Fang PF, Li X, Yan Y *et al.* Connecting the dots in self-supervised learning: A brief survey for beginners. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 37(3): 507–526 May 2022. DOI 10.1007/s11390-022-2158-x

Connecting the Dots in Self-Supervised Learning: A Brief Survey for Beginners

Peng-Fei Fang[1,2] (方鹏飞), Xian Li[1] (李　贤), Yang Yan[1,3] (燕　阳), Shuai Zhang[1,3] (章　帅)
Qi-Yue Kang[1] (康启越), Xiao-Fei Li[1] (李晓飞), and Zhen-Zhong Lan[1] (蓝振忠)

[1] *School of Engineering, WestLake University, Hangzhou 310030, China*
[2] *College of Engineering and Computer Science, Australian National University, Canberra, ACT 2601, Australia*
[3] *College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China*

E-mail: {fangpengfei, lixian, yanyang, zhangshuai, kangqiyue, lixiaofei, lanzhenzhong}@westlake.edu.cn

Received January 15, 2022; accepted May 18, 2022.

Abstract The artificial intelligence (AI) community has recently made tremendous progress in developing self-supervised learning (SSL) algorithms that can learn high-quality data representations from massive amounts of unlabeled data. These methods brought great results even to the fields outside of AI. Due to the joint efforts of researchers in various areas, new SSL methods come out daily. However, such a sheer number of publications make it difficult for beginners to see clearly how the subject progresses. This survey bridges this gap by carefully selecting a small portion of papers that we believe are milestones or essential work. We see these researches as the "dots" of SSL and connect them through how they evolve. Hopefully, by viewing the connections of these dots, readers will have a high-level picture of the development of SSL across multiple disciplines including natural language processing, computer vision, graph learning, audio processing, and protein learning.

Keywords artificial intelligence (AI), dot, self-supervised learning (SSL), survey

1 Introduction

> "You can't connect the dots looking forward; you can only connect them looking backwards. So you have to trust that the dots will somehow connect in your future."
> — Steve Jobs①.

In the last few years, the artificial intelligence (AI) community has witnessed a boom in self-supervised learning (SSL), a class of algorithms that can learn meaningful representations② without manually labeled data. These methods have significantly improved the performance of a variety of AI-related tasks [1–3]. Research fields like natural language processing (NLP) [4], computer vision (CV) [5, 6], and speech recognition [7] have all witnessed breakthroughs through the use of self-supervised methods. With the rapid growth of computational power, modern neural architectures endowed with self-supervised algorithms can even improve supervised models trained with over a million labeled data [6].

Having its advantages in representation learning, SSL has become a popular research topic. Recently, thousands of papers [1, 2] have been published each year, and such a massive number of publications make it difficult for researchers, especially newcomers, to find out genuinely inspiring articles and gain an overall picture of how SSL evolves.

In addition to many related publications, the intriguing property of SSL allows it to raise interdisciplinary. That said, innovations of SSL can appear in any of the application fields because of its wide us-

Survey
Special Section on Self-Learning with Deep Neural Networks
This work was supported by the Key Research and Development Program of Zhejiang Province of China under Grant No. 2021-C03139.
① Jobs S. 'You've got to find what you love,' Jobs says. https://news.stanford.edu/2005/06/14/jobs-061505/, December 2021.
② In the remainder of this manuscript, we will use the terms "representation" and "embedding" interchangeably.

age. Researchers these days often get ideas from related fields. For example, both context predictions [8] and wav2vec [9] were inspired by the famous Word2Vec [10] algorithm. Similarly, Mockingjay [11] and MAE [6] are the audio and visual version of BERT [4], respectively. Therefore, interdisciplinary integration requires researchers to keep track of papers across all related research fields.

Usually, surveys are good resources for beginners to learn a particular field quickly and comprehensively. However, due to the tendency to include more papers, these surveys themselves are becoming lengthy and difficult to digest. Actually, when these surveys list their main contributions, "comprehensive" and "detailed" are often keywords [2, 12]. In addition, because new papers come out daily, some methods listed in these surveys will quickly become outdated and lack reference significance. In order to make our survey easy to understand, we select a handful of milestone papers and important work from each field. We call these papers "dots", and connect them instead of listing or categorizing them. By connecting these dots, we clarify how SSL evolves and how different research fields inspire one another.

Our criterion for selecting these dots is that they must be the top-cited papers in the fields. We first determine the famous work for the feature engineering for each field. We restrain ourselves from selecting those papers published after 2013, because deep learning, the activator of representation learning, gets popular since 2013. Before then, training deep neural networks (DNNs) with graphics processing unit (GPU) was not trendy and researchers spent many of their efforts reducing computational complexity. Hence they paid less attention to the SSL algorithm itself. Table 1 gives an overview of the work presented in this survey.

In this brief survey, we review and connect the work from NLP, CV, graph learning, audio processing, and protein learning. By looking at the links of these SSL methods from different fields, we can see the followings. 1) Supervised learning contributes significantly to the development of SSL. Major neural network architectures like residual networks and transformers resulting supervised learning research are essential to SSL. 2) SSL methods in NLP like Word2Vec [10] and BERT [4] inspired most SSL methods in other fields. 3) The gains in hardware are the main driving force of SSL methods as they are computationally demanding.

We structure the rest of this paper as follows. From Section 2 to Section 6, we present the recent advances of SSL in NLP, CV, graph learning, audio processing, and protein learning, respectively. In Section 7 and Section 8, we discuss the existing survey articles and the future trends of SSL. We conclude our article in Section 9.

2 Self-Supervised Learning in Natural Language Processing

The language data, i.e., text, is a typical sequence of word-tokens, and is easily accessible through the Internet. However, annotating the label for text is very expensive. For example, on Google AI Platform, assigning a class label for a piece of text with 50 words is about 4x as expensive as classifying an image[③]. This motivates researchers to invent effective SSL methods to learn language representations without using text labels. Its progress diagram is shown in Fig.1.

After a few decades of research efforts, several key ingredients for SSL are identified. Some of the very important factors are global modeling and large size in terms of both network and data. However, all of these factors demand large computation resources. In this section, we will learn how researchers in the field, constrained by computational resources, gradually gather together and improve the above ingredients.

The journey is not without peril. One great temptation that researchers must resist is optimizing for a specific task. Although such improvements can be useful at the time when the invention was made, they have little, sometimes even negative, contribution towards the elusive final goal of solving language understanding problems. Many researchers go astray along this journey.

2.1 Traditional Feature Engineering

The Bag-of-Words (BoW) model yields a simple realization for the text representation [13]. It presents the text as a set of its words and calculates the frequency of occurrence per word. TF-IDF is another word embedding technique [14]. It calculates the two statistical values, term frequency (TF) and inverse document frequency (IDF), and multiplies those two values, such that it can indicate the significance of the rare words in the document. These simple methods have attained considerable improvement in language modeling and

[③] https://cloud.google.com/ai-platform/data-labeling/pricing, Dec. 2021.

Table 1. Overview of the Dots of Self-Supervised Learning Methods in Natural Language Processing, Computer Vision, Graph Learning, Audio Processing, and Protein Learning

Field	Local Modeling	Global Modeling		
		Discriminative Modeling	Generative Modeling	Discriminative & Generative
Natural language processing	Word2Vec [10] BoW [13] TF-ID [14]	BERT [4] Collobert *et al.* [15] SA-LSTM [16] ELMO [17] ULMFiT [18] RoBERTa [19] XLNeT [20] ALBERT [21] ELECTRA [22] DeBERTa [23] T5 [24]	GPT [25] GPT-3 [26]	–
Computer vision	Patch position prediction [8] SIFT [27] HOG [28] SURF [29] Exemplar CNN [30] Counting [31] Jigsaw puzzle [32] Unsupervised tracking [33] Ego-motion [34]	DIM [35] AM-DIM [36] CMC [37] MoCo [38] SimCLR [39] MoCo v2 [40] SimSiam [41] DINO [42] BYOL-A [43]	MAE [6] BEiT [44] MaskFeat [45]	–
Graph learning	DeepWalk [46] node2vec [47]	DGI [48] InfoGraph [49] GRACE [50] CMVR [51] GROVER [52]	Graph-GAN [53] GPT-GNN [54]	–
Audio processing	Mel-spectrograms [55]	HuBERT [7] CPC [56] wav2vec [57] COLA [58] CLAR [59] CLMR [60] BYOL-A [43] vq-wav2vec [61] wav2vec 2.0 [62]	Mockingjay [11] APC [63] VQ-APC [64] TERA [65]	wav2vec-C [9] PASE [66] PASE+ [67]
Protein learning	PCF [68] PSSM [69] ProtVec [70]	CPCProt [71]	TAPE [72] UniRep [73] SeqVec [74] UDSMProt [75] PLUS [76] ESM [77] ProtTrans [78] PMLM [79] HJRSS [80] MSA transformer [81]	–

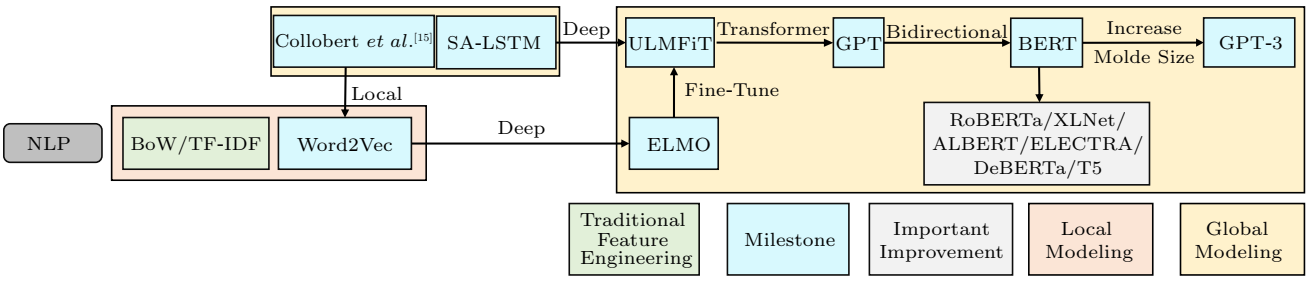


Fig.1. Diagram of the progress of SSL in the NLP field.

text classification tasks. However, they cannot understand the semantic meaning of the text, limiting the capacity of machine understanding.

2.2 Early Attempts on Global Modeling

In the deep learning era, let us begin our recount with the seminal work, proposed by Collobert *et al.* in 2011 [15]. Prior to this work, the majority of the state-of-the-art systems were built upon task-specific features. This work addresses each task independently with linear models on top of features that contain a large body of manually designed linguistic knowledge. Different from these systems, Collobert *et al.* learned contextualized intermediate representations through language modeling (LM), i.e., estimating the acceptability of a word given the previous words in a sentence [15]. Through this classical pretext task in language learning, [15] performs full network pre-training on convolutional neural networks (CNNs) and fine-tuned the network on multiple downstream benchmarks. This visionary design includes both full-network pre-training and long contextualized embedding. However, without using GPUs, this approach was still too expensive to compute. Early attempts on global modeling were limited due to the computational constraint.

