annotated protein sequences for various tasks. UniRef [37] has been a frequently used protein sequence database since its first release in 2004. On the basis of different sequence identity levels, UniRef is divided into three different protein sequence subsets: UniRef100, UniRef90 and UniRef50, which meet different task requirements.

DrugBank [45] is a bioinformatic–cheminformatic database that combines the information of drugs and targets. For drugs, DrugBank provides drug chemical structures, pharmacological effects, protein targets, drug–drug interactions, etc. As for protein targets, DrugBank stores related information, such as protein sequence, structure and approach. ChEMBL [48] is another outstanding database that provides reliable information of compound and target, which obtains bioactivity data and structures for small molecules from a variety of journals. Similar to ChEMBL, BindingDB [46] is also an open database that extracts data from scientific literature. BindingDB database primarily provides binding affinities between compound and target protein, focusing on drug–target proteins.

Pretraining model

With the development of NLP technology, pretraining models have gradually become a hot field in deep learning. Owing to the improvement of software and proposed new methods, pretraining models have substantially achieved state-of-the-art results in almost all NLP tasks [59]. Several methods have made contributions to biological sequence-related tasks as a pretraining model, to improve the performance and speed up the training process. In the early years, neural network models [11, 14, 15] have occupied the mainstream of pretraining models, which generate word vectors for representing sequences. In recent years, with the introduction of attention mechanism and development of Transformer-based methods [17, 18], pretraining language models succeeded in many fields. Table 2 summarizes four types of popular pretraining models and their brief description. In this section, we introduce four categories of pretraining models for biological sequence data that have been used in surveyed papers: CNN, word2vec, LSTM and Transformer.

Convolutional neural network

CNN [11], one of the classic neural network structures in deep learning, has outstanding performance in many fields. Inspired by local receptive field mechanism, CNN uses convolution operations to extract features with other network structure to crop features and transform output. The specific architecture of CNN determines the unique advantages in the Computer Vision (CV)

Table 1. List of biological sequence databases and datasets

Category	Dataset	Year	Entities	Description	URL (Source)
Protein	PDB [32]	1971	2.5-dimensional structure of biological macromolecules	Protein structure database, containing 3D structures obtained through experiments	http://www.rcsb.org
	SWISS-PROT [33]	1986	Protein sequence	Annotated protein sequence database	http://www.uniprot.org
	SCOP [34]	1994	Protein sequences and structures	Database of protein structure classification according to the spatial characteristics of protein domains	http://scop2.mrc-lmb.cam.ac.uk
	Pfam [35]	1995	Protein sequence	Protein family database, including annotations and sequences of 17 929 protein families	http://pfam.xfam.org/
	UniProt [36]	2002	Protein sequence	A database consisting of a large number of labeled and unlabeled primary protein sequences	http://www.uniprot.org
	UniRef [37]	2004	Protein sequence	Unlabeled big data protein sequence	http://www.uniprot.org
	DisProt [38]	2007	Protein sequence	The database of disordered proteins	https://www.disprot.org/
	SCOPe [39]	2012	Protein structural relationships	59 514 protein database (PDB) entries, including more than 65% of the protein structures in the PDB	http://scop.berkeley.edu/
	BFD [40]	2018	Protein sequences	Largest set of protein sequences	https://metaclust.mmseqs.org/.
	ProteinNet [41]	2019	Protein sequences and structure	A standardized dataset for machine learning of protein structure	https://github.com/aqlaboratory/proteinnet
Nucleic acid	GENCODE [42]	2003	Genome annotation	Documented the functional annotation of the genome	https://www.gencodegenes.org.
	circRNADb [43]	2016	circRNAs sequences	Contains 32 914 human circRNAs	http://reprod.njmu.edu.cn/circrnadb
	TargetFinder [44]	2016	Enhancer–promoter interactions	Contains enhancer and promoter interactions in six human cell lines (GM12878, HUVEC, HeLa-S3, IMR90, K562, NHEK)	https://github.com/shwhalen/targetfinder
Interaction	DrugBank [45]	2006	Drug–target associations	17 000 high-quality standard drug–target associations	https://www.drugbank.ca/
	BindingDB [46]	2007	Compound–protein interaction	39 747 positive instances and 31 218 negative instances	http://www.bindingdb.org/bind/
	STITCH [47]	2007	Compound–protein interaction	Interactions between more than 30 000 small molecule compounds and 2.6 million proteins from 1133 species	http://stitch.embl.de
	ChEMBL [48]	2009	Drug–target associations	Collected 12 482 targets, 1.879 million compounds and a total of 155 million pieces of biological activity information	https://www.ebi.ac.uk/chembl/
	HIPPIE [49]	2012	Protein–protein interactions	Human PPI dataset with standardized scoring	http://cbdm.uni-mainz.de/hippie/
	KIBA [50]	2014	Target–ligand associations	467 targets and 52 498 ligands collected from ChEMBL and STITCH	https://pubs.acs.org/doi/abs/10.1021/ci400709d
	GLASS [51]	2015	GPCR-ligand associations	A large number of experimentally verified GPCR-ligand associations	http://zhanglab.ccmb.med.umich.edu/GLASS/.

Table 2. Summary of popular pretraining models for biological sequences

Category	Algorithm	Author	Year	Description	Source code
CNN	CNN [11]	Lecun *et al.*	1998	A common deep learning network architecture inspired by biological natural visual cognitive mechanisms	
word2vec	word2vec [14]	Mikolov et.al	2013	A well-known unsupervised method to learn high-quality embedded vectors representations of words	code.google.com/p/word2vec
	doc2vec [60]	Mikolov et.al	2014	An improved unsupervised embedding method for variable length texts	https://radimrehurek.com/gensim/auto_examples/index.html
	BioVec [61]	Asgari *et al.*	2015	A new method designed for embedded representation of biological sequences	http://dx.doi.org/10.7910/DVN/JMFHTN
	dna2vec [62]	Ng *et al.*	2017	A method to gain DNA k-mer embedded representation	https://pnpnpn.github.io/dna2vec/
LSTM	LSTM [15]	Hochreiter *et al.*	1997	A special RNN network designed to solve the long sequences	
	seq2seq [63]	Sutskever *et al.*	2014	An encoder–decoder LSTM model for outputting sequences with uncertain length	https://github.com/google/seq2seq
	AWD-LSTM [64]	Merity *et al.*	2018	A weight-decreasing LSTM that uses DropConnect as a form of cyclic regularization for hidden weights	https://github.com/salesforce/awd-lstm-lm
	ELMo [16]	Peters *et al.*	2018	A general method for learning high-quality deep context-dependent representations from bi-LM	http://allennlp.org/elmo
	SeqVec [65]	Heinzinger *et al.*	2019	A new method to represent protein sequences as continuous vectors	https://github.com/mheinzinger/SeqVec
	UniRep [66]	Alley *et al.*	2019	An mLSTM-encoded representation method trained on 24 million protein sequences	https://github.com/churchlab/UniRep
Transformer	Transformer [17]	Vaswani *et al.*	2017	Solve the sequence to sequence problem and replace LSTM with a full attention structure	https://github.com/tensorflow/tensor2tensor/blob/master/tensor2tensor/models/transformer.py
	Bert [18]	Devlin *et al.*	2018	Bidirectional language model based on Transformer	https://github.com/google-research/bert
	Transformer-XL [67]	Dai *et al.*	2019	A variant of Transformer to solve the problem of long sequences	https://github.com/kimiyoung/transformer-xl
	XLNet [68]	Yang *et al.*	2019	A generalized autoregressive pretraining model	https://github.com/zihangdai/xlnet
	RoBERTa [69]	Liu *et al.*	2019	A Bert model with improved pretraining procedure	https://github.com/pytorch/fairseq
	ALBERT [70]	Lan *et al.*	2019	A lite Bert model with parameter sharing mechanism	https://github.com/google-research/ALBERT.

