A Tutorial on Energy-Based Learning

Chunpai Wang

August 30, 2018

Energy-Based Models (EBMs) capture dependencies between variables by association a scalar energy to each configuration of the variables.

- **Inference** consists in fixing the value of observed variables and finding configurations of the remaining variables that minimize the energy. For example, the variables X are observed, and we would like to find the configurations of labels Y, such that $E(X, Y)$ is minimum.
- **Learning** consists in finding an energy function in which observed **configurations** of the variables are given lower energies than unobserved ones. Why ? How could you know the energy of unobserved variables ? We would like the observed configuration to be the lowest, otherwise, there exists a better ground-truth labels, where we assume nothing wrong on our ground-truth labels labeled by human.
- A **loss functional**, minimized during learning, is used to measure the quality of the available energy functions. We can use different energy functions and loss functionals for different problems.

Advantages of EBMs:

- Probabilistic models must be properly normalized, which sometimes requires evaluating intractable integrals over the space of all possible variable configurations. **Since EBMs have no requirement for proper normalization, this problem is naturally circumvented.** For example, the loopy belief propagation can be understood as minimizing the Bethe Approximation on Gibbs Free energy.
- EBMs can be viewed as **a form of non-probabilistic factor graphs**, and they provide considerably more flexibility in the design of architectures and training criteria than probabilistic approaches.
- Many popular linear models can be reformulated in the EBM framework, such as max-margin Markov networks and conditional random fields.

1 Energy-Based Inference

Once we have learned the energy function $E(Y, X)$, which measures the "goodness" of each possible configuration of X and Y, we can find the optimal configurations for unobserved Y given the observed input X that minimize the $E(Y, X)$:

$$Y^* = \arg\min_{Y \in \mathcal{Y}} E(Y, X) \tag{1}$$

When the size of the set $\mathcal{Y}$ is small, we can simply compute $E(Y, X)$ for all possible values of $Y \in \mathcal{Y}$ and pick the smallest. When the cardinality or dimension of Y is large, exhaustive search impractical, a specific strategy, called the *inference procedure*, must be employed to find the Y that minimize $E(Y, X)$, which has no guarantee on global optimal solution. We can use "smart" inference procedures, such as min-sum, Viterbi, min-cut, belief propagation, gradient descent, etc ..

Understanding the structure is a global decision in which several local decisions play a role but there are mutual dependencies on their outcome. It is essential to make coherent decisions in a way that takes the interdependencies of variables Y into account. CRFs allow manually define some feature functions that interconnect the unobserved variables, or capture the global structure, right ? From the perspective of energy-based learning, we can view that

2 Converting Energies to Probabilities

Energies are uncalibrated measured in arbitrary units, thus the energies of two separately-trained systems cannot be combined directly. But we can calibrate energies by turning them into probabilities first. The simplest and most common method for turning a collection of arbitrary energies into a collection of numbers between 0 and 1 whose sum (or integral) is 1 is through the Gibbs distribution.

$$P(Y|X) = \frac{e^{-\beta E(Y,X)}}{\int_{y \in \mathcal{Y}} e^{-\beta E(Y,X)}} \tag{2}$$

where β is an arbitrary positive constant. It should be noted that sometimes the denominator is intractable, thus we should avoid probabilistic modeling when the application does not require. Energy model is more general than probabilistic model.

3 Energy-Based Training

3.1 Architecture

We are given a set of training data $S = \{(X^i, Y^i) : i = 1...P\}$, where X^i is the input for $i-th$ training sample, and Y^i is the corresponding desired answer. Training is to find a best energy function in a family of functions $\mathcal{E}$, indexed by a parameter W

$$\mathcal{E} = \{E(W, Y, X) : W \in \mathcal{W}\} \tag{3}$$

How can we decide which energy function is the best ? We need a way to assess the quality of any particular energy function, based solely on two elements: **the training set, and our prior knowledge about the task**. The quality measure is called the $loss functional$ and denoted $\mathcal{L}(E, S)$. For example, the architecture of the neural network decide the family of function, and we can decide the loss function as $L(W, S)$. The learning problem is simply to find the W that minimizes the loss

$$W^* = \min_{W \in \mathcal{W}} \mathcal{L}(W, S) \tag{4}$$

For most cases, the loss functional is defined as follows:

$$\mathcal{L}(E, S) = \frac{1}{P} \sum_{i=1}^{P} L(Y^i, E(W, \mathcal{Y}, X^i)) + R(W) \tag{5}$$

where $E(W, \mathcal{Y}, X^i)$ is the predicted output and Y^i denotes the desired output, and $R(W)$ is the regularizer, which can be used to embed our prior knowledge about which energy functions in our chosen family are preferable to others before we see the training data. What are the prior knowledge on the neural network model ? Fine-tuning on the pre-trained model ?