2.3 Falling Back into Local Modeling

As the global modeling in [15] was computationally demanding, this method was not so popular as another local, shallow, and lightweight method called Word2Vec [10], which appeared two years later in 2013. In order to train on a larger dataset, Mikolov *et al.* [10] shortened the input window size to 5 and only used a one-layer MLP as the network. Specifically, they used a linear network to either predict middle words from four neighboring words (CBOW) or the four neighboring words from the middle word (skip-gram). The trade-off among the data size, network size, and context size was a huge success at that time. In fact, it was so successful that follow-up researchers even called into question "the importance of the full neural network structure for learning useful word representation". However, although good word embeddings helped to improve the performance of language tasks in general, they also encouraged researchers to design task-specific networks on top of pre-training embeddings. A number of follow-up studies went astray along that direction and later on were proved to be meaningless.

2.4 Global Modeling Regaining Its Strength

It was not until 2015 that Dai and Le revisited the full network pre-training strategy in the SA-LSTM model [16]. They used both LM and sequence autoencoder which reconstructs the input sequence from the hidden states as their pre-training target and showed that sequence autoencoder was a better choice as a pre-training model. It was also observed that a well pre-trained model did significantly improve the performance of multiple downstream tasks. However, their networks were still shallow and only pre-trained on a relatively small target dataset, which limited the power of pre-training. Full-network pre-training still did not get popular.

2.5 Full Prosperity of Global Modeling

Finally, in 2018, the NLP community met its year of wonder. In February, Peters *et al.* introduced a new type of deep contextualized word representation called ELMO [17]. Compared with CBOW and skip-gram, ELMO was pre-trained on a larger network, and used a bidirectional language model and a much longer context. It pre-trained a fix contextualized word embedding through a 2-layer bidirectional LSTM. At roughly the same time, a similar method called ULMFiT was proposed by Howard and Ruder [18]. Instead of using a fixed word embedding, they proposed to fine-tune the deep pre-trained network and further increased the depth of the network. In June, Radford *et al.* suggested to pre-train a transformer decoder, which is a typical architecture of modern language model [25]. This new method was called GPT and it had a much larger network compared with previous approaches. The ability to model contexts was also stronger compared with the LSTM or CNN methods. Devlin *et al.* proposed the now well-known BERT model [4]. In BERT, the model pre-training is realized by a new pretext task, namely, the masked language model (MLM), i.e., masking some of the words in the inputs and recovering them through the network. BERT also further increases the data scale by including both the book corpus and the Wikipedia dataset. The simplicity of the model, its significant performance improvements, and the easy-to-use toolkits made BERT extremely popular.

On top of BERT, there are several variants including RoBERTa [19], XLNeT [20], ALBERT [21], ELECTRA [22], DeBERTa [23], T5 [24], etc. Most of these models increase the data size, increase the model

size, or design a better objective function. For example, as compared with BERT, RoBERTa is trained on larger datasets with longer training time; ALBERT is a much wider network (although the network size is smaller through sharing the parameter cross layers); ELECTRA replaces the MLM task with a task of predicting whether a token is generated or not, so that it calculates losses in all the token position; DeBERTa combines LM and MLM objectives together, and T5 converts all text-based language problems into a text-to-text format and trains a much larger model (1.5 billion parameters).

Although the above methods gain significant performance improvement over BERT, they still follow the pre-training fine-tune scheme, where the labeled data is needed. In contrast to the above approaches, GPT-3 [26], proposed by Brown *et al.*, shows that when the pre-trained model is larger enough, one can remove the necessity of fine-tuning the model, and instructions including basic information about the task can generate appealing text. This progress from large networks brought the SSL in language to a brand new high level.

2.6 Epilogue in NLP

Looking back, improved network architecture like Transformer [82] and bigger datasets have fueled a revolution in SSL. Transformer [82], although initially invented for machine translation — a typical supervised method, enabled us to create a much larger and deeper network in SSL and currently is the main architecture for most of the SSL tasks. Because SSL with Transformer can continuously benefit from larger architectures and larger quantities of data, one of the biggest trends for SSL in NLP has been the ever-increasing model size [83].

Also, the development of SSL in NLP has a great influence on other research fields. For example, the Transformer architecture is also preferable as a feature extractor in the CV field. In the following Sections 3–6, we will briefly introduce the important dots of SSL in other research fields.

3 Self-Supervised Learning in Computer Vision

Learning discriminative representations of visual data (e.g., image embedding or video embedding) in a self-supervised fashion have been considered as an important problem in the CV community. In CV applications, the input data is images or a sequence of video frames, composed of well-structured discrete RGB values. However, labeling a large number of images is expensive, and the cost increases dramatically for the task of pixel-level prediction, i.e., semantic segmentation. For example, annotating the pixel-level label per $2\,048 \times 1\,024$ image costs more than 1.5 hours [86]. To avoid the reliance on human effort for the data, SSL, therefore, becomes a useful tool to build a pre-trained feature extractor.

By looking back at the progress of SSL in visual data, its development is in line with this intuition from local modeling to global modeling (see its progress diagram in Fig.2). However, different from NLP, the local modeling in CV only happens in the data part. For the network part, the pre-training is on the whole network

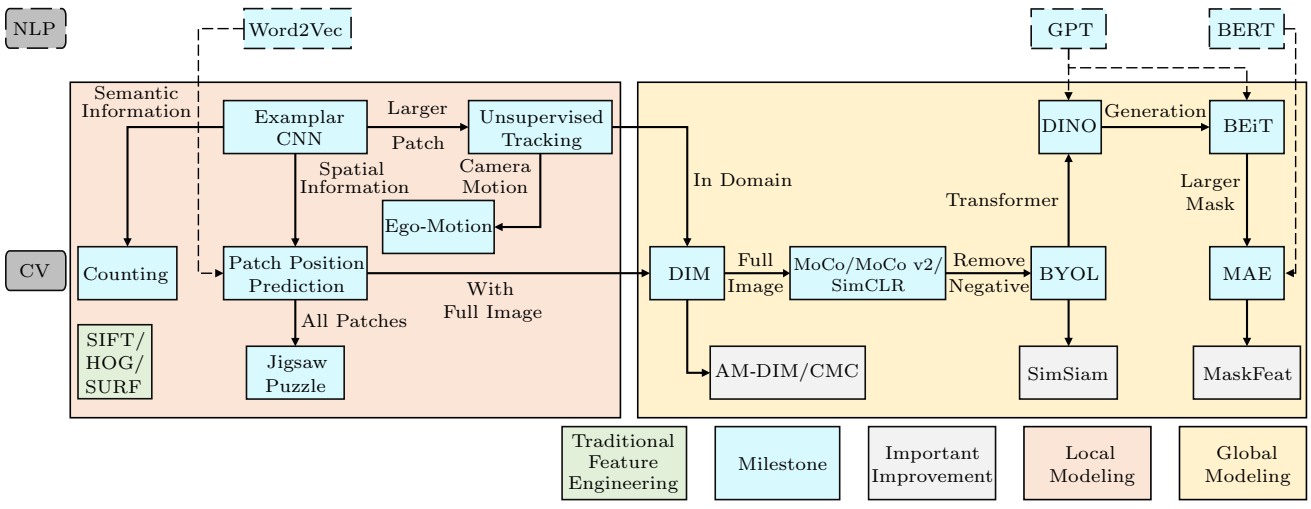


Fig.2. Diagram of the progress of SSL in the CV field. Note that the dashed boxes and the dashed arrows indicate methods from other research fields and the inspiration paths, respectively.

from the beginning. We conjecture that it is because of the efficient computation of CNNs and the high accuracy bar brought by the supervised pre-training on ImageNet [87].

3.1 Traditional Feature Engineering

The traditional feature engineering for visual data creates the image descriptors. The scale-invariant feature transform (SIFT) [27] was proposed by Lowe in 2004. SIFT is invariant to image transformation (e.g., scaling or rotation); hence, it can perform reliable matching between different views of an object. HOG improves SIFT by counting occurrences of gradient orientation in localized portions of images [28]. Beyond SIFT descriptor, a fast algorithm, called SURF, was further invented [29]. Its feature descriptor is based on the summation of the Haar wavelet response around the point of interest. It leverages the multi-resolution pyramid technique to realize the blurring effect and to guarantee the scale-invariant property of the interesting point.

3.2 Local Modeling on Patches

Early work of learning highly-discriminative visual representations leveraged the local cues within images. The work, exemplar CNN, samples a set of 32×32 patches from the same image and applies various data transformations to each patch [30]. These patches from one image are grouped into one category, such that a network can be trained to discriminate between a set of categories. Understanding the visual concepts is necessary for the feature extractor and the Counting in [31] defines a counting rule in the pre-text task, which trains the network to recognize visual primitives, e.g., noses, eyes, by means of correctly predicting the counting relationship.

To make the network understand both the scenes and objects, another famous work utilized the spatial position of patches of one image as labels and model the SSL as a task to predict the spatial relationship between patches [8]. In doing so, a pair of patches are sampled per image and is fed to a network which is required to predict the relative position of two patches to learn more inherent visual information. Following the intuition that doing a complex task well requires more knowledge, a jigsaw puzzle game, where the objective trains the network to place shuffled patches back to their original positions, was further proposed as a pretext task in SSL [32]. In [32], all patches are shuffled and independently fed into an encoder, such that the encoder can jointly learn the feature embeddings of patches and the associated spatial arrangement. More challenging settings were also investigated in [88, 89].

The initial development using local features has shown positive results for SSL. Increasing the receptive field of images becomes a possible way for further study.

In [33], unsupervised tracking is performed as a pretext task. That is, the visual tracker provides a query patch and a positive patch from the same video and samples a negative patch from other videos, such that the patch features can be optimized by the triplet loss [82]. In contrast to tracking the moving objects, estimating the motion from the camera for videos (a.k.a., ego-motion) is further considered later in [34], where the objective is to synthesize a targeted view using the depth and pose features. These video-based methods can learn robust features but have difficulty applying them to image tasks, because of the domain gap between videos and images.

The SSL for visual data at an early age was achieved by defining complex pretext tasks, and most of the pretext tasks used the local features of an image/video. Even though it has achieved considerable improvement, it has difficulty in encoding the holistic representation. However, objects within image/video are well-structured, and such structured information indeed affects the representation power. This issue can be addressed by discriminative modeling or generative modeling, which learns the global representation of images.

