

An Overview and Perspectives On Bidirectional Intelligence: Lmsr Duality, Double IA Harmony, and Causal Computation

Lei Xu, *Fellow, IEEE*

Abstract—Advances on bidirectional intelligence are overviewed along three threads, with extensions and new perspectives. The first thread is about bidirectional learning architecture, exploring five dualities that enable Lmsr six cognitive functions and provide new perspectives on which a lot of extensions and particularly flexible Lmsr are proposed. Interestingly, either or two of these dualities actually takes an important role in recent models such as U-net, ResNet, and DenseNet. The second thread is about bidirectional learning principles unified by best yIng-yAng (IA) harmony in BYY system. After getting insights on deep bidirectional learning from a bird-viewing on existing typical learning principles from one or both of the inward and outward directions, maximum likelihood, variational principle, and several other learning principles are summarised as exemplars of the BYY learning, with new perspectives on advanced topics. The third thread further proceeds to deep bidirectional intelligence, driven by long term dynamics (LTD) for parameter learning and short term dynamics (STD) for image thinking and rational thinking in harmony. Image thinking deals with information flow of continuously valued arrays and especially image sequence, as if thinking was displayed in the real world, exemplified by the flow from inward encoding/cognition to outward reconstruction/transformation performed in Lmsr learning and BYY learning. In contrast, rational thinking handles symbolic strings or discretely valued vectors, performing uncertainty reasoning and problem solving. In particular, a general thesis is proposed for bidirectional intelligence, featured by BYY intelligence potential theory (BYY-IPT) and nine essential dualities in architecture, fundamentals, and implementation, respectively. Then, problems of combinatorial solving and uncertainty reasoning are investigated from this BYY IPT perspective. First, variants and extensions are suggested for AlphaGoZero like searching tasks, such as traveling salesman problem (TSP) and attributed graph matching (AGM) that are turned into Go like problems with help of a feature enrichment technique. Second, reasoning activities are summarized under guidance of BYY IPT from the aspects of constraint satisfaction, uncertainty propagation, and path or tree searching. Particularly, causal potential theory is proposed for discovering causal direction, with two roads developed for its implementation.

Index Terms—Autoencoder, LMSER, duality, outward attention, associative recall, concept formation, imagining, pattern transformation, STD vs LTD, RPCL, skip connection, feedback, variational, least redundancy, Bayesian Ying Yang, IA system, best harmony, best matching, image thinking, rational thinking, intelligence potential theory, Alpha-TSP, Alpha-AGM, graph matching, ME Player, BYY Follower, constraint satisfaction, causal potential theory.

I. INTRODUCTION

Bidirectional deep learning is featured by using an outbound deep neural networks to generate desired patterns, while the generative networks are driven by inner code or representation by an inbound deep neural network with its inputs in patterns that are either similar to or simpler than patterns that are generated. Typically, bidirectional deep learning performs various transformations, such as language to language, text to image, text to sketch, sketch to image, image to image, 2D image to 3D image, past to future, image to caption, image to sentence, music to dance,..., etc, on which there has been a wave of ever increasing interests in recent years.

Efforts of bidirectional deep learning can be backtracked to the later eighties and the early nineties in the last century. Autoencoder (or also called auto-association) [1] and Lmsr reconstruction [2], [3] are two typical examples.

Earliest efforts on autoencoder can be backtracked to papers by [4]–[6] that make autoencoder to learn internal representations of sensor-motor coupling, speech, and image. All the three base on three layer networks, i.e., a network with merely one hidden layer as illustrated in Fig. 1(a), simply with heteroassociation $X \rightarrow Z$ replaced by auto-association $X \rightarrow \hat{X}$ with \hat{X} representing a reconstruction X , i.e., X takes a dual role as both the input and the output. Unknown parameters in the network is learned by gradient-based back-propagation. Alternatively, Bourlard & Kamp considered that all the units of the hidden layer are linear and thus solved unknown parameters by singular value decomposition [1]. Moreover, they extended their study to handle nonlinear hidden units approximately by linearizing each unit. Also, the case of linear hidden units was studied in 1989 by Baldi and Hornik [7] along a similar direction but with more detailed mathematical analyses.

Subsequently, in the nineties of the last century and particularly in the first decade of this century, it becomes common practices to perform autoencoder by networks of more than three layers as illustrated in Fig. 1(a) and (b). The underlying principle of autoencoder is making $X \rightarrow f(X) =$

Manuscript received June 20, 2019; accepted June 26, 2019. This work was supported by the Zhi-Yuan Chair Professorship Start-up Grant (WF220103010) from Shanghai Jiao Tong University. Recommended by Associate Editor Changyin Sun.

Citation: L. Xu, “An overview and perspectives on bidirectional intelligence: Lmsr duality, double IA harmony, and causal computation,” *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 4, pp. 865–893, Jul. 2019.

L. Xu is with the Center for Cognitive Madmnes and Computational Health (CMAcH), School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, and also with Neural Computation Research Centre in Brain and Intelligence Science-Technology Institute, Shanghai Zhangjiang National Lab, China (e-mail: lxu@cs.sjtu.edu.cn).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/JAS.2019.1911603

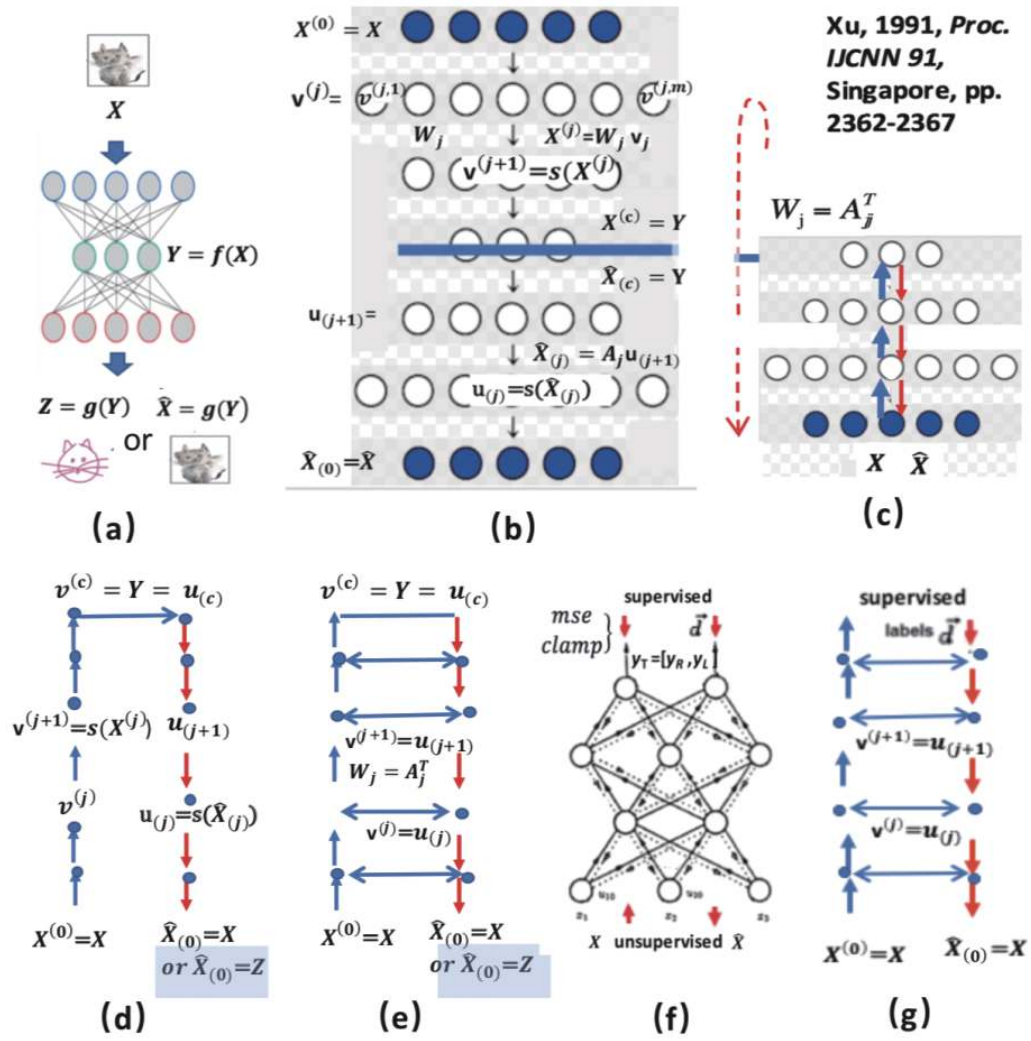


Fig. 1. From Autoencoder to Lmsr reconstruction. (a) Auto-association by three layer networks in early studies. (b) Autoencoder by networks of more than three layers in a symmetric architecture. (c) Lmsr architecture obtained from folding autoencoder along the central layer Y . (d) Autoencoder makes a direct cascading of $X \rightarrow Y$ and $Y \rightarrow \hat{X}$ by a simple circle. It becomes a forward multilayer net for $\hat{X}_{(0)} = Z$. (e) Lmsr improves the direct cascading into a distributed cascading by adding shortcut connections between the paired neurons per layer. (f) Supervised learning and unsupervised learning handled jointly and benefited mutually. (g) Duality in top-down teaching flow versus bottom-up self-organisation flow.

$Y \rightarrow g(Y) = \hat{X}$ to be approximately an identical mapping, i.e., $g(f(X)) \approx 1$, subject to a nature that a same architecture takes a dual role that is used for the direction $X \rightarrow Y$ and $Y \rightarrow \hat{X}$, which is shortly named Duality in Bidirectional Architecture (DBA). Specifically, $f(X) = Y$ is a composite function implemented by a multilayer architecture with each layer performing an one layer function parametrised by weight matrix W_j , and $g(Y) = \hat{X}$ is also a composite function implemented by a same number of layers with each layer performing parametrised by weight matrix A_j . As further sketched in Fig. 1(d), except DBA that the part from Y to \hat{X} (i.e., red line) is symmetrical to the one from X to Y (blue line), the entire architecture is featured by a simple circle without extra constraints.

Usually, Y has a much reduced dimension, and thus $f(X)$ is not invertible in a strict mathematical sense. Autoencoder aims at $g(Y)$ for a direct cascading approximation of $g(f(X)) \approx 1$. Similar situations may be found in many real fields, e.g., a classical closed-loop control system and in a

forward-inverse optics model of visual cortical areas [8]. Another typical example of early efforts on bidirectional deep learning is Least Mean Square Error Reconstruction (Lmsr) proposed in 1991 [2], as illustrated in Fig. 1(f), which improves such a direct cascading into a distributed cascading as illustrated in Fig. 1(e).

The mapping $X \rightarrow \hat{X}$ is approximately implemented by a bidirectional architecture of more than three layers with the DBA nature too, coming from folding autoencoder along the central layer (i.e., one for Y) as illustrated in Fig. 1(c), which merges two corresponding layers such that $v^{(j)} = u_{(j)}$ and $A_j = W_j^T$. The vector equality $v^{(j)} = u_{(j)}$ means that the value of each elements takes a dual role of belonging to the corresponding paired neurons. Collectively, it leads to the symmetry in weight matrix, for which we approximately have an identical mapping per layer j when $A_j W_j^T = I$. Shortly, we call this nature Duality in Paired Neurons (DPN). The matrix equality $A_j = W_j^T$ means that each connection between a pair

of neurons takes a dual role for each of two directions. We call this nature the duality in connection weight (DCW). It has been shown recently in [9] that the natures DPN and DCW make Lmsr outperforms autoencoder significantly. Summarised in Table I there are six types of duality, namely, DBA, DCW, DPN, DPD, DAM, and DSP. The last three are also featured by Lmsr. DPD and DAM will be introduced in Section II-A and Fig. 3, while DSP is illustrated in Fig. 1(f) and (g), that is, making supervised and unsupervised learning handled jointly and benefited mutually.

Helmholtz machine [10], [11] and BYY learning [12], [13] represent two major progresses of bidirectional deep learning in the middle nineties of the last century. Helmholtz machine also considers an architecture as sketched in Fig. 1(e) but with different sets of weights for A_j and W_j^T respectively, that is, it shares the above DPN nature but without DCW, though this DCW nature was also found ten year later in the stacked RBMs [14], [15]. Recently, the DPN nature is found in REDNet [16] under the name of symmetric skip connections and is also closely related to forward skip connections in U-net [17] though there was no backward skip connections, as illustrated in Table I. Actually, with auto-association $X \rightarrow \hat{X}$ changed into heteroassociation $X \rightarrow Z$ as sketched in Fig. 1(e), such a DPN idea is also closely related to the ones in ResNet [18] and DenseNet [19].

Beyond the underlying principle of autoencoder and Lmsr for making $X \rightarrow \hat{X}$, Helmholtz machine takes the distribution $q(Y)$ in consideration by maximising the likelihood $q(X) = E_{q(Y)}q(X|Y) = \int q(X|Y)q(Y)dY$ by a generative model. However, except for some simplest generative models, each pattern X can be generated in exponentially many ways. It is thus intractable to making Maximum Likelihood (ML) learning on $q(X)$. In order to ease the difficulty of computing, Helmholtz machine approximately uses a multilayer networks to implement a factorial distribution in place of Bayesian posteriori $p(Y|X) = q(X|Y)q(Y)/p(X)$, and minimise the Helmholtz free energy, which is equivalently maximising a lower bound of the likelihood $q(X) = E_{q(Y)}q(X|Y)$. Actually, this approach is later and also presently called the variational approach in various studies [20]–[22].

About the same period that Helmholtz machine was proposed, Lmsr network is extended into a general probabilistic framework called Bayesian Ying Yang (BYY) system, as illustrated in Fig. 2. BYY system models two directions in a complementary way, with its encoding part $X \rightarrow Y$ modelled by a probabilistic model $p(Y|X)p(X)$ named YAng machine or Machine yAng (M_A) and with its decoding part $Y \rightarrow X$ modelled by a probabilistic model $q(X|Y)q(Y)$ named YIng machine or Machine yIng (M_I). The Ying and Yang approximately implement mutually inverse mappings $X \rightarrow Y$ and $Y \rightarrow X$ in a probabilistic sense. The ideal case is $p(Y|X)p(X) = q(X|Y)q(Y)$ that becomes true if Yang is given by $p(Y|X) = q(X|Y)q(Y)/p(X)$, which becomes equivalent to ML learning of the generative model $p(X) = E_{q(Y)}q(X|Y)$. However, we generally have $p(Y|X)p(X) \neq q(X|Y)q(Y)$. Instead of approximating maximum likelihood that Helmholtz machine aims at, we aim at the mutual matching between $p(Y|X)p(X)$ and $q(X|Y)q(Y)$, measured by either Kullback-

Leibler divergence or harmony measure. Also, we may cascade layers in an architecture as in Fig. 1(g), with two directions between two consecutive layers in a probabilistic Ying Yang pair as suggested in 1995 [24].

This BYY learning provides not only a unified framework that includes Helmholtz machine (or called variational approach) and typical existing learning approaches as special cases but also a new road towards to new learning methods with automatic model selection [23], [25]–[27]. Later in this paper, relations between BYY learning and several typical bidirectional learning methods are summarized with not only new insights on existing typical methods, but also a number of Lmsr extensions and several new progresses on BYY learning, plus a three-dimensional taxonomy proposed for synthesising and creating.

In the past decade, extensive studies have been made on bidirectional deep learning, under the names of variational autoencoders [20]–[22], deep generative models [28], [29], generative adversarial networks [30], [31], U-net [17], densely connected networks [19], and some combination [32]. An incomplete list of application topics include languages [33], [34], sketches [35], [36], [37], Styles [38], robot motion [39], image [40], [18], [41], 3D images [42], image inpainting [43], diagnosis prediction in healthcare [44], and future data [45], [46].

The rest of the paper is organised as follows. In Section II, Lmsr is further reviewed on its major equations, five types of duality, six cognitive functions, with new insights and a quite large number of extensions. In Section III, insights on deep bidirectional learning are obtained from a bird-viewing on typical learning principles from one or both of the inward and outward directions. Also, relations among maximum likelihood, variational principle, Lmsr learning, and Bayesian Ying Yang learning are further elaborated. In Section IV, a number of typical learning principles are summarized as exemplars of the principles of BYY best matching and BYY best harmony, together with new perspectives on advanced topics. Section V proceeds from deep bidirectional learning to deep bidirectional intelligence, driven by LTD for parameter learning and STD for image thinking and rational thinking in harmony. A BYY intelligence potential theory is proposed, provided with a number of solutions suggested to the problems of AlphaGoZero, TSP, AGM, and uncertainty reasoning. Particularly, causal potential theory is proposed for discovering causal direction, together with two roads for its implementation. Finally, concluding remarks are provided in Section VI.

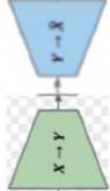
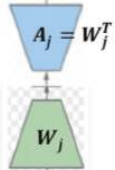
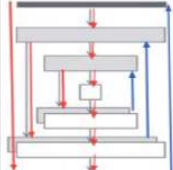
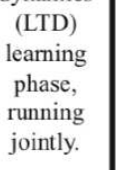
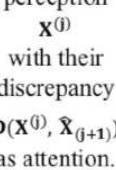
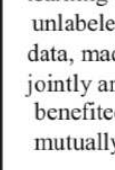
II. LMSER NETWORKS FOR BIDIRECTIONAL LEARNING

A. Lmsr: Implementation Essences and Possible Functions

The implementation of Lmsr learning is a stable dynamic process that consists of short term dynamics (STD) of perception phase and long term dynamics (LTD) of learning phase, that is, featured by a duality of paired dynamics (DPD) listed in Table I.

As illustrated in Fig. 3(a), the perception phase consists of information flows in both forward propagation and backward

TABLE I
TYPES OF DUALITY OR SYMMETRY IN LMSEr

Duality in bidirectional architecture (DBA)	Duality in connection weight (DCW)	Duality in paired neurons (DPN)	Duality in paired dynamics (DPD)	Duality in attention mechanism (DAM)	Duality in supervision paradigm (DSP)
same architecture for direction $Y \rightarrow X$ and direction $Y \rightarrow \hat{X}$. 	each connection links a pair of neurons in a dual role of two directions, i.e., for layer j , we have 	neurons of paired layers merged with each $v_j = u_j$ in a dual role of each other. 	short term dynamics (STD) in perception phase versus long term dynamics (LTD) learning phase, running jointly. 	during STD, each layer gets top-down reconstruction $\hat{X}_{(j+1)}$ versus bottom-up perception $X^{(j)}$ with their discrepancy $D(X^{(j)}, \hat{X}_{(j+1)})$ as attention. 	supervised top teaching by labeled data versus bottom unsupervised learning of unlabeled data, made jointly and benefited mutually. 

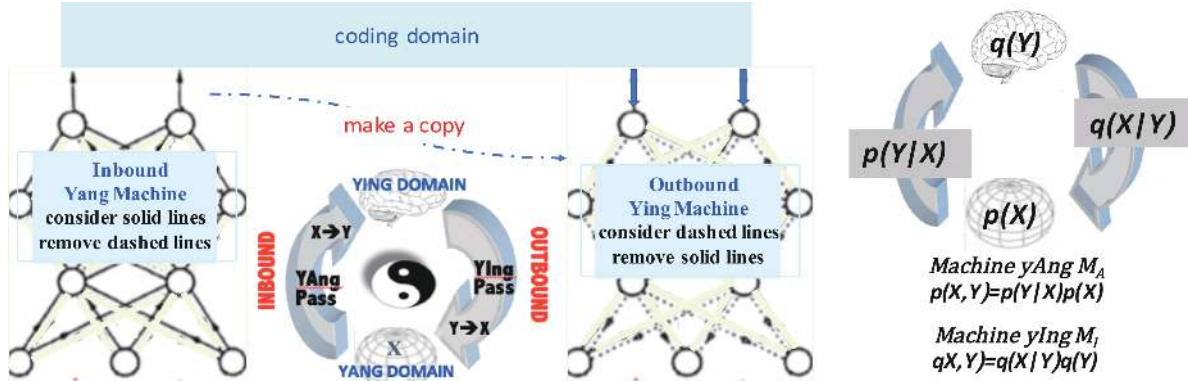


Fig. 2. Bayesian Ying-Yang learning proposed firstly in 1995. “Ying” is spelled “Yin” in the current Chinese Pin Yin system that could be backtracked to over 400 years from the initiatives by M. Ricci and N. Trigault. But, the length of ‘Yin’ lost its harmony with Yang, thus ‘Ying’ is preferred since 1995 [12]. Further details are referred to a section named *a modern perspective on Ying-Yang and WuXing* in Ref.[23].

propagation, which constitute a STD process of variations of neurons in network with the current weights fixed, resulting in two outputs. One is a label y_T given on the top layer as a perception of the current input x at the bottom. The other is a reconstruction u_0 of x given at the bottom. The difference $u_0 - x$ is examined to judge whether this networks should trigger the learning phase to learn the novel information contained in x , which plays a sort of attention role.

It has been shown in [2], [3] that this STD is stable and will reach an equilibrium process of $d\vec{z}_k/dt = 0$ when $A_j = W_j^T$ holds for every layer j , such that each layer is stabilised at $\vec{z}_k = s(\vec{y}_k + \vec{u}_k)$. However, not only it needs time to reach equilibrium but also computation is discretely step by step. In implementation, we may approximately repeat the following updating

$$\vec{z}_k^{(t+1)} = (1 - \eta)\vec{z}_k^{(t)} + \eta s(\vec{y}_k + \vec{u}_k) \quad (1)$$

for a few steps, where $\eta = 1/\tau > 0$ is a small stepsize and $s(\mathbf{v}) = [s(v_1), \dots, s(v_m)]^T$ for $\mathbf{v} = [v_1, \dots, v_m]^T$.