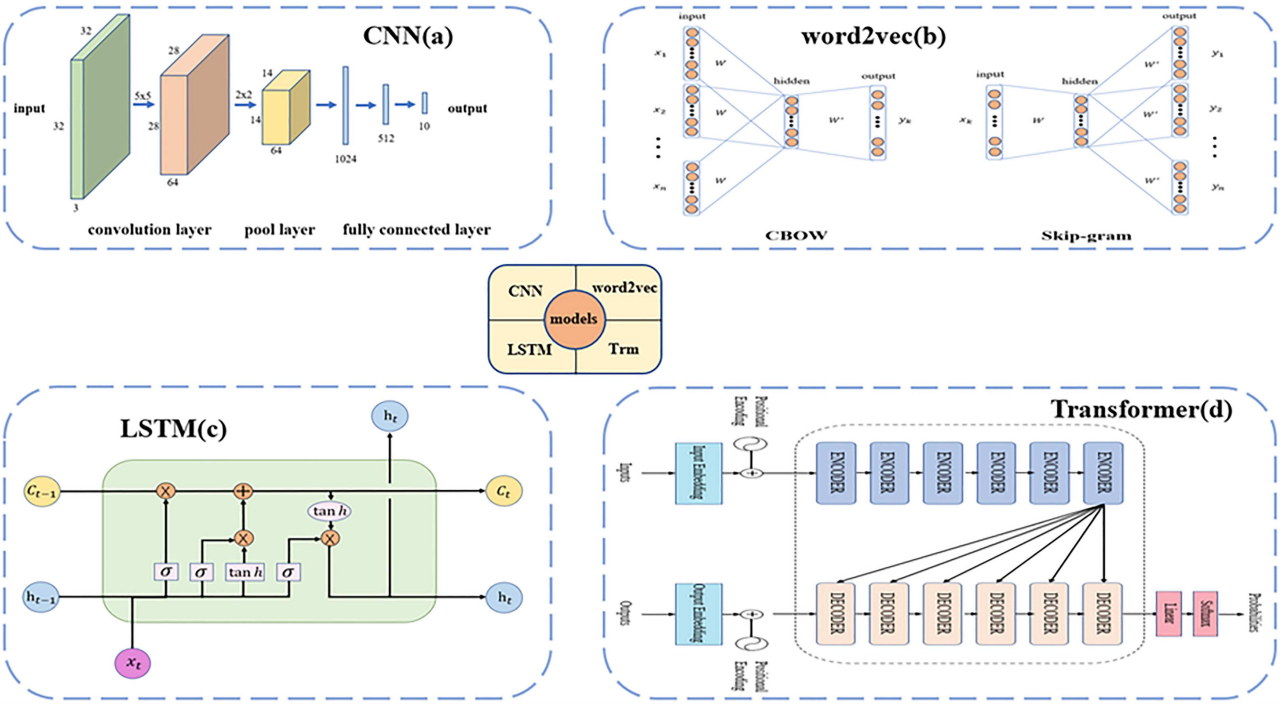

Figure 2. Four types of pretraining models for biological sequences. It introduces four popular pretraining models: CNN(a), LSTM(b), Word2vec(c) and Transformer(d).

field. Over the years, CNN has also become a frequently used neural network model in other deep learning fields.

CNN is composed of three primary neural layers, namely, the convolutional layer, pooling layer and fully connected layer [71]. The simple architecture of CNN is shown in Figure 2(A). In the convolutional layer, multiple convolution kernels perform convolution operations on the input matrix and intermediate feature maps and then transmit the result represented by the feature matrix to the next layer for operation. The pooling layer is used to reduce the feature map dimensions and the number of network parameters, thereby speeding up the network training. The fully connected layer that is always located at the end can convert the two-dimensional feature map into one-dimensional feature vector, which reflects the results of tasks.

Recently, CNN has been applied in obtaining information from biological sequences for corresponding tasks. Through pretraining on other datasets and transferring to target datasets, the CNN-based model not only utilizes data characteristics from different datasets but also achieves remarkable results on target tasks with less training time.

Word2vec

Mikolov *et al.* [14] proposed word2vec in 2013, a well-known unsupervised method to learn high-quality embedded vectors to represent words. By designing two context word prediction tasks, the word2vec model learns the low-dimensional embedding representation of each word, which reflects the context and sematic information of words among sequences.

Word2vec comprises two important models: Skip-gram and Continuous Bag of Words (CBOW). Figure 2(B) shows the architectures of Skip-gram and CBOW. The training object of the Skip-gram model is predicting context words based on the target word, in which the input is the target word, and the output is the context words. Different from Skip-gram, the CBOW model can predict the target word based on context words, which changes the input as the surrounding context and output as the target word. The embedded representation of words is the low-dimensional vectors that are mapped by the hidden layer, which is the by-product of Skip-gram and CBOW. In particular, the Skip-gram model is described as follows:

$$h = Wx_k \tag{1}$$

$$(y_1, y_2, \ldots, y_n) = W'h \tag{2}$$

where x_k is the input representing the target word; $(y_1, y_2, \ldots, y_n)$ is the output representing the context words; and h denotes the hidden representation with W and W′ representing different weights. The CBOW model is introduced as follows:

$$h = AVG\,(W\,(x_1 + x_2 + \cdots + x_k)) \tag{3}$$

$$y_k = W'h \tag{4}$$

where $x_1, x_2, \ldots, x_n$ is the input representing the context words; y_k is the output representing the target words; and h, W and W′ denote the hidden representation and weight. In addition, two training strategies in word2vec are proposed for reducing computational cost and speeding up training time, namely, Hierarchical Softmax and Negative Sampling.

For tasks based on biological sequences, pretraining word2vec-based models can capture syntax and semantic information among biological sequences. After pretraining on large unlabeled biological sequence datasets, word2vec-based models generate high-quality embedding vectors to represent biological sequences, which significantly improve performance after being used in downstream tasks.

Long short-term memory

LSTM [15] is an improved RNN model, which obtains not only information from single input but also contextual information from other input, having advantages in processing long sequences. The LSTM model can extract the semantic and grammatical information in the sequences with mapping sequences into low-dimensional vector space.

Figure 2(C) illustrates the internal structure of LSTM. Compared with RNN, LSTM establishes more delivery states (hidden state and cell state) to transport information, which addresses the gradient explosion and disappearance problem in training long sequences. Structurally, the input of the current unit and states, which are passed from previous units, jointly control the current output and states. The specific process can be described as follows: suppose x_t is the input of LSTM unit t, and h_{t-1} is the hidden state passed from previous units, the input information and three function gates can be obtained as follows:

$$z = \tan h\left(W' \text{contact}\left(x_t, h_{t-1}\right)\right) \tag{5}$$

$$z^i = \sigma\left(W^i \text{contact}\left(x_t, h_{t-1}\right)\right) \tag{6}$$

$$z^f = \sigma\left(W^f \text{contact}\left(x_t, h_{t-1}\right)\right) \tag{7}$$

$$z^o = \sigma\left(W^0 \text{contact}\left(x_t, h_{t-1}\right)\right) \tag{8}$$

where W', W^i, W^f and W^0 represent different weights; σ denotes sigmoid function and z denotes input information. Three function gates are identified: input gate (z^i) controls the information needed to be retained; forget gate (z^f) controls the information that should be forgotten and output gate (z^o) controls the information that will be outputted. Next, suppose c_{t-1} is the cell state passed from previous units, the current output and states are obtained as follows:

$$c_t = z^f * c_{t-1} + z^i * z \tag{9}$$

$$h_t = z^o * \tan h\left(c_t\right) \tag{10}$$

$$y_t = \sigma\left(W h_t\right) \tag{11}$$

where c_t and h_t denote the cell state and hidden state of LSTM unit t, respectively, which will be passed to the next unit. y_t represents the current output, which can be used for tasks. $*$ denotes the matrix Hadamard product.