3.3 Discriminative Global Modeling from Augmented Data

In the discriminative modeling, the basic idea relies on the Noise Contrastive Estimation (NCE) [85]. In NCE, a positive pair only contrasts with one negative pair, which is similar to the triplet loss [84]. A more general formulation is called InfoNCE [56], where a positive pair contrasts with many negative pairs④. In the NCE or InfoNCE framework, two main groups of approaches are studied to realize SSL, i.e., mutual information estimation and contrastive learning scheme. In the following, we will briefly introduce those two types of methods.

④For reading friendly, mathematical formulations are omitted in this article. We refer the interested readers to [56, 85] for more details.

3.3.1 Mutual Information Estimation

The methods using mutual information (MI) achieve SSL by jointly estimating and maximizing MI, and MI can also be presented by the NCE value. Intuitively, maximizing the MI of two variables can align the associated distributions. In the CV field, the variables can be modeled as different views of images. The seminal work, Deep InfoMax (DIM), models the variables as global context features and local region features [35]. That said, maximizing the MI between global features and local features forces the network to encode the consistent information of global and local features of images.

Exploration of better ways to model the variables for MI was studied in the following work [36, 37]. Augmented Multiscale DIM (AM-DIM), applies various augmentation skills to context and region features of the same image, thereby enforcing the deep network to learn a high-level image representation that is robust against the diversity of data transformation [36]. In [37], contrastive multiview coding (CMC) calculates the MI value between global features, and such features are encoded from the same images with different views. This setting enables networks to learn the view-invariant factors of images. Even though CMC optimizes the MI as the objective, it has a fundamental difference from DIM and AM-DIM in that CMC considers the global-to-global MI, while infoMax and AM-DIM optimize the global-to-local MI.

3.3.2 Contrastive Learning Scheme

Learning with a contrastive scheme is also a natural idea in supervised representation learning [84, 90, 91] and has been studied extensively in recent years for SSL. In [38], He *et al.* developed MoCo, which adopts two encoders to the same image, leading to a positive pair. MoCo also proposes a momentum contrastive scheme, which significantly enlarges the number of negative pairs. Despite its effectiveness, creating positive pairs without using data augmentation makes the encoder easy to distinguish positive pairs. This issue is addressed by another seminal work, SimCLR, proposed by Chen *et al.* [39]. SimCLR establishes a general framework for SSL using a contrastive scheme [39]. Similar to CMC [37], SimCLR adopts 10 data augmentation techniques and each positive pair can be constructed by applying two random augmentations to the same image. More importantly, the authors also conducted heuristic experiments to study the correct usage of contrastive loss. To be specific, it is observed that a large batch size, non-linear projection heads, deeper networks, and more training steps are essential factors for a good practice of contrastive loss. The MoCo v2 justifies the effectiveness of such training methods by integrating them into the MoCo framework [40].

In contrast to work in [39, 40] adopting more negative pairs in infoNCE loss, both BYOL and SimSiam avoid collapsing solutions during the optimization process even without using negative pairs [41, 92]. It is observed from BYOL [92] that using a static key encoder (referred to as target encoder) can avoid the collapse because the static network is not trained. With such an observation, BYOL trains a query encoder (referred to as online encoder) as in the common practice and iteratively updates the key encoder with a moving average of the query network. The same idea also occurs in SimSiam [41], whereas two encoders are identical, and a projection head is added to one of the encoders, creating two views of features.

The success of Transformer architecture (i.e., BERT) in the NLP field suggests using Transformer as an alternative building block of the backbone network, which is verified in the Vision Transformer (ViT) [93]. DINO further bridges the gap between ViT and SSL, i.e., training a ViT in a self-supervised manner, and reveals that the Transformer architecture can learn class-specific semantic information [42]. DINO follows the form of self-distillation that contains a teacher network and a student network and optimizes the objective of the cross-entropy loss calculated between the features from the student and the central feature from the teacher.

Discriminative modeling indeed makes significant progress as a pre-training technique, and the recognition accuracy on ImageNet is very close to the supervised learning. However, because all of these approaches are built upon the concept of distinguishing the augmented data from all other data, it is not so difficult as generative tasks in general. Its further improvement is stepped by using the generative ideas from NLP.

3.4 Generative Modeling Through Recovering Missing Image Patches

Inspired by the significant progress of generative modeling in NLP, one can also consider adopting such models (e.g., GPT or MLM) as candidates to learn image representations.

In [5], the image is operated via downsampling and flattening, obtaining a 1D sequence, which is then fed to a generative model, i.e., GPT, to realize the pixel generation objective. Despite that iGPT only generates low-resolution images, it shows its potential that it achieves SOTA performance as compared with its competitors in low-resolution representation learning. Recovering masked pixels, which mimics the pipeline in MLM, is also studied in BEiT [44]. Training a BEiT consists of two steps, with the first step that an auto-encoder is applied to tokenize the patch features. Then the masked image modeling (MIM) is used as a pre-training task, which trains the network to predict the masked visual tokens. A simpler yet effective method, MAE, proposed by He *et al.*, further simplifies the training paradigm in that it trains an asymmetric auto-encoder to construct the masked patches [6]. Appealing performance is observed that the trained auto-encoder can recover images with only 25% visible patches. Recent work, termed MaskFeat, proves that the model's prediction in the feature space (i.e., HOG features) is much better than that in pixel spaces [45].

3.5 Epilogue in CV

With the large-scale application of Transformer in the field of CV, the development of SSL in NLP and CV is getting approach. Although the development of SSL in CV bears the imprint of NLP, CV has also begun to feedback on the development of NLP. For example, SimCSE [94] uses dropout as minimal data augmentation for sentence embedding and applies contrastive learning on top of pre-trained model like BERT [4] or RoBERTa [19]. The resulting pre-training model significantly outperforms the original models.

4 Self-Supervised Learning in Graph Learning

Graph data is presented by a set of nodes, with linked ones being related. Unlike other formats of data, the graph can model a number of graph-structured data, e.g., the social networks, molecules, knowledge graphs. Addressing problems with graph data is not easy and the emergence of graph neural networks (GNNs) makes the solutions flexible and easier. That said, once the input data is modeled as a graph, the GNNs provide a powerful framework for the tasks at hand, e.g., node predication, edge prediction, or graph predication. The recent trend also shows promising results by employing the SSL on GNNs for pre-training. Its various applications make the progress from local modeling for node-level tasks to global modeling for graph-level tasks, shown in Fig.3.

4.1 Traditional Feature Engineering

An early solution of learning graph embeddings uses walks to traverse the graph and aggregates the connected node representations. This is known as DeepWalk [46], and it learns the node representations by leveraging the skip-gram from Word2Vec [10].

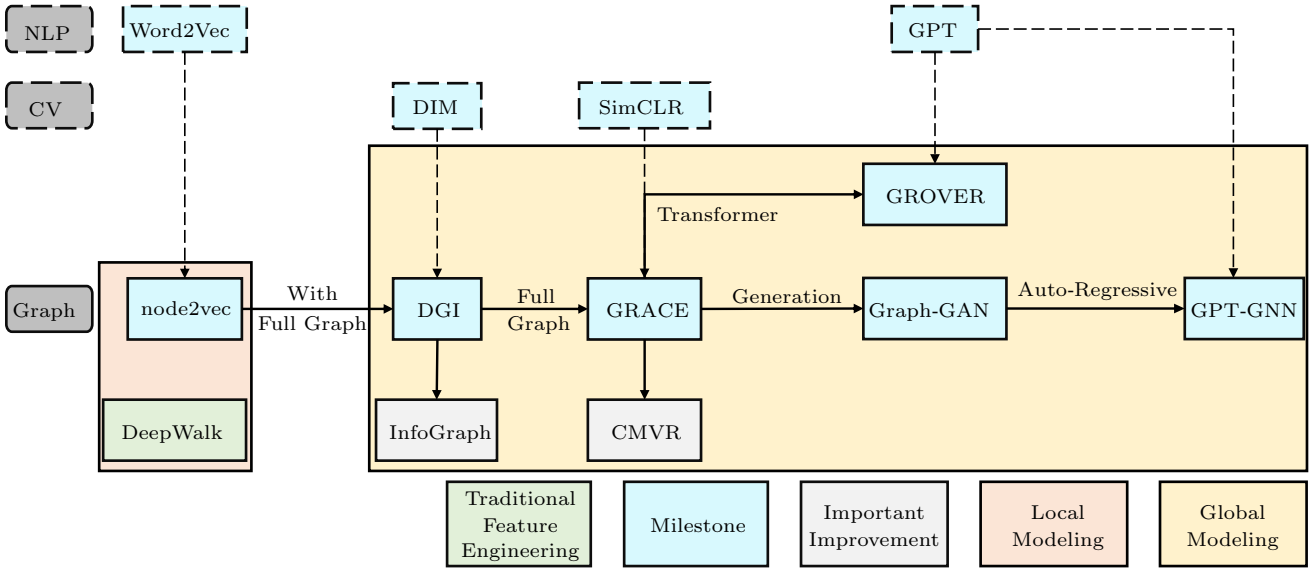


Fig.3. Diagram of the progress of SSL in the graph learning field. Note that the dashed boxes and the dashed arrows indicate the methods from other research fields and the inspiration paths, respectively.

4.2 Local Modeling as a Way of Embedding Nodes

In dealing with the SSL for graph data, it comes to mind that the local modeling can be a straightforward choice as that in the NLP and CV fields. Similar to Word2Vec in NLP, node2vec was proposed for graph data [47]. Using the network neighborhoods of nodes as supervision signal, node2vec establishes node presentations that keep the connection relationship the same between the graph space and the embedding space. As in the CV field, graph learning also includes discriminative and generative modeling.

4.3 Discriminative Modeling Maximizing the Similarity of Different Views

The discriminative modeling of SSL for graph-structured data also follows closely the progress of the visual data, where the main categories are MI estimation and the contrastive learning scheme.

The graph counterpart of DIM, termed Deep Graph Infomax (DGI), was developed in [48]. In DGI, a graph convolutional network (GCN) is trained to learn node representations by the infoNCE objectives, thereby maximizing MI between the local patch representations and the global graph representations. In practice, the local patch representation is the high-level node feature, aggregated from the node and its neighborhoods, and the global graph representation is summarized by the readout function over node features. A similar idea was extended to InfoGraph [49], where a GIN is trained to encode the graph representations.

Its further improvement derives from the success of the contrastive scheme [50, 51]. A simple attempt is performed in GRACE [50] that the representation per node is optimized by maximizing the agreement of two graph views, where the graph views are constructed by removing edges to neighbors and masking node features. In [51], a new method, namely, CMVR, investigates a new method to create different views per graph. Given a raw graph, another view is created by graph diffusion. The origin graph and the augmented graph are fed to two separated GCN encoders respectively, to obtain both node features and graph features, which are then optimized by contrasting the node representations from one view to the graph representations of another view. In contrast to establishing various views for graphs, GROVER defines a pretext task that predicts the contextual properties of the node/edge and adopts Transformer, jointly learning representations for graphs [52].