After each perception phase, learning phase may be

triggered by the difference $u_0 - x$. As illustrated in Fig. 3(b), the learning phase is a long term dynamic process of weight updating to adapt the novel information carried by the inputs, targeting at either a best reconstruction of inputs, e.g., minimising the Mean Square Error J by the 1st equation, or both this best reconstruction and a best matching between perception and teaching on the top layer, e.g., minimising the joint MSE J by the 2nd equation. In implementation, gradient descent approach is used to deriving learning rule. As illustrated in Fig. 3(d), information flows in forward propagation and backward propagation are actually coupled, in analog to light propagation in layered media. Some approximation for decoupling was adopted in [2], [3] for getting the learning rule illustrated in Fig. 3(b). Interestingly, the 1st term of this learning rule is equivalent to Hinton’s wake-sleep algorithm, plus the 2nd term as one additional correcting term.

Firstly addressed in Sections 5 & 6 in the 1991 paper [2], as summarised in Table II, *Lmse*r proceeded beyond autoencoder with the following possible functions:

Perception phase

$$\tau \frac{d\vec{z}_k}{dt} = -\vec{z}_k + S(\vec{y}_k + \vec{u}_k)$$

$$\vec{z}_k = S(\vec{y}_k + \vec{u}_k)$$

$$S(y) = \text{diag}[s(y_1), \dots, s(y_m)]$$

$$\vec{y}_k = W_k \vec{z}_{(k-1)}$$

$$\vec{u}_k = W_{k+1}^T \vec{z}_{k+1}$$

$$\vec{u}_0 = W_1 \vec{z}_1$$

Xu, 1991,
Proc. IJCNN 91,
Singapore,
pp. 2362-67

(a)

Learning phase

$$J = E(\|\vec{x} - \vec{u}_0\|^2)$$

$$J = \frac{1}{2} E(\|\vec{x} - \vec{u}_0\|^2) + \frac{1}{2} E(\|\vec{d} - y_L\|^2)$$

$$\tau^w \frac{\partial w_{ijk}}{\partial t} = - \frac{\partial J}{\partial w_{ijk}}$$

$$D_k^s = \text{diag}[(s'_k(y_{ik} + u_{ik}))]$$

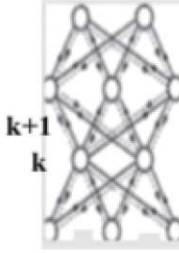
$$\Delta W_k \propto \vec{z}_k \varepsilon_{k-1}^T D_{k-1}^s + D_k^s \vec{\varepsilon}_k \vec{z}_{k-1}^T$$

$$\vec{\varepsilon}_0 = \vec{x} - \vec{u}_0, \quad \vec{\varepsilon}_k = W_k \vec{\varepsilon}_{k-1}$$

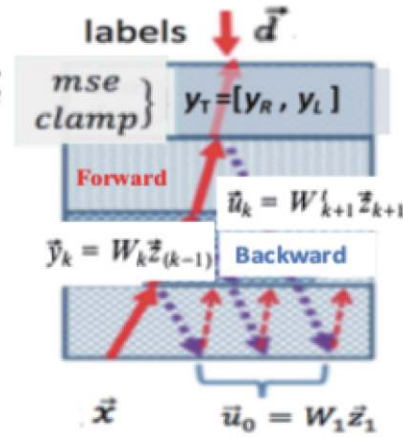
(b)

Notation Translation	
Fig. 1	here
$j+1$	k
$v^{(j+1)} = u^{(j+1)}$	\vec{z}_k
$\hat{X}_{(j+1)}$	\vec{u}_k
$X^{(j)}$	\vec{y}_k

Xu 1993,
Neural Networks,
vol 6, pp. 627-48



(c)

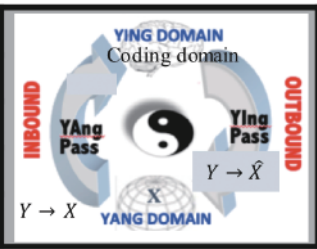


(d)

Fig. 3. Lmsr works in two phases. (a) Perception phase is a short term dynamic (STD) process in bidirectional propagation. (b) Learning phase is a long term dynamic (LTD) process that minimizes MSE via gradient descending that is implemented by local updating rule. (c) Notations translation to ones in Fig. 1(a). (d) An analog illustration by light propagation in layered medias.

TABLE II
POSSIBLE FUNCTIONS THAT LMSER MAY PERFORMS

Bottom input X	Coding domain	Bottom output
pattern	• recognizing	reconstruction
partial input	• concept abstracting	associative recall
	labels or thinking trace	imaginary emergence
saliency	attention	pre-activation

**(a) Pattern recognition with rejection**

Each input pattern \mathbf{x} will be recognised by a label \mathbf{y}_T output on the top, acting as a classifier. The difference is that there is also an output \mathbf{u}_0 at the bottom. Setting a threshold on the discrepancy between \mathbf{x} and its reconstruction on the bottom layer, we may reject those very different input patterns as unknown or unlearned yet, from which we get another perspective to measure how much confidence we trust the classification by the label \mathbf{y}_T .

(b) Concept abstraction and formation

For inputs without labels, each input pattern \mathbf{x} gets an abstract representation on the top concept domain. Top-down reconstruction can verify whether concepts interpret the corresponding inputs well. Since each layer performs a linear transform and then a sigmoid rescaling, which preserves the neighbourhood or topological relation. This nature facilitates concept forming and organising in the top encoding domain. Such a nature makes *Lmsr* and auto-encoder become superior

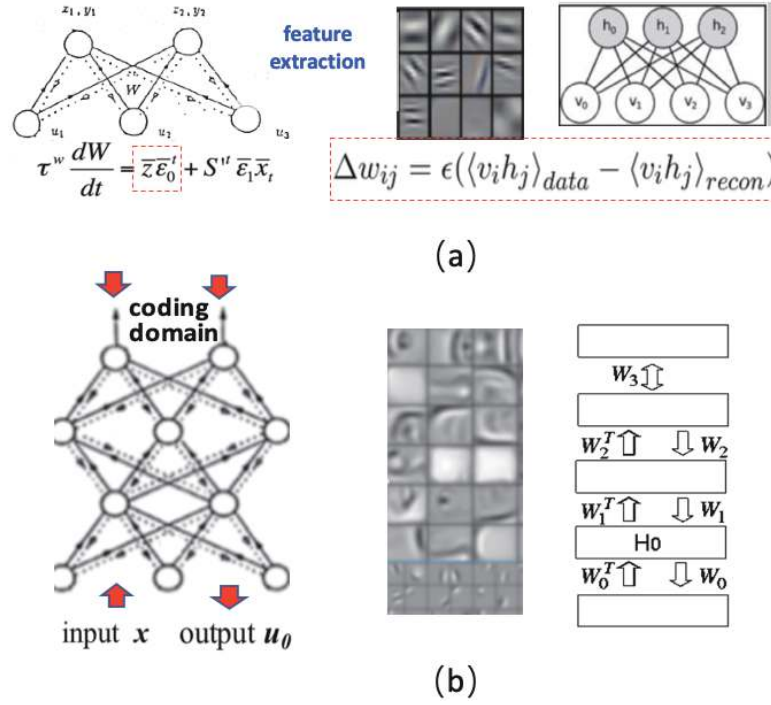


Fig. 4. Lmsr versus stacked RBMs. (a) Each layer in Lmsr and in stacked RBMs [14], [15] learns templates of feature extraction, by a learning rule that consists of a term equivalent to Hinton's wake-sleep algorithm (i.e., ones in red colored dashed box), plus one correcting term in Lmsr. (b) Multiple layers get internal representations of inputs, similar to the one obtained by the stacked RBMs, where each layer uses weight sharing $A_j = W_j^T$ too.

to those models in lack of such preservation, e.g., variational autoencoder [21], deep generative model [28], and generative adversarial networks [30].

(c) Reconstruction and associative memory

Given a full pattern \mathbf{x} as an input, the bottom output \mathbf{u}_0 acts as a reconstruction of \mathbf{x} . Given an input partially from pattern \mathbf{x} , the bottom output \mathbf{u}_0 can be regarded as associative recall of a partial input. The DCW nature $A_j = W_j^T$ in Table I makes learning to catch invertible structures under input patterns for restoring these structures by partial inputs and discarding those irregular details and disturbances, as well as even adversarial attacks.

(d) Saliency, attention, and imaginary recall

Those elements or parts that \mathbf{x} differs from \mathbf{u}_0 significantly can be detected as saliency, which activates attention to the saliency for further analysis and learning. Attention may also be aroused from top concepts activated by perception from other medias (e.g., hearing, text during getting image as \mathbf{x}). Top level concepts may also be activated by internal thinking. When such activating strengths are strong enough, top-down reconstruction \mathbf{u}_0 acts as imaginary recall.

(e) Cortical field template and feature map

It has been shown theoretically and experimentally [2], [3] that learning makes each neuron or unit become a selective feature detector (like the orientation cell in cortical field) and units on each layer form a feature detecting map as illustrated in Fig. 4(a). Layer by layer, feature maps are formed hierarchically as illustrated in Fig. 4(b).

(f) Pattern transform

Given an input pattern \mathbf{x} (either of vector, image, speech, text, etc.) to reconstruct \mathbf{z} (either of vector, image, speech,

text, etc.), the task of pattern transform implements nonlinear mapping, which actually makes autoencoder return back to act as a multilayer perceptron. Comparing Fig. 1(d) and Fig. 1(e), we observe that Lmsr may improve transform performances due to DCW and DPN nature.

In the era of the early 1990's, computing power and sample size is very limited, which was far from being able to support simulation of the above functions. By that time, computer simulation was merely made on one layer *Lmsr*, which demonstrates how learning makes neurons become orientation cell like feature detectors in the cortical field [2]. Recently in [9], many others of the above functions have been verified. Particularly, the DCW and DPN nature (see Table I) indeed make Lmsr outperform autoencoder considerably on several real benchmark datasets, especially on samples of small sizes, polluted by disturbances, and adversarial attacks. The contribution by DPN is generally much bigger than the one by DCW, while DCW typically provides further improvements.

B. Lmsr Extensions

Lmsr may be further improved in three aspects. First, conceptualisation in the encoding space may be improved in the following three ways.

(1) RPCL-Lmsr

Improvement is made in three stage. The first stage makes Lmsr as usual. The second stage uses the resulted Lmsr network to map all the samples of X into the corresponding inner codes in the Y domain on the top layer. As illustrated in Fig. 5(a), RPCL [47] is made on these inner codes to group them into clusters with each cluster representing one concept. The third stage makes learning on the weights in Lmsr

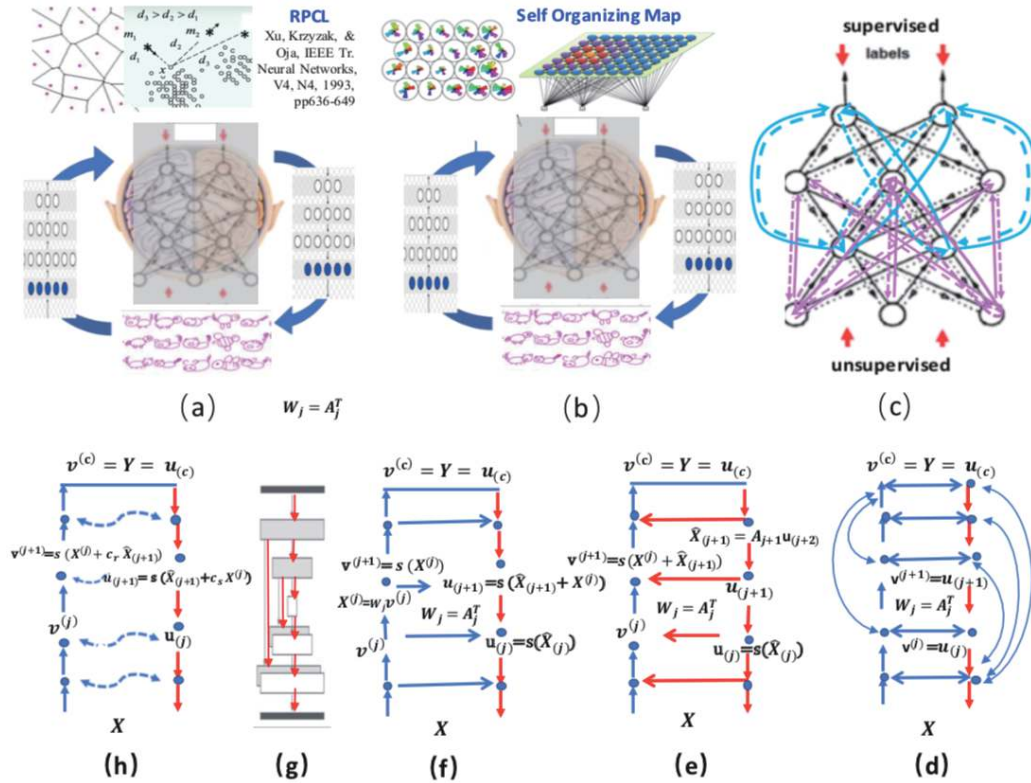


Fig. 5. Lmser extensions. (a) Combining Lmser and RPCL to form clusters of concepts. (b) Combining Lmser and SOM that preserves topology of clusters. (c) Getting skip connected Lmser by adding direct symmetrical links between every pair or some pairs of layers in networks. (d) Densely connected Lmser. (e) Removing the right arrow of every \leftrightarrow link in Fig. 1(e) leads us to a feedback Lmser. (f) Removing the left arrow of every \leftrightarrow link in Fig. 1(e) leads us to a fast-lane Lmser. (g) U-net [17]. (h) Combining feedback Lmser and fast-lane Lmser leads a Flexible Lmser (see Table 3).

network and RPCL learning on the coding clusters jointly. Specifically, a sample X is randomly picked from samples, and then mapped into a code Y that not only modifies the winner and rival clusters by RPCL. The modified winner cluster's center is propagated top-down to yield a reconstruction \hat{X} , and then the Lmser rule is used to update network weights to reduce the error $X - \hat{X}$. It follows from the DCW nature that the bottom up pathway is updated for considering the next sample. Such adaptive learning is made for an enough long period until the overall reconstruction error can be further improved.

(2) RPCLVQ-Lmser

For input samples of multiple classes with part of samples having labels and part of samples having not, further improvement may be made on the above second stage in a way that clustering is made by RPCL for unlabelled samples and LVQ [48] for labelled samples.

(3) SOM-Lmser

As addressed at the end of a recent paper [49], bidirectional multilayer networks used by Lmser and auto-encoder has a favourable nature that preserves neighbourhood or topological relation, which facilitates concept forming and organising in the top encoding domain and thus is superior to those models in lack of such preservation, e.g., variational autoencoder [21], deep generative model [28], generative adversarial networks [30]. To enhance such topology preservation, the above three stage learning may be improved with RPCL and/or LVQ in the second stage being replaced by Kohonen Self-Organising

Map (SOM) [50], as illustrated in Fig. 5(b).

The second aspect consists of the following alternatives that come from enhancements or variants of the DPN and DCW nature given in Table I:

(4) Skip connected Lmser and Densely connected Lmser

As already reviewed at the end of Section II-A, the DPN nature has been taking crucial roles in several recent popular studies. We may enhance the DPN nature to get skip connected Lmser by adding direct symmetrical links between every pair or some pairs of layers in Fig. 1(e) and (f). Next, adding more skip connections between layers further lead to Densely connected Lmser as illustrated in Fig. 5(d).

(5) Recurrent net and Feedback Lmser

Disabling the DCW nature (i.e., $A_j = W_j^T$) and modifying the DPN nature $\mathbf{v}^{(j+1)} = s(X^{(j)} + \hat{X}_{(j+1)}) = \mathbf{u}_{(j+1)}$ into $\mathbf{v}^{(j+1)} = s(X^{(j)} + \hat{X}_{(j+1)}) \neq s(\hat{X}_{(j+1)}) = \mathbf{u}_{(j+1)}$ (i.e., removing the right arrow of every \leftrightarrow link in Fig. 1(e)), we are lead to recurrent autoencoder and recurrent net, which is known to be good at catching temporal dependence among patterns such as speech, video, drawing, sketch, etc. Instead, keeping the DCW nature $A_j = W_j^T$ but modifying the DPN nature into $\mathbf{v}^{(j+1)} = s(X^{(j)} + c_{r\rho} \hat{X}_{(j+1)})$ and $s(\hat{X}_{(j+1)}) = \mathbf{u}_{(j+1)}$, we are lead to a feedback Lmser listed in Table III, which stably and selectively tunes to merely those relevant feedbacks controlled by $c_{r\rho}$.

(6) U-net, DenseNet, and Fast-lane Lmser

Disabling the DCW nature (i.e., $A_j = W_j^T$) and modifying the DPN nature into $\mathbf{v}^{(j+1)} = s(X^{(j)}) \neq s(X^{(j)} + \hat{X}_{(j+1)}) = \mathbf{u}_{(j+1)}$

TABLE III
FLEXIBLE LMSEr AS A UNIFIED FORMULATION

$$\mathbf{v}^{(j+1)} = s(\mathbf{X}^{(j)} + c_R t_\rho \hat{\mathbf{X}}_{(j+1)}), \mathbf{u}_{(j+1)} = s(\hat{\mathbf{X}}_{(j+1)} + c_S t_\rho \mathbf{X}^{(j)}), \quad \rho = \mathbf{X}_{(j+1)}^T \mathbf{X}^{(j)} / (\|\hat{\mathbf{X}}_{(j+1)}\| \|\mathbf{X}^{(j)}\|), \gamma \geq 0$$

Type	Mechanism	c_R Recurrent	c_S Shortcut	$W_j = A_j^T$ Transfer	t_ρ tuning
Integrated Lmse		learned c_R	learned c_S	Yes	$t_\rho = \tanh(\gamma\rho)$
Lmse		$c_R=1$	$c_S=1$	Yes	$t_\rho = 1$
Fast-lane Lmse		$c_R=0$	$c_S=1$	Yes	$t_\rho = \tanh(\gamma\rho)$
U-net, DenseNet		$c_R=0$	$c_S=1$	No	$t_\rho = 1$
Feedback Lmse		$c_R=1$	$c_S=0$	Yes	$t_\rho = \tanh(\gamma\rho)$
Recurrent net		$c_R=1$	$c_S=0$	No	$t_\rho = 1$
Autoencoder		$c_R=0$	$c_S=0$	No	any t_ρ

Diagram illustrating the Flexible Lmse mechanism. It shows a bottom-up pattern $\mathbf{X}^{(j)}$ and a top-down pattern $\hat{\mathbf{X}}_{(j+1)}$. Dashed links represent feedback (c_R) and shortcut (c_S) interactions. The interaction strength is modulated by t_ρ . The final output is $\mathbf{v}^{(j+1)} = \mathbf{Y} = \mathbf{u}^{(j+1)}$. The input $\mathbf{X}^{(0)} = \mathbf{X}$ and the reconstruction $\hat{\mathbf{X}}_{(0)} = \mathbf{X}$ or $\hat{\mathbf{X}}_{(0)} = \mathbf{Z}$ are shown at the bottom.

Remarks:

- (1) c_R, c_S indicate the strength of feedback and shortcut interactions by the dashed links, ρ indicates the similarity between the bottom-up pattern $\mathbf{X}^{(j)}$ and the top-down pattern $\hat{\mathbf{X}}_{(j+1)}$, which is first rescaled by a parameter $\gamma \geq 0$ and then sharpen by \tanh function to get t_ρ .
- (2) t_ρ modulates the strength of interaction that acts jointly with c_R, c_S . When $\rho \approx 1$ and thus $t_\rho \approx 1$, the two patterns are similar and integrated directly to get \mathbf{u}, \mathbf{v} ; when $\rho \approx -1$ and thus $t_\rho \approx -1$, the two patterns are similar but in different signs, and thus integrated after correcting sign to get \mathbf{u}, \mathbf{v} .
- (3) If the two patterns are irrelevant, ρ is small and thus $t_\rho \approx 0$, the interaction can be ignored.

(i.e., removing the left arrow of every \leftrightarrow link in Fig. 1(e)), we are lead to ones similar to recent popularised U-net [17] and densely connected networks [19]. Instead, keeping $A_j = W_j^T$ but modifying the DPN nature into $\mathbf{v}^{(j+1)} = s(\mathbf{X}^{(j)})$ and $\mathbf{u}_{(j+1)} = s(\hat{\mathbf{X}}_{(j+1)} + c_S t_\rho \mathbf{X}^{(j)})$, we are lead to a fast-lane Lmse listed in Table III, with fast-lanes provided to merely those relevant shortcuts controlled by $c_S t_\rho$.

(7) Integrated Lmse and Flexible Lmse

Combining feedback Lmse and fast-lane Lmse leads an integrated Lmse that unifies all these discussed architectures. As summarised in Table III, not only pre-setting the parameters c_R, c_S , and t_ρ will lead to all the above cases, but also getting these parameters via learning further leads to a flexible Lmse for optimal performances.

(8) Causal Lmse

As discussed on Fig. 6(a) and (b) and Fig. 6(k) and (l) in a recent paper [49] as well as at the end of that paper, the reconstruction or generative mapping $Y \rightarrow \hat{X}$ contains a principal structure that represents causal topology. It is this causal topology that takes crucial roles. We may use a causal discovery method to prune off those non-causal connections and also extra duplicated causal connections for $Y \rightarrow \hat{X}$. Then, it follows from the DBA nature and DCW nature in Table I we get the symmetrical connects for the mapping $X \rightarrow Y$, which is equivalent to making Lmse under the constraint that weights of those non-causal connections and extra duplicated causal connections are forced to zero.

(9) Alternative implementation of the DCW nature

The nature $A_j = W_j^T$ aims at an identical mapping $X^{(j)} = \hat{X}_{(j+1)}$ which works merely for an orthogonal matrix W_j .

We may consider the architecture as in Fig. 1(e) and replace $A_j = W_j^T$ by directly enforcing $X^{(j)} = \hat{X}_{(j+1)}$ for every j via implementing the learning phase illustrated in Fig. 3(b).