A widely used variant of LSTM is bidirectional long short-term memory (Bi-LSTM), which obtains semantic information in two directions from long sequences. Bi-LSTM performs better in language modeling while extracting comprehensive information from sequences on the basis of LSTM.

The LSTM model gains success in processing long sequences. However, the training strategy and model structure limit the embedding representation of words generated by LSTM, which cannot represent polysemous words in different context. To overcome abovementioned shortcomings, Peters *et al.* [16] proposed an embedding model based on deep bidirectional language model (Bi-LM), named ELMO, to generate embedding vectors for representing corresponding words according to contextual information. The architecture of the language model in ELMO is presented in Figure 3(A). After embedding, the input sequences are encoded by the Bi-LSTM layers. The output of each LSTM layer is used as the context-dependent word vectors of each word. The final word vectors are generated by linearly combining word vectors of different layers, which represent specific meaning of words in specific context. Different from taking words as input in previous models, the input of ELMO is a sentence. Therefore, ELMO dynamically generates the word vectors on the basis of the context in sentences instead of generating fixed word vectors for words in different sequences.

The LSTM-based model has evident advantages in dealing with long sequences. Thus, it is always used for embedding long biological sequences into low-dimensional vectors as pretraining models.

Transformer

Inspired by remarkable performance of attention mechanism [72] in many fields, the Transformer model is proposed on the basis of attention mechanism in the NLP field. Instead of using traditional CNN and RNN models, Vaswani *et al.* [17] creatively proposed Transformer, which is a full attention mechanism network. The architecture of the attention layer leads to the parallelism and long-term dependence of the Transformer model.

Figure 2(D) shows the initial structure of Transformer. The Transformer model has a typical encoder–decoder architecture, which is composed of multihead self-attention and feedforward neural network (FNN). In Transformer, sequences are first represented by low-dimensional vectors after word embedding and positional embedding. In encoder, embedding vectors are encoded through N-EncoderLayer, which has a multihead self-attention layer and FNN layer in each layer. In decoder, vectors are decoded through N-DecoderLayer that adds a masked multihead self-attention layer compared with EncoderLayer.

In particular, the multihead self-attention in Transformer can be described using the following equations. Suppose X is the input, the three vectors for calculating attention are as follows:

$$Q = W^Q X \tag{12}$$

$$K = W^K X \tag{13}$$

$$V = W^V X \tag{14}$$

where Q, K and V represent the query vector, key vector and value vector, respectively; W^Q, W^K and W^V are the different weights to calculate different vectors. In addition, suppose Att_i is the i-head attention, output A can be obtained as follows:

$$Att_i = Attention\left(Q_i, K_i, V_i\right) = \sigma\left(\frac{Q_i K_i^T}{\sqrt{d_k}}\right) V_i \tag{15}$$

$$A = W contact\left(Att_1, Att_2, \ldots, Att_i\right) \tag{16}$$

where d_k represents the dimension number in Q; σ denotes the sigmoid function and W denotes the weight.

In recent years, Devlin *et al.* [18] proposed a breakthrough multitask pretraining model on the basis of Transformer, Bidirectional Encoder Representations from Transformers (Bert), to learn high-quality vector representation of words. The structure of Bert is presented in Figure 3(B). The Bert model is made up of Bidirectional Transformer (Bi-Transformer) blocks; thus, representation generated by Bert is based on context information of all layers. Compared with previous embedding methods, Bi-Transformer in Bert can capture bidirectional information

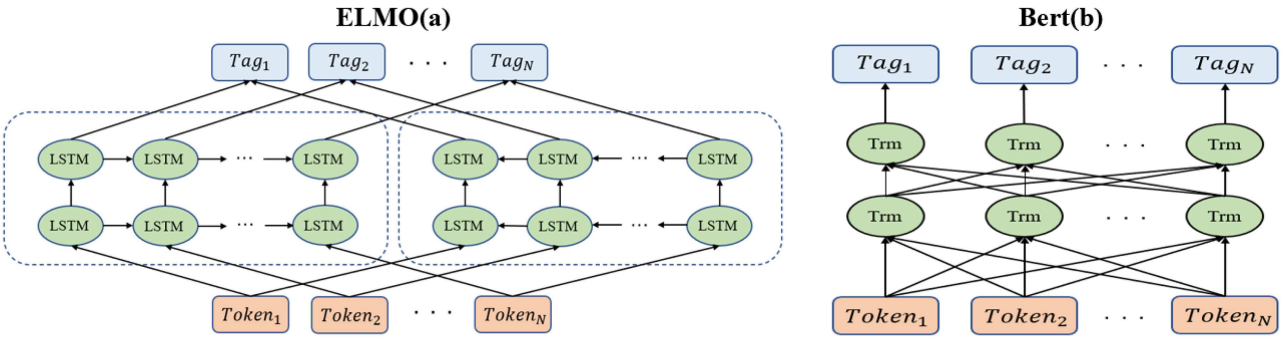

Figure 3. ELMO and Bert architecture. It introduces two popular pretraining models: ELMO(a) and Bert(b).

among the sequences more thoroughly. The pretrained task in Bert is Masked Language Model (MLM) and Next Sentence Prediction (NSP), which are used for obtaining embedding representation of words in self-supervised learning. Consequently, multitasks enable Bert to learn sequence information in different views. The Bert model pretrained on large corpus can provide other methods with outstanding embedding representation of words, which improve model performance and reduce training time. In addition, pretrained Bert can be applied in various tasks after fine-tuning according to special tasks, thereby obtaining excellent results.

The full attention mechanism structure enables the Transformer-based models to generate embedding vectors according to the importance of context information, which represent the members in biological sequences. After training on magnanimous unlabeled biological sequences, the pretraining Transformer-based model can provide embedding representation with rich features of biological sequences, which are beneficial to downstream tasks.

Application of pretraining model

In general, most of the pretraining models for biological sequence data are used for sequence-embedded representation because of the difficulty in obtaining labeled data. Given the sequence $w_1, w_2, \cdots, w_K$, where each token (w_k) represents a word, the embedding process can be described as follows:

$$[V_1, V_2, \cdots, V_K] = f_{emb}(w_1, w_2, \cdots, w_K) \tag{17}$$

where f_{emb} represents the pretrained embedding models and V_K is an embedded low-dimensional vector for representing w_K. Embedded representation can extract features of biological sequences and express such features in another form of vectors. In addition, the feature matrix of biological sequences that is created by embedded representation improves the generalization ability of the models, thereby speeding up the training process and achieving remarkable final output.

Table 3 lists some recent studies that used pretraining models for embedded representation to improve the performance of downstream tasks. In this section, we review some proposed pretraining models for biological sequence data and their application on downstream tasks. Concurrently, we summarize these methods on the basis of the type of pretraining models: CNN, word2Vec, LSTM and Transformer.