4.4 Generative Modeling via Generating Graph Components

Generative modeling on graph data relies on two pipelines, i.e., generative-adversarial and auto-regressive [53, 54]. Under the framework of GANs, Graph-GAN is composed of two networks, i.e., a generator and a discriminator [53]. For each node, the generator aims to learn the underlying connectivity of all nodes and generates a graph as a fake sample. Then the discriminator can tell the connectivity of true pairs and false pairs.

The generative pre-training is realized in GPT-GNN [54]. To achieve so, self-supervised attributed graph generation is defined as a pre-training pretext task. By the generative process for both node attributes and edges, the network can capture the inherent dependency of the underlying graphs, thereby producing powerful representations.

4.5 Epilogue in Graph Learning

As suggested by the above SSL methods, we can find that researches in both NLP and CV are the sources of ideas for the SSL in graph learning, though the format of graph data is significantly different from that of text and images. This indeed shows the importance of interdisciplinary research. We believe the generative modeling over the Transformer architecture [82] would be an important direction to explore.

5 Self-Supervised Learning in Audio Processing

Audio data is a format of time sequence being continuous in both time and amplitude. To facilitate analysis, the audio signal is normally split into clips with duration varying from hundreds of milliseconds to several seconds depending on the task at hand. According to the frequency spread, the audio signal is sampled in time with a rate of, e.g., 16 kHz⑤. Assuming that the signal is stationary (with invariant frequency components) in one frame, each sampled audio clip is further split into frames with a constant frame length, e.g., 10 milliseconds. The raw audio samples can be directly fed to a neural network as input, or alternatively, a feature vector can be extracted for each frame in the frequency domain, e.g., the log-Mel (log-magnitude in

⑤It means the audio signal consists of 16 thousand samples per second.

Mel-Frequency) feature. With this feature representation, an audio clip is represented as a matrix with axes of frequencies and time frames, which is called a spectrogram. Audio units, such as speech phones, sound events, and music notes, have varying lengths and normally occupy multiple frames. The application over audio data includes clip-level tasks and frame-level tasks.

Its SSL pre-training has been developed rapidly since 2019, and many ideas are inspired by the NLP/CV field. Due to the fact that audio frames have strong temporal dependencies, including short-term dependencies due to the signal smoothness within audio units, and long-term dependencies between audio units reflecting the semantic information, the modeling of SSL mainly focuses on discriminative modeling and generative modeling for the contextual/global embedding. (Refer to Fig.4 for its progress diagram.)

5.1 Traditional Feature Engineering

In the traditional feature engineering, the audio representations can be represented by the Mel-spectrograms [55], which are calculated from the log-magnitude spectrum. Due to the property of the spectrum features, it can preserve both the frequency resolution and amplitude of a signal.

5.2 Discriminative Modeling via Contrastive Scheme

The discriminative modeling in audio data minimizes a pretext classification loss. In [56], targeting the task of future frame prediction, CPC aims to correctly classify the positive frames (future k frames) from a set of negative frames (other frames in the audio). Pre-training for a downstream task, i.e., speech recognition, is realized by wav2vec [57], where MI between the speech context embedding and the future frame embeddings is maximized. Extending the idea from SimCLR [39], some methods, e.g., COLA [58], CLAR [59] and CLMR [60], propose to create positive samples in the contrastive objective for the clip-level feature learning. Similar to SimCLR, CLAR applies various data augmentations to the same audio clip, leading to a positive pair [59]. The follow-up work, CLMR [60], uses the same strategy for the music data. While in COLA, the positive pair is defined as two segments in the same audio recordings [58]. Note in both cases, for an anchor sample, any different audio clips in a mini-batch are selected as negative samples. Considering the fact that in the audio data, negative samples are possibly similar to the anchor sample in some scenarios, BYOL-A [43], the audio version of BYOL [92], removes the negative pairs in the contrastive learning.

Research efforts were also made to benefit the powerful Transformer architecture as a feature extractor for audio data. However, it brought the issue that unlike the words in a text with discrete tokens, audio frames are real-number vectors. In vq-wav2vec [61] and wav2vec 2.0 [62], the real-number hidden units of audio frames are clustered via either Gumbel-Softmax or online k-means algorithms, so as to assign a discrete token to each audio frame. With these discrete tokens,

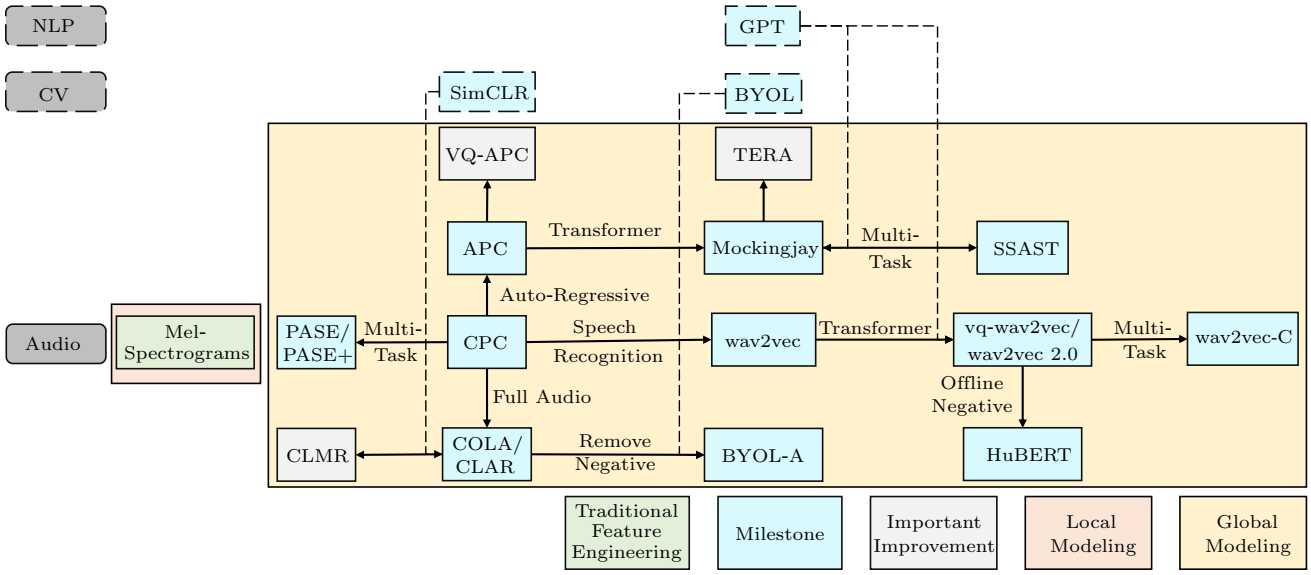


Fig.4. Diagram of the progress of SSL in the audio learning field. Note that the dashed boxes and the dashed arrows indicate the method from other research fields and the inspiration paths, respectively.

it is ready to use the BERT [4] model for SSL of audio data. The follow-up work, HuBERT, applies the offline clustering method to produce the discrete tokens [7].

5.3 Generative Modeling via Audio Reconstruction

The development of SSL in the NLP and CV fields also feeds many ideas in generative modeling for audio data. Early studies on audio SSL adopt the classic denoising autoencoder [95, 96] by embedding the input to a bottleneck hidden representation and then reconstructing the input from the hidden representation. APC [63] and VQ-APC [64] follow the line of autoregressive learning used for LM. Different from CPC [56] and wav2vec [57] that use the contrastive classification loss, APC and VQ-APC directly predict the input feature of future frames and use the ℓ_1 loss between the true feature and the predicted one. Mockingjay mimics BERT to predict the masked input feature of one frame conditioning on both past and future frames, and also uses the ℓ_1 loss between the true feature and the predicted one [11]. TERA extends Mockingjay by not only masking frames, but also masking frequencies and contaminating spectrogram with noise [65].

5.4 Multi-Task Modeling as Joint Discriminative and Generative Training

Both discriminative modeling and generative modeling boost the representation power of SSL via multi-task training. PASE [66] and its improvement PASE+ [67] jointly train a model for regression and discriminative tasks. To better preserve meaningful information in the latent space, wav2vec-C [9] was developed to reconstruct the audio signal from the latent space, in conjunction with the training target of contrastive loss in wav2vec 2.0. Splitting the audio spectrogram into patches, SSAST learns audio representations supervised by contrastive loss and generative loss in the BERT model [97].

5.5 Epilogue in Audio Processing

Given the fact that the audio data can be processed either by a sequence of frames or a spectrogram, the ideas from both the NLP and CV fields promote the development of the SSL for audio data, again showing the necessity and potential of interdisciplinary research. Although audio data is continuous by nature, currently the superior performance is still achieved with discriminative learning by constructing a pretext classification task. Generative learning alone, or combined with discriminative learning, has not yet been very successfully developed.

6 Self-Supervised Learning in Protein Learning

The protein sequence is composed of ordered amino acids sequentially, and each protein consists of 20 common types of elements and several uncommon ones [72]. The evolution process selects the protein with a suitable function, thereby biasing the protein distribution, and such distribution results in special dependencies among amino acids in protein [98]. The dependency property can be used to define the pretext tasks for SSL.

The `protein structure` results from the complicated physical and chemical interactions among amino acids, such that a protein with a specific function folds into a specific shape in space. In the protein learning community, the multiple sequence alignment (MSA) is a useful tool to identify the dependencies of the protein [99]. The `MSA` consists of a group of aligned homologous sequences, which includes the co-evolution pattern, and such co-evolution patterns can indicate dependencies. Recently, the language models built on top of `protein sequences` have also encoded the dependencies of amino acids within a protein sequence. (Refer to Fig.5 for an illustration of concepts in protein data.)

Therefore, the `protein structure`, `MSA`, as well as the `protein sequence`, can be used to identify the dependencies in protein, and the mapping function can be learned by SSL. Its progress is suggested in Fig.6.

6.1 Traditional Feature Engineering

The initial method, namely Protein Coding Features (PCF), to represent the protein structure is determined by the protein sequence of amino acid residues [68]. That is, each amino acid is encoded by a one-hot vector, with only the element of the amino acid being a non-zero value. The MSA information [99] can also be leveraged to represent the protein features. For example, the PSSM for homologous sequences calculates the substitution log-likelihood of the occurrence per amino acid at each position [69].