The third aspect examines the discrepancy between $X^{(j)}$ and $\hat{X}_{(j+1)}$ for every layer j and particularly between input X and its reconstruction at the bottom layer, based on which we may implement some attention mechanisms to further improve Lmse in the following four ways:

(10) Attention: unexpected input and saliency

The discrepancy between $X^{(j)}$ and $\hat{X}_{(j+1)}$ for every layer j may be measured or integrated as vigilance signals of unexpected inputs and saliencies on these inputs, not only for robust learning to discard disturbances and attacks, but also for modulated learning via pre-activations and inhibitions by top-down attention.

(11) Pipeline Lmse and mixture Lmse

When the reconstruction discrepancy between input X and its reconstruction becomes larger than a given threshold, forcing the current Lmse net to adapt this X may considerably wash away what have been already learned. Instead, we may deliver this X to the other one of a number of Lmse networks in a pipeline similar to that illustrated in Fig. 4(b) of the paper [51]. As a result, a pipeline of Lmse networks are learned to describe samples with each sample being allocated to an appropriate Lmse net in a sequent way. Alternatively, a mixture of Lmse networks may also be used in a way similar to Gaussian mixture [52] or mixture of experts [53].

(12) Discrepancy guided implementation of perception phase and learning phase

We may implement perception phase on a layer with the

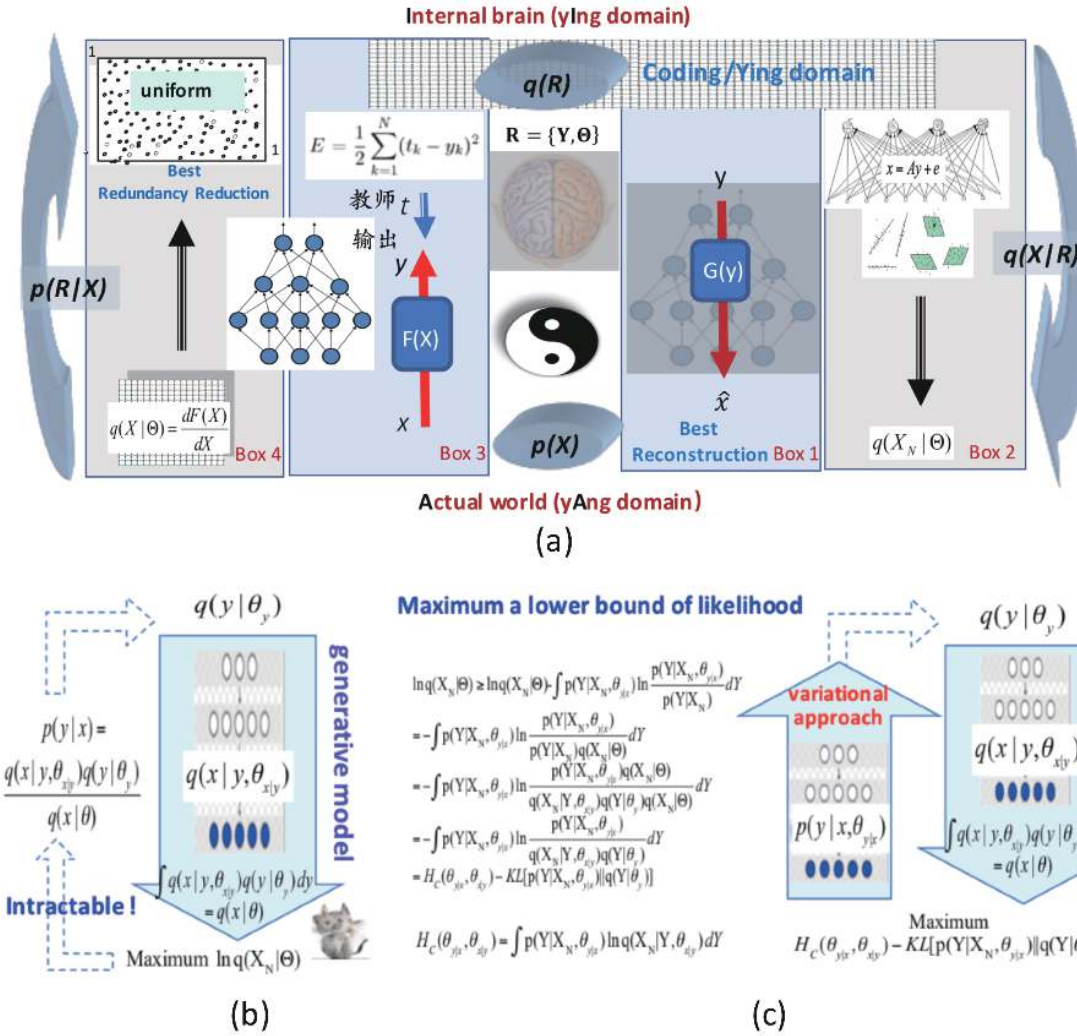


Fig. 6. Learning principles from a bidirectional perspective.

biggest discrepancy until the discrepancy of every layer getting smaller than a threshold or stabilized. Also, we may implement learning phase by updating a layer with the biggest discrepancy until the discrepancy of every layer getting smaller than a threshold, instead of starting learning from the bottom layer or randomly as suggested in [2], [3].

(13) Layered pipeline Lmser

When the discrepancy between $X^{(j)}$ and $\hat{X}_{(j+1)}$ for a layer j can not be further reduced during the perception phase and learning phase, but is still bigger than a threshold, we may form a pipeline on this layer by attaching this layer j with alternative layer that gets its $X^{(j)}$ from its previous layer $j-1$ and its $\hat{X}_{(j+1)}$ from the next layer $j+1$. Then, the perception phase and learning phase are implemented on this alternative layer. As a result, we get an extension of Lmser by layered pipelines. In the recall phase, we select the best one on each pipeline with the least discrepancy between $X^{(j)}$ and $\hat{X}_{(j+1)}$.

III. FROM BIDIRECTIONAL LEARNING TO BAYESIAN YING YANG LEARNING

A. Bird-view on Learnings: Unidirectional vs Bidirectional

Further insights on deep bidirectional learning can be

obtained from overviewing typical existing learning principles systematically. A birdview is illustrated in Fig. 6 from not only each direction separately but also two directions jointly, outlined from four aspects.

First, modelling along the direction $Y \rightarrow X$ is featured typically by the following three existing ways:

(1) Generative regression

Given paired samples, the mapping $Y \rightarrow X$ aims at generating X from Y , as illustrated in Box-1 of Fig. 6, modelled by nonlinear regression or conditional probability for a best fitting of samples of X . Actually, the mapping $Y \rightarrow X$ has been seldom considered because it maps from low dimension to high dimension. However, it does act as one ingredient in those studies addressed in Section II-B.

(2) Generative model

Given samples merely at the end of X , as illustrated in Box-2 of Fig. 6, the direction $Y \rightarrow X$ is modelled by the maximum likelihood on

$$q(X|\Theta) = E_{q(Y|\Theta_Y)} q(X|Y, \Theta_{X|Y}) = \int q(X|Y, \Theta_{X|Y}) q(Y|\Theta_Y) dY \quad (2)$$

that best fits samples of X , in order to estimate the parameter $\Theta = \{\Theta_{X|Y}, \Theta_Y\}$. This method suffers extensive computational

costs due to the integral over Y .

(3) Bayesian approach

The direction $Y \rightarrow X$ is modelled by the maximum marginal likelihood

$$\begin{aligned} q(X) &= E_{q(\Theta)} q(X|\Theta) = \int q(X|\Theta) q(\Theta) d\Theta, \\ q(X) &= E_{q(R)} q(X|R) = \int q(X|R) q(R) dR \end{aligned} \quad (3)$$

that best fits samples of X , in order to obtain a model selection criterion. Typical examples include Bayesian inference criterion (BIC) [54], minimum description length (MDL) [55], [56], and those studies made under the name Bayesian approach [57]. Typically implemented together with the above generative learning, the integral over Θ is even much more computational extensive than the one for the integral over Y . Thus, rough approximation is adopted for handling integrals.

Actually, all the above three are specific implementation of the principle of *best reconstruction* illustrated in Box-1 of Fig. 6, that is, the outward direction $Y \rightarrow X$ generates a best reconstruction or description of samples of X . Typically, the principle is implemented in two ways. One is simply generating \hat{X} with the best measured by minimising the MSE error $E\|X - \hat{X}\|^2$ or one of L_p error, as made in Lmser and autoencoder. The other is generating a distribution q_X of X that is a best approximation of the unknown true distribution of $p(X)$ with the best in term of minimising the following Kullback divergence

$$\begin{aligned} KL(p\|q) &= E_{p(X)} \ln \frac{p(X)}{q(X)} = \int p(X) \ln \frac{p(X)}{q(X)} dX = \chi - H, \\ H &= E_{p(X)} \ln q(X) = \int p(X) \ln q(X) dX, \\ \chi &= E_{p(X)} \ln p(X) = \int p(X) \ln p(X) dX, \end{aligned} \quad (4)$$

which is equivalent to maximising L for a given $p(X)$. Typically, a given sample set $\{X_t\}_{t=1}^N$ is used to approximate $p(X)$ by the following kernel estimation

$$p_h(X) = \frac{1}{N} \sum_{t=1}^N G(X|X_t, h^2 I), \quad (5)$$

where $G(u|\mu, \Sigma)$ is a Gaussian distribution of u with the mean μ and the covariance matrix Σ , and $h > 0$ is a small smoothing parameter.

Particularly, we have $p_0(X) = \lim_{h \rightarrow 0} p_h(X)$ becomes the empirical distribution. In this case, maximising L becomes minimising the MSE error $E\|X - \hat{X}\|^2$ when $q_X = G(X|\hat{X}, \sigma^2 I)$, the above maximum likelihood learning when $q_X = q(X|\Theta)$ by Eq.(2), and Bayesian approach when $q_X = q(X)$ by Eq.(3). Moreover, when $p_h(X)$ with $h \neq 0$, we get so called data-smoothing extension of the ML learning [58], [59].

Second, modelling along the direction $X \rightarrow Y$ is also featured by three typical approaches as follows:

(4) Supervised learning

Given paired samples or called labeled data, the A-mapping direction $X \rightarrow Y$ is modelled by nonlinear regression or conditional probability. Such a mapping aims at abstraction Y of input X , as illustrated in Box-3, implementing those conventional tasks of deep learning [14], [15].

(5) Best redundancy reduction

Given samples of merely at the end of X , as illustrated in Box-4, the direction $X \rightarrow Y$ is conceptually modelled by a mapping $F(X)$ that makes Y become uniformly distributed, because $dF(X)/dX$ is just the distribution of X . The fact that Y is uniformly distributed means that there is no redundancy among Y , which motivates the principle of best redundancy for the reduction mapping $X \rightarrow Y$, e.g., implemented by INFOR-MAX [60]. Relaxing slightly, this principle is approximately implemented by requiring the components of Y become mutually independent, e.g., by the minimum mutual information (MMI) [61]. Both MMI and INFOR-MAX have been widely adopted in the studies of independent component analysis (ICA), especially those via a linear mapping $y = Wx$ [62], [63].

(6) Bayesian Self-Organisation

In addition to having samples at the end of X , we are also given a priori distribution $q(Y)$, the direction $X \rightarrow Y$ is modelled by a mapping $F(X)$ or a probabilistic model $p(Y|X)$ via minimising

$$\begin{aligned} KL(p\|q) &= E_{p(Y)} \ln \frac{p(Y)}{q(Y)} = \int p(Y) \ln \frac{p(Y)}{q(Y)} dY, \\ p(Y) &= \frac{1}{N} \sum_{i=1}^N p(Y|X_i) \end{aligned} \quad (6)$$

which was proposed for visual processing under the name Bayesian Self-Organisation (BSO) [64].

Third, both the direction $X \rightarrow Y$ and the direction $Y \rightarrow X$ are jointly modelled by an approximation of inverse relationship in one of the following ways:

(7) $X \rightarrow Y \rightarrow X$ approximately making an identical mapping

Autoencoder, Lmser, and all those addressed in Section II all belong to this sort.

(8) The unknown mapping approximating the inverse of the prefixed one

Given paired samples of X, Y , we may regard either that the sample pairs come from one underlying mapping $X \rightarrow Y$ and its inverse mapping is approximated as illustrated by Box 1 in Fig. 6(a) (e.g., see (1) Generative regression) or that the sample pairs come from one underlying mapping $Y \rightarrow X$ and its inverse mapping is approximated as illustrated by Box 2 in Fig. 6(a) (e.g., see (4) Supervised learning). Moreover, assume that X comes from Y from the uniform distribution or a given $q(Y)$ through one underlying mapping $Y \rightarrow X$, its inverse mapping $X \rightarrow Y$ is approximated by either the above (5) Best redundancy reduction or the above Bayesian Self-Organisation.

(9) Bidirectional inverse relationship $p(Y|X)p(X) = q(X|Y)q(Y)$

With $Y \rightarrow X$ by the generative model by Eq.(2), this relationship implies the Bayesian posterior $p(Y|X) = q(X|Y, \Theta_{X|Y})q(Y|\Theta_Y)/q(X|\Theta)$ for the mapping $X \rightarrow Y$, as illustrated in Fig. 6(b), which is usually computationally intractable.

(10) Variational learning and Variational Bayes

The idea of this variational approach was firstly proposed by Hinton and colleague in their Helmholtz machine [11]. As illustrated in Fig. 6(b), the variational learning uses a

multilayer networks to implement $p(Y|X)$ in place of the computationally intractable Bayesian posteriori $p(Y|X)$ and maximises a lower bound of likelihood as illustrated in Fig. 6(b), which approaches the likelihood as $p(Y|X)$ approaches $p(Y|X)$. Moreover, such an idea may also be used for Bayesian approach on $q(X)$ by Eq.(3), under the name of variational Bayes.

Last but not last, relaxing the inverse relationship $p(Y|X)p(X) = q(X|Y)q(Y)$, we consider Bayesian Ying-Yang learning previously introduced in Fig. 2 and Section I, seeking the mutual matching between $p(Y|X)p(X)$ and $q(X|Y)q(Y)$, for which a systematical overview with some novel proposals will be made in Section IV.

B. Bayesian Ying Yang Learning, Variational Approach, and Lmser Learning

As previously addressed in Section I and particularly in Fig. 2, about the same period that Helmholtz machine was proposed, Lmser network is extended into a general probabilistic framework called Bayesian Ying Yang (BYY) system [12], [23], which is actually a probabilistic model for the first front activities that are supported by the second layer featuring the structure \mathbf{k} . As shown in Fig. 7, the internal representation $R = \{Y, \Theta, \mathbf{k}\}$ generally consists of Short term memory (STM) representation Y , Long term memory (LTM) for all the parameters Θ and the coding domain structure \mathbf{k} . It follows from the left table and the right table in the figure that combinations of different specific settings of four components in the front layer lead to a number of specific formulations, covering several existing learning models.

The one in Fig. 2 is one direct extension of Lmser architecture into BYY system, simply with

$$\begin{aligned} p(Y|X) &= G(Y|f(X, W), \sigma_f^2 I), \quad q(X|Y) = G(X|g(Y, W^T), \sigma_g^2 I), \\ \text{or } p(Y|X) &= \prod_j f_j(X, W)^{y_j} [1 - f_j(X, W)]^{1-y_j}, \\ y_j &= 0 \text{ or } 1, \quad 0 \leq f_j \leq 1, \\ f(X, W) &= [f_1(X, W), \dots, f_k(X, W)]^T, \quad Y = [y_1, \dots, y_k]^T, \end{aligned} \quad (7)$$

where $f(X, W), g(Y, W^T)$ are given by the forward mapping and backward mapping, respectively, still featured by two phases in Fig. 3 and the dualities in Table I. Moreover, $q(Y|\Theta_Y)$ is either a Gaussian or nonGaussian for a real Y , while $q(Y|\Theta_Y) = \prod_j q_j^{y_j} [1 - q_j]^{1-y_j}$ for a binary Y .

Instead of approximating maximum likelihood that Helmholtz machine or Variational approach aims at, a mutual harmony between Ying and Yang is sought for and measured via a triple-Relation $H_\mu(P||Q)$ about dP, dQ , and $d\mu$ as shown in Fig. 8, which is a simplified version of Fig. 5 in [26]. P, Q are both σ -finite measures on the same measure space that supports both p_{M_A} and q_{M_P} , and μ is one benchmarking reference σ -finite measure that describes a volume or capacity about this joint space; while each of the Radon-Nikodym derivative $dP/d\mu, dQ/d\mu$ describes the relative configuration against the benchmark μ and represents the relative density against μ , measured on a differential piece of this space. A local harmoniousness is described by a product $f(dQ/d\mu)dP/d\mu$ with a scale adjustment by $f(r)$ that

monotonically increases with r . Further details are referred to Section IV-A and especially Eq. (21) in [23].

When μ is a Lebesgue measure, and P, Q are probability measures on the probability space, as shown by Box ①a and further Box ①c in Fig. 8, it follows that $H_\mu(P||Q)$ becomes $H = E_{p(X)} \ln q(X)$ in Eq.(11), named as *harmony functional*. Here, we get an insight that this harmony measure actually describes a triple-relation among three measures dP, dQ , and $d\mu$ at the special case that μ is Lebesgue, though it appears to be and also typically regressed as describing a bi-relation.

This triple-Relation among dP, dQ , and $d\mu$ includes two typical bi-Relations as its degenerated cases. One is $dQ = dP$ that

$$H_\mu(P||P) = \int \frac{dP}{d\mu} f\left(\frac{dP}{d\mu}\right) d\mu = \int f\left(\frac{dP}{d\mu}\right) dP = \chi_P, \quad (8)$$

which is the self-harmoniousness, the negative entropy of $dP/d\mu$, and the representation compactness of the measure P , indicating the vigour of the system described by P or called Q_i , referred to its Chinese character in Fig. 7 according to Chinese Ying Yang philosophy. We use Greece symbol χ to denote this quantity since the pronunciation of the Chinese character Q_i is somewhat similar to the one for χ .

The other bi-Relation is obtained by letting $d\mu = dP$ that leads to

$$H_P(P||Q) = \int f\left(\frac{dQ}{dP}\right) dP, \quad (9)$$

which describes the discrepancy between P and Q .

Considering $f(r) = \ln r$ and focusing on the front layer, $H_\mu(P||Q)$, $H_\mu(P||P)$, and $H_P(P||Q)$ become $H(p||q)$, χ_P , and $KL(p||q)$ respectively in a relationship as follows:

$$\begin{aligned} H(p||q) &= E_{p(X,Y)} \ln q(X, Y) = \int p(X, Y) \ln q(X, Y) dXdY \\ &= \chi_P - \mathbf{KL}(p||q), \\ \chi_P &= E_{p(X,Y)} \ln p(X, Y) = H(p||p), \\ KL(p||q) &= E_{p(X,Y)} \ln \frac{p(X, Y)}{q(X, Y)} = \int p(X, Y) \ln \frac{p(X, Y)}{q(X, Y)} dXdY, \end{aligned} \quad (10)$$

where $KL(p||q)$ is the Kullback-Leibler divergence that measures the discrepancy between Ying and Yang [12], [13], [65], [23], while $H(p||q)$ measures the harmony between them. Best harmony [66]–[68] is made by $\max H(p||q)$ in a sense that seeks a best matching via $\min \mathbf{KL}(p||q)$ in a most tacit manner by minimising the information $-H(p||p)$ transferred by Yang or equivalently maximising the vigour χ_P carried by the Yang.

For the BYY system illustrated in Fig. 7, $KL(p||q)$ becomes

$$\begin{aligned} KL_h(p||q, \Theta) &= E_{p_h(X)} E_{p(Y|X)} \ln \frac{p(Y|X)p_h(X)}{q(X|Y, \Theta_{X|Y})q(Y|\Theta_Y)} \\ &= \int p(Y|X)p_h(X) \ln \frac{p(Y|X)p_h(X)}{q(X|Y, \Theta_{X|Y})q(Y|\Theta_Y)} dXdY \\ &= E_{p_h(X)} V_L(X|\Theta) + E_{p_h(X)} \ln p_h(X), \\ V_L(X|\Theta) &= E_{p(Y|X)} \ln \frac{p(Y|X)}{q(X|Y, \Theta_{X|Y})q(Y|\Theta_Y)} \\ &= \int p(Y|X) \ln \frac{p(Y|X)}{q(X|Y, \Theta_{X|Y})q(Y|\Theta_Y)} dY. \end{aligned} \quad (11)$$

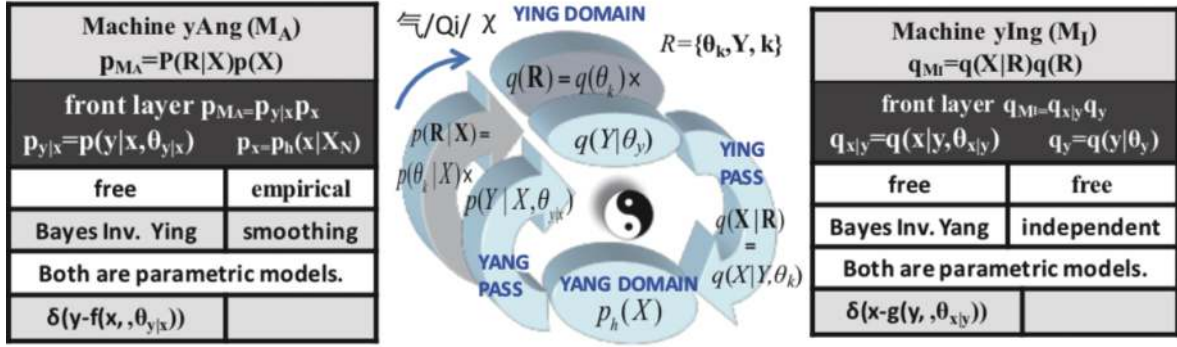


Fig. 7. Generative model, Variational approach, and Bayesian Ying Yang system Combinations of different specific settings of four components in Bayesian Ying Yang (BYY) system lead to a number existing learning models.