Methods based on CNN

Previously, CNN was often used in CV to deal with image information. Recently, some studies [22, 73] have used CNN to pretrain biological sequence in transfer learning. Zhuang *et al.* [73] proposed a simple CNN model on the basis of DNA sequences to predict EPI. The simple model contains two input channels for enhancer and promoter sequences, followed by a convolution layer to encode sequences and a max-pool layer in each sequence. After matrix operations, a fully connected layer connects two results and finally outputs the EPI probability after dropout and sigmoid activation. The model is first pretrained with DNA sequences from other cell lines, where feature information was extracted from other sequences, and then trained with DNA sequences from target cell lines for CPI prediction. Playe *et al.* [22] designed a chemogenomic neural (CN) network, a deep neural network model that takes the embedded representation of the protein sequences and molecular graphs as input to predict DTI. The embedded representation of protein sequences is obtained by CNN encoder pretrained on DBEColi dataset from DrugBank database, which performs better than Bi-LSTM in test. As shown in the results, the CN model performs well in DTI prediction on large datasets.

Methods based on Word2vec

In NLP field, some embedded methods [14, 60] are developed on the basis of word2vec, which can be applied to embedded biological sequences. Concurrently, many methods [61, 62] particularly designed for biological sequences are proposed in recent years. Some studies [25, 54, 58, 75, 81] have employed word2vec-based methods for embedded representation of biological sequences, which are pretrained over large corpus and performed well on downstream tasks.

Benefit by great performance of word2vec in embedded representation, some methods [54, 58] select word2vec as a pretrained model for biological sequences. Chen *et al.* [54] proposed a novel method, namely, TransformerCPI for CPI prediction, which is fed with protein sequences and compound sequences. The model converts the protein sequences into a real-value 100-dimensional vector by word2vec pretrained on UniProt database, to extract features and represent protein sequences. Chaabane *et al.* [58] used the Skip-gram model (word2vec) to generate an embedded matrix of RNA sequences for circular RNA classification. During embedding, word2vec is first pretrained with RNA sequences and fine-tuned during subsequent training.

Although word2vec performed well in representing words, it still cannot avoid the order and semantics of words in

Table 3. List of recent works employing pretraining models

Author	Year	Dataset	Method	Pretraining model	Application	Code
Based on CNN						
Zhuang *et al.* [73]	2019	SPEID	CNN	CNN	EPI	https://github.com/zzUMN/Combine-CNN-Enhancer-andPromoters
Playe *et al.* [22]	2020	DrugBank, DBHuman, DBEColi	CN		DTI	https://github.com/bplaye/NNk_DT
Based on word2vec						
Chen *et al.* [54]	2020	UniProt, DrugBank, BindingDB, GPCR, Kinase	TransformerCPI	word2vec	CPI	https://github.com/lifanchen-simm/transformerCPI
Chaabane *et al.* [58]	2020	circRNADb, GENCODE	circDeep		Circular RNA classification	https://github.com/UofLBioinformatics/circDeep
Yang *et al.* [74]	2018	UniProt	GP	doc2vec	Four tasks	https://github.com/fhalab/embeddings_reproduction/.
Deznabi *et al.* [75]	2020	PhosphoSitePlus	DeepKinZero	ProtVec	Kinase–phosphosite associations	thttps://github.com/Tastanlab/DeepKinZero.
Hong *et al.* [25]	2020	TargetFinder (EPI)	EPIVAN	dna2vec	EPI	https://github.com/hzy95/EPIVAN.
Based on LSTM						
Bepler *et al.* [53]	2019	SCOPe, Pfam	SSA	LSTM	Protein structural similarity	https://github.com/tbepler/protein-sequence-embedding-iclr2019
Karimi *et al.* [76]	2019	Pfam, BindingDB, UniRef, STITCH	DeepAffinity	seq2seq	CPI	https://github.com/Shen-Lab/DeepAffinity
Strodthoff *et al.* [26]	2020	SWISS-PROT, DEEPre, ECPre, EC40, EC50	UDSMProt	AWD-LSTM	Protein classification	https://github.com/nstrodt/UDSMProt.
Amelia *et al.* [55]	2020	PDB, SWISS-PROT, CAFA3	Several methods	SeqVec	Protein function prediction	https://github.com/stamakro/GCN-for-Structure-and-Function.
Based on Transformer						
Rives *et al.* [77]	2019	UniProt, UniParc, CB5135926_filtered	Bert	Bert	Variant activity prediction	
Vig *et al.* [78]	2020	ProteinNet, TAPE	Bert	Bert	Interpretability	https://github.com/salesforce/provis
Nambiar *et al.* [79]	2020	SWISS-PROT, HIPPIE	PRoBERTa	RoBERTa	PFC PPI	
Elnaggar *et al.* [80]	2020	UniRef100, BFD	ProtTrans	Bert, Transformer-XL XLNet, ALBERT	HPC in protein LMs	https://github.com/agemagician/ProtTrans

sentences. Mikolov *et al.* [60] proposed doc2vec, an improved unsupervised embedding method for sequences with variable length to address the challenges of word2vec. Yang *et al.* [74] used pretrained doc2vec to embed protein sequences into 64-dimensional space, which is first trained on unlabeled protein sequences from UniProt. The experimental results show that the embedded representation of protein sequences performs well for predicting protein property.

Word2vec makes great contributions to the embedded representation of biological sequences as a pretrained model. Consequently, some word2vec-based methods [61, 62] designed particularly for embedding biological sequences have been proposed in recent years. Asgari *et al.* [61] proposed Bio2vec, which is an embedded method particularly for biological sequences. Bio2vec can generate continuously distributed representation of biological sequences using a pretrained Skip-gram model, which is divided into Protvec (for protein) and Genvec (for gene) according to different training objects. Deznabi *et al.* [75] applied Protvec that trained protein sequences from SWISS-PROT to embed phosphosite into 1300-dimensional vector, which provides embedded representations of phosphosite for kinase–phosphosite association prediction. Ng *et al.* [62] obtained DNA k-mer-embedded representation by word2vec trained with human DNA sequences, which is called dna2vec. Hong *et al.* [25] proposed a novel method called EPIVAN to predict EPI with only genomic sequences. For representing enhancer and promoter, EPIVAN used the DNA vectors generated by pretrained dna2vec to encode DNA sequences.

Methods based on LSTM

Word2vec is an effective model in generating vectors representing words, but it only provides limited help for embedding long sequences. Another effective neural network model for processing sequence information is LSTM. Given its advantages, many LSTM-based embedding methods [15, 16, 63–66] are proposed and used for representing biological sequences as pretrained models in some methods [26, 53, 55, 76].

Bepler *et al.* [53] proposed SSA frame, which predicts protein structural similarity from amino acid sequences. Their proposed method maps protein sequences to embed vectors by Bi-LSTM models pretrained on protein sequences in Pfam. Sutskever *et al.* [63] proposed seq2seq, an encoder–decoder LSTM model that used attention mechanism to output sequences with uncertain length. In seq2seq, sequences are first embedded into vectors with fixed length by an LSTM model and then converted into ideal sequences by another LSTM model. Karimi *et al.* [76] designed a semisupervised deep learning model, which predicts compound–protein affinity with unlabeled and labeled data, namely, DeepAffinity. For leveraging rich information from compound and protein, they used a seq2seq model pretrained on unlabeled sequences to embed labeled sequences.

Merity *et al.* [64] regularized and optimized an LSTM model by using DropConnect on hidden-to-hidden weights and presented ASGD Weight-Dropped LSTM (AWD-LSTM), which performs better in embedding sequences than LSTM. Strodthoff *et al.* [26] proposed UDSMProt pretrained on unlabeled protein sequences, which classified proteins from sequences. UDSMProt used AWD-LSTM as a pretrained language model to understand the good embedding of protein sequences, which also performs well when transferring to other three tasks. Alley *et al.* [66] build unified representation (UniRep) from massive unlabeled amino acid sequences by a multi-LSTM model. As shown in their results, the statistical representation of protein sequences contains rich semantical information, which can be broadly applied to other methods as pretrained embedded representation.