6.2 Local Modeling and Discriminative Modeling by Using Amino Acids

The protein data is a sequence of amino acids, which is similar to the language data. That said, creating discriminative protein representation can follow the success in the NLP field, such that the

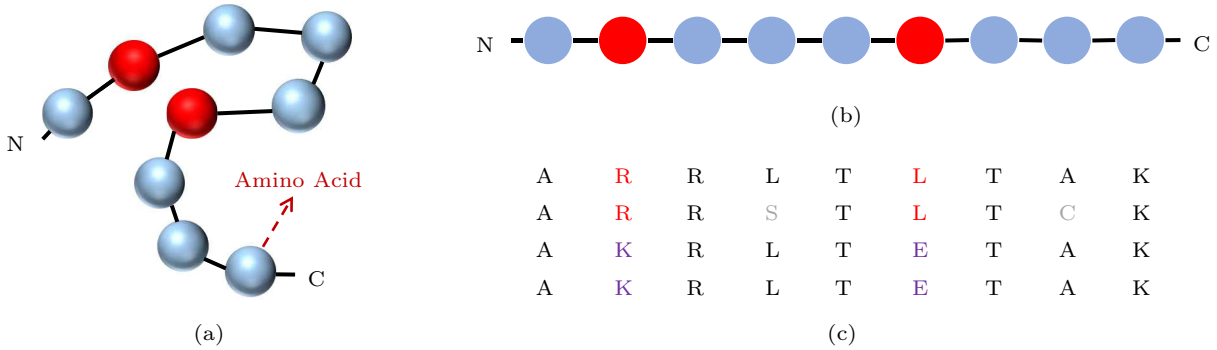


Fig.5. Illustration of basic concepts of the protein data. (a) Protein structure. (b) Protein sequence. (c) Multiple sequence alignment (MSA). In (a), the basic component of the protein data is an amino acid. In (c), the co-evolution occurs in two spatial neighbored amino acids (see two red amino acids in (a)), and MAS can indicate the co-evolution of the protein.

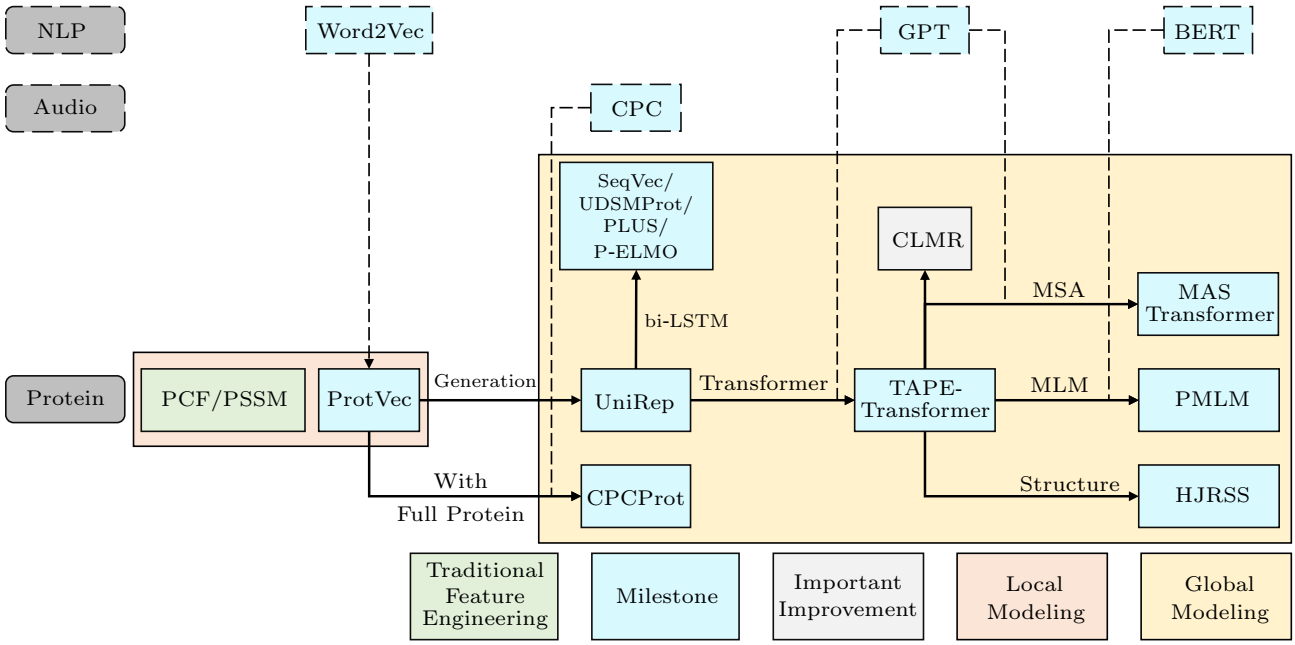


Fig.6. Diagram of the progress of SSL in the protein learning field. Note that the dashed boxes and the dashed arrows indicate the methods from other research fields and the inspiration paths, respectively.

methodology includes learning the local amino acids feature and the global protein context feature. Inspired by Word2Vec [10], its local modeling is studied in ProtVec [70]. ProtVec groups every three contiguous amino acids of the protein as a word and employs the Word2Vec technique to train a context-independent embedding of the protein. Following the development of representation learning in NLP and CV fields, the global modeling of protein data is also studied. In the discriminative modeling, the contrastive scheme benefits the protein data to establish its representation via optimizing the infoNCE objective in CPCProt [71]. Following the framework of the CPC contrastive model [56], CPCProt maximizes the MI between embeddings of a protein fragment (i.e., local feature) and its context (i.e., global feature). The only difference here is that CPCProt replaces the image patch with a certain number of amino acids.

6.3 Generative Modeling by Treating Each Amino Acid as a Word

In protein learning, generative modeling dominates the community, probably attributed to the fact that it is difficult to define a positive sample for an anchor protein. In generative modeling, many ideas are derived from the NLP field. Following the format of language data, protein, a sequence of amino acids, can be modeled as a sequence of tokens and the sequence contains the long-range dependencies of protein.

UniRep [73] takes a single amino acid as a word and uses the multiplicative LSTM [100] to train a generative model in an auto-regressive manner [101]. Its further improvement of the representation power adopts the bi-LSTM models, such as SeqVec [74], UDSMProt [75], PLUS [76] and P-ELMO [102]. The parallel computation, realized by Transformer, is also studied in TAPE [72] and more variants of Transformer-type models are investigated in ESM [77] and ProtTrans [78].

Inspired by the MLM protocol in NLP pre-training, He *et al.* proposed the Pairwise Masked Language Model (PMLM) and empirically justified that the pre-training model incorporating PMLM is particularly good at capturing co-evolutionary dependencies [79]. This improvement is mainly attributed to the fact that the model of the joint probability of a pair of masked tokens is much more delicate than the product of the probability of a single masked token in the conventional masked language model.

Considering the structure information in the model, a de-noising task is defined in HJRSS [80] as a pretext task for pre-training, where a network is trained to recover both the token and the structure from the masked tokens and the disturbed structure.

To understand more dependencies of amino acids, one can also resort to MSA information. In doing so, an MSA transformer is trained under the MLM protocol [81]. In the MSA transformer, the dependencies on all amino acids in sequences of an MSA are built by the axis attention.

6.4 Epilogue in Protein Learning

As we can see, SSL methods for protein modeling largely follow the development of SSL in NLP. This is easy to understand as proteins are the language of nature and they are also one-dimensional sequential data. However, different from human language, proteins have structures and MSA. How to leverage this additional information would be much more interesting than simply applying SSL methods in language modeling to protein modeling. We believe that MSA transformer [81] is just a beginning, and we are looking forward to more exciting breakthroughs.

7 Related Work

SSL has been the choice of learning representations for various formats of data in the learning community, and their research progress has been extensively summarized in a number of survey papers [1, 3, 12, 103]. In this section, we will briefly introduce the existing survey work for inter-disciplines, i.e., NLP, CV, graph learning, protein learning and audio processing.

The work in [1] takes a look into these methodologies of SSL and groups them into three categories: generative SSL, contrastive SSL, and generative-contrastive SSL. Following this categorization, the SSL on the applications of CV, NLP, and graph is considered in the survey. Employing SSL as means of model pre-training, a recent work [12] established a hands-on guide for understanding, using, and developing pre-trained models (PTM) on various NLP tasks. Another important component of PTM, network models, is reviewed in [3], where the training objective, model architectures, over-parameterization issue, etc., are thoroughly introduced over BERT-like architectures. A unified framework using contrastive learning as the objective for representations is surveyed in [104]. A new promising paradigm, dubbed prompt learning, is systematically studied in [105]. The success of Transformer in NLP also inspires the researchers in CV to develop a better visual feature extractor, i.e., Vision Transformer (ViT), and their progress is reported in the latest manuscripts [106–108].

The progress of SSL on the graph-structured data is also studied in many articles [103, 109–111]. The survey provides comprehensively studied mainstream learning settings in graph neural networks (GNNs), i.e., supervised learning, self-supervised learning, and semi-supervised learning [109]. In [110], Xie *et al.* summarized the SSL in GNNs and split the methodologies into two groups, namely, the contrastive model and the predictive model. The superiority of SSL in GNN is justified in [111] that SSL brings better generalization and robustness to GNNs. A deep understanding of the training methods w.r.t. different pretext tasks on graph-structured data is also empirically evaluated [103].

Endowing the capacity of identifying the protein sequences with optimized properties to AI tools also gains increasing interest in the biological field, and the learning methodologies using deep neural networks are also extensively surveyed [112, 113]. Targeting the goal to generate protein sequences, the articles [114, 115] summarize the methods of generative models.

Remark 1. *In contrast to existing work, our survey wants to thin the existing surveys, and mainly focuses on the milestone work in SSL, thereby building the connection to the dots. To be specific, the difference can be summarized as follows.*

- *First, most surveys presented the development*

of SSL in an individual field (e.g., natural language processing[2]*, computer vision*[116]*, graph learning*[103, 109–111] *or protein learning*[112–115]*), or only a few fields*[1, 104, 117]*, while in our work, we thoroughly discussed the SSL over multiple disciplines (see the comparison in Table 2).*

• *Second, existing surveys comprehensively presented the papers, making them lengthy and difficult to digest. In contrast, our survey only selected a handful of milestone papers and important work from each field, making our survey easy to understand and the development path clear.*

• *Third, instead of merely listing or categorizing the papers in existing surveys, our article also connected the main ideas via inter-disciplines, such that readers can understand how SSL evolves and how different research fields inspire each other.*

8 Discussions and Future Directions

In this section, we would like to discuss the main challenges and potential solutions for SSL.

Network Architecture and Knowledge Transfer. Recent studies have showed that the Transformer-type architecture consistently improves SSL in different fields. However, the success of Transformer-type architecture relies on the heavy parameters of the model. For example, the parameter size of GPT-3 is up to 175 billion for language understanding models[26] and the parameter number of DeepNet is up to 3.8 billion for vision tasks[120]. Thus deploying such a large model on mobile devices is not easy. That said, it is necessary to develop efficient architectures, e.g., neural architecture search (NAS), or algorithms, e.g., knowledge distillation, network pruning, to leverage the knowledge from large models. Also, to address the issue of out-of-date knowledge in machine[121, 122], it is also useful to develop self-supervised continual learning algorithms that can endow the model to learn knowledge in a lifelong manner.