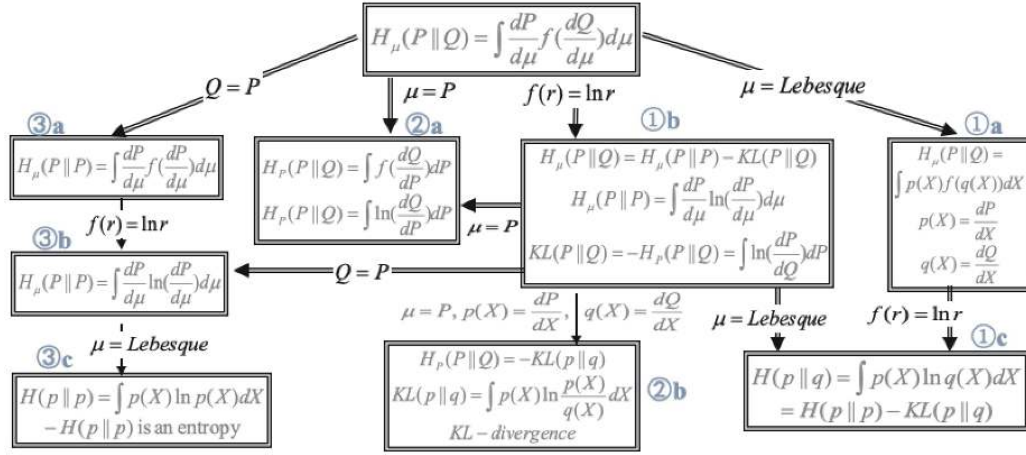


Fig. 8. Harmony functional: A unified scheme for bi-entity proximity. Combinations of different specific settings of four components in Bayesian Ying Yang (BYY) system lead to a number existing learning models.

It follows that $\min_{\Theta} KL_h(p||q, \Theta)$ is equivalent to $\min_{\Theta} E_{p_h(X)} V_L(X|\Theta)$ because $E_{p_h(X)} \ln p_h(X)$ does not contain Θ , and also further equivalent to $\min_{\Theta} V_L(X_N|\Theta)$ as $h \rightarrow 0$, that is, Ying Yang best matching is equivalent to the variational learning (see the 2nd line of equations in Fig. 6).

Also, we may get

$$\begin{aligned} H_h(p||q, \Theta) &= E_{p_h(X)} E_{p(Y|X)} \ln [q(X|Y, \Theta_{X|Y}) q(Y|\Theta_Y)] \\ &= \int p(Y|X) p_h(X) \ln [q(X|Y, \Theta_{X|Y}) q(Y|\Theta_Y)] dX dY, \\ \max_{h \rightarrow 0} H_h(p||q, \Theta) &\text{ leads to } \\ p(Y|X_N, \Theta_{Y|X}) &= \delta(Y - f(X_N, \Theta_{Y|X})) \text{ and } \\ \max_{\Theta} H_{gLmser}(\Theta) &= [\ln q(X_N|Y, \Theta_{X|Y}) + \ln q(Y|\Theta_Y)]_{Y=f(X_N, \Theta_{Y|X})}. \end{aligned} \quad (12)$$

When $q(Y|\Theta_Y)$ can be approximately regarded as being irrelevant to Y (i.e., a uniform distribution) and $q(X_N|Y, \Theta_{X|Y}) = G(X_N|g(Y, \Theta_{X|Y}), \sigma^2 I)$, we have that $\max_{\Theta} H_{gLmser}(\Theta)$ is approximately equivalent to learning deep autoencoder

$$\min_{\Theta_{X|Y}, \Theta_{Y|X}} \|X_N - g(f(X_N, \Theta_{Y|X}), \Theta_{X|Y})\|^2, \quad (13)$$

when $f(X_N, \Theta_{Y|X})$ and $g(Y|\Theta_{X|Y})$ are implemented by deep networks, from which we are further lead to Lmser by weight sharing $A = W^T$ and neuron pairing $\eta_i = \zeta_i$.

In summary, we have

- Variational learning can be regarded as a special case of BYY best matching by $\min_{\Theta} KL_h(p||q, \Theta)$. The general case also covers other existing methods, as addressed in the next subsection, especially in Fig. 9 and Table IV.

- Deep autoencoder and Lmser can be approximately regarded as a special case of $\max_{\Theta} H_{gLmser}(\Theta)$ without a regularisation term $\ln q(f(X_N, \Theta_{Y|X})|\Theta_Y)$. The general case of BYY best harmony by $\max_{\Theta} H_h(p||q, \Theta)$ covers not only existing learning methods but also new learning methods, as to be addressed in the next subsection, especially in Fig. 10 and Table V.

IV. BAYESIAN YING YANG LEARNING: EXISTING STUDIES AND BEYOND

A. BYY Best Matching: Exemplars

BYY best matching learning was first proposed in 1995 by using Kullback-Leibler divergence to measure the discrepancy between Ying and Yang [12], [13], shortly denoted by BYY-KL. In addition to merely considering minimisation, we may also consider maximisation, there are four combinations to handle $KL(p||q)$. Two of them make sense while the other two do not. We may also consider $H(p||q)$, there are four combinations too. Again, one half makes sense while the other does not, as summarised in Fig. 9(a), where $KL(p||q)$ is in a notation $K(M_A, M_I)$.

$\text{Min}_{M_A} \text{Min}_{M_I} K(M_A, M_I)$	$\text{Max}_{M_A} \text{Min}_{M_I} K(M_A, M_I)$	$\text{Min}_{M_A} \text{Max}_{M_I} K(M_A, M_I)$	$\text{Max}_{M_A} \text{Max}_{M_I} K(M_A, M_I)$
$\text{Min}_{M_A} \text{Min}_{M_I} K(M_I, M_A)$	$\text{Max}_{M_A} \text{Min}_{M_I} K(M_I, M_A)$	$\text{Min}_{M_A} \text{Max}_{M_I} K(M_I, M_A)$	$\text{Max}_{M_A} \text{Max}_{M_I} K(M_I, M_A)$

(Xu, NIPS1995, A Unified Learning Scheme: Bayesian-Kullback Ying Yang Machine)

$\text{Min}_{M_I} K$ drives M_I to fit p_h (i.e., samples of x), which make sense and is thus denoted by .

Add Min_{M_A} on $\text{Min}_{M_I} K$ drives M_A to enhance that M_A fits p_h and thus denoted by .

Add Max_{M_A} on $\text{Min}_{M_I} K$ to wipe out the contribution from M_A such that ' M_I fits p_h ' is robust and thus may be named Robust learning (RL), which is denoted by .

$\text{Max}_{M_I} K$ drives M_I away from p_h (i.e., samples of x), which does not make sense and is denoted by .

(a)

$$K_{AI} = K(M_A, M_I) = KL(p_{M_A} \| q_{M_I}),$$

$$K_{AI} = E_{p_{M_A}} \ln \frac{p_{M_A}}{q_{M_I}} = \chi_{M_A} - H_{AI},$$

$$\chi_{M_A} = E_{p_{M_A}} \ln p_{M_A} = \chi_{p_h} + E_{p_h} \chi_{p_{y|x}},$$

$$\chi_{p_h} = E_{p_h} \ln p_h,$$

$$E_{p_h} \chi_{p_{y|x}} = E_{p_h} E_{p_{y|x}} \ln p_{y|x},$$

$$H_{AI} = E_{p_{M_A}} \ln q_{M_I} = H_{AI}^i + H_{AI}^c,$$

$$H_{AI}^i = E_{p_h} E_{p_{y|x}} \ln q_{y|y} = E_{p_y} \ln q_{y|y},$$

$$p_y = E_{p_h} p_{y|x},$$

$$H_{AI}^c = E_{p_h} E_{p_{y|x}} \ln q_{x|y}.$$

Yang	Ying
M_A, p, p_{M_A}	M_I, q, q_{M_I}

$$p_h(X_N) = \prod_i G(x | x_i, hI),$$

$$p_h(x) = \frac{1}{N} \sum_i G(x | x_i, hI).$$

$$\chi_{p_h} \xrightarrow{h \rightarrow 0} -0.5d \ln(2\pi eh),$$

$$E_{p_h} \chi_{p_{y|x}} \xrightarrow{h \rightarrow 0} \frac{1}{N} \sum_i E_{p_{y|x_i}} \ln p_{y|x_i},$$

$$H_{AI}^i \xrightarrow{h \rightarrow 0} \frac{1}{N} \sum_i E_{p_{y|x_i}} \ln q_{y|y},$$

$$p_y \xrightarrow{h \rightarrow 0} \frac{1}{N} \sum_i p_{y|x_i},$$

$$H_{AI}^c \xrightarrow{h \rightarrow 0} \frac{1}{N} \sum_i E_{p_{y|x_i}} \ln q_{x|y},$$

(b)

$$K_{IA} = K(M_I, M_A) = KL(q_{M_I} \| p_{M_A}),$$

$$K_{IA} = E_{q_{M_I}} \ln \frac{q_{M_I}}{p_{M_A}} = \chi_{M_I} - H_{IA},$$

$$\chi_{M_I} = E_{q_{M_I}} \ln q_{M_I} = \chi_{q_y} + E_{q_y} \chi_{q_{x|y}},$$

$$\chi_{q_y} = E_{q_y} \ln q_y,$$

$$E_{q_y} \chi_{q_{x|y}} = E_{q_y} E_{q_{x|y}} \ln q_{x|y},$$

$$H_{IA} = E_{p_{M_A}} \ln p_{M_A} = H_{IA}^i + H_{IA}^c,$$

$$H_{IA}^i = E_{q_y} E_{q_{x|y}} \ln p_h = E_{q_x} \ln p_h,$$

$$q_x = \int q_{x|y} q_y dy,$$

$$H_{IA}^c = E_{q_y} E_{q_{x|y}} \ln p_{y|x},$$

Fig. 9. Bayesian Ying Yang learning via KL divergence (shortly, BYY-KL). (a) Major interactions of Ying-Yang via KL divergence (b) Two directional KL measures for Ying-Yang interactions.

TABLE IV
FIVE SPECIAL CASES OF BYY-KL AND RELATIONS TO TYPICAL EXISTING METHODS

A	B	C	D	E
<p>$\text{Min}_{M_A, M_I} K(M_A, M_I)$ becomes equivalent to minimize $KL(p_h(x) \ q(x \theta_I))$, Smooth ML - G when $h=0 \Rightarrow$ maximize $\ln q(X_N q_I)$ ML - G</p> <p>$\text{Min}_{M_A, M_I} K(M_I, M_A) \Leftrightarrow$ $\text{Min } KL(q(x \theta_I) \ p_h(x))$ Variant of Maxent</p>	<p>$\because E_{p(Y X_N, \theta_{yI})} \ln p(Y X_N, \theta_{yI})$ is irrelevant, $\text{Min}_{M_A, M_I} K(M_A, M_I)$ $\xrightarrow{h \rightarrow 0}$ maximize $\ln q(X_N f(X_N, \theta_{yI}), \theta_{yI}) + \ln q(f(X_N, \theta_{yI}) \theta_y)$ Regularized Reconstruction</p> <p>$\xrightarrow{\text{Special Cases}}$ LMSER</p>	<p>$\text{Min}_{M_A, M_I} K(M_A, M_I) \Leftrightarrow \text{Min}_{M_A, M_I} K(M_A, M_I) - E_{p_h} \ln p_h \xrightarrow{h \rightarrow 0}$ variational learning that minimize $E_{p(Y X_N, \theta_{yI})} \ln \frac{p(Y X_N, \theta_{yI})}{q(Y \theta_y)} - E_{p(Y X_N, \theta_{yI})} \ln q(X_N Y, \theta_{yI})$ VL - G</p> <p>$\text{Min}_{M_A, M_I} K(M_I, M_A) \Leftrightarrow$ minimize $K(M_I, M_A) = \chi_{M_I} - H_{IA}$ New</p>	<p>$\text{Min}_{M_A, M_I} K(M_I, M_A) \Leftrightarrow$ maximize $E_{q(y)} \ln p_h(g(y \theta_{yI})) + E_{q(y)} \ln p(y g(y \theta_{yI})) - E_{q(y)} \ln q(y)$ Bidirectional Reconstruction</p>	<p>$\text{Min}_{M_A, M_I} K(M_A, M_I) \Leftrightarrow$ minimize $KL(p(y \theta_A) \ q(y \theta_y))$ INFORMAX - ICA MMI - ICA BSO - vision</p> <p>$p(y \theta_A) = \int p(y x, \theta_{yI}) p_h(x) dx$</p> <p>$\text{Min}_{M_A, M_I} K(M_I, M_A) \Leftrightarrow$ minimize $KL(q(y \theta_y) \ p(y \theta_A))$</p>

Given in Fig. 9(b) are detailed components of $KL(p \| q)$ and $KL(q \| p)$, while given in Table IV are five special cases of BYY-KL, corresponding to five special structures. The first three come from three types of specific structures of $p(Y|X)$. Type A leads to two existing methods on Yang domain,

namely ML and a variant of Maxent [69], plus possible extensions [59] by using a smoothed density $p_h(X)$ with $h \neq 0$ instead of directly using data X_N . The last three shown in Table IV come from three types of specific structures of $q(X|Y)$. Type E leads to two existing typical methods on the

$\text{Max}_{M_A} \text{Max}_{M_I} H(M_A, M_I)$	$\text{Min}_{M_A} \text{Max}_{M_I} H(M_A, M_I)$	$\text{Min}_{M_A} \text{Max}_{M_I} H(M_A, M_I)$	$\text{Min}_{M_A} \text{Min}_{M_I} H(M_A, M_I)$
$\text{Max}_{M_A} \text{Max}_{M_I} H(M_I, M_A)$	$\text{Min}_{M_A} \text{Max}_{M_I} H(M_I, M_A)$	$\text{Min}_{M_A} \text{Max}_{M_I} H(M_I, M_A)$	$\text{Min}_{M_A} \text{Min}_{M_I} H(M_I, M_A)$

(a)

$$H_{AI} = H(M_A, M_I) = \chi_{M_A} - KL(p_{M_A} \| p_{M_I}) = E_{p_{M_A}} \ln q_{M_I},$$

When the variance of $p_{y|x}$ is not lower bounded, i.e., $\text{Var}(p_{y|x}) \geq 0$, we get that

$$\bullet \text{Max}_{M_A} H(M_A, M_I) \xrightarrow{\text{Var}(p_{y|x}) \geq 0} \text{a } \delta\text{-Yang mapping } p_{y|x} = \delta(y - f(x, \theta_{y|x})),$$

$\text{Max}_{M_A, M_I} H(M_A, M_I)$ is accordingly named **δ -Yang BYY-HL**,

which is actually equivalent to $\text{Min}_{M_A, M_I} K(M_A, M_I)$ on BYY system (B), i.e.

Regularized Reconstruction that maximizes $\ln q(X|f(X_N, \theta_{y|x}), \theta_{x|y}) + \ln q(f(X_N, \theta_{y|x})|X_N)$.

$$\bullet \text{Min}_{M_A} \text{Max}_{M_I} H(M_A, M_I) \text{ is equivalent to } \text{Max}_{M_A} \text{Min}_{M_I} K(M_A, M_I) \text{ on BYY system (B),}$$

which wipes out the contribution from M_A such that ' M_I fits p_h ' is **δ -Yang BYY-RL**.

(b)

$$H_{IA} = H(M_I, M_A) = \chi_{M_I} - KL(q_{M_I} \| p_{M_A}) = E_{q_{M_I}} \ln p_{M_A}, \text{ similarly we get}$$

$$\bullet \text{Max}_{M_A} H(M_A, M_I) \xrightarrow{\text{Var}(q_{x|y}) \geq 0} \text{a } \delta\text{-Ying mapping } q_{x|y} = \delta(x - g(y, \theta_{x|y})),$$

$\text{Max}_{M_A, M_I} H(M_I, M_A)$ is accordingly named **δ -Ying BYY-HL**,

which is equivalent to $\text{Min}_{M_I, M_A} K(M_I, M_A)$ on BYY system (D), i.e. **Bidirectional Reconstruction**

that maximizes $E_{q(y)} \ln p_h(g(y|\theta_{x|y})) + E_{q(y)} \ln p(y|g(y|\theta_{x|y})) - E_{q(y)} \ln q(y)$.

$$\bullet \text{Min}_{M_I} \text{Max}_{M_A} H(M_I, M_A) \Leftrightarrow \text{Max}_{M_I} \text{Min}_{M_A} K(M_I, M_A) \text{ on BYY system (D), i.e., } \delta\text{-Ying BYY-RL.}$$

(c)

Fig. 10. Bayesian Ying Yang harmony learning (shortly, BYY-HL). (a) Major interactions of Ying-Yang harmony (b) Type AI : relation of BYY-HL to BYY-KL. (c) Type IA: relation of BYY-HL to BYY-KL.

TABLE V
MAJOR INTERACTIONS OF YING-YANG IN BALANCED SYMMETRY

$\text{Min}_{M_A} \text{Min}_{M_I} K_M$	$\text{Max}_{M_A} \text{Min}_{M_I} K_M$	$\text{Min}_{M_A} \text{Max}_{M_I} K_M$	$\text{Max}_{M_A} \text{Max}_{M_I} K_M$
$\text{Max}_{M_A} \text{Max}_{M_I} H_M$	$\text{Max}_{M_A} \text{Min}_{M_I} H_M$	$\text{Min}_{M_A} \text{Max}_{M_I} H_M$	$\text{Min}_{M_A} \text{Min}_{M_I} H_M$

$$K_M = \omega K(M_A, M_I) + (1 - \omega) K(M_I, M_A) \quad 0 \leq \omega \leq 1 \quad H_M = \omega H(M_A, M_I) + (1 - \omega) H(M_I, M_A)$$

$$KL(M_A \| M_I) = (1 - \tau) KL^u(M_A \| M_I) + \tau KL^s(M_A \| M_I), \quad H(M_A, M_I) = (1 - \tau) H^u(M_A, M_I) + \tau H^s(M_A, M_I)$$

$$KL(M_I \| M_A) = (1 - \tau) KL^u(M_I \| M_A) + \tau KL^s(M_I \| M_A), \quad H(M_I, M_A) = (1 - \tau) H^u(M_I, M_A) + \tau H^s(M_I, M_A)$$

where $0 \leq \tau \leq 1$ and superscripts: u denotes "unsupervised", s denotes "supervised".

inner coding domain or Ying domain, namely ICA [63] and BSO [64] mentioned previously in Section III-B. Type C locates at the middle with both $p(Y|X)$ and $q(X|Y)$ in a general structure, which leads to variational learning (VL) previously mentioned after Eq.(11).

Type B and Type D are points where BYY-KL meets BYY best harmony learning by $\max H(p||q)$ (shortly denoted by BYY-HL). Replacing **KL** by **H** and swapping 'max' and 'min', we can turn Fig. 9(a) into Fig. 10(a). It follows from Eq.(12) that $\max H(p||q)$ automatically drives a $p(Y|X)$ that is free of structure become a δ structure of Type B, which may be shortly referred as δ -Yang BYY-HL. Actually, it further develops autoencoder and Lmser into $\max_{\Theta} H_{g\text{Lmser}}(\Theta)$ with a priori regularisation $\ln q(f(X_N, \Theta_{Y|X})|\Theta_Y)$ added in for regularization. In this case, χ_p becomes irrelevant to learning, and thus Type B in Table IV implements δ -Yang BYY-KL that is actually equivalent to δ -Yang BYY-HL. Similarly, with $KL(q||p)$ in place of $KL(p||q)$, Type D in Table IV implements δ -Ying BYY-KL that is equivalent to δ -Ying BYY-HL,

aiming at a bidirectional reconstruction.

B. BYY Best Harmony: Exemplars

The BYY learning was firstly made on $\min KL(p||q)$ in 1995 [12], [13] and also in subsequent years. Since 2000 [66], studies further proceeded to consider both $\min KL(p||q)$ and $\max H(p||q)$. Also, at the end of the NIPS paper [13], a simplified version of $\max_p \min_q KL(p||q)$, or $\max_{M_A} \min_{M_I} K(M_A, M_I)$ in Fig. 9(a), was particularly suggested in a two step implementation. Its first step implements $\max_p K(p||q)$ that drives a free $p(Y|X)$ to become one δ structure, and its second step implements $\min_q KL(p||q)$. This is actually equivalent to alternation of two step implementation of $\max_p H(p||q)$ and $\max_q H(p||q)$ for $\max_{p,q} H(p||q)$ or $\max H(p||q)$. Readers are referred to [23], [25]–[27] for two decades of developments on BYY-KL and BYY-HL.

Strictly speaking, the above two step implementation is not exactly equivalent to jointly $\max_p \min_q KL(p||q)$ that drives a free $p(Y|X)$ to a δ structure of Type B but becomes equivalent

to

$$\begin{aligned} \min_{\Theta_Y|X} \max_{\Theta_X|Y, \Theta_Y} H_{gLmsr}(\Theta) \\ = [\ln q(X_N|Y, \Theta_X|Y) + \ln q(Y|\Theta_Y)]_{Y=f(X_N, \Theta_Y|X)}, \end{aligned} \quad (14)$$

where maximisation makes Ying to best fit samples while minimisation seeks the fitting more robust, and thus named as δ -Yang BYY-RL as illustrated in Fig. 10(b). Similarly, we also get δ -Ying BYY-RL as illustrated in Fig. 10(c).

In general, when $p(Y|X)$ is neither free nor in a δ structure, the above equivalences may not always hold. BYY-HL improves BYY-KL by its model selection nature.