Due to the outstanding performance of EMLO [16] in sequences processing, the pretraining ELMO model also performed well in embedding biological sequences. Heinzinger *et al.* [65] proposed Seq2Vec, a novel embedding model based on ELMO pretrained on UniRef50, to represent protein sequences by continuous vectors. This method can also be used as pretrained model for embedding biological sequences in other methods. Amelia *et al.* [55] used embedded representation of protein sequences with additional protein contact map to predict protein function, in which high-quality embedded vectors are generated by LSTM-based SeqVec model pretrained on PDB database.

Methods based on transformer

Although the LSTM-based methods have achieved good results in embedding biological sequence as pretrained models, they are still limited in training long sequences. In recent years, many models [17, 18, 67–70] based on Transformer have been proven to perform well in embedded representation of biological sequence as pretrained models, particularly after Bert proposed.

Rives *et al.* [77] trained Bert on 86 billion amino acids from 250 million sequences. In their experiment, raw protein sequences are mapped into representation space reflecting biological structure at many levels. The representations offer various information and feature of proteins, which can be extracted and used by other methods according to downstream tasks. Vig *et al.* [78] focused on interpretability of embedded representation learned by Transformer architectures (Bert). Their results show that attention mechanism can capture the folding structure of proteins and target binding sites and focus on biophysical properties.

Recent advances in Transformer-based methods made it valid to embed biological sequence as high-quality vector representation. Dai *et al.* [67] improved the Transformer model in accepting variable length context in language modeling, proposing Transformer-XL. With novel segment-level recurrence mechanism and positional encoding scheme, Transformer-XL performed well in capturing long-term dependency and processing context fragmentation. Yang *et al.* [68] overcame the limitations of MLM in Bert by replacing the autoregressive model with autoencoding model simultaneously and designing a novel generalized autoregressive pretraining method, namely, XLNet. Given the two-stream self-attention mechanism and integrating advantages of Transformer-XL, XLNet outperformed Bert in various NLP tasks. Liu *et al.* [69] proposed an improved Bert model-RoBERTa, which adjusted training details of Bert and achieved dynamic masking mechanism. Lan *et al.* [70] proposed ALBERT, a lite Bert with cross-layer parameter sharing and factorized embedding parameterization, thereby speeding up the training phase. For embedding protein sequences, Nambiar *et al.* [79] designed PRoBERTa, a neural network architecture based on RoBERTa. After pretraining on SWISS-PROT database and fine-tuning, their method performs well in protein family classification and protein interaction prediction. Elnaggar *et al.* [80] combined Transformer-based models with high-performance computing to map protein sequences as embedding vectors, namely, ProtTrans. In their experiment, researchers trained four models (Transformer-XL, XLNet, BERT and Albert) on 93 billion amino acids from 2.1 billion protein sequences. As shown in the results, Transformer-based models pretrained on a large amount of labeled data extracted the

biophysical information of proteins and achieved good results in various downstream tasks.

Scheme and benchmark

In this section, we introduce a novel pretraining scheme for protein sequences and a multitask benchmark for protein embedding methods. Hopefully, pretraining scheme and protein embedding benchmark can provide novices with a way, in which researchers can quickly design methods for protein sequences and evaluate the performance of protein embedding models.

Min *et al.* [82] proposed a novel pretrained scheme for protein sequences, namely, PLUS, in which embedding models are pretrained with protein-specific pretraining task to obtain information in unlabeled protein sequences. It reflects the difference between protein sequences and natural language sequences. PLUS contains two pretraining tasks (MLM and Same Family Prediction), which obtain sequence information and protein-specific information. In particular, protein sequences are first masked 15% at random and then transformed to embedding vectors by representation models with two pretraining tasks. The transferability of PLUS enables various embedding models for biological sequences to be pretrained and fine-tune on downstream tasks, such as Bi-LSTM and Transformer. With the help of PLUS, researchers can focus on designing their pretraining embedding methods regardless of auxiliary tasks and training procedures.

Rao *et al.* [83] collected and designed a multitask standard benchmark, Tasks Assessing Protein Embeddings (TAPE), to make up the gaps in standardized evaluation indicators and datasets for protein semisupervised learning. TAPE reflects multiple functions of the protein sequences and evaluates the pretraining models for protein sequence from multiple aspects. TAPE consists of five semisupervised learning tasks relevant to proteins (secondary structure prediction, contact prediction, remote homology detection, fluorescence landscape prediction and stability landscape prediction), which cover three areas of protein biology: structure prediction, evolutionary understanding and protein engineering. For datasets, TAPE provides an unlabeled protein sequence dataset constructed from Pfam database and many supervised preprocessed datasets for downstream tasks. In addition, the experimental results indicate that self-supervised pretraining models for biological sequences can significantly improve the performance on downstream tasks. These tasks and protein datasets in TAPE can be used for evaluating the performance of protein pretraining methods in multiaspects, which offers a fair and open benchmark for measuring the effectiveness of pretraining models.

Challenges and future directions

Pretraining models that are not related to specific tasks are obtained by self-supervised learning on large-scale data. With the emergence of pretraining models and their successful applications in fields such as NLP [84], CV depicts the power of pretraining technology. Pretraining models are applicable to almost all tasks that rely on large amounts of data, particularly unlabeled data.

Regarding biological sequence data, pretraining models can generate embedded representation that reflects the semantic information of biological sequence after training on large corpus, which speeds up training process, improves performance on downstream tasks and supports new tasks with fine-tune. However, despite the success of pretraining models for biological sequence in recent years, such models still face challenges and need further development in this field. Herein, we summarize challenges and potential future directions in pretraining models for biological sequence.

Data

Pretraining models for biological sequence require large amount of data to learn sufficient features in sequences, but reliable biological sequence data are not enough. Although high-throughput sequencing technology has brought various new sequence data [85, 86], it still cannot meet the developing pretraining models, particularly DNA and RNA sequences. In addition, the expensive cost of obtaining labeled data and lack of negative samples hinder the transfer of pretraining models for biological sequence in many tasks. On the one hand, the breakthrough of sequencing technology in the biological field may alleviate these problems. On the other hand, the multimodal pretraining model is a good solution. Compared with previous methods, multimodal pretraining models [87] fuse abstract feature from different types of data such as sequence, image and graph, which learn good feature representation from multimodal data while making up for the lack of sequence data. Therefore, multimodal pretraining models can make full use of more data and perform better on downstream tasks. We hypothesize that more multimodal pretraining methods are proposed for biological sequence in the future. Meta-learning is a novel learning strategy, which helps the model learn information quickly with a small number of samples. The main idea in meta-learning is making models to learn based on previous experience and knowledge. Combining meta-learning with pretraining models for biological sequences would be a potential future direction.

Pretraining tasks

Based on specific pretraining tasks, pretraining models can learn abundant feature representation on large datasets. A variety of NLP pretraining models [18, 69, 88] used LM or MLM as a pretraining task. Some NLP pretraining models [68, 70, 89–91] are also transferred to biological sequences. However, tasks in NLP reflect partial characteristics of biological sequences [82]. Simultaneously, a single task has limited effects on pretraining models. For the former, more tasks reflecting specific information in biological sequences are proposed. In addition, a new future direction is contrastive learning [92], learning the semantic information from similarity and difference of sequence pairs. Recently, many contrastive learning pretraining models [93–96] are proposed, which also have a good prospect in biological sequence data. For the latter, one improvement that can be made is using multitask pretraining models [97] instead of single pretraining models. The models in multitask learning are trained through a set of related tasks, which improve the generalization ability of the models. By taking the relation and difference between different tasks into consideration, multitask pretraining models perform better than single-task models. Multitask pretraining models become increasingly popular, when single-task models have been unable to meet pretraining requirements gradually.

