self-supervised learning

Authors

Alexander Jung*, Department of Computer Science, Aalto University, Espoo, Finland
*Corresponding author: alex.jung@aalto.fi

Abstract

Self-supervised learning constructs the labels of a training set from the data points themselves. Some features of a data point are withheld and used as its label while the rest remain features, so what results is an ordinary supervised learning problem, solved by empirical risk minimization (ERM) on data that no annotator has touched. In natural language processing (NLP) the withheld feature is the next token, which is how a large language model (LLM) is trained. In computer vision it is a set of pixels, deleted from an image and predicted from the patches left visible.

Definition

An open image database returns thousands of photographs for a keyword such as *cat*, and a web crawl returns billions of sentences. Neither comes with a label saying what should be predicted. A prediction task can be made from any one of them: hide the last word of a sentence, and the words before it become a question whose answer is already known. Self-supervised learning constructs labels this way, from the data point itself. de Sa (1993) uses the term for a classifier whose label for one sensory modality is derived from a co-occurring input in another.

Some of the features of a data point are withheld and used as the label; the remaining ones stay as the features. The result is an ordinary supervised learning problem, solved by empirical risk minimization (ERM), and its training set needs nothing beyond the raw data. A single unlabeled collection yields as many training sets as there are ways of hiding part of a data point (see Fig. 1).

In natural language processing (NLP) a data point is a stretch of text, its features are the tokens $v^{(1)}, \ldots, v^{(n)}$ and its label is the token $v^{(n+1)}$ that follows. Every position in every document is one labeled data point, so a corpus of m tokens gives on the order of m of them. This is how a large language model (LLM) is trained (Brown et al., 2020). Masking a token in the middle and predicting it from both sides withholds a different feature of the same data point (Devlin et al., 2019).

In computer vision the withheld features are pixels: an image is split into patches, most of them are deleted, and the method predicts the missing pixel values from those that remain. He et al. (2022) delete 75% of the patches and still recover recognizable content. A deep net solving either task computes a composition $\widehat{h} = s \circ \phi$, where the feature transformation $\phi\colon \mathcal{X} \to \mathbb{R}^d$ sends a data

point to a vector of activations and the map s sends that vector to the withheld feature. The withheld part can be predicted only from what ϕ has kept: a next token cannot be recovered from a vector that has discarded the context, nor a deleted patch from one that has discarded its surroundings. Fitting the composition is therefore what shapes ϕ, even though ϕ appears nowhere in the loss. He et al. (2022), Sect. 3 name the two halves: their encoder is ϕ and sees only the patches left visible, while their decoder is s and rebuilds the pixels from its output. What restricts ϕ there is the deletion of most of the input, not a narrow layer.

In both domains it is ϕ that is carried over. The map s is dropped, and a small replacement for it is fitted on the output of ϕ using the few labels the later task does have (see transfer learning, foundation model). Self-supervised learning is therefore one answer to the same shortage that motivates semi-supervised learning (SSL): there, unlabeled data points assist a label-scarce problem directly; here, they are turned into a labeled problem of their own first.

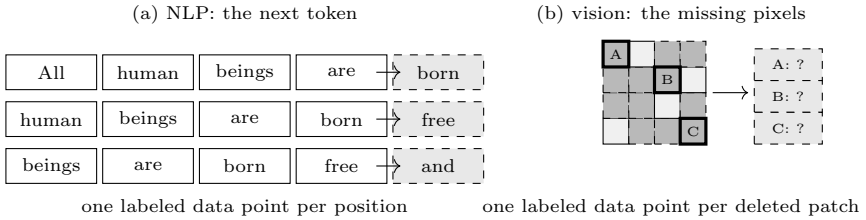

Figure 1: Two ways of constructing labels from unlabeled data points, and in both a single source yields many. (a) NLP: a sentence is read through a sliding window, and every position gives a labeled data point whose features are the tokens in the window and whose label is the token after it (shaded). (b) Computer vision: an image is split into patches and most are deleted (shaded); each deleted patch, three of them outlined here, is one labeled data point whose label is its own pixel values. Neither task needs a human annotator

Cross References

feature, label, supervised learning, SSL, LLM, NLP, token, transfer learning, foundation model

References

1. Brown et al. (2020). Language Models are Few-Shot Learners. In: Adv. Neural Inf. Process. Syst., pp. 1877–1901.
2. Devlin et al. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Proc. 2019 Conf. North American Chapter Assoc. Computational Linguistics: Human Lang. Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186.

3. He et al. (2022). Masked Autoencoders Are Scalable Vision Learners. In: 2022 IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 15979–15988.

4. de Sa (1993). Learning Classification with Unlabeled Data. In: Adv. Neural Inf. Process. Syst. (NIPS), pp. 112–119.