Typically, some structural constraint $C(p, q) = 0$ is imposed on Ying and Yang to regularise learning, some examples are listed below:

- (1) $C(p, q) = KL(p(X, Y) \| q(X, Y)) = 0$, which leads to Lagrange-like implementations of BYY harmony learning [27];
- (2) $C(p, q) = U(p(X, Y)) - U(q(X, Y)) = 0$ under a uncertainty measure $U(p)$, e.g., as reviewed by Eqs.(9) and (11) in [27];
- (3) $Var(p) = U(p) = U(q) = Var(q)$, where $Var(p)$ is the variance of x for $p(x)$;
- (4) $\chi(p) = U(p) = U(q) = \chi(q)$, where $\chi(p)$ is the negative entropy of p .

Typically, under the constraint $Var(p(Y|X)) = c$ or equivalently $\chi_{path}(X) = \int p(Y|X) \ln p(Y|X) dY = c$, best harmony and best matching become equivalent, that is, we have

$$\begin{aligned} \max_{\text{s.t. } Var(p(Y|X))=c} H(p \| q) \text{ is equivalent to } \min_{\text{s.t. } Var(p(Y|X))=c} \mathbf{KL}(p \| q), \end{aligned} \quad (15)$$

which trends to δ -Yang BYY-KL and δ -Ying BYY-HL as c gradually reduces to 0.

C. Advanced Topics: Symmetry, Mixture, GAN, and Creative Outputs

Advances of BYY-HL and BYY-KL are further made along four directions. The first is considering combinations of three sorts of varieties, namely (M_A, M_I) versus (M_I, M_A) , BYY-HL versus BYY-KL, and samples of supervised versus unsupervised. Simply, we consider linear combination as illustrated in Table V. In implementation, we handle the integral over Y with help of approximation by Taylor expansion, see Sect.4.3 in [23] and in [26], as well as Table 2 in [26].

The second direction is considering both $p(Y|X)$ and $q(Y|\Theta_Y)$ by mixture modelling [53], [70]–[73], as illustrated in Fig. 11.

The third direction is about generative adversarial networks (GAN) [30], [31], which has been recently a popular model to generate samples, especially image samples. As shown in Fig. 12(a), the key idea of GAN is treating generated samples $\{\hat{x}_t\}$ with a label 0 for fake together with real samples $\{x_t\}$ with a label 1 for real. A discriminator is trained to well classify the two types, while generative model is trained to get $\{\hat{x}_t\}$ to deteriorate the classification performance of the discriminator, until the adversarial reaches an equilibrium.

Illustrated on the right side of Fig. 12(a) is a BYY-GAN system for GAN like learning, which is degenerated from the BYY system illustrated in Fig. 12(b) with two types of inner

codes y, ς , where y is a code same as one in autoencoder and Lmsr, and ς is a label for classification. The corresponding BYY-HL learning is $\max_{M_A, M_I} H_{GAL}$. This degeneration has two features. One is that there is no information flow from Yang path to Ying domain. The other is $p_h(x)_{near x_t}$ that considers the real sample and the fake sample jointly.

The corresponding mathematical formulation is given in Fig. 12(c), from which we observe that $\max_{M_A, M_I} H_{IA}$ makes Yang path to discriminate real/fake and Ying path to generate \hat{x}_t to approach x_t as closely as possible. That is, it performs a GAN like task via BYY-HL, which is not adversarial but actually in harmony. In other words, adversary is not an intrinsic nature but just the other one of two aspects to view such tasks. Interestingly, the BYY system illustrated in Fig. 12(b) together with $\max_{M_A, M_I} H_{GAL}$ integrates the strength of GAN like learning where x_t and \hat{x}_t are not linked via Ying domain and the strength of autoencoder/Lmsr like learning where x_t and \hat{x}_t are linked via $x_t \rightarrow y_t \rightarrow x_t$, covering not only sample generation but also those tasks performed by Lmsr.

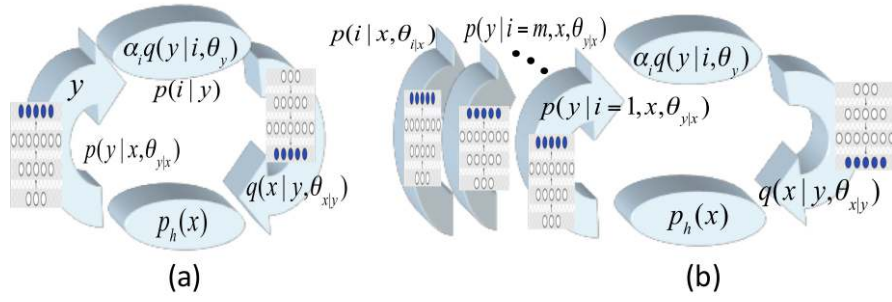
Last but not least, the fourth direction is developing bidirectional deep system for synthesising and creative outputs, as further addressed in sequel.

In the same year of AI Dartmouth 1956 symposium, the first proposed model of the famous Bloom Educational Target Pyramid was also initiated. The Pyramid describes the six levels of cognitive ability in human education, placing the synthesis and assessment capabilities at the top two. The term synthesis was changed to the term creativity and further shifted to the top of the Pyramid in its 2001 revision. The development of artificial intelligence and creativity research is closely intertwined. Since 1999, AAAI, IJCAI, ECAI, and other AI related conferences in Europe and the United States have held symposiums on creative intelligence each year.

The first wave of artificial intelligence in China began in the early 1980s, when artificial intelligence dominated by symbolic reasoning reached a climax internationally. At that time, Chinese scientist Qian Xuesen already began to advocate thinking sciences, including artificial intelligence. He thought that image thinking plays a leading role in creative process, and foresighted studying image thinking will be a breakthrough point for thinking sciences [74]. Pan developed the thought in 1996 [75] and believed that reasoning research proceeded gradually from deductive logic to visual reasoning and demonstrated such a loosening tendency in a reasoning process, which gradually leads reasoning research to thinking simulation. Moreover, he proposed a synthesis reasoning model, expounded the relationship with image thinking, and compared the characteristics with traditional reasoning.

Recent studies relate to indirectly or directly synthesis reasoning and image thinking, and involve various aspects such as news writing, composing and painting, dance animation, advertising videos, and intelligent design.

As illustrated in Fig. 13(a), BYY learning can handle various transform problems from an input x into an output z . Typically, both x and z are represented by multidimensional arrays (e.g., vector or 1D image, image, 3D image, etc.). The flow $x \rightarrow z$ indicates an inferring or a step of thinking directly in image format. Specifically, synthesizing and creating are



$$\begin{aligned}
 & \text{Max}_{M_A, M_I} H(M_A, M_I), \\
 & H(M_A, M_I) = H(p(y|x, \theta_{yx}) p(i|y, \theta_{iy}) p_h(x) \| q(x|y, \theta_{xy}) q(y|i, \theta_y) \alpha_i) \\
 & = \sum_i \int p(y|x, \theta_{yx}) p(i|y, \theta_{iy}) p_h(x) \ln[q(x|y, \theta_{xy}) q(y|i, \theta_y) \alpha_i] dx dy, \text{ for Fig. (a),} \\
 & H(M_A, M_I) = H(p(y|x, i, \theta_{yx,i}) p(i|x, \theta_{ix}) p_h(x) \| q(x|y, \theta_{xy}) q(y|i, \theta_y) \alpha_i) \\
 & = \sum_i \int p(y|x, i, \theta_{yx,i}) p(i|x, \theta_{ix}) p_h(x) \ln[q(x|y, \theta_{xy}) q(y|i, \theta_y) \alpha_i] dx dy, \text{ for Fig. (b),} \\
 & \text{Max}_{M_A, M_I} H(M_I, M_A) \\
 & H(M_I, M_A) = H(q(x|y, \theta_{xy}) q(y|i, \theta_y) \alpha_i \| p(y|x, \theta_{yx}) p(i|y, \theta_{iy}) p_h(x)) \\
 & = \sum_i \int q(x|y, \theta_{xy}) q(y|i, \theta_y) \alpha_i \ln[p(y|x, \theta_{yx}) p(i|y, \theta_{iy}) p_h(x)] dx dy, \text{ for Fig. (a),} \\
 & H(M_I, M_A) = H(q(x|y, \theta_{xy}) q(y|i, \theta_y) \alpha_i \| p(y|x, i, \theta_{yx,i}) p(i|x, \theta_{ix}) p_h(x)) \\
 & = \sum_i \int q(x|y, \theta_{xy}) q(y|i, \theta_y) \alpha_i \ln[p(y|x, i, \theta_{yx,i}) p(i|x, \theta_{ix}) p_h(x)] dx dy, \text{ for Fig. (b),}
 \end{aligned}$$

Fig. 11. Mixture-of-experts and BYY learning. (a) Mixture modelling merely in the encoding domain. (b) Mixture modelling in both Yang pass and encoding domain.

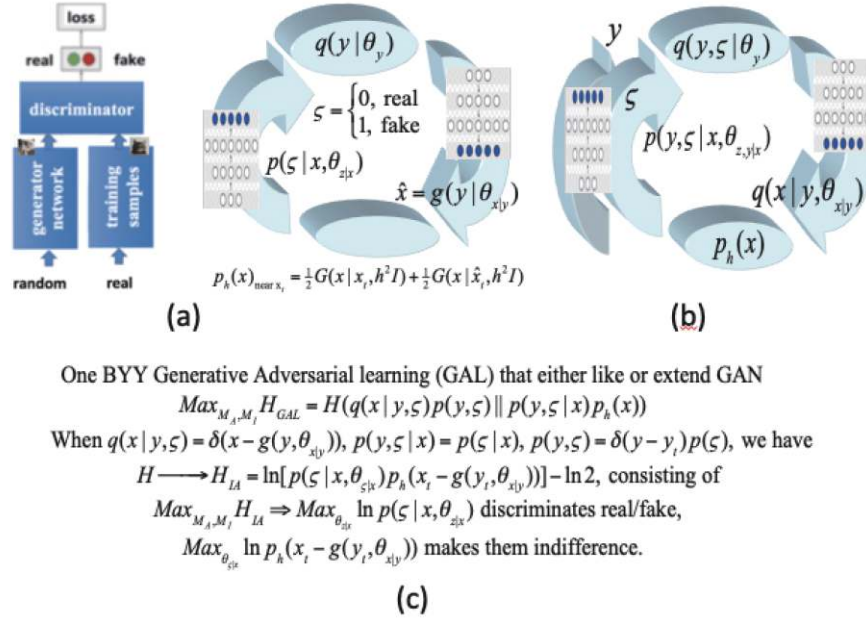


Fig. 12. Generative adversarial networks (GAN) and BYY learning. (a) A BYY system for GAN like learning (BYY-GAN). (b) BYY system for both GAN like learning and Lmsr like learning. (c) The corresponding mathematical formulations

performed with help of certain operators in coding domain. For example, the code c_A for Pattern A and the code c_B for Pattern B are combined or integrated to yield a creative code c_C by interpolation, extrapolation, and transform via some linear or convex combination. Then, the code c_C drives the Ying path or the I-mapping to generate the corresponding pattern in the data domain, where some evaluating criterion or procedure is used to check whether this pattern is taken as an acceptable creation..

There are various ways to make synthesising and creating,

each of which may be classified by taxonomy in a three-dimensional classification ICO array, as illustrated in Fig. 13(c), where I is the abbreviation of input or ideas, O is the abbreviation of output or originality, and C refers to a creative operation in the field of creative coding or contemplation. For each specific case, an input I of a specific value or instance is mapped to the C domain, and then a specific creative operation combines one or more specific instances in the C domain to be used as the code key that further activates the decoder to generate a specific creative output.

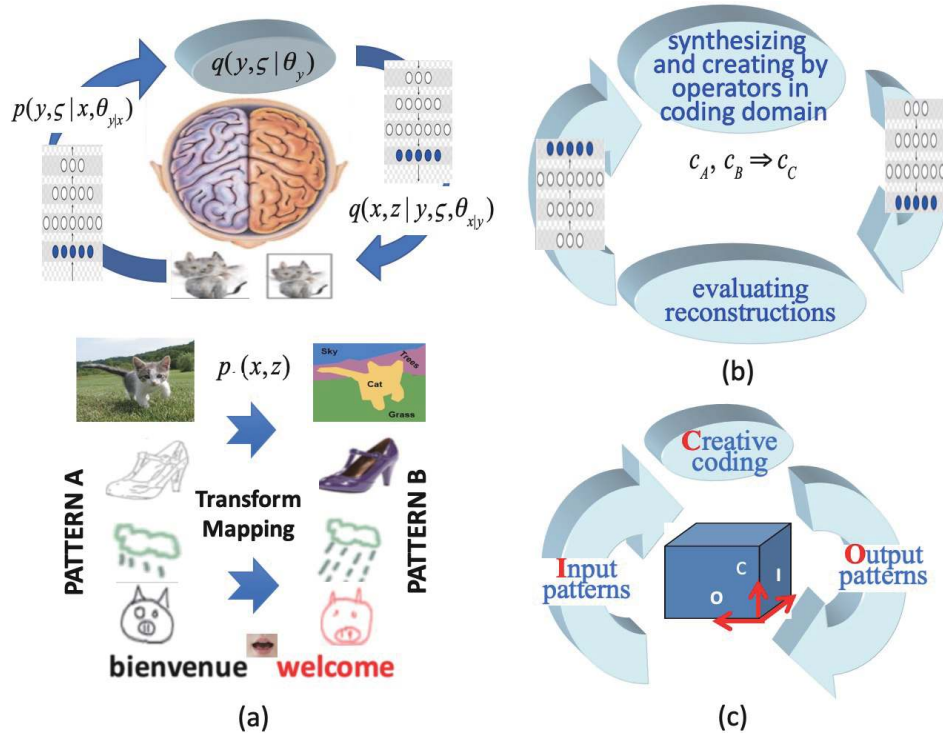


Fig. 13. BYY learning: reconstruction, transform problems, and creative tasks. (a) BYY learning for various transform problems. (b) Deep synthesising and deep creating (c) An ICO taxonomy for synthesising and creating tasks.

V. DEEP BIDIRECTIONAL INTELLIGENCE

A. IA System and BYY Intelligence Potential Theory

In addition to merely bidirectional learning, an intelligent system is intrinsically bidirectional as a whole. As illustrated in Fig. 14, any system alive in its environment needs not only inward cognition from the actual visible world or yAng domain (A-domain) but also outward interaction from the invisible internal brain or yIng domain (I-domain) back to yAng domain. Inward direction is featured by A-mapping $X \rightarrow Y$ that involves various *abstraction* or *cognition* activities related to words with an initial character “A”, such as Assort (recognise, classify, etc.) with Y taking one of discrete values, Aware (percept, conscious, etc.) with Y being certain coding patterns (vectors, icons, etc), and Abstract (induce, conceptualise, etc.), which activate the internal brain *thinking* that further drives the outward yIng pathway (shortly I-mapping) $Y \rightarrow Z$ to *implement* intelligent activities. In short, an intelligent system is one exemplar of yIng yAng system or shortly IA system.

Typically, intelligent activities are further grouped into two categories as follows:

Image Thinking. the mapping $X \rightarrow Z$ starts from and ends at the real world through continuous mapping functions $X \rightarrow Y^c$ and $Y^c \rightarrow Z$, featured by an information flow in a format of multidimensional arrays and especially image sequence, that is, as if thinking process was displayed in the real world. As addressed by a few paragraphs at the end of the last section, recent popular AI studies either indirectly or directly relate to image thinking, and thus echo Qian's views on the importance of image thinking [74] and Pan's prediction on the reasoning tendency [75].

In sequel, we further detail image thinking activities from the following two aspects:

(a) $Z = X$ that is, $X \rightarrow Z$ approximates one identical mapping such that Z is a reconstruction that approximates X as closely as possible, which not only verifies or calibrates the inward cognition and the corresponding internal representation but also guides learning on unknown parameters in both the pathways and Ying domain. After learning, Ying pathway generates a data distribution that approximates the distribution of training samples of X , and thus this part is often called generative model. This activity sets up a foundation that makes the intelligent system gain basic information processing ability for cognition and interaction with its world, via building up an A-mapping $X \rightarrow Y^c$ and an I-mapping $Y^c \rightarrow Z$ with help of certain bidirectional learning methods, e.g., autoencoder, Lmsr, and others.

(b) $Z \neq X$ that is, $X \rightarrow Z$ generates various patterns of Z for interpreting, interacting, and communicating with its outside, which may be further considered in the following scenarios:

- abstraction mapping $X \rightarrow Z$ maps from a high complexity image into a much simpler one such that Z reflects key structure or feature of X . An example is image segmentation.
- enrichment mapping $X \rightarrow Z$ maps from a low complexity pattern into a high complexity pattern such that Z represents a sort of enrichment of X . An example is image super resolution.
- pattern transformation $X \rightarrow Z$ maps from one pattern or style to another one, e.g., language translation and art style transform, while the complexity of X and Z are not different from each other considerably.
- imagining and creating $X \rightarrow Z$ maps from one pattern to one with a large dissimilarity, but still share some essential spirit, which may be classified by the ICO array in Fig.13.

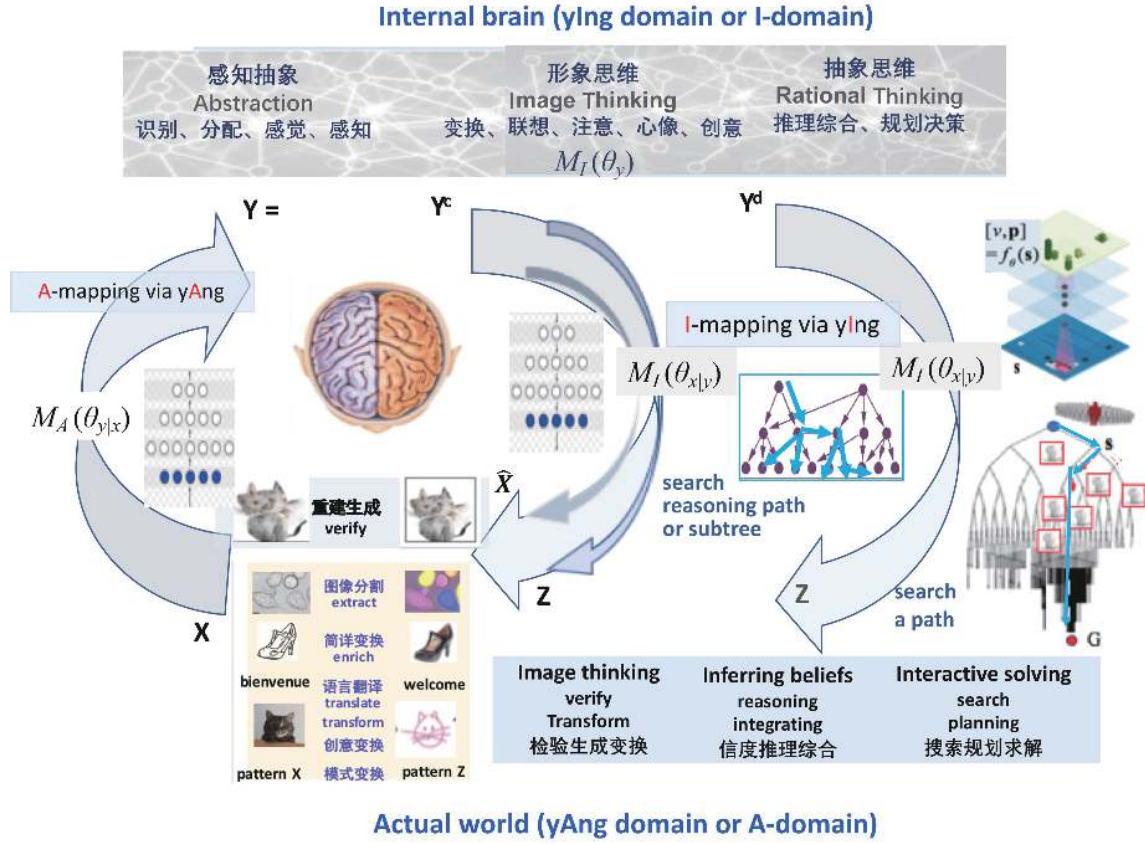


Fig. 14. Deep bidirectional Intelligence via Ying Yang or IA system.

Rational Thinking. $X \rightarrow Z$ involves internal brain activities in the I-domain of Y to handle abstracted or symbolised concepts represented in a discrete structure $M_I(\Theta_{x|y})$ (e.g., string, tree, and networks as illustrated in Fig. 14), which outcome Y^b that is usually a set of ordered binary variables or structured in a tree. The process of $X \rightarrow Y^b$ is internal and may further trigger a specific process $Y^b \rightarrow Z$ to its external world. Two typical types of rational thinking are the following two main themes of AI studies:

- (a) inferring beliefs e.g., logic reasoning, Bayesian networks, constraint relaxation, and causal analysis;
- (b) interactive deep solving e.g., search, decision, and planning.

Both the two themes encountered intractable computing difficulties in the traditional AI studies. In recent years, AlphaGo [76] and AlphaGoZero [77] have achieved surprisingly significant advances for tackling these difficulties, featured by using an A-mapping $X \rightarrow Y^c$ to help rational thinking $Y^c \rightarrow Y^b$. Further details will be addressed in the subsequent two subsections.