Pretraining models

Existing pretraining models for biological sequences are derived from NLP domain. After the Transformer architecture came out, Transformer-based pretraining models in NLP reached a new

height and faced some new challenges. The most prominent problem is too many parameters in Transformer models, which requires expensive computing resources and long time to fit the pretraining models. An interesting future direction is to propose new architectures to overcome the disadvantages of Transformer models. In recent years, some valid training strategies are designed for compressing pretraining models, such as model trimming [98], parameter sharing [70], etc. At present, Knowledge Distillation (KD) [99] is a novel research direction in reducing pretraining models. Two models are identified in KD: student (small) model and teacher (large) model, in which the student model is obtained through transferring knowledge from a trained teacher model. KD has an ability to transfer a large model to a small model that retains the performance close to the large model. In addition to model compression, new pretraining model architecture and interpretability of pretraining models are popular future directions. Knowledge graphs also have been applied in the prediction of drug repurposing [100], disease genes [101, 102], circular RNAs [103] miRNAs [104, 105]. It would be interesting to study using knowledge graphs for the pretraining of biological sequences.

Conclusion

In this paper, we provided a review that aimed to introduce recent development and studies on pretraining models for biological sequence data. In general, we included in this review the background of pretraining models for biological sequence, a brief introduction to biological sequence data and corresponding datasets, popular pretraining models in previous works, application of pretraining models for biological sequences, a novel scheme and benchmark on pretraining models for protein sequences, and challenges and future directions.

In particular, we first illustrated the deep learning background of pretraining models for biological sequences, containing the role of biological sequence data and introduction of pretraining models. Then, we made a brief introduction of biological sequences and several notable biological sequence databases. We also collected and presented some datasets with brief description and available link. Next, we proposed a classification scheme for pretraining models and reviewed the literature on the basis of the categories of pretraining models. Moreover, we separately introduced the corresponding structure, features and mechanisms of pretraining models. We further detailed some methods for downstream tasks with proposed pretraining models to explain the application of pretraining models, such as DTI, EPI, PPI [106, 107], protein function prediction [108–111] and RNA classification [112]. In addition, we provided a novel pretraining scheme and benchmark for protein sequences, which helped researchers to design and verify their methods. Finally, we discussed existing challenges and popular future research directions of pretraining models for biological sequence to guide future works. We hope that this survey can provide readers with a general understanding toward this field, some resources for conducting research and feasible ideas for future research in pretraining models for biological sequence data.

Key Points

- Summarize popular pretraining models for biological sequences based on four categories: CNN, word2vec, LSTM and transformer.
- Present some applications of pretraining models for biological sequences on downstream tasks to explain the role of pretraining models.
- Discuss the challenges and future research directions in pretraining models for biological sequences.

Funding

The work was supported in part by National Natural Science Foundation of China (61872309, 61972138, 62002111), in part by the Fundamental Research Funds for the Central Universities (531118010355), in part by China Postdoctoral Science Foundation (2019 M662770), in part by Hunan Provincial Natural Science Foundation of China (2020JJ4215), in part by Key Research and Development Program of Changsha (kq2004016) and in part by Changsha Municipal Natural Science Foundation (kq2014058).

Conflict of interest

The authors confirm that this article content has no conflict of interest.

References

1. Jurtz VI, Johansen AR, Nielsen M, *et al*. An introduction to deep learning on biological sequence data: examples and solutions. *Bioinformatics* 2017;**33**(22):3685–90.
2. Shen Z, Zhang Q, Han K, *et al*. A deep learning model for RNA-protein binding preference prediction based on hierarchical LSTM and attention network. *IEEE/ACM Trans Comput Biol Bioinform* 2020;1–1.
3. Zhang T, Wu Q, Zhang Z. Probable pangolin origin of SARS-CoV-2 associated with the COVID-19 outbreak. *Curr Biol* 2020;**30**(7):1346–1351.e2.
4. Zhou Y, Wang F, Tang J, *et al*. Artificial intelligence in COVID-19 drug repurposing. *The Lancet Digital Health* 2020;**2**(12):e667–76.
5. Soranzo N, Hou Y, Shen J, *et al*. A network medicine approach to investigation and population-based validation of disease manifestations and drug repurposing for COVID-19. *PLoS Biol* 2020;**18**(11):e3000970.
6. Wu J, Liu J, Li S, *et al*. Detection and analysis of nucleic acid in various biological samples of COVID-19 patients. *Travel Med Infect Dis* 2020;**37**:101673.
7. Zhang Z-Y, Zhang Z-Y, Yang Y-H, *et al*. Design powerful predictor for mRNA subcellular location prediction in Homo sapiens. *Brief Bioinform* 2020;**22**(1):1–10.
8. Dao F-Y, Lv H, Zulfiqar H, *et al*. A computational platform to identify origins of replication sites in eukaryotes. *Brief Bioinform* 2020;**22**(2).
9. Liu X, Zhang F, Hou Z, *et al*. Self-supervised learning: generative or contrastive arXiv preprint arXiv:2006.08218. 2020.
10. Zou Q, Lin G, Jiang X, *et al*. Sequence clustering in bioinformatics: an empirical study. *Brief Bioinform* 2020;**21**(1):1–10.
11. Lecun Y, Bottou L, Bengio Y, *et al*. Gradient-based learning applied to document recognition. *Proc IEEE* 1998;**86**(11):2278–324.
12. Lv H, Dao FY, Guan ZX, *et al*. Deep-Kcr: accurate detection of lysine crotonylation sites using deep learning method. *Brief Bioinform* 2020. doi: 10.1093/bib/bbaa255.