3.2 Loss Functional

A loss functional should be designed in a way such that energy functions that give the lowest energy to the correct answer and higher energy to all other (incorrect) answers. Conversely, energy functions that do not assign the lowest energy to the correct answers would have a high loss. Since the inference algorithm selects the Y with the lowest energy, the learning procedure must shape the energy surface so that the desired value of Y has lower energy than all other (undesired) values.

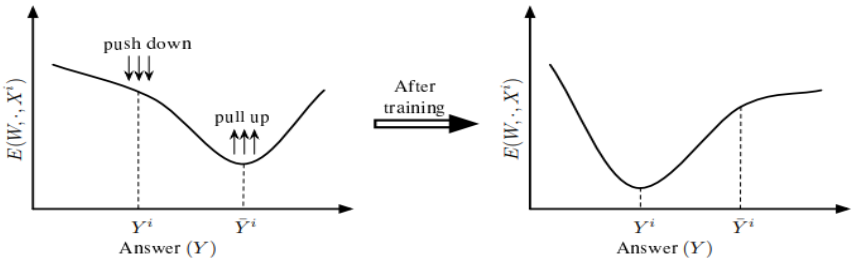

Figure 1: Y^i denotes by the correct answer; $\bar{Y}^i$ denotes by the most offending incorrect answer, i.e. the answer that has the lowest energy among all the incorrect answers. To define this answer in the continuous case, we can simply view all answers within a distance ϵ of Y^i as correct, and all answers beyond that distance as incorrect. With a properly designed loss function, the learning process will push down on $E(W, Y^i, X^i)$ and pull up on the incorrect energies, particularly on $E(W, \bar{Y}^i, X^i)$. **However, the loss function must satisfy some conditions in order to be guaranteed to shape the energy surface correctly.**

When we optimize an energy-based model, minimizing the energy of a given data is usually inappropriate. In particular, such an objective function may be unbounded. It may not distinguish two patterns, one is good and the other is very good, as both of the two patterns have the minimum energy.
An objective function of an energy-based model should have a contrastive term, which natually appear in the objective function of a probabilistic model. For example, to maximize the average log-likelihood or minimize

the average negative log-likelihood of a set of configurations D with respect to a Boltzmann machine is given by

$$-\frac{1}{|D|}\sum_{\mathbf{x}\in D}\log P_\theta(\mathbf{x}) = \frac{1}{|D|}\sum_{\mathbf{x}\in D}E_\theta(\mathbf{x}) - \log\sum_{\tilde{\mathbf{x}}}(-E_\theta(\tilde{\mathbf{x}})) \tag{6}$$

where the second term is a contrastive term. In particular, to minimize this objective function, we should not only reduce the energy of the patterns in D but also increase the energy of the patterns not in D. However, the summation over exponential configurations, and is usually intractable.

3.3 Four-Main Components

Given a training set $\mathcal{S}$, building and training an energy-based model involves designing four components:

1. The architecture: the internal structure of $E(W, Y, X)$
2. The inference algorithm: the method for finding a value of Y that minimizes $E(W, Y, X)$ for any given X
3. The loss function: $\mathcal{L}(\mathcal{W}, \mathcal{S})$ measures the quality of an energy function using the training set.
4. The learning algorithm: the method for finding W that minimizes the loss functional over the family of energy functions $\mathcal{E}$, given the training set.

References

[1] Yann LeCun's Tutorial https://www.youtube.com/watch?v=LK7OEgrxJiY

[2] Osogami, Takayuki. "Boltzmann machines and energy-based models." arXiv preprint arXiv:1708.06008 (2017).