Insights on the IA system may be further obtained from observing three groups of dualities summarised in Table VI. The first group consists of extensions of the first three dualities in Table I, featuring the hardware part of the IA system, that is, DBA of Lmser is extended into duality of Ying Yang as shown in Fig. 7, DPN is extended into DFF, and DCW may be directly adopted. The second group features how the IA system works by three core dualities. The first is

the front STM space versus the LTM back space, which accommodates the second duality of short term dynamics (STD) vs long term dynamics (LTD). Extending DSL in Table I, STD varies quickly and frequently to deal with short term activities in the front space, while LTD changes knowledges by updating Θ in the back LTM space for parameters Θ that stores knowledges in long term. Moreover, as further extension of DST in Table I, STD and LTD are guided by theories in the third duality, with details given as follows:

$$KL(p||q) = E_{p(X)}KL(p||q, X), \quad H(p||q) = E_{p(X)}H(p||q, X),$$

$$KL(p||q, X) = E_{p(Z|X)}E_{p(Y|X,Z)} \ln \frac{p(Y|X)p(Z, X)}{q(Z|Y, X)q(Y|X)}, \quad Y = \{Y^c, Y^b\}$$

$$H(p||q, X) = E_{p(Z|X)}E_{p(Y|X,Z)} \ln [q(Z|Y, X)q(Y|X)]$$

$$H_\eta(p||q, X) = H(p||q, X) - \eta KL(p||q, X)$$

$$STD \text{ theory: } \max_{p(Y|X,Z) \text{ s.t. } \mathcal{P}_{Y|XZ}} H_\eta(p||q, X),$$

$\mathcal{P}_{Y|XZ}$ is a set of $p(Y|X, Z)$ that satisfies certain constraints,

$$LTD \text{ theory: } \max_{\Theta_q} H(p||q),$$

Θ_q consists of parameters in q ,

(16)

where η is a coefficient like a temperature that starts at a high value and gradually anneals to zero as LTD makes $H(p||q)$ approach a maximum, and readers are referred to Eq. (13) in

TABLE VI
DUALITIES 3-3 IN DEEP BIDIRECTIONAL INTELLIGENCE

Dualities in architecture					
DIA in an overall view Duality of yIng vs yAng Extension of DBA in Tab.1		DFB for two successive layers Duality of forward vs backward Extension of DCW in Table 1		DFP within the paired layer Duality of fast-lane vs feedback Extension of DPN in Table 1	
Dualities in fundamentals			Dualities in interaction/implementation		
DLT: Duality in learning theory	DPD: Duality in paired dynamics	DAM: Duality in attention mechanism	DSP: Duality in supervision paradigm	DIT: Duality in image thinking	DRT: Duality in rational think
$\max_p [Y X, Z] H_\eta(p q, X)$ in front layer for perception in short term (ST) memory versus $\max_{\theta} H(p q)$ subject to DIA in back layer as learning in long term (LT) memory	ST dynamics $\tau_Y \frac{\partial p(Y X, Z)}{\partial t} \propto \frac{\partial H_\eta(p q, X)}{\partial p(Y X, Z)}$ versus LT dynamics $\tau_L \frac{\partial \theta}{\partial t} \propto \nabla_{\theta} H(\theta)$ (Detailed DPD in Table 1)	get similarity $S(X^{(j)}, \hat{X}_{(j+1)})$ or discrepancy $D(X^{(j)}, \hat{X}_{(j+1)})$ to switch over STD vs LTD adapt vs forget alert vs ignore (Detailed DAM in Table 1)	learning $X \rightarrow Y$ for classifying or encoding in supervised on labeled data versus unsupervised (DSP in Tab. 1)	$Y \rightarrow \hat{X} \approx X$ verify learning $X \rightarrow Y$ versus auto-recall generate $Y \rightarrow Z$ for pattern transform	$Y \rightarrow Y^b \rightarrow Z$ gets a path for decision, planning, and solving versus $Y \rightarrow Y^b$ reasoning & networks relaxation

[27] and its discussions. When $\eta = 0$, STD and LTD target at the same objective, i.e., best harmony. Without constraints of $\mathcal{P}_{Y|XZ}$, a free $p(Y|X, Z)$ tends to become deterministic $\delta(Y - f(X, Z))$ that may overfit the current sample X , which may make $H(p|q)$ quenches at a local maximum. When $\eta > 0$, STD makes a free $p(Y|X, Z)$ tend:

$$p(Y|X, Z) = \frac{[q(Z|Y, X)q(Y|X)]^{\frac{1+\eta}{\eta}}}{\int [q(Z|Y, X)q(Y|X)]^{\frac{1+\eta}{\eta}} dY}, \quad (17)$$

from which we see again that it tends to be deterministic as $\eta \rightarrow 0$ and to be Bayes posteriori as $\eta \rightarrow \infty$. More sophisticatedly, it has been addressed in [27] and especially around its Eq. (29) and Fig. 4 that η reflects one attention mechanism of the intelligent system, taking a large value when the system feels unfamiliar with the current observations.

Though Eq.(17) may be directly used in some cases, $p(Y|X, Z)$ by Eq.(17) is not available in many applications where the integral incurs for computational intractability. Typically, there are two alternative directions to proceed. One is using a parametric $p(Y|X, Z, \Theta_p)$ and making STD by

$$\frac{\partial \Theta_p}{\partial t} \propto \frac{\partial H_\eta(p|q, X)}{\partial \Theta_p}. \quad (18)$$

The other direction is getting a best value Y^* from q and considering

$$p(Y|X, Z, Y^*) \text{ with its mean or mode at } Y^* = \arg \max_Y [q(Z|Y, X)q(Y|X)]. \quad (19)$$

Specifically, when Y is a real vector or array, as usually encountered in image thinking, one example is Eqs. (35) and (36) in [26]; while when Y is a binary vector or array, Y^* is obtained by a discrete optimisation, as usually encountered in rational thinking, some further details will be addressed in the next subsection.

Jointly, the STD+LTD theory by Eq.(16) and dualities summarised in Table VI come up with a general thesis about bidirectional intelligence, named as BYY intelligence potential theory (BYY-IPT), with its key points summarised as follows:

- $H(p|q)$ in Eq.(16) indicates the system's intelligence potentiality of managing what it is encountering;
- As new events or samples are input to an IA-system, like J of Lmsr in Fig. 3, $KL(p||q, X)$ in Eq.(16) measures how much the system's potentiality drops, due to unknowns contained in inputs;
- There is an intrinsic mechanism that triggers less-energy-consuming STD dynamics to get p and perform intelligent activities, e.g., perception/abstraction, imaging thinking, and rational thinking, to reduce the stress caused by unknowns and to bring this potentiality drop back;
- When there are too much unknowns to be managed by the STD dynamics, the mechanism triggers more energy-

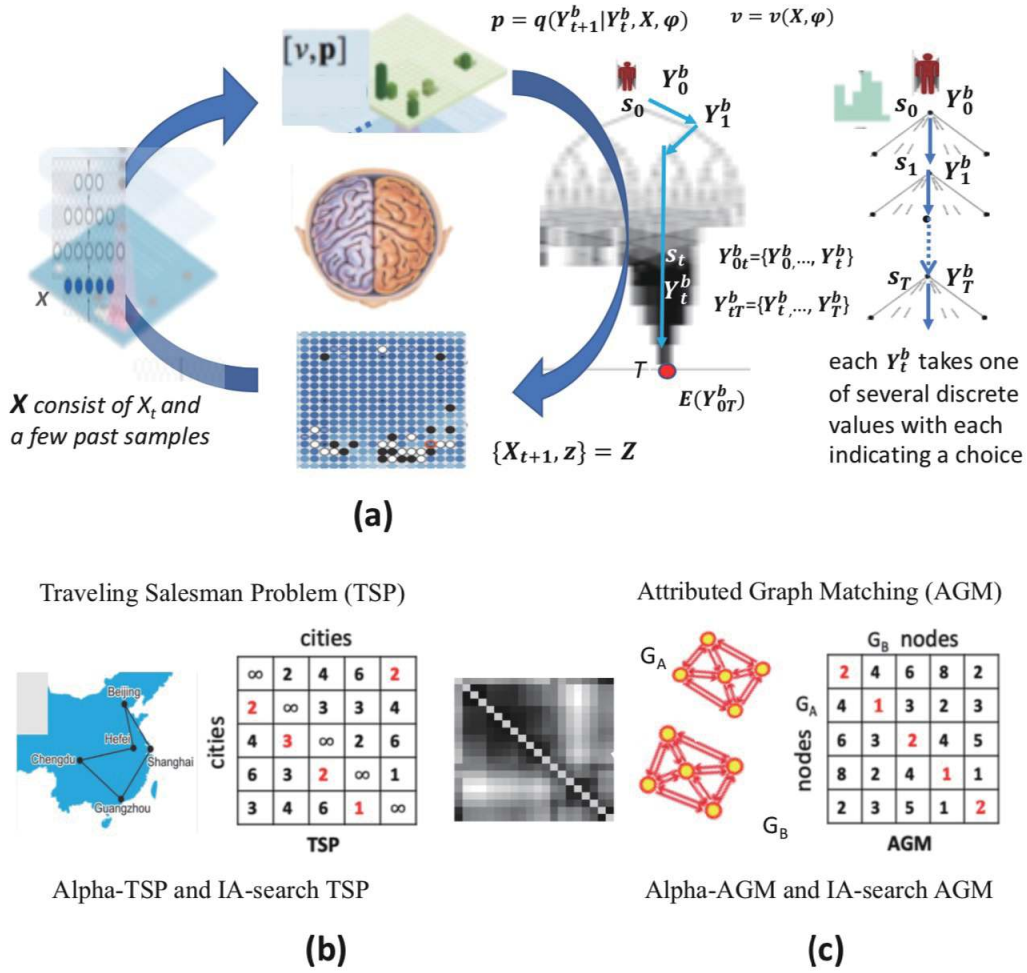


Fig. 15. AlphaGoZero and extensions from BYY-IPT perspective.

consuming LTD dynamics to update q to increase the intelligence potentiality $H(p||q)$, such that the STD dynamics continues and brings the potentiality drop back.

B. Deep Bidirectional Tree Searching

No doubly, Go-game, traveling salesman problem (TSP), attributed graph matching (AGM) are three of famous NP hard problem solving tasks. Typically, they are all formulated into tree search, as illustrated in Fig. 15(a). The solving process starts from the root to a target T at the depth d , featured by a sequence of steps with each selecting the next one among a number m of child nodes. Such a sequence is denoted by $Y_{0:t}^b = \{Y_0^b, \dots, Y_t^b\}$ with each Y_t^b taking one of several discrete values and each value indicating a choice. Each sequence $Y_{0:t}^b$ is associated with a measure $E(Y_{0:t}^b)$ with its structure and parameters Θ implicitly specified by the conditions or constraints that the target T satisfies, such that $E(Y_{0:t}^b)$ achieves its minimum for a sequence $Y_{t:T}^b$ that t ends at the target T .

For any sequence $Y_{0:d}^b$ that does not end at T but at an alternative node that locates also at the depth d , we have $E(Y_{0:d}^b) \geq E(Y_{0:T}^b)$ with $\Delta_d = E(Y_{0:d}^b) - E(Y_{0:T}^b)$ indicating a loss. It is a discrete process that searches $Y_{0:T}^b$ among m^d candidate sequences of $Y_{0:d}^b$ locating at the depth d , suffering a

complexity of the order $O(m^d)$, which is computationally intractable. This is a typical scenario encountered not only in solving Go-game, TSP, and AGM, but also in many combinatorial optimization tasks. Implementing such a discrete optimization is merely a part of the I-mapping study as illustrated in Fig. 14.

AlphaGo [76] and AlphaGoZero [77] represent one most popular achievement in the recent decade, breaking this traditional computing challenge by adding in A-mapping to perform a deep bidirectional tree searching. As illustrated in Fig. 15(a), there are three key points. First, a configuration X of Go chessboard (i.e., a 2D image) naturally specifies a state or node in the searching tree. In other words, a rather abstractly or symbolically represented state is attached with a representation X in a format that is able to accommodate rich information such that any two different states can get represented differently. Second, a Monte Carlo tree search (MCTS) is made by probabilistic policy $p = q(Y_{t+1}^b | Y_t^b, X)$ as a priori for searching ahead this state to collect the v -values for certain depth with their weighted average as an estimation of Q value. Third, Y_{t+1}^b , i.e., the next state to be expanded, is selected according to

$$\arg \max_{s_{t+1}} \{p(s_{t+1} | s_t) + [1 + N(s_t, s_{t+1})]Q(s_t, s_{t+1})\}, \quad (20)$$

in [76], [77], where $N(s_t, s_{t+1})$ is the number of times that

searching passes from s_t to s_{t+1} . This is conceptually consistent to make STD by Eq.(19) for selecting Y_{t+1}^b with $X = X_t$, $Y = \{v, Y_{t+1}^b, Y_{0:t}^b\}$ and $Z = \{z, X_{t+1}\}$ with $z = 1$ for winning and $z = 0$ for losing, such that

$$\begin{aligned} q(Y|X) &= q(v, Y_{t+1}^b, Y_{0:t}^b|X) = q(Y_{t+1}^b, Y_{0:t}^b|v, X_t)q(v|Y_{t+1}^b, Y_{0:t}^b, X_t) \\ &= q(Y_{t+1}^b|Y_t^b, X_t, \phi)q(v|Y_{t+1}^b, Y_{0:t}^b, X_t)q(Y_{0:t}^b|X_t, \phi), \\ q(Z|Y, X) &= q(X_{t+1}|z, v, Y_{t+1}^b, Y_{0:t}^b, X_t)q(z|v, Y_{t+1}^b, Y_{0:t}^b, X_t) \\ &\propto q(z|v, Y_{t+1}^b, Y_{0:t}^b, X_t), \end{aligned} \quad (21)$$

at the special cases that $q(Y_{0:t}^b|v, X_t, \phi) = q(Y_{0:t}^b|X_t, \phi)$ and $q(Y_{t+1}^b|Y_t^b, v, X_t, \phi) = q(Y_{t+1}^b|Y_t^b, X_t, \phi)$, and the last line comes from noticing that X_{t+1} comes deterministically from Y_{t+1}^b, X_t by the playing rules.

However, z is unavailable until the end of game. Instead, STD by Eq.(19) for selecting Y_{t+1}^b simply becomes

$$\arg \max_{Y_{t+1}^b} \ln q(Y|X) = [\ln q(Y_{t+1}^b|Y_t^b, X_t, \phi) + \ln q(v|Y_{t+1}^b, Y_{0:t}^b, X_t)], \quad (22)$$

where the two terms are conceptually consistent to their counterparts in Eq.(20).

We further consider the LTD learning in Eq.(16). Similarly, we can get that $H(p||q, X)$ in Eq.(16) becomes

$$\begin{aligned} H(p||q, X) &= E_{p(Y_{t+1}^b|Y_t^b, X_t, X_{t+1})p(Y_t^b|X_t, X_{t+1})} \ln [q(Y|X)q(Z|Y, X)] \\ &= H_A(p||q, X) + H_B(p||q, X) + E_{p(Y_t^b|X_t, X_{t+1})} H_C(p||q, Y_t^b), \\ H_A(p||q, X) &= E_{p(v|Y_{t+1}^b, Y_t^b, X_t)} \ln q(v|Y_{t+1}^b, Y_t^b, X_t), \\ H_B(p||q, X) &= E_{p(Y_t^b|X_t, X_{t+1})} \ln q(Y_t^b|X_t, \phi), \\ H_C(p||q, Y_t^b) &= E_{p(Y_{t+1}^b|Y_t^b, X_t, X_{t+1})} \\ &\quad \times \ln [q(Y_{t+1}^b|Y_t^b, X_t, \phi)q(z|v, Y_{t+1}^b, Y_t^b, X_t)]. \end{aligned} \quad (23)$$

As an example of Eq.(19), AlphaGoZero gets the posteriori $p(Y_{t+1}^b|Y_t^b, X_t, X_{t+1})$ by $\pi(s_{t+1}|s_t)$ obtained in a process that expands every son of s_t to subtrees by MCTS, where $\pi(s_{t+1}|s_t)$ is proportional to the number of nodes in each subtree. Further considering a Gaussian distribution $q(z|v, Y_{t+1}^b, Y_t^b, X_t)$, we may observe that maximising $H_C(p||q, Y_t^b)$ of the LTD learning is equivalent to minimising $(z - v)^2 - \pi \ln p$.

In a summary, STD dynamics and LTD learning in Eq.(16) are closely related to the algorithm of AlphaGoZero [77]. Further studies may be conducted at least from three aspects as follows:

- Examining pros and cons of making action by Eq.(20) versus Eq.(22), beyonds just knowing that two are conceptually consistent;
- Maximizing $H_B(p||q, X)$ reduces the entropy or the complexity of $p(Y_t|X_{t+1}, X_t) = q(Y_t|X_t)$, which is a favorable nature of BYY harmony learning [27]. It is interesting to investigate whether we may add in this term to improve the performance of alphaGoZero like problem solving.
- Recalling the path consistency nature addressed in [49] on its Eqs. (8)–(10), the v -values on one optimal path should be identical at one optimal value v^* or randomly deviates around

v^* subject to certain distribution. Simply, we may let v^* to be the average of the v -values on one path segment that we currently have found. Let

$$\ln q(v|Y_{t+1}^b, Y_t^b, X_t) \propto |v - v^*|^\gamma, \quad (24)$$

we observe that maximizing $H_A(p||q, X)$ imposes such path consistency intrinsically in the BYY IPT learning by Eq.(16).

• Taking the constraint $\mathcal{P}_{Y|XZ}$ in consideration of the STD, e.g., considering that $q(Y_{t+1}^b|Y_t^b, X_t, \phi)$ is doubly stochastic as follows:

$$\sum_{Y_{t+1}^b} q(Y_{t+1}^b|Y_t^b, X_t, \phi) = 1, \quad \sum_{Y_t^b} q(Y_{t+1}^b|Y_t^b, X_t, \phi) = 1. \quad (25)$$

Additionally, summarised by Table 1 in [49] is a family of variants and extensions of tree search technique used in AlphaGo [76] and AlphaGoZero [77], providing a number of topics for further investigation.

Next, we proceed to consider two other variants of AlphaGoZero. One is the following mixture-of-expert model [53], [78], [71]

$$\begin{aligned} q(Y_{t+1}^b|Y_t^b, X_t, \phi) &= \sum_j q(j|\phi_g)q(Y_{t+1}^b|Y_t^b, X_t, \phi_j), \\ v &= \sum_k q(k|\psi_g)f(X_t, \psi_k), \end{aligned} \quad (26)$$

where the gating networks $q(k|\phi_g), q(j|\psi_g)$, as well as all the expert networks $q(Y_{t+1}^b|Y_t^b, X_t, \phi_j)$, $f(X_t, \psi_k)$ can be all implemented by a same deep neural networks with multiple output ends. Learning is still made by getting gradients through the above equations. The other parts of implementations can be same as those in [76], [77] as well as recent extensions summarised by Fig. 3 in [49]. Such a ME based extension maybe named as ME Player.

The another is BYY FOLLOWER that uses a follower model q to learn from a master player p by maximizing $E_{p(X)}H(p||q, X)$ with

$$H(p||q, X) = E_{p(v|X_t)} \ln q(v|X_t) + E_{p(Y_{t+1}^b|Y_t^b, X_t)} \ln q(Y_{t+1}^b|Y_t^b, X_t, \phi), \quad (27)$$

which consists of two phases. In the first phase, the master plays per step to train the follower via increasing $H(p||q, X_t)$ by gradient learning, performed for a large enough number of games. In the second phase, the learned follower model q is used in most steps to learn from a master player p by the above maximisation. Occasionally, learning is made via playing by the master p . Such occasions could be picked randomly with a probability that gradually reduces to zero. Also, such occasion may be detected by fast lookahead playing or the discrepancy between the follower q and the master p . Such a two phase method is still applicable if we do not get p but see the moves that the master makes, simply letting $p(v|X_t) = 1$ and setting $p(Y_{t+1}^b|Y_t^b, X_t)$ to 1 at the choice that the move is made and to 0 at all the other choices.

In addition to playing Go, decades of efforts have been made on both TSP [79], [80] in the field of operational research and AGM [81], [82], [83] in the field of artificial intelligence. As mentioned in the previous subsection, these

typical rational thinking problems can not avoid intractable computing difficulties. The breaking through made by AlphaGo [76] and AlphaGoZero [77] is solving the problems by integrating rational thinking and image thinking with perception of image pattern X by an A-mapping $X \rightarrow Y$ to help discrete rational thinking, and also with an I-mapping $Y \rightarrow X$ to let internal abstract process to be displayed by image pattern X in its observation world.

As proposed recently in the last paragraph of the subsection “Deep learning, path consistency, and domain knowledge embedding” in [49], tasks of TSP and many problem solving tasks are formulated in a representation that may facilitate conventional computing but is difficult for making deep learning network to distinguish different states because state representation is usually discrete, symbolic, and conceptual. To extend the breakthrough by AlphaGo and AlphaGoZero, *feature enrichment* is proposed to explore, opposing to the direction of compressing data and/or removing redundancy. That is, we enrich compressed or simplified representation by restoring topological, neighbourhood, and association information into one of two enriched formats, e.g., one is letting a 2D or higher dimensional image as a state representation.

Also, it was suggested in [49] to turn samples of time series to 2D images by time-frequency analysis, to turn a state representation for TSP into a format of 2D image, and to consider natural language understanding by associating words with its corresponding speech signals or image patterns. Recently, the first suggestion has been already verified by one team including the present author on detecting of Coronary Artery Disease and Congestive Heart Failure from Electrocardiogram, with a significant improvement in accuracy.

In sequel, we propose a practical way for handling TSP and AGM. Illustrated in Fig. 15(b) and (c) are the state representation formats for TSP and AGM, respectively. Each of the problems becomes a playing on the corresponding game-board. For TSP of n cities, the digit locating at each position (i, j) is the distance between the city i and the city j . The playing rule is each time picking one pair among $0.5n(n-1)$ pairs of cities, swapping two cities of the picked pair, and getting winning when the sum f of cyclic sub-diagonal elements reaches the minimum.

Usually, the minimum value is unknown and estimated by the best traveling circle C_{best} and its corresponding distance sum f^* up to now. Each time, C_{best} and f^* are renewed if the current $f > f^*$. Observing $f > f^*$ consecutively for a pre-specified long period, we declare the wining and regard C_{best} and f^* as the solution. Moreover, we denote either a win with $v^* = 1$ if this f^* is smaller than a pre-specified threshold, otherwise a failure with $v^* = 0$.