13. Dao FY, Lv H, Zhang D, *et al*. DeepYY1: a deep learning approach to identify YY1-mediated chromatin loops. *Brief Bioinform* 2020. doi: 10.1093/bib/bbaa356.
14. Mikolov T, Sutskever I, Chen K, *et al*. Distributed representations of words and phrases and their compositionality. In: *Advances in Neural Information Processing Systems*, 2013.
15. Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput* 1997;**9**(8):1735–80.
16. Peters ME, Neumann M, Iyyer M, *et al*. Deep contextualized word representations arXiv preprint arXiv:1802.05365. 2018.
17. Vaswani A, Shazeer N, Parmar N, *et al*. Attention is all you need. In: *Advances in Neural Information Processing Systems*, 2017.
18. Devlin J, Chang M-W, Lee K, *et al*. Bert: pre-training of deep bidirectional transformers for language understanding arXiv preprint arXiv:1810.04805. 2018.
19. Otter DW, Medina JR, Kalita JK. A survey of the usages of deep learning for natural language processing. *IEEE Transactions on Neural Networks and Learning Systems* 2020;1–21.
20. le NQK, Yapp EKY, Ho QT, *et al*. iEnhancer-5Step: identifying enhancers using hidden information of DNA sequences via Chou's 5-step rule and word embedding. *Anal Biochem* 2019;**571**:53–61.
21. Lin X, Quan Z, Wang ZJ, *et al*. A novel molecular representation with BiGRU neural networks for learning atom. *Brief Bioinform* 2020;**21**(6):2099–111.
22. Playe B, Stoven V. Evaluation of deep and shallow learning methods in chemogenomics for the prediction of drugs specificity. *J Chem* 2020;**12**(1):11.
23. Zeng X, Zhu S, Hou Y, *et al*. Network-based prediction of drug–target interactions using an arbitrary-order proximity embedded deep forest. *Bioinformatics* 2020;**36**(9): 2805–12.
24. Zeng X, Zhu S, Lu W, *et al*. Target identification among known drugs by deep learning from heterogeneous networks. *Chem Sci* 2020;**11**(7):1775–97.
25. Hong Z, Zeng X, Wei L, *et al*. Identifying enhancer–promoter interactions with neural network based on pretrained DNA vectors and attention mechanism. *Bioinformatics* 2020;**36**(4):1037–43.
26. Strodthoff N, Wagner P, Wenzel M, *et al*. UDSMProt: universal deep sequence models for protein classification. *Bioinformatics* 2020;**36**(8):2401–9.
27. Fu X, Cai L, Zeng X, *et al*. StackCPPred: a stacking and pairwise energy content-based prediction of cell-penetrating peptides and their uptake efficiency. *Bioinformatics* 2020;**36**(10):3028–34.
28. Yang W, Zhu XJ, Huang J, *et al*. A brief survey of machine learning methods in protein sub-Golgi localization. *Current Bioinformatics* 2019;**14**(3):234–40.
29. Tan JX, Li SH, Zhang ZM, *et al*. Identification of hormone binding proteins based on machine learning methods. *Math Biosci Eng* 2019;**16**(4):2466–80.
30. DeLano, W.L., *The PyMOL Molecular Graphics System*. http://www.pymol.org, 2002.
31. Zhu X-J, Feng CQ, Lai HY, *et al*. Predicting protein structural classes for low-similarity sequences by evaluating different features. *Knowledge-Based Systems* 2019;**163**:787–93.
32. Berman HM, Westbrook J, Feng Z, *et al*. The protein data bank. *Nucleic Acids Res* 2000;**28**(1):235–42.
33. Boeckmann B, Bairoch A, Apweiler R, *et al*. The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Res* 2003;**31**(1):365–70.
34. Lo Conte L, Ailey B, Hubbard TJ, *et al*. SCOP: a structural classification of proteins database. *Nucleic Acids Res* 2000;**28**(1):257–9.
35. Finn RD, Bateman A, Clements J, *et al*. Pfam: the protein families database. *Nucleic Acids Res* 2014;**42**(D1):D222–30.
36. Consortium U. UniProt: a worldwide hub of protein knowledge. *Nucleic Acids Res* 2019;**47**(D1):D506–15.
37. Suzek BE, Wang Y, Huang H, *et al*. UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics* 2015;**31**(6):926–32.
38. Hatos A, Hajdu-Soltész B, Monzon AM. *et al*., DisProt: intrinsic protein disorder annotation in 2020. *Nucleic Acids Res* 2020;**48**(D1):D269–76.
39. Fox NK, Brenner SE, Chandonia J-M. SCOPe: structural classification of proteins—extended, integrating SCOP and ASTRAL data and classification of new structures. *Nucleic Acids Res* 2014;**42**(D1):D304–9.
40. Steinegger M, Söding J. Clustering huge protein sequence sets in linear time. *Nat Commun* 2018;**9**(1):1–8.
41. AlQuraishi M. ProteinNet: a standardized data set for machine learning of protein structure. *BMC Bioinformatics* 2019;**20**(1):1–10.
42. Frankish A, Diekhans M, Ferreira AM, *et al*. GENCODE reference annotation for the human and mouse genomes. *Nucleic Acids Res* 2019;**47**(D1):D766–73.
43. Zhao YW, Lai HY, Tang H, *et al*. Prediction of phosphothreonine sites in human proteins by fusing different features. *Sci Rep* 2016;**6**(1):34817.
44. Whalen S, Truty RM, Pollard KS. Enhancer–promoter interactions are encoded by complex genomic signatures on looping chromatin. *Nat Genet* 2016;**48**(5):488–96.
45. Wishart DS, Knox C, Guo AC, *et al*. DrugBank: a comprehensive resource for in silico drug discovery and exploration. *Nucleic Acids Res* 2006;**34**(suppl_1):D668–72.
46. Liu T, Lin Y, Wen X, *et al*. BindingDB: a web-accessible database of experimentally determined protein–ligand binding affinities. *Nucleic Acids Res* 2007;**35**(suppl_1): D198–201.
47. Kuhn M, von Mering C, Campillos M, *et al*. STITCH: interaction networks of chemicals and proteins. *Nucleic Acids Res* 2007;**36**(suppl_1):D684–8.
48. Gaulton A, Bellis LJ, Bento AP, *et al*. ChEMBL: a large-scale bioactivity database for drug discovery. *Nucleic Acids Res* 2012;**40**(D1):D1100–7.
49. Alanis-Lobato G, Andrade-Navarro MA, Schaefer MH. HIPPIE v2. 0: enhancing meaningfulness and reliability of protein–protein interaction networks. *Nucleic Acids Res* 2016;gkw985.
50. Tang J, Szwajda A, Shakyawar S, *et al*. Making sense of large-scale kinase inhibitor bioactivity data sets: a comparative and integrative analysis. *J Chem Inf Model* 2014;**54**(3):735–43.
51. Chan WK, Zhang H, Yang J, *et al*. GLASS: a comprehensive database for experimentally validated GPCR-ligand associations. *Bioinformatics* 2015;**31**(18):3035–42.
52. Gregory S, Barlow KF, McLay K, *et al*. The DNA sequence and biological annotation of human chromosome 1. *Nature* 2006;**441**(7091):315–21.
53. Bepler, T. and B. Berger, Learning protein sequence embeddings using information from structure. arXiv preprint arXiv:1902.08661, 2019.
54. Chen L, Tan X, Wang D, *et al*. TransformerCPI: improving compound–protein interaction prediction by sequence-based deep learning with self-attention mechanism and