Pre-Training Tasks. Recent advances of SSL are converged to the generative modeling, e.g., GPT-3[26] in NLP, MAE[6] in CV, or GPT-GNN in graph learning[54], and gained considerable achievements. Nevertheless, the SOTA pre-training strategies require either deeper architecture or large-scale data, resulting in expensive training cost. To mitigate this issue, it is possible to investigate efficient pre-training tasks, like ELECTRA[22]. In addition, another promising avenue to improve the model efficiency is to align models with user intent, e.g., InstructGPT[123], such that the aligned model can save parameters while reaching a good performance, which is on par with large-scale models.

Model Reliability. The model reliability has been a big issue in the deep learning community since the decision-making process of such deep architectures is non-transparent, such that the understanding of unseen sample for the pre-trained model is unpredictable, limiting its deployment in real practice. The models'

Table 2. Overview of Differences of Our Work and Existing Surveys

Survey	Year	Natural Language Processing	Computer Vision	Graph Learning	Audio Processing	Protein Learning
Liu *et al.*[1]	2021	✓	✓	✓	–	–
Han *et al.*[2]	2021	✓	–	–	–	–
Rogers *et al.*[3]	2020	–	–	–	–	–
Qiu *et al.*[12]	2020	✓	–	–	–	–
Jin *et al.*[103]	2020	–	–	✓	–	–
Le-Khac *et al.*[104]	2020	✓	✓	✓	✓	–
Waikhom and Patgiri[109]	2021	–	–	✓	–	–
Xie *et al.*[110]	2021	–	–	✓	–	–
You *et al.*[111]	2020	–	–	✓	–	–
Gao *et al.*[112]	2020	–	–	–	–	✓
Defresne *et al.*[113]	2021	–	–	–	–	✓
Strokach and Kim[114]	2021	–	–	–	–	✓
Wu *et al.*[115]	2021	–	–	–	–	✓
Jing and Tian[116]	2021	–	✓	–	–	–
Mao[117]	2020	✓	✓	–	✓	–
Jaiswal *et al.*[118]	2021	✓	✓	–	–	–
Liu *et al.*[119]	2021	–	–	✓	–	–
Ours	2022	✓	✓	✓	✓	✓

reliability can be improved by adversarial attacks or adversarial defenses. While it still remains an open problem, requiring further studies to understand the model and improve its robustness.

9 Conclusions

Self-supervised learning (SSL) is an important step in the road to improving the understanding of AI machines. The research community made numerous efforts to push the boundary of development, recorded by hundreds of publications. It is not easy for researchers, especially beginners, to follow and understand the progress in their own subjects. In this brief article, we built a path of the important dots in SSL development on various data, i.e., text, image, and graph. This not only showed the progress of SSL in each subject but also clearly explored the interaction of the development between subjects, e.g., the Transformer architecture invented in the NLP field inspired the development of ViT in the CV field, or the contrastive learning pipeline in CV field can also be extended in the graph/audio learning field. Beyond the high-level picture of SSL, we also believe that the development of individual subjects can be inspired by other subjects, and the research over cross-subjects is a useful way to produce impact work.

Open Access

References

[1] Liu X, Zhang F, Hou Z, Mian L, Wang Z, Zhang J, Tang J. Self-supervised learning: Generative or contrastive. *IEEE Transactions on Knowledge and Data Engineering*. DOI: 10.1109/TKDE.2021.3090866.

[2] Han X, Zhang Z, Ding N *et al.* Pre-trained models: Past, present and future. *AI Open*, 2021, 2: 225-250. DOI: 10.1016/j.aiopen.2021.08.002.

[3] Rogers A, Kovaleva O, Rumshisky A. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 2020, 8: 842-866. DOI: 10.1162/tacl_a_00349.

[4] Devlin J, Chang M W, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, June 2019, pp.4171-4186. DOI: 10.18653/v1/N19-1423.

[5] Chen M, Radford A, Child R, Wu J, Jun H, Luan D, Sutskever I. Generative pretraining from pixels. In *Proc. the 37th International Conference on Machine Learning*, July 2020, pp.1691-1703.

[6] He K, Chen X, Xie S, Li Y, Dollár P, Girshick R. Masked autoencoders are scalable vision learners. arXiv:2111.06377v3, 2021. https://arxiv.org/abs/2111.06377, December 2021.

[7] Hsu W N, Bolte B, Tsai Y H H, Lakhotia K, Salakhutdinov R, Mohamed A. HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2021, 29: 3451-3460. DOI: 10.1109/TASLP.2021.3122291.

[8] Doersch C, Gupta A, Efros A A. Unsupervised visual representation learning by context prediction. In *Proc. the 2015 IEEE International Conference on Computer Vision*, December 2015, pp.1422-1430. DOI: 10.1109/ICCV.2015.167.

[9] Sadhu S, He D, Huang C W, Mallidi S H, Wu M, Rastrow A, Stolcke A, Droppo J, Maas R. wav2vec-C: A self-supervised model for speech representation learning. In *Proc. the 22nd Annual Conference of the International Speech Communication Association*, August 30-September 3, 2021, pp.711-715. DOI: 10.21437/Interspeech.2021-717.

[10] Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. arXiv:1301.3781v3, 2013. https://arxiv.org/abs/1301.3781, December 2021.

[11] Liu A T, Yang S W, Chi P H, Hsu P C, Lee H Y. Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders. In *Proc. the 2020 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2020. DOI: 10.1109/ICASSP40776.2020.9054458.

[12] Qiu X, Sun T, Xu Y, Shao Y, Dai N, Huang X. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 2020, 63(10): 1872-1897. DOI: 10.1007/s11431-020-1647-3.

[13] Harris Z S. Distributional structure. *Word*, 1954, 10(2/3): 146-162. DOI: 10.1080/00437956.1954.11659520.

[14] Rajaraman A, David Ullman. Mining of Massive Datasets. Cambridge University Press, 2011.

[15] Collobert R, Weston J, Bottou L, Karlen M, Kavukcuoglu K, Kuksa P. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 2011, 12: 2493-2537.

[16] Dai A M, Le Q V. Semi-supervised sequence learning. In *Proc. the 28th International Conference on Neural Information Processing Systems*, December 2015, pp.3079-3087.

[17] Peters M E, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L. Deep contextualized word representations. In *Proc. the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, June 2018, pp.2227-2237. DOI: 10.18653/v1/N18-1202.

[18] Howard J, Ruder S. Universal language model fine-tuning for text classification. In *Proc. the 56th Annual Meeting of the Association for Computational Linguistics*, July 2018, pp.328-339. DOI: 10.18653/v1/P18-1031.

[19] Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Stoyanov V. RoBERTa: A robustly optimized BERT pretraining approach. arXiv:1907.11692, 2019. https://arxiv.org/abs/1907.11692, December 2021.

[20] Yang Z, Dai Z, Yang Y, Carbonell J, Salakhutdinov R R, Le Q V. XLNet: Generalized autoregressive pretraining for language understanding. In *Proc. the 33rd International Conference on Neural Information Processing Systems*, December 2019, pp.5754-5764.

[21] Lan Z, Chen M, Goodman S, Gimpel K, Sharma P, Soricut R. ALBERT: A lite BERT for self-supervised learning of language representations. In *Proc. the 8th International Conference on Learning Representations*, April 2020.

[22] Clark K, Luong M T, Le Q V, Manning C D. ELECTRa: Pre-training text encoders as discriminators rather than generators. In *Proc. the 8th International Conference on Learning Representations*, April 2020.

[23] He P, Liu X, Gao J, Chen W. DeBERTa: Decoding-enhanced BERT with disentangled attention. In *Proc. the 9th International Conference on Learning Representations*, May 2021.

[24] Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, Zhou Y, Li W, Liu P J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 2020, 21: Article No. 140.

[25] Radford A, Narasimhan K, Salimans T, Sutskever I. Improving language understanding by generative pre-training. Technical Report, OpenAI, 2018. https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf, December 2021.

[26] Brown T, Mann B, Ryder N *et al.* Language models are few-shot learners. In *Proc. the Annual Conference on Neural Information Processing Systems*, December 2020, pp.1877-1901.

[27] Lowe D G. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2004, 60(2): 91-110. DOI: 10.1023/B:VISI.0000029664.99615.94.

[28] Dalal N, Triggs B. Histograms of oriented gradients for human detection. In *Proc. the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 2005, pp.886-893. DOI: 10.1109/CVPR.2005.177.

[29] Bay H, Ess A, Tuytelaars T, Gool L V. Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 2008, 110(3): 346-359. DOI: 10.1016/j.cviu.2007.09.014.

[30] Dosovitskiy A, Fischer P, Sringenberg J T, Riedmiller M, Box T. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016, 38(9): 1734-1747. DOI: 10.1109/TPAMI.2015.2496141.

[31] Noroozi M, Pirsiavash H, Favaro P. Representation learning by learning to count. In *Proc. the International Conference on Computer Vision*, October 2017, pp.5898-5906. DOI: 10.1109/ICCV.2017.628.

[32] Noroozi M, Favaro P. Unsupervised learning of visual representations by solving jigsaw puzzles. In *Proc. the 14th European Conference on Computer Vision*, October 2016, pp.69-84. DOI: 10.1007/978-3-319-46466-4_5.

[33] Wang X, Gupta A. Unsupervised learning of visual representations using videos. In *Proc. the IEEE International Conference on Computer Vision*, December 2015, pp.2794-2802. DOI: 10.1109/ICCV.2015.320.

[34] Zhou T, Brown M, Snavely N, Lowe D G. Unsupervised learning of depth and ego-motion from video. In *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, July 2017, pp.1851-1860. DOI: 10.1109/CVPR.2017.700.

[35] Hjelm R D, Fedorov A, Lavoie-Marchildon S, Grewal K, Bachman P, Trischler A, Bengio Y. Learning deep representations by mutual information estimation and maximization. In *Proc. the 7th International Conference on Learning Representations*, May 2019.

[36] Bachman P, Hjelm R D, Buchwalter W. Learning representations by maximizing mutual information across views. arXiv:1906.00910, 2019. https://arxiv.org/abs/1906.00910, December 2021.

[37] Tian Y, Krishnan D, Isola P. Contrastive Multiview coding. In *Proc. the 16th European Conference on Computer Vision*, August 2020, pp.776-794. DOI: 10.1007/978-3-030-58621-8_45.

[38] He K, Fan H, Wu Y, Xie S, Girshick R. Momentum contrast for unsupervised visual representation learning. In *Proc. the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2020, pp.9726-9735. DOI: 10.1109/CVPR42600.2020.00975.

[39] Chen T, Kornblith S, Norouzi M, Hinton G. A simple framework for contrastive learning of visual representations. In *Proc. the 37th International Conference on Machine Learning*, July 2020, pp.1597-1607.