For AGM, the digit locating at each position (i, j) on the game-board is the matching cost between the node i of graph G_A and the node j of graph G_B . The playing rule is swapping the node pairing of two node pairs per time until the sum of diagonal elements reaches the minimum. Since the minimum is usually unknown, we tackle the problem in a way similar to that for TSP.

With help of the reformulations above, TSP and AGM can be tackled in the same way as ones for implementing AlphaGo [76] and AlphaGoZero [77], as well as those discussed in Eqs.(20)–(23). Also, we further enforce that $q(Y_{t+1}^b | Y_t^b, X_t, \phi)$ satisfies the doubly stochastic constraint by Eq.(25). We are thus lead to a new direction for TSP and AGM with the resulted methods named as Alpha-TSP and Alpha-AGM. Also, instead of considering the binary value $v = 1$ or 0 , we may alternatively consider the value f directly by using ones in the deep IA-Search family (see Table I in Ref.[49]) and replacing the loss function by Eq. (8) or Eq. (9) in [49], with the resulted methods named as IA-search TSP and IA-search AGM.

C. Deep Bidirectional Reasoning

In a wide sense, reasoning is made on a networks with a set of nodes $\Xi = \{\xi_1, \xi_2, \dots, \xi_M\}$ and a set of edges, as illustrated in Fig. 16(a). Each edge connecting two nodes takes a role that is either a directional dependency similar to a *pre* \rightarrow *con* rule or generally a bidirectional dependence or constraint. Jointly, there are a set of constraints. Premises and conclusion can also be regarded as that the nodes of a subset $\Xi_I \subset \Xi$ are assigned to prefixed values, which further adds in a set of boundary constraints. The reasoning can be regarded as a process of seeking whether all the constraints are satisfied by assigning values to every node in the rest part $\Xi - \Xi_I$.

The extent about how much the constraints are satisfied can be measured by an energy like cost $E(\Xi | \Theta_r)$. With Ξ_I initialised to prefixed values and with those in $\Xi - \Xi_I$ initialised randomly, typically $E(\Xi | \Theta_r)$ is much bigger than E_0 because many of constraints have not been satisfied yet. Reasoning is thus a process of seeking

$$\min_{\Xi, \text{ subject to those prefixed in } \Xi_I} E(\Xi | \Theta_r), \quad (28)$$

One process that makes an exact reasoning corresponds to a satisfaction of all the constraints, reaching the global minimum E_0 . Typically, there are uncertainties and conflicts among the constraints. What we encounter is uncertainty reasoning with that $E(\Xi | \Theta_r)$ merely reaching a minimum bigger than E_0 .

One most popular example is called Bayesian networks with $E(\Xi | \Theta_r) = -\ln p(\Xi | \Theta_r)$. The joint probability distribution $p(\Xi | \Theta_r) = p(\xi_1, \xi_2, \dots, \xi_M | \Theta_r)$ is represented on a directed acyclic graph (DAG), on which reasoning can be conducted by an effective propagation procedure [84], [85].

Beyond DAG, the situation becomes much complicated if $p(\xi_1, \xi_2, \dots, \xi_M | \Theta_r)$ is on a network that is not DAG. Still, reasoning is meaningful for some types of $p(\xi_1, \xi_2, \dots, \xi_M | \Theta_r)$, e.g., when $-\ln p(\xi_1, \xi_2, \dots, \xi_M | \Theta_r)$ is specified by an energy likes ones either in the classic Hopfield networks [86] or in Boltzmann machine [87]. Making reasoning by Eq.(28) is refined as maximising the following posteriori

$$\max_{\xi_1, \xi_2, \dots, \xi_M, \text{ subject to those prefixed in } \Xi_I} p(\xi_1, \xi_2, \dots, \xi_M | \Theta_r). \quad (29)$$

Alternatively, reasoning based on a multivariate joint distribution may also be understood from a uncertainty propagation perspective, namely, given the changing of one or more marginal distribution $p(\xi_j | \Theta_r)$ that describes how ξ_j

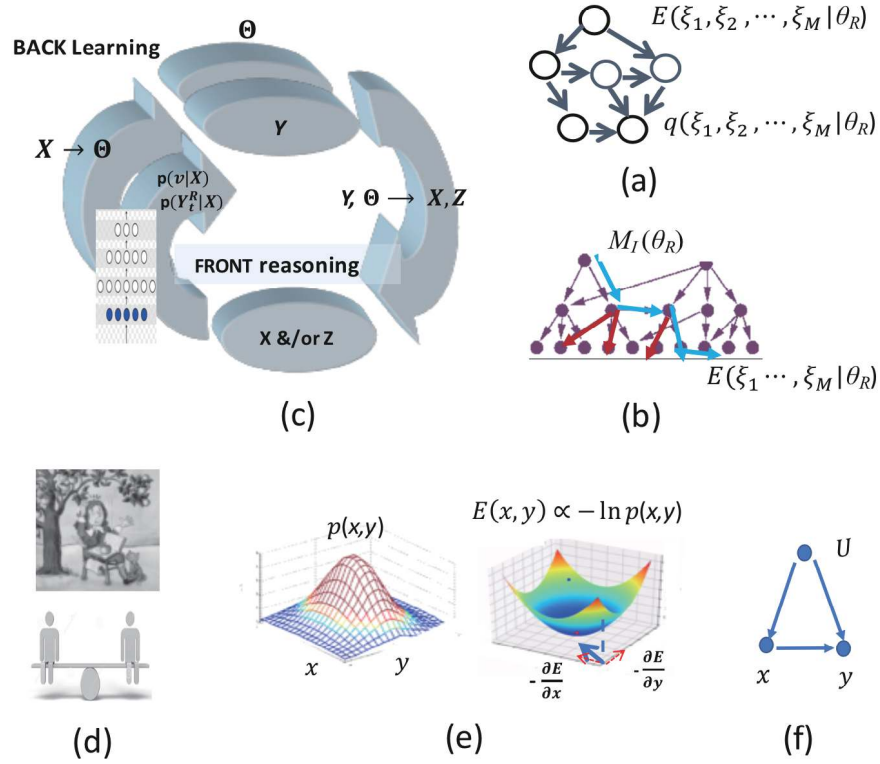


Fig. 16. Deep bidirectional reasoning and causal potential theory. (a) Reasoning that is made on networks. (b) Reasoning that is featured by searching a path or subtree. (c) Deep bidirectional reasoning from BYY-IPT perspective. (d) Apple falls and balance loses caused by physics mechanism (e) Causal potential theory. (f) The confounder problem.

varies with specified uncertainty, the task is propagating such changing over the entire distribution $p(\xi_1, \xi_2, \dots, \xi_M | \Theta_r)$, such that all the changed marginal distribution $p(\xi | \Theta_r), \xi \in \Xi$ are solved.

Strictly speaking in a classical sense, reasoning is featured by a path or subtree as illustrated in Fig. 16(b), starting from certain premises or preconditions to reach some expected conclusion or consequence, based on a set of reasoning rules as a knowledge base. One typical example is a classical logical reasoning with the knowledge base consisting of a set of logical rules in a format $pre \rightarrow con$. At each node in a search tree, the number of child nodes to be considered relates to the current situation and the related rules in the knowledge base. An involved search tree can be very large both in depth and width, and suffers the computationally intractable challenge. One other example is a production rule based reasoning in the so called expert system, which was one main theme in traditional AI studies. Though in a format $pre \rightarrow con$, each rule goes beyond a logical rule, permitting uncertainty and even including some conflicts. Still, the reasoning process suffers the challenges of intractable computation and of managing uncertainties and conflicts.

Alternatively, reasoning path may be featured by seeking a sequence $Y_{0:t}^b = \{Y_0^b, \dots, Y_t^b\}$ with each Y_t^b taking one of several discrete values and each value indicating a child node, in order to minimise a measure $E(Y_{0:t}^b | \Theta)$ with its structure and parameters Θ implicitly specified by premises, consequences, and the rules along the path. Similarly, reasoning subtree may be featured by seeking a tree T^b to minimise a measure

$E(T^b | \Theta)$. Both the problems, as well as ones by Eq.(28) and Eq.(29), can be regarded as examples of typical rational thinking problems, also suffering intractable computing difficulties of typical combinatorial optimisation.

Similar to ones for TSP and AGM in Fig. 15(b) and (c), we may tackle the difficulties by deep bidirectional reasoning from BYY-IPT perspective, as illustrated in Fig. 16(c), that is, integrating rational thinking and image thinking with image pattern X perceived by an A-mapping $X \rightarrow Y$ to help rational reasoning, and then an I-mapping $Y \rightarrow X$ to let internal abstract process to be displayed by image pattern X in its observation world, for which a key point is whether we can find a formulation to set up a correspondence between each internal state and a 2D or higher dimensional image X .

In such a bidirectional reasoning, intractable computation may also be tackled by the back learning in Fig. 16(c), which adapts parameters Θ such that the configuration of $E(\bullet | \Theta)$ changes to make its optimisation become easier.

D. Causal Potential Theory

In the previous subsection, reasoning is based a given set of edges or the networks they form. However, how to discover these edges and the networks from samples is not an easy task. Particularly, it is an important but difficult problem to discover each directed or causal edge to build up a DAG from samples. The rest of this subsection will address Causal Potential Theory (CPT), which is proposed recently for discovering causal direction of each edge [88].

Extensive efforts have been made on detecting causal

direction, evaluating causal strength, and discovering causal structure from observations. Examples include not only the studies based on conditional independence and DAG [84], [85], [89]–[92], but also those on path analysis [93], RCM [94], SEM [95], ANM [96], LiNGAM [97], PNL [98], and CGNN [99], as well as ones for discovering star-structure [100] and identifying the so called ρ -diagram [49]. More or less, these efforts share a similar direction of thinking. First, one presumes a causal structure (e.g., simply one direction in the simplest case or a DAG in a sophisticated situation) for one multivariate distribution either modelled in parametric form or partly inspected via statistics, subject to certain constraints. Second, one uses observation data to learn the parametric model or estimate the statistics, and examine whether the model fits observations and the constraints are satisfied, based on which we verify whether the presumed causal structure describes externally observations well. Typically, a set of causal structures are presumed as candidates among which the best is searched.

In analogy to physics, CPT regards causality as an intrinsic kinetic nature caused by a causal potential energy [88]. Without losing generality, we start at considering a cause-effect relation between a pair of variables x, y in an environment U . Instead of presuming a causal structure (i.e., one specific direction), one estimates a nonparametric distribution $p_U(x, y) = p(x, y|U)$ from samples of x, y and gets the corresponding causal potential energy in an analogy to Gibbs distribution, such that an event occurring at x, y is associated with the following gradient field:

$$[g_x, g_y] = \left[\frac{\partial E_U}{\partial x}, \frac{\partial E_U}{\partial y} \right], \quad E_U = E_U(x, y) = -\ln p(x, y|U), \quad (30)$$

which depicts probability change rate of an event occurring at x, y versus at $x + dx, y + dy$, which causes information flow towards a location or area where events occur in high chances or equivalently with a lowest causal potential.

If the random varying of y and such a potential driven change by g_x are strongly dependent with each other while the random varying of x and the potential driven change by g_y are independent or weakly dependent, we may infer that g_x drives both the changes of x and y while g_y drives mainly the change of y and feebly the change of x . In other words, we may infer the causal direction $x \rightarrow y$. Similarly, we may infer the causal direction $y \rightarrow x$.

One should not confuse that independence between a random variable x and a random variable g_y (shortly $x \perp\!\!\!\perp g_y$) with

$$g_{yx} = \frac{\partial g_y}{\partial x} = \frac{\partial^2 E}{\partial y \partial x} = 0, \text{ which further leads to } \frac{\partial^2 E}{\partial y \partial x} = \frac{\partial^2 E}{\partial x \partial y} = \frac{\partial g_x}{\partial y} = g_{xy} = 0. \quad (31)$$

Though $g_{yx} = 0$ or equivalently $g_y(x) = g_y$ leads to $p(x, g_y) = p(x)p(g_y|x) = p(x)p(g_y)$, but it is not true inversely, knowing $p(g_y|x) = p(g_y)$ can not guarantee $g_y(x) = g_y$. Though $g_{yx} = g_{xy}$ generally holds, getting one of $p(x, g_y) = p(x)p(g_y)$ and $p(g_x, y) = p(g_x)p(y)$ does not necessarily get that the other holds too.

Table VII shows two conceptual roads for analysing CPT causality. *Road_A* is proceeded by testing an answer of yes (Y) or no (N) on the mutual independence between g_y, x and also the one between g_x, y , resulting in four types of Y-N combinations. The first two types indicate two types of causality. The third type Y-Y indicates the independence $\perp\!\!\!\perp$ between x, y , i.e., no relation. The last type N-N indicates ‘unclear’, and study may be further made on whether causal relation occurs locally or even reciprocally in local regions of x, y though there is no causal relation detected globally. In implementation, both the roads can be made in several choices.

For *Road_A*, $[g_x, g_y]$ may simply come from a kernel estimate by

$$p_U(x, y) = p_h(x, y) = \frac{1}{N} \sum_{t=1}^N G([x, y] | [x_t, y_t], h^2 I), \quad (32)$$

where $G(u|\mu, \Sigma)$ is a Gaussian of mean μ and covariance Σ .

Then, we need a tool to make independent test on four types of Y-N combinations in Table VII. In a rough approximation, one may merely consider the 2nd order independence. Simply we consider

$$\begin{aligned} r_{xg_y} &= E(x - Ex)(g_y - Eg_y), \text{ or} \\ r_{xg_y}^* &= \frac{r_{xg_y}}{\sqrt{E(x - Ex)(x - Ex)E(g_y - Eg_y)(g_y - Eg_y)}}, \\ r_{yg_x} &= E(y - Ey)(g_x - Eg_x), \text{ or} \\ r_{yg_x}^* &= \frac{r_{yg_x}}{\sqrt{E(g_x - Eg_x)(g_x - Eg_x)E(y - Ey)(y - Ey)}}, \end{aligned} \quad (33)$$

and generally we consider the following KL measure:

$$D_{xg_y} = E_{p(x, g_y)} \ln \frac{p(x, g_y)}{p(x)p(g_y)}, \quad D_{g_x, y} = E_{p(g_x, y)} \ln \frac{p(g_x, y)}{p(g_x)p(y)}. \quad (34)$$

Based on the measures, we may make one hypothesis test on one of choices given in Table VIII.

Instead of Eq.(32), we may get $p_U(x, y)$ by one presumed causal structure, e.g.,

$$p_U(x, y) = \begin{cases} p_h(x)p(y|x, \phi_{y|x}) & \text{for } x \rightarrow y \\ p_h(y)p(x|y, \phi_{x|y}) & \text{for } y \rightarrow x; \end{cases}$$

where $p(u|v, \phi) = \begin{cases} \sum_j \alpha_j G(u - \mu_j | f(v, \phi), \sigma_j^2), & \text{example 1} \\ s^u(v, \phi) [1 - s(v, \phi)]^{1-u}, & \text{example 2.} \end{cases}$

$$p_h(\xi) = \frac{1}{N} \sum_{t=1}^N G(\xi | \xi_t, h^2), \text{ and } 0 < s(r) < 1 \text{ is sigmoid.} \quad (35)$$

Accordingly, we may modify Eq.(33) into

$$\begin{aligned} r_{xg_y} &= \int p_h(x)p(y|x, \phi_{y|x})(x - Ex)(g_y - Eg_y)dx dy, \\ r_{yg_x} &= \int p_h(y)p(x|y, \phi_{x|y})(y - Ey)(g_x - Eg_x)dx dy, \end{aligned} \quad (36)$$

such that the presumed causal structure may be integrated into consideration.

For *Road_B*, the problem is turned into supervised learning tasks with x, y as inputs into neural net that fits two gradient components $[g_x, g_y]$. Each component is fit by different neural nets, with each or both of x, y as inputs, respectively. An

TABLE VII
TWO ROADS FOR ANALYZING CPT CAUSALITY

$\nabla_u E_U$	$y \rightarrow x$	$x \rightarrow y$	$x \perp\!\!\!\perp y$	$x \text{ ? } y$
g_x	$\perp\!\!\!\perp y$	$\perp\!\!\!\perp y$	$\perp\!\!\!\perp y$	$\perp\!\!\!\perp y$
g_y	$\perp\!\!\!\perp x$	$\perp\!\!\!\perp x$	$\perp\!\!\!\perp x$	$\perp\!\!\!\perp x$

Road A

$\nabla_u E_U$	$y \rightarrow x$	$x \rightarrow y$	$x \perp\!\!\!\perp y$	$x \text{ ? } y$
$g_x =$	$\xi(x, y) + \varepsilon$	$\xi(x) + \varepsilon$	$\xi(x) + \varepsilon$	$\xi(x, y) + \varepsilon$
$g_y =$	$\eta(y) + \varepsilon$	$\eta(x, y) + \varepsilon$	$\eta(y) + \varepsilon$	$\eta(x, y) + \varepsilon$

Road B

appropriate one is chosen according to both its fitting and simplicity, based on which four types of outcomes are listed in Table VII. Instead of function fitting based on Gaussian fitting error ε , we consider the maximum likelihood on $p(u|v, \phi)$ given in Eq.(35) with u replaced by g_x or g_y and v replaced by either or both of x, y .

In all the above considerations, cause-effect relation between a pair of variables x, y may be affected by its environment U . To reduce some misleading effect caused by U , we may further consider $p(x, y) = \int p(x, y|U)p(U)dU$ and $E(x, y) = -\ln p(x, y)$ through the following relation

$$[g_x, g_y] = \left[\frac{\partial E(x, y)}{\partial x}, \frac{\partial E(x, y)}{\partial y} \right] = \frac{\int \left[\frac{\partial E_U}{\partial x}, \frac{\partial E_U}{\partial y} \right] p(x, y|U)p(U)dU}{\int p(x, y|U)p(U)dU}, \quad (37)$$

that is, we get g_x, g_y by averaging over the different environments. Given only a set of samples on (x, y) , we may consider resample from this set to roughly simulate different environments, with help of some resampling methods [101].

Regarding U as one entity as a whole, one actually encounters the well known confounder problem as illustrated in Fig. 16(f), for which one may possibly integrate recent results based on the ρ -diagram illustrated in Fig. 5 of [49] and also in Fig. 8 in [88].

VI. CONCLUDING REMARKS

An extensive overview and new perspectives have been made on bidirectional intelligence. Here, we summarise them in the following six aspects, according to dualities given Table I and further elaborated in Table VI:

(1) Dualities in Table I make Lmsr work well

Lmsr shares the duality DBA with autoencoder and also possesses four new dualities. First, DPN takes an important role in U-net and DenseNet too. It is further noted in Table VI that DPN is a special case of DFF that leads to an extension named Flexible Lmsr in Table III, with its special cases covering Lmsr and autoencoder as well as several extensions (e.g., including U-net and DenseNet). Specifically, the feedback makes bottom-up perception more robust, while the fast-lane provides top-down reconstruction with more details. Second, DCW implies approximately an invertible constraint between two subsequent layers. Interestingly, such a spirit is recently found in RevNet [102] and i-revnet [103], which may

also be clearly observed DFB in Table VI that includes DCW as a special case. Additionally, DCW is also an example that is called weight sharing or learning transfer in the past decade. Third, DPN and DCW not only make Lmsr outperform autoencoder considerably but also incorporate DPD, DAM and DSP to jointly make a number of possible cognitive functions [2], [3], which are all interestingly verified by experiments recently.

(2) Duality DBA exemplifies that learning is bidirectional

Not only autoencoder and Lmsr are early examples of bidirectional learning, but also typical existing learning principles may all be regarded as special or degenerated cases of bidirectional learning, as summarised by a bird view illustrated in Fig. 6(a). Moreover, the dualities DIA, DLT, and DPD in BYY harmony learning provide a unified framework that accommodates maximum likelihood, variational principle, and several others, with new perspectives on advanced topics. Also, a dozen of typical learning methods are summarized as exemplars of the BYY harmony learning, in terms of BYY best matching and BYY best harmony.

(3) A dozen of Lmsr extensions come from exploring and extending dualities

First, as illustrated in Fig. 5(a)–(c), exploring the dualities DPD, DSP in Table I and their extended versions in Table VI, as well as ST memory vs LT memory in the duality DIA in Table VI, leads to extensions named RPCL-Lmsr, RPCLVQ-Lmsr, and SOM-Lmsr for conceptualisation in the encoding space or I-domain. Second, elaborating the dualities DPN, DCW in Table I and their extended versions DFF, DFB in Table VI leads to those extensions illustrated in Fig. 5(d)–(h) and summarised in Table III, as well as unified under the name Flexible Lmsr. Third, further improvements of Lmsr may be obtained with help of appropriate uses of the duality DAM in Table I and its extended version in Table VI, resulting in a number of attention based extensions introduced at the end of Section II.

(4) Dualities in Table VI depict bidirectional intelligence and BYY IPT thesis as illustrated in Fig. 14.

First, the duality DAM prescribes the architecture of IA system as illustrated in Fig. 6, while the dualities DFF, DFB further equip Lmsr like structure to implement A-mapping and I-mapping, respectively, as illustrated in Fig. 2(b) and subsequently in Table IV, as well as in Fig. 9–11. Second, the duality DLT features the BYY intelligence potential theory (BYY-IPT) given by Eq.(16), and the duality DPD features the long term dynamics (LTD) versus the short term dynamics (STD). Implementations of intelligent activities are driven by STD that depicts changes of Y in the ST memory (STM), subject to the current value of Θ in the LT memory (LTM), which is supported by LTD for parameter learning on Θ . Moreover, the duality DAM coordinates the switching between STD vs LTD and trades off adapting vs forgetting and vigilance vs ignorance. Third, the dualities DSP, DIT, and DRT characterize the STD implementations of three main themes of intelligent activities, namely, perception, image thinking, and rational thinking, as illustrated in Fig. 14.