label reversal experiments. *Bioinformatics* 2020;**36**(16): 4406–14.
55. Villegas-Morcillo A, Makrodimitris S, van Ham RCHJ, *et al*. Unsupervised protein embeddings outperform hand-crafted sequence and structure features at predicting molecular function. *bioRxiv* 2020.
56. Watson JD, Crick FH. The structure of DNA. In: *Cold Spring Harbor Symposia on Quantitative Biology*. Cold Spring Harbor Laboratory Press, 1953.
57. Khalifa NEM, Taha MHN, Ezzat Ali D, *et al*. Artificial intelligence technique for gene expression by tumor RNA-Seq data: a novel optimized deep learning approach. *IEEE Access* 2020;**8**:22874–83.
58. Chaabane M, Williams RM, Stephens AT, *et al*. circDeep: deep learning approach for circular RNA classification from other long non-coding RNA. *Bioinformatics* 2020;**36**(1):73–80.
59. Dong L, Yang N, Wang W, *et al*. Unified language model pre-training for natural language understanding and generation. In: *Advances in Neural Information Processing Systems*, 2019.
60. Le, Q. and T. Mikolov. Distributed representations of sentences and documents. In: *International Conference on Machine Learning*. 2014.
61. Asgari E, Mofrad MR. Continuous distributed representation of biological sequences for deep proteomics and genomics. *PLoS One* 2015;**10**(11):e0141287.
62. Ng P. dna2vec: consistent vector representations of variable-length k-mers arXiv preprint arXiv:1701.06279. 2017.
63. Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks. In: *Advances in Neural Information Processing Systems*, 2014.
64. Merity S, Keskar NS, Socher R. Regularizing and optimizing LSTM language models arXiv preprint arXiv:1708.02182. 2017.
65. Heinzinger M, Elnaggar A, Wang Y, *et al*. Modeling aspects of the language of life through transfer-learning protein sequences. *BMC Bioinformatics* 2019;**20**(1):723.
66. Alley EC, Khimulya G, Biswas S, *et al*. Unified rational protein engineering with sequence-based deep representation learning. *Nat Methods* 2019;**16**(12):1315–22.
67. Dai Z, Yang Z, Yang Y, *et al*. Transformer-xl: attentive language models beyond a fixed-length context arXiv preprint arXiv:1901.02860. 2019.
68. Yang Z, Dai Z, Yang Y, *et al*. Xlnet: Generalized autoregressive pretraining for language understanding. In: *Advances in Neural Information Processing Systems*, 2019.
69. Liu Y, Ott M, Goyal N, *et al*. Roberta: a robustly optimized bert pretraining approach arXiv preprint arXiv:1907.11692. 2019.
70. Lan Z, Chen M, Goodman S, *et al*. Albert: a lite bert for self-supervised learning of language representations arXiv preprint arXiv:1909.11942. 2019.
71. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. *Commun ACM* 2017;**60**(6):84–90.
72. Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate arXiv preprint arXiv:1409.0473. 2014.
73. Zhuang Z, Shen X, Pan W. A simple convolutional neural network for prediction of enhancer–promoter interactions with DNA sequence data. *Bioinformatics* 2019; **35**(17):2899–906.
74. Feng P, Yang H, Ding H, *et al*. iDNA6mA-PseKNC: identifying DNA N(6)-methyladenosine sites by incorporating nucleotide physicochemical properties into PseKNC. *Genomics* 2018.
75. Deznabi I, Arabaci B, Koyutürk M, *et al*. DeepKinZero: zero-shot learning for predicting kinase–phosphosite associations involving understudied kinases. *Bioinformatics* 2020;**36**(12):3652–61.
76. Karimi M, Wu D, Wang Z, *et al*. DeepAffinity: interpretable deep learning of compound–protein affinity through unified recurrent and convolutional neural networks. *Bioinformatics* 2019;**35**(18):3329–38.
77. Rives A, Meier J, Sercu T, *et al*. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *bioRxiv* 2019;622803.
78. Vig J, Madani A, Varshney LR, *et al*. Bertology meets biology: interpreting attention in protein language models. arXiv preprint arXiv:2006.15222. 2020.
79. Nambiar A, Liu S, Hopkins M, *et al*. Transforming the language of life: transformer neural networks for protein prediction tasks. In: *Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*. 2020.
80. Elnaggar A, Heinzinger M, Dallago C, *et al*. ProtTrans: towards cracking the language of Life's code through self-supervised deep learning and high performance computing arXiv preprint arXiv:2007.06225. 2020.
81. Feng P, Yang H, Ding H, *et al*. iDNA6mA-PseKNC: identifying DNA N6-methyladenosine sites by incorporating nucleotide physicochemical properties into PseKNC. *Genomics* 2019;**111**(1):96–102.
82. Min S, Park S, Kim S, *et al*. Pre-training of deep bidirectional protein sequence representations with structural information arXiv preprint arXiv:1912.05625. 2019.
83. Rao R, Bhattacharya N, Thomas N, *et al*. Evaluating protein transfer learning with TAPE. In: *Advances in Neural Information Processing Systems*, 2019.
84. Qiu X, Sun T, Xu Y, *et al*. Pre-trained models for natural language processing: a survey. arXiv preprint arXiv:2003.08271. 2020.
85. Zhang T, Tan P, Wang L, *et al*. RNALocate: a resource for RNA subcellular localizations. *Nucleic Acids Res* 2017;**45**(D1):D135–8.
86. Liang ZY, Lai HY, Yang H, *et al*. Pro54DB: a database for experimentally verified sigma-54 promoters. *Bioinformatics* 2017;**33**(3):467–9.
87. Baltrušaitis T, Ahuja C, Morency L-P. Multimodal machine learning: a survey and taxonomy. *IEEE Trans Pattern Anal Mach Intell* 2018;**41**(2):423–43.
88. Baevski A, Edunov S, Liu Y, *et al*. Cloze-driven pretraining of self-attention networks. arXiv preprint arXiv:1903.07785. 2019.
89. Joshi M, Chen D, Liu Y, *et al*. Spanbert: improving pre-training by representing and predicting spans. *Trans Assoc Comput Linguist* 2020;**8**:64–77.
90. Lewis M, Liu Y, Goyal N, *et al*. Bart: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:1910.13461. 2019.
91. Wang W, Bi B, Yan M, *et al*. Structbert: incorporating language structures into pre-training for deep language understanding arXiv preprint arXiv:1908.04577. 2019.

92. Arora S, Khandeparkar H, Khodak M, *et al*. A theoretical analysis of contrastive unsupervised representation learning arXiv preprint arXiv:1902.09229. 2019.
93. Chen T, Kornblith S, Norouzi M, *et al*. A simple framework for contrastive learning of visual representations arXiv preprint arXiv:2002.05709. 2020.
94. He, K., Fan H, Wu Y. *et al*. Momentum contrast for unsupervised visual representation learning. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020.
95. Oord AVD, Li Y, Vinyals O. Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748. 2018.
96. Qiu, J., Chen Q, Dong Y. *et al*. Gcc: graph contrastive coding for graph neural network pre-training. In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2020.
97. Caruana R. Multitask learning. *Mach Learn* 1997;**28**(1): 41–75.
98. Gordon MA, Duh K, Andrews N. Compressing BERT: studying the effects of weight pruning on transfer learning. arXiv preprint arXiv:2002.08307. 2020.
99. Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531. 2015.
100. Zeng X, Song X, Ma T, *et al*. Repurpose open data to discover therapeutics for COVID-19 using deep learning. *J Proteome Res* 2020;**19**(11):4624–36.
101. Zeng X, Liao Y, Liu Y, *et al*. Prediction and validation of disease genes using HeteSim scores. *IEEE/ACM Trans Comput Biol Bioinform* 2017;**14**(3):687–95.
102. Jin S, Zeng X, Xia F, *et al*. Application of deep learning methods in biological networks. *Brief Bioinform* 2020; **22**(3).
103. Zeng X, Lin W, Guo M, *et al*. A comprehensive overview and evaluation of circular RNA detection tools. *PLoS Comput Biol* 2017;**13**(6):e1005420.
104. Zou Q, Li J, Song L, *et al*. Similarity computation strategies in the microRNA-disease network: a survey. *Brief Funct Genomics* 2016;**15**(1):55–64.
105. Zhang X, Zou Q, Rodriguez-Paton A, *et al*. Meta-path methods for prioritizing candidate disease miRNAs. *IEEE/ACM Trans Comput Biol Bioinform* 2017;**16**(1):283–91.
106. Wei L, Xing P, Zeng J, *et al*. Improved prediction of protein-protein interactions using novel negative samples, features, and an ensemble classifier. *Artif Intell Med* 2017;**83**:67–74.
107. Wei L, Wan S, Guo J, *et al*. A novel hierarchical selective ensemble classifier with bioinformatics application. *Artif Intell Med* 2017;**83**:82–90.
108. Su R, Liu X, Xiao G, *et al*. Meta-GDBP: a high-level stacked regression model to improve anticancer drug response prediction. *Brief Bioinform* 2020;**21**(3):996–1005.
109. Su R, Liu X, Wei L. MinE-RFE: determine the optimal subset from RFE by minimizing the subset-accuracy-defined energy. *Brief Bioinform* 2020;**21**(2):687–98.
110. Su R, Hu J, Zou Q, *et al*. Empirical comparison and analysis of web-based cell-penetrating peptide prediction tools. *Brief Bioinform* 2020;**21**(2):408–20.
111. Qiang X, Zhou C, Ye X, *et al*. CPPred-FL: a sequence-based predictor for large-scale identification of cell-penetrating peptides by feature representation learning. *Brief Bioinform* 2020;**21**(1):11–23.
112. Wei L, Liao M, Gao Y, *et al*. Improved and promising identification of human MicroRNAs by incorporating a high-quality negative set. *IEEE/ACM Trans Comput Biol Bioinform* 2014;**11**(1):192–201.