[40] Chen X, Fan H, Girshick R, He K. Improved baselines with momentum contrastive learning. arXiv:2003.04297, 2020. https://arxiv.org/pdf/2003.04297.pdf, December 2021.

[41] Chen X, He K. Exploring simple Siamese representation learning. In *Proc. the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2021, pp.15750-15758. DOI: 10.1109/CVPR46437.2021.01549.

[42] Caron M, Touvron H, Misra I, Jégou H, Mairal J, Bojanowski P, Joulin A. Emerging properties in self-supervised vision transformers. In *Proc. the 2021 IEEE/CVF International Conference on Computer Vision*, October 2021, pp.9650-9660. DOI: 10.1109/ICCV48922.2021.00951.

[43] Niizumi D, Takeuchi D, Ohishi Y, Harada N, Kashino K. BYOL for audio: Self-supervised learning for general-purpose audio representation. arXiv:2103.06695, 2021. https://arxiv.org/abs/2103.06695, December 2021.

[44] Bao H, Dong L, Wei F. BEiT: BERT pre-training of image transformers. arXiv:2106.08254, 2021. https://arxiv.org/abs/2106.08254, December 2021.

[45] Wei C, Fan H, Xie S, Wu C Y, Yuille A, Feichtenhofer C. Masked feature prediction for self-supervised visual pre-training. arXiv:2112.09133v1, 2021. https://arxiv.org/abs/2112.09133, December 2021.

[46] Perozzi B, Al-Rfou R, Skiena S. DeepWalk: Online learning of social representations. In *Proc. the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, August 2014, pp.701-710. DOI: 10.1145/2623330.2623732.

[47] Grover A. node2vec: Scalable feature learning for networks. In *Proc. the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data*, August 2016, pp.855-864. DOI: 10.1145/2939672.2939754.

[48] Veličković P, Fedus W, Hamilton W L, Liò P, Bengio Y, Hjelm R D. Deep graph infomax. In *Proc. the 7th International Conference on Learning Representations*, May 2019.

[49] Sun F Y, Hoffman J, Verma V, Tang J. InfoGraph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *Proc. the 8th International Conference on Learning Representations*, April 2020.

[50] Zhu Y, Xu Y, Yu F, Liu Q, Wu S, Wang L. Deep graph contrastive representation learning. arXiv:2006.04131, 2020. https://arxiv.org/abs/2006.04131v1, December 2021.

[51] Hassani K, Khasahmadi A H. Contrastive multi-view representation learning on graphs. In *Proc. the 37th International Conference on Machine Learning*, July 2020, pp.4116-4126.

[52] Rong Y, Bian Y, Xu T, Xie W, Wei Y, Huang W, Huang J. Self-supervised graph transformer on large-scale molecular data. In *Proc. the Annual Conference on Neural Information Processing Systems*, December 2020.

[53] Wang H, Wang J, Wang J, Zhao M, Zhang W, Zhang F, Xie X, Guo M. GraphGAN: Graph representation learning with generative adversarial nets. In *Proc. the 32nd AAAI Conference on Artificial Intelligence*, February 2018, pp.2508-2515.

[54] Hu Z, Dong Y, Wang K, Chang K W, Sun Y. GPT-GNN: Generative pre-training of graph neural networks. In *Proc. the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, August 2020, pp.1857-1867. DOI: 10.1145/3394486.3403237.

[55] Zhang B, Leitner J, Thornton S. Audio recognition using Mel spectrograms and convolution neural networks. Technical Report, Dept. Electrical and Computer Engineering, University of California. http://noiselab.ucsd.edu/ECE228_2019/Reports/Report38.pdf, December 2021.

[56] Oord A, Li Y, Vinyals O. Representation learning with contrastive predictive coding. arXiv:1807.03748, 2019. https://arxiv.org/abs/1807.03748, January 2022.

[57] Schneider S, Baevski A, Collobert R, Auli M. wav2vec: Unsupervised pre-training for speech recognition. arXiv:1904.05862, 2019. https://arxiv.org/abs/1904.05862, December 2021.

[58] Saeed A, Grangier D, Zeghidour N. Contrastive learning of general-purpose audio representations. In *Proc. the 2021 IEEE International Conference on Acoustics, Speech and Signal Processing*, June 2021, pp.3875-3879. DOI: 10.1109/ICASSP39728.2021.94135.

[59] Al-Tahan H, Mohsenzadeh Y. CLAR: Contrastive learning of auditory representations. In *Proc. the 24th International Conference on Artificial Intelligence and Statistics*, April 2021, pp.2530-2538.

[60] Spijkervet J, Burgoyne J A. Contrastive learning of musical representations. arXiv:2103.09410, 2021. https://arxiv.org/abs/2103.09410, September 2021.

[61] Baevski A, Schneider S, Auli M. vq-wav2vec: Self-supervised learning of discrete speech representations. arXiv:1910.05453, 2020. https://arxiv.org/abs/1910.05453v1, February 2022.

[62] Baevski A, Zhou H, Mohamed A, Auli M. wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Proc. the Annual Conference on Neural Information Processing Systems*, December 2020.

[63] Chung Y A, Hsu W N, Tang H, Glass J. An unsupervised autoregressive model for speech representation learning. arXiv:1904.03240, 2019. https://arxiv.org/abs/1904.03240, December 2021.

[64] Chung Y A, Tang H, Glass J. Vector-quantized autoregressive predictive coding. arXiv:2005.08392, 2020. https://arxiv.org/abs/2005.08392, December 2021.

[65] Liu A T, Li S W, Lee H y. TERA: Self-supervised learning of transformer encoder representation for speech. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2021, 29: 2351-2366. DOI: 10.1109/TASLP.2021.3095662.

[66] Pascual S, Ravanelli M, Serrá J, Bonafonte A, Bengio Y. Learning problem-agnostic speech representations from multiple self-supervised tasks. arXiv:1904.03416, 2019. DOI: https://arxiv.org/abs/1904.03416, December 2021.

[67] Ravanelli M, Zhong J, Pascual S, Swietojanski P, Monteiro J, Trmal J, Bengio Y. Multi-task self-supervised learning for robust speech recognition. In *Proc. the 2020 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2020, pp.6989-6993. DOI: 10.1109/ICASSP40776.2020.9053569.

[68] Wang S, Peng J, Ma J, Xu J. Protein secondary structure prediction using deep convolutional neural fields. *Scientific Reports*, 2016, 6(1f): Article No. 18962. DOI: 10.1038/srep18962.

[69] Altschul S F, Madden T L, Schäffer A A, Zhang J, Zhang Z, Miller W, Lipman D J. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Research*, 1997, 25(17): 3389-3402. DOI: 10.1093/nar/25.17.3389.

[70] Asgari E, Mofrad M R. Continuous distributed representation of biological sequences for deep proteomics and genomics. *PLoS One*, 2015, 10(11): Article No. e0141287. DOI: 10.1371/journal.pone.0141287.

[71] Lu A X, Zhang H, Ghassemi M, Moses A M. Self-supervised contrastive learning of protein representations by mutual information maximization. *bioRxiv*, 2020. DOI: 10.1101/2020.09.04.283929.

[72] Rao R, Bhattacharya N, Thomas N, Duan Y, Chen X, Canny J, Abbeel P, Song Y S. Evaluating protein transfer learning with tape. In *Proc. the Annual Conference on Neural Information Processing Systems*, December 2019.

[73] Alley E C, Khimulya G, Biswas S, AlQuraishi M, Church G M. Unified rational protein engineering with sequence-based deep representation learning. *Nature Methods*, 2019, 16(12): 1315-1322. DOI: 10.1038/s41592-019-0598-1.

[74] Heinzinger M, Elnaggar A, Wang Y, Dallago C, Nechaev D, Matthes F, Rost B. Modeling aspects of the language of life through transfer-learning protein sequences. *BMC Bioinformatics*, 2019, 20(1): Article No. 723. DOI: 10.1186/s12859-019-3220-8.

[75] Strodthoff N, Wagner P, Wenzel M, Samek W. UDSMProt: Universal deep sequence models for protein classification. *Bioinformatics*, 2020, 36(8): 2401-2409. DOI: 10.1093/bioinformatics/btaa003.

[76] Min S, Park S, Kim S, Choi H S, Lee B, Yoon S. Pre-training of deep bidirectional protein sequence representations with structural information. *IEEE Access*, 2021, 9: 123912-123926. DOI: 10.1109/ACCESS.2021.3110269.

[77] Rives A, Meier J, Sercu T, Goyal S, Lin Z, Liu J, Guo D, Ott M, Zitnick C L, Ma J, Fergus R. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 2021, 118(15): Article No. e2016239118. DOI: 10.1073/pnas.2016239118.

[78] 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:2007.06225, 2020. https://arxiv.org/abs/2007.06225, December 2021.

[79] He L, Zhang S, Wu L *et al.* Pre-training co-evolutionary protein representation via a pairwise masked language model. arXiv:2110.15527, 2021. https://arxiv.org/abs/2110.15527. October 2021.

[80] Mansoor S, Baek M, Madan U, Horvitz E. Toward more general embeddings for protein design: Harnessing joint representations of sequence and structure. *bioRxiv*, 2021. DOI: 10.1101/2021.09.01.458592.

[81] Rao R, Liu J, Verkuil R, Meier J, Canny J F, Abbeel P, Sercu T, Rives A. MSA transformer. *bioRxiv*, 2021. DOI: 10.1101/2021.02.12.430858.

[82] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, Kaiser Ł, Polosukhin I. Attention is all you need. In *Proc. the Annual Conference on Neural Information Processing Systems*, December 2017, pp.5998-6008.

[83] Bender E M, Gebru T, McMillan-Major A, Shmitchell S. On the dangers of stochastic parrots: Can language models be too big? In *Proc. the 2021 ACM Conference on Fairness, Accountability, and Transparency*, March 2021, pp.610-623. DOI: 10.1145/3442188.3445922.

[84] Schroff F, Kalenichenko D, Philbin J. FaceNet: A unified embedding for face recognition and clustering. In *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, June 2015, pp.815-823. DOI: 10.1109/CVPR.2015.7298682.

[85] Gutmann M, Hyvärinen A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proc. the 13th International Conference on Artificial Intelligence and Statistics*, May 2010, pp.297-304.

[86] Cordts M, Omran M, Ramos S *et al.* The Cityscapes dataset for semantic urban scene understanding. In *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, June 2016, pp.3213-3223. DOI: 10.1109/CVPR.2016.350.

[87] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks. In *Proc. the 25th International Conference on Neural Information Processing Systems*, December 2012, pp.1097-1105.

[88] Kim D, Cho D, Yoo D, Kweon I S. Learning image representations by completing damaged jigsaw puzzles. In *Proc. the IEEE Winter Conference on Applications of Computer Vision*, March 2018, pp.793-802. DOI: 10.1109/WACV.2018.00092.

[89] Wei C, Xie L, Ren X, Xia Y, Su C, Liu J, Tian Q, Yuille A L. Iterative reorganization with weak spatial constraints: Solving arbitrary jigsaw puzzles for unsupervised representation learning. In *Proc. the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2019, pp.1910-1919. DOI: 10.1109/CVPR.2019.00201.

[90] Sohn K. Improved deep metric learning with multi-class N-pair loss objective. In *Proc. the Annual Conference on Neural Information Processing Systems*, December 2016, pp.1857-1865.

[91] Khosla P, Teterwak P, Wang C, Sarna A, Tian Y, Isola P, Maschinot A, Liu C, Krishnan D. Supervised contrastive learning. In *Proc. the 34th Conference on Neural Information Processing Systems*, December 2020, pp.18661-18673.

[92] Grill J B, Strub F, Altché F *et al.* Bootstrap your own latent—A new approach to self-supervised learning. In *Proc. the Annual Conference on Neural Information Processing Systems*, December 2020, pp.21281-21284.

[93] Dosovitskiy A, Beyer L, Kolesnikov A *et al.* An image is worth 16x16 words: Transformers for image recognition at scale. In *Proc. the 9th International Conference on Learning Representations*, May 2021.

[94] Gao T, Yao X, Chen D. SimCSE: Simple contrastive learning of sentence embeddings. arXiv:2104.08821, 2021. https://arxiv.org/abs/2104.08821, December 2021.

[95] Xu Y, Huang Q, Wang W, Foster P, Sigtia S, Jackson P J B, Plumbley M D. Unsupervised feature learning based on deep models for environmental audio tagging. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2017, 25(6): 1230-1241. DOI: 10.1109/TASLP.2017.2690563.

[96] Chorowski J, Weiss R J, Bengio S, Oord A. Unsupervised speech representation learning using WaveNet autoencoders. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2019, 27(12): 2041-2053. DOI: 10.1109/TASLP.2019.2938863.

[97] Gong Y, Lai C I J, Chung Y A, Glass J. SSAST: Self-supervised audio spectrogram transformer. arXiv:2110.09784, 2021. https://arxiv.org/abs/2110.09784, October 2021.

[98] Göbel U, Sander C, Schneider R, Valencia A. Correlated mutations and residue contacts in proteins. *Proteins: Structure, Function, and Bioinformatics*, 1994, 18(4): 309-317. DOI: 10.1002/prot.340180402.

[99] Chen J, Chaudhari N S. Bidirectional segmented-memory recurrent neural network for protein secondary structure prediction. *Soft Computing*, 2006, 10(4): 315-324. DOI: 10.1007/s00500-005-0489-5.

[100] Krause B, Lu L, Murray I, Renals S. Multiplicative LSTM for sequence modelling. arXiv:1609.07959, 2016. https://arxiv.org/abs/1609.07959, December 2021.

[101] Suzek B E, Wang Y, Huang H, McGarvey P B, Wu C H, Consortium U. UniRef clusters: A comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics*, 2015, 31(6): 926-932. DOI: 10.1093/bioinformatics/btu739.

[102] Bepler T, Berger B. Learning protein sequence embeddings using information from structure. arXiv:1902.08661, 2019. https://arxiv.org/abs/1902.08661, February 2022.

[103] Jin W, Derr T, Liu H, Wang Y, Wang S, Liu Z, Tang J. Self-supervised learning on graphs: Deep insights and new direction. arXiv:2006.10141, 2020. https://arxiv.org/abs/2006.10141, December 2021.

[104] Le-Khac P H, Healy G, Smeaton A F. Contrastive representation learning: A framework and review. *IEEE Access*, 2020, 8: 193907-193934. DOI: 10.1109/ACCESS.2020.3031549.

[105] Liu P, Yuan W, Fu J, Jiang Z, Hayashi H, Neubig G. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. arXiv:2107.13586, 2021. https://arxiv.org/abs/2107.13586, December 2021.

[106] Lin T, Wang Y, Liu X, Qiu X. A survey of transformers. arXiv:2106.04554, 2021. https://arxiv.org/abs/2106.04554, December 2021.

[107] Khan S, Naseer M, Hayat M, Zamir S W, Khan F S, Shah M. Transformers in vision: A survey. arXiv:2101.01169, 2021. https://arxiv.org/abs/2101.01169, October 2021.

[108] Liu Y, Zhang Y, Wang Y, Hou F, Yuan J, Tian J, Zhang Y, Shi Z, Fan J, He Z. A survey of visual transformers. arXiv:2111.06091, 2021. https://arxiv.org/abs/2111.06091, November 2021.

[109] Waikhom L, Patgiri R. Graph neural networks: Methods, applications, and opportunities. arXiv:2108.10733, 2021. https://arxiv.org/abs/2108.10733, December 2021.

[110] Xie Y, Xu Z, Zhang J, Wang Z, Ji S. Self-supervised learning of graph neural networks: A unified review. arXiv:2102.10757, 2021. https://arxiv.org/abs/2102.10757, March. 2022.

[111] You Y, Chen T, Wang Z, Shen Y. When does self-supervision help graph convolutional networks? In *Proc. the 37th International Conference on Machine Learning*, July 2020, pp.10871-10880.

[112] Gao W, Mahajan S P, Sulam J, Gray J J. Deep learning in protein structural modeling and design. *Patterns*, 2020, 1(9): Article No. 100142. DOI: 10.1016/j.patter.2020.100142.

[113] Defresne M, Barbe S, Schiex T. Protein design with deep learning. *International Journal of Molecular Sciences*, 2021, 22(21): Article No. 11741. DOI: 10.3390/ijms222111741.

[114] Strokach A, Kim P M. Deep generative modeling for protein design. arXiv:2109.13754, 2021. https://arxiv.org/abs/2109.13754, December 2021.

[115] Wu Z, Johnston K E, Arnold F H, Yang K K. Protein sequence design with deep generative models. *Current Opinion in Chemical Biology*, 2021, 65: 18-27. DOI: 10.1016/j.cbpa.2021.04.004.

[116] Jing L, Tian Y. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021, 43(11): 4037-4058. DOI: 10.1109/TPAMI.2020.2992393.

[117] Mao H H. A survey on self-supervised pre-training for sequential transfer learning in neural networks. arXiv:2007.00800, 2020. https://arxiv.org/abs/2007.00800v1, December 2021.

[118] Jaiswal A, Babu A R, Zadeh M Z, Banerjee D, Makedon F. A survey on contrastive self-supervised learning. *Technologies*, 2021, 9(1): Article No. 2. DOI: 10.3390/technologies9010002.

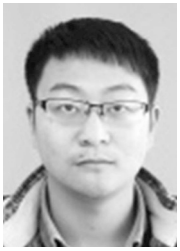
[119] Liu Y, Pan S, Jin M, Zhou C, Xia F, Yu P S. Graph self-supervised learning: A survey. arXiv:2103.00111, 2021. https://arxiv.org/abs/2103.00111, February 2022.

[120] Wang H, Ma S, Dong L, Huang S, Zhang D, Wei F. DeepNet: Scaling transformers to 1,000 layers. arXiv:2203.00555, 2022. https://arxiv.org/abs/2203.00555, March 2022.

[121] Qin Y, Zhang J, Lin Y, Liu Z, Li P, Sun M, Zhou J. ELLE: Efficient lifelong pre-training for emerging data. arXiv:2203.06311, 2022. https://arxiv.org/abs/2203.06311, March 2022.

[122] Li Z, Hoiem D. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018, 40(12): 2935-2947. DOI: 10.1109/TPAMI.2017.2773081.

[123] Ouyang L, Wu J, Jiang X. Training language models to follow instructions with human feedback. arXiv:2203.02155v1, 2022. https://arxiv.org/abs/2203.02155, March 2022.

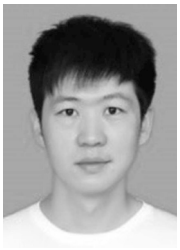


Peng-Fei Fang received his B.E. degree in automation from Hangzhou Dianzi University (HDU), Hangzhou, in 2014, and his M.E. degree in mechatronics from Australian National University (ANU), Canberra, in 2017. He is currently pursuing his joint Ph.D. degree with ANU and the Data61-CSIRO. He is also a visiting scholar with Westlake University, Hangzhou. His research interests include computer vision and machine learning.

Xian Li received his Ph.D. degree in pattern recognition and intelligence systems from University of Science and Technology of China, Hefei, in 2015. He is currently a postdoctoral researcher at Westlake University, Hangzhou. His research interests include audio self-supervised learning, speech and singing voice synthesis, and music generation.

Yang Yan is currently a Ph.D. candidate in computer science and technology in the Joint Graduate Program at Westlake University, Hangzhou, and Zhejiang University, Hangzhou. He received his M.S. and B.S. degrees in software engineering from Chongqing University of Posts and Telecommunications, Chongqing, in 2021 and 2018, respectively. His current research interests include deep learning, computer vision, and cross-modality learning.

Shuai Zhang is a Ph.D. candidate in computer science in the Joint Graduate Program at Westlake University, Hangzhou, and Zhejiang University, Hangzhou, currently. He received his M.S. and B.S. degrees in mathematics and applied mathematics from University of Chinese Academy of Sciences, Beijing, in 2017, and University of Science and Technology Beijing, Beijing, in 2014, respectively. His research interests include representation learning and natural language processing.



Qi-Yue Kang received his B.S. and Ph.D. degrees in chemistry and environmental geography from China Agricultural University, Beijing, and Peking University, Beijing, in 2016 and 2021, respectively. He is a postdoctoral fellow in the School of Engineering, Westlake University, Hangzhou. His current research interests include prediction of molecular properties and protein variation effects.

Xiao-Fei Li received his Ph.D. degree from Peking University, Beijing, in July 2013. He is currently an assistant professor with Westlake University, Hangzhou. Before, he was with INRIA Grenoble Rhône-Alpes, France, as a postdoctoral researcher from February 2014 to January 2016, and as a starting research scientist from February 2016 to December 2019. His research interests lie in the field of acoustic, audio and speech signal processing, including the topics of speech denoising, dereverberation, separation and localization, sound/speech semi-supervised and self-supervised learning, sound field reproduction, and personal sound zone.

Zhen-Zhong Lan received his Ph.D. degree from Carnegie Mellon University, Pittsburgh, in May 2017. He is currently an assistant professor with Westlake University, Hangzhou. From 2018 to 2020, he worked at Google AI, Los Angeles, as an NLP researcher. His current research interests lie in data-driven video and natural language understanding. He is also a member of the committees for several of the premier computer vision and multimedia conferences, including CVPR, ICCV, ECCV, and ACM MM.