(5) Dualities DSP and DIT coordinate tasks of perception versus image thinking

TABLE VIII
SEVERAL CHOICES OF STATISTICS FOR DECIDING FOUR TYPES

statistics / type	$y \rightarrow x$	$x \rightarrow y$	$x \perp y$	$x ? y$
r_{xg_y}, r_{g_xy}	$r_{xg_y} \neq 0, r_{g_xy} = 0$	$r_{xg_y} = 0, r_{g_xy} \neq 0$	$r_{xg_y} = 0, r_{g_xy} = 0$	$r_{xg_y} \neq 0, r_{g_xy} \neq 0$
$\gamma = r_{g_xy} / r_{xg_y} $	$\gamma \ll 1$	$\gamma \gg 1$	$\gamma = 0/0$	$\gamma \approx 1$
$r_{xg_y}^*, r_{g_xy}^*$	$r_{xg_y}^* \neq 0, r_{g_xy}^* = 0$	$r_{xg_y}^* = 0, r_{g_xy}^* \neq 0$	$r_{xg_y}^* = 0, r_{g_xy}^* = 0$	$r_{xg_y}^* \neq 0, r_{g_xy}^* \neq 0$
$\gamma^* = r_{g_xy}^* / r_{xg_y}^* $	$\gamma^* \ll 1$	$\gamma^* \gg 1$	$\gamma^* = 0/0$	$\gamma^* \approx 1$
D_{xg_y}, D_{g_xy}	$D_{xg_y} > 0, D_{g_xy} = 0$	$D_{xg_y} = 0, D_{g_xy} > 0$	$D_{xg_y} = 0, D_{g_xy} = 0$	$D_{xg_y} > 0, D_{g_xy} > 0$
$\gamma_D = D_{g_xy}/D_{xg_y}$	$\gamma_D \ll 1$	$\gamma_D \gg 1$	$\gamma_D = 0/0$	$\gamma_D \approx 1$

The A-mapping $X \rightarrow Y$ implements perception with discrete Y^b for classification and real vector Y^c as an inner code of X , while learning $X \rightarrow Y^b$ is featured by duality DLP, namely, learned either from labeled data in a supervised way or from unlabelled data in an unsupervised way together with learning $Y^b \rightarrow \hat{X}$ to reconstruct X , such that the perception $X \rightarrow Y^b$ is verified if $X \rightarrow Y^b \rightarrow \hat{X} \approx X$ well, as previously made by autoencoder and Lmsr. In addition to those previously addressed Lmsr functions, $X \rightarrow Y^b \rightarrow \hat{X} \approx X$ also acts as a benchmark for the IA system to implement image thinking that is, it is just one hand of the Duality DIT. The other hand of image thinking implements $Y^c \rightarrow Z$ for various tasks of pattern-to-pattern (PtoP) transform, as illustrated by the loop $X \rightarrow Y^c \rightarrow Z$ in Fig. 14.

(6) *Dualities DSP and DRT coordinate tasks of perception versus rational thinking*

Instead of making one PtoP transform, tasks of rational thinking are depicted by searching a sequence $Y_{0:t}^b = \{Y_0^b, \dots, Y_t^b\}$ with each being either a binary variable or vector. Such a search is featured by the duality DRT with one type of Y_t^b for solving a combinatorial task and other types for implementing uncertainty reasoning. Under the general framework of bidirectional intelligence as illustrated in Fig. 14, this searching is made in coordination with the A-mapping that provides $p(Y_{0:t}^b|X)$ in either or both of supervised and unsupervised manners, namely featured by the duality DSP, which inherits and develops one key spirit of AlphaGo.

Beyond all the above aspects, in sequel, we use two more paragraphs to address the new direction featured by the coordination of A-mapping and I-mapping. Classically, rational thinking tasks were regarded as high level thinking activities in nature of abstract, advanced, and accuracy, which misled to a lot of computationally intractable challenges. In contrast, the coordination of A-mapping and I-mapping signifies that many or most of rational thinking tasks may not only associate with inaccuracy and heuristics as advocated by heuristic search studies in the traditional AI studies [104], [105] but also intrinsically interweave with image thinking and particularly perception and recognition of what are encountered in its environment. Many classic computationally intractable problems can be reconsidered along such a new direction, with help of feature enrichment, that is, let these tasks to be associated with much enriched inputs (e.g., 2D or higher dimensional images) such that appropriate features are systematically extracted by deep learning networks, instead of heuristically formulated by human.

The last three subsections in Section V all proceed along this direction. First, ME Player, BYY Follower, and BYY IPT guided extensions are suggested for implementing AlphaGoZero tree searching. Second, traveling salesman problem (TSP) and attributed graph matching (AGM) are turned into Go like games, with help of chessboard like feature enrichments, resulting in algorithms Alpha-TSP and Alpha-AGM as well as their variants IA-search TSP and IA-search AGM. Third, not only reasoning activities are summarized from the aspects of constraint satisfaction, uncertainty propagation, path and tree searching, but also a causal potential theory is addressed for discovering causal direction of each edge, together with two roads suggested for implementing such causality discovery.

REFERENCES

- [1] H. Bourlard and Y. Kamp, "Auto-association by multilayer perceptrons and singular value decomposition," *Biol. Cybern.*, vol. 59, no. 4-5, pp. 291-294, Sep. 1988.
- [2] L. Xu, "Least MSE reconstruction by self-organization. I. Multi-layer neural-nets," in *Proc. Int. Joint Conf. Neural Networks*, Singapore, 1991, pp. 2362-2367.
- [3] L. Xu, "Least mean square error reconstruction principle for self-organizing neural-nets," *Neural Networks*, vol. 6, no. 5, pp. 627-648, Oct. 1993.
- [4] D. H. Ballard, "Modular learning in neural networks," in *Proc. 6th National Conf. Artificial Intelligence*, Seattle, USA, 1987, pp. 279-284.
- [5] J. L. Elman and D. Zipser, "Learning the hidden structure of speech," *J. Acoust. Soc. Am.*, vol. 83, no. 4, pp. 1615-1626, Apr. 1988.
- [6] P. G. Cottrell, P. Munro, and D. Zipser, "Image compression by back propagation: An example of extensional programming," in *Models of Cognition: A Review of Cognition Science*, N. E. Sharkey, Ed. Norwood, USA: Ablex, 1989, pp. 208-240.
- [7] P. Baldi and K. Hornik, "Neural networks and principal component analysis: Learning from examples without local minima," *Neural Networks*, vol. 2, no. 1, pp. 53-58, Dec. 1989.
- [8] M. Kawato, H. Hayakawa, and T. Inui, "A forward-inverse optics model of reciprocal connections between visual cortical areas," *Network: Comput. Neural Syst.*, vol. 4, no. 4, pp. 415-422, Oct. 1993.
- [9] W. E. A. Huang, "Deep LMSER learning with symmetric weights and neuron sharing," 2018.
- [10] G. E. Hinton, P. Dayan, B. J. Frey, and R. M. Neal, "The "wake-sleep" algorithm for unsupervised neural networks," *Science*, vol. 268, no. 5214, pp. 1158-1161, May 1995.
- [11] P. Dayan, G. E. Hinton, R. M. Neal, and R. S. Zemel, "The Helmholtz machine," *Neural Comput.*, vol. 7, no. 5, pp. 889-904, Sep. 1995.
- [12] L. Xu, "Bayesian-Kullback coupled Ying-Yang machines: Unified learnings and new results on vector quantization," in *Proc. Int. Conf. Neural Information Processing*, Beijing, China, 1995, pp. 977-988.
- [13] L. Xu, "A unified learning scheme: Bayesian-Kullback Ying-Yang machine," in *Proc. 8th Int. Conf. Neural Information Processing*

Systems, Denver, USA, 1996, pp. 444-450.

- [14] G. E. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527-1554, Jul. 2006.
- [15] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504-507, Jul. 2006.
- [16] X. J. Mao, C. H. Shen, and Y. B. Yang, "Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections," in *Proc. 30th Int. Conf. Neural Information Processing Systems*, Barcelona, Spain, 2016, pp. 2802-2810.
- [17] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Int. Conf. Medical Image Computing and Computer-Assisted Intervention*, Munich, Germany, 2015, pp. 234-241.
- [18] K. M. He, X. Y. Zhang, S. Q. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. 2016 IEEE Conf. Computer Vision and Pattern Recognition*, Las Vegas, USA, 2016, pp. 770-778.
- [19] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. 2017 IEEE Conf. Computer Vision and Pattern Recognition*, Honolulu, USA, 2017, pp. 4700-4708.
- [20] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," arXiv preprint arXiv: 1312.6114, 2013.
- [21] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85-117, Jan. 2015.
- [22] C. K. Sønderby, T. Raiko, L. Maaløe, S. K. Sønderby, and O. Winther, "Ladder variational autoencoder," in *Proc. 30th Conf. Neural Information Processing Systems*, Barcelona, Spain, 2016, pp. 3738-3746.
- [23] L. Xu, "Bayesian Ying-Yang system, best harmony learning, and five action circling," *Front. Electr. Electron. Eng. China*, vol. 5, no. 3, pp. 281-328, Sep. 2010.
- [24] L. Xu, "New advances on the Ying-Yang machine," in *Proc. 1995 Int. Symp. Artificial Neural Networks*, Taiwan, China, 1995, pp. 7-12.
- [25] L. Xu, "Codimensional matrix pairing perspective of BYY harmony learning: hierarchy of bilinear systems, joint decomposition of data-covariance, and applications of network biology," *Front. Electr. Electron. Eng. China*, vol. 6, no. 1, pp. 86-119, Mar. 2011.
- [26] L. Xu, "On essential topics of BYY harmony learning: Current status, challenging issues, and gene analysis applications," *Front. Electr. Electron. Eng.*, vol. 7, no. 1, pp. 147-196, Mar. 2012.
- [27] L. Xu, "Further advances on Bayesian Ying-Yang harmony learning," *Appl. Inform.*, vol. 2, pp. 5, Dec. 2015.
- [28] D. J. Rezende, S. Mohamed, I. Danihelka, K. Gregor, and D. Wierstra, "One-shot generalization in deep generative models," in *Proc. 33rd Int. Conf. Machine Learning*, New York, USA, 2016.
- [29] S. J. Zhao, J. M. Song, and S. Ermon, "Learning hierarchical features from deep generative models," in *Proc. 34th Int. Conf. Machine Learning*, Sydney, Australia, 2017, pp. 4091-4099.
- [30] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. 27th Int. Conf. Neural Information Processing Systems*, Montreal, Canada, 2014, pp. 2672-2680.
- [31] S. Gurumurthy, R. K. Sarvadevabhatla, and R. V. Babu, "DeLiGAN: Generative adversarial networks for diverse and limited data," in *Proc. 2017 IEEE Conf. Computer Vision and Pattern Recognition*, Honolulu, USA, 2017.
- [32] L. Mescheder, S. Nowozin, and A. Geiger, "Adversarial variational Bayes: Unifying variational autoencoders and generative adversarial networks," in *Proc. 34th Int. Conf. Machine Learning*, Sydney, Australia, 2017.
- [33] I. V. Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau, "Building end-to-end dialogue systems using generative hierarchical neural network models," in *Proc. 30th AAAI Conf. Artificial Intelligence*, Phoenix, Arizona, 2016, pp. 3776-3783.
- [34] I. V. Serban, A. Sordoni, R. Lowe, L. Charlin, J. Pineau, A. Courville, and Y. Bengio, "A hierarchical latent variable encoder-decoder model for generating dialogues," in *Proc. 31st AAAI Conf. Artificial Intelligence*, San Francisco, USA, 2017, pp. 3295-3301.
- [35] P. Ballester and R. Matsumura Araujo, "On the performance of GoogLeNet and AlexNet applied to sketches," in *Proc. 30th AAAI Conf. Artificial Intelligence*, Phoenix, Arizona, 2016, pp. 1124-1128.
- [36] S. Reed, Z. Akata, S. Mohan, S. Tenka, B. Schiele, and H. Lee, "Learning what and where to draw," in *Proc. 29th Conf. Neural Information Processing Systems*, Barcelona, Spain, 2016, pp. 217-225.
- [37] Y. J. Chen, S. K. Tu, Y. Q. Yi, and L. Xu, "Sketch-pix2seq: a model to generate sketches of multiple categories," arXiv preprint arXiv: 1709.04121, 2017.
- [38] T. Nakamura and R. Goto, "Outfit generation and style extraction via bidirectional LSTM and autoencoder," arXiv preprint arXiv: 1807.03133, 2018.
- [39] A. Augello, E. Cipolla, I. Infantino, A. Manfre, G. Pilato, and F. Vella, "Creative robot dance with variational encoder," arXiv preprint arXiv: 1707.01489, 2017.
- [40] E. Denton, S. Chintala, A. Szlam, and R. Fergus, "Deep generative image models using a Laplacian pyramid of adversarial networks," in *Proc. 28th Int. Conf. Neural Information Processing Systems*, Montreal, Canada, 2015, pp. 1486-1494.
- [41] P. Isola, J. Y. Zhu, T. H. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. 2017 IEEE Conf. Computer Vision and Pattern Recognition*, Honolulu, USA, 2017, pp. 1125-1134.
- [42] J. J. Wu, C. K. Zhang, T. F. Xue, B. Freeman, and J. Tenenbaum, "Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling," in *Proc. 29th Conf. Neural Information Processing Systems*, Barcelona, Spain, 2016, pp. 82-90.
- [43] G. L. Liu, F. A. Reda, K. J. Shih, T. C. Wang, A. Tao, and B. Catanzaro, "Image inpainting for irregular holes using partial convolutions," in *European Conf. Computer Vision*, Munich, Germany, 2018.
- [44] F. L. Ma, R. Chitta, J. Zhou, Q. Z. You, T. Sun, and J. Gao, "Dipole: Diagnosis prediction in healthcare via attention-based bidirectional recurrent neural networks," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, Halifax, Canada, 2017, pp. 1903-1911.
- [45] C. Vondrick and A. Torralba, "Generating the future with adversarial transformers," in *Proc. 2017 IEEE Conf. Computer Vision and Pattern Recognition*, Honolulu, USA, 2017.
- [46] Z. F. Zhang, Y. Song, and H. R. Qi, "Age progression/regression by conditional adversarial autoencoder," in *Proc. 2017 IEEE Conf. Computer Vision and Pattern Recognition*, Honolulu, USA, 2017.
- [47] L. Xu, A. Krzyzak, and E. Oja, "Rival penalized competitive learning for clustering analysis, RBF net, and curve detection," *IEEE Trans. Neural Networks*, vol. 4, no. 4, pp. 636-649, Jul. 1993.
- [48] T. Kohonen, "Learning vector quantization," in *Self-organizing Maps*, T. Kohonen, Eds. Berlin, Heidelberg, Germany: Springer, 1995, pp. 175-189.
- [49] L. Xu, "Deep bidirectional intelligence: AlphaZero, deep IA-search, deep IA-infer, and TPC causal learning," *Appl. Inform.*, vol. 5, no. 1, pp. 5, Dec. 2018.
- [50] T. Kohonen, "The self-organizing map," *Proc. IEEE*, vol. 78, no. 9, pp. 1464-1480, Sep. 1990.
- [51] L. Xu, "Adding learned expectation into the learning procedure of self-organizing maps," *Int. J. Neural Syst.*, vol. 1, no. 3, pp. 269-283, Apr. 1990.
- [52] M. A. F. Figueiredo and A. K. Jain, "Unsupervised learning of finite mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 3, pp. 381-396, Mar. 2002.
- [53] L. Xu, M. I. Jordan, and G. E. Hinton, "An alternative model for mixtures of experts," in *Advances in Neural Information Processing Systems*, J. D. Cowan, G. Tesauro, and J. Alspector, Eds. Cambridge: MIT Press, 1995, pp. 903-912.
- [54] G. Schwarz, "Estimating the dimension of a model," *Ann. Stat.*, vol. 6, no. 2, pp. 461-464, Mar. 1978.
- [55] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, no. 5, pp. 465-471, Sep. 1978.
- [56] J. Rissanen, *Information and Complexity in Statistical Modeling*. New York, USA: Springer, 2007.
- [57] D. J. MacKay, "A practical Bayesian framework for backpropagation

- networks," *Neural Comput.*, vol. 4, no. 3, pp. 448–472, May 1992.
- [58] L. Xu, "Bayesian Ying Yang system and theory as a unified statistical learning approach: (I) unsupervised and semi-supervised learning," in *Brain-like Computing and Intelligent Information Systems*, S. Amari and N. Kassabov, Eds. Berlin, Germany: Springer-Verlag, 1997, 241–274.
 - [59] L. Xu, "Data smoothing regularization, multi-sets-learning, and problem solving strategies," *Neural Networks*, vol. 16, no. 5–6, pp. 817–825, Jun.–Jul. 2003.
 - [60] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Comput.*, vol. 7, no. 6, pp. 1129–1159, Nov. 1995.
 - [61] S. Amari, A. Cichocki, and H. H. Yang, "A new learning algorithm for blind signal separation," in *Proc. 8th Int. Conf. Neural Information Processing Systems*, Denver, USA, 1995, 757–763.
 - [62] L. Xu, "Independent subspaces," in *Encyclopedia of Artificial Intelligence*, J. Ramón, R. Dopico, J. Dorado, and A. P. Sierra, Eds. Hershey, USA: IGI Global, 2009, pp. 892–901.
 - [63] L. Xu, "Independent component analysis and extensions with noise and time: a Bayesian Ying-Yang learning perspective," *Neural Inf. Process. Lett. Rev.*, vol. 1, no. 1, pp. 1–52, Oct. 2003.
 - [64] A. L. Yuille, S. M. Smirnakis, and L. Xu, "Bayesian self-organization," in *Proc. 6th Int. Conf. Neural Information Processing Systems*, Denver, USA, 1993, pp. 1001–1008.
 - [65] L. Xu, "Ying-yang learning," in *The Handbook of Brain Theory and Neural Networks*, M. A. Arbib, Ed. Cambridge, USA: MIT Press, 2002, 1231–1237.
 - [66] L. Xu, "BYX \Sigma-\Pi factor systems and harmony learning," in *Proc. Int. Conf. Neural Information Processing*, Taejeon, Korea, 2000, pp. 548–558.
 - [67] L. Xu, "Best harmony, unified RPCL and automated model selection for unsupervised and supervised learning on Gaussian mixtures, three-layer nets and ME-RBF-SVM models," *Int. J. Neural Syst.*, vol. 11, no. 1, pp. 43–69, Feb. 2001.
 - [68] L. Xu, "BYX harmony learning, independent state space, and generalized APT financial analyses," *IEEE Trans. Neural Networks*, vol. 12, no. 4, pp. 822–849, Jul. 2001.
 - [69] E. T. Jaynes, *Probability Theory: The Logic of Science*. New York, USA: Cambridge University Press, 2003.
 - [70] L. Xu and M. I. Jordan, "On convergence properties of the EM algorithm for Gaussian mixtures," *Neural Comput.*, vol. 8, no. 1, pp. 129–151, Jan. 1996.
 - [71] L. Xu, "RBF nets, mixture experts, and Bayesian Ying-Yang learning," *Neurocomputing*, vol. 19, no. 1–3, pp. 223–257, Apr. 1998.
 - [72] L. Xu and S. I. Amari, "Combining classifiers and learning mixture-of-experts," in *Encyclopedia of Artificial Intelligence*, J. Ramón, R. Dopico, J. Dorado, and A. P. Sierra, Eds. Hershey, USA: IGI Global, 2008, pp. 318–326.
 - [73] L. Xu, "Learning algorithms for RBF functions and subspace based functions," in *Handbook of Research on Machine Learning, Applications and Trends: Algorithms, Methods, and Techniques*, E. S. Olivas, J. D. M. Guerrero, M. Martinez-Sober, J. R. Magdalena-Benedito, and A. J. S. López, Eds. Hershey, USA: IGI Global, 2009, pp. 60–94.
 - [74] X. S. Qian, "On thinking sciences," *Chin. J. Nat.*, no. 8, pp. 563–567, 572–640, 1983.
 - [75] Y. H. Pan, "The synthesis reasoning," *Pattern Recognition and Artificial Intelligence*, vol. 9, no. 3, pp. 201–208, 1996.
 - [76] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, Jan. 2016.
 - [77] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. T. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, and D. Hassabis, "Mastering the game of Go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–259, Oct. 2017.
 - [78] M. I. Jordan and L. Xu, "Convergence results for the EM approach to mixtures of experts architectures," *Neural Networks*, vol. 8, no. 9, pp. 1409–1431, 1995.
 - [79] M. Jünger, G. Reinelt, and G. Rinaldi, "The traveling salesman problem," in *Handbooks in Operations Research and Management Science*, Amsterdam, Netherlands: Elsevier, 1995, pp. 225–330.
 - [80] C. Y. Dang and L. Xu, "A globally convergent Lagrange and barrier function iterative algorithm for the traveling salesman problem," *Neural Networks*, vol. 14, no. 2, pp. 217–230, Mar. 2001.
 - [81] W. H. Tsai and K. S. Fu, "Error-correcting isomorphisms of attributed relational graphs for pattern analysis," *IEEE Trans. Syst., Man, Cybern.*, vol. 9, no. 12, pp. 757–768, Dec. 1979.
 - [82] L. Xu and E. Oja, "Improved simulated annealing, Boltzmann machine, and attributed graph matching," in *European Association for Signal Processing Workshop*, Sesimbra, Portugal, 1990, pp. 151–160.
 - [83] L. Xu and S. Klasa, "A PCA-like rule for pattern classification based on attributed graph," in *Proc. 1993 Int. Conf. Neural Networks*, Nagoya, Japan, 1993, pp. 1281–1284.
 - [84] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, USA: Morgan Kaufmann, 1988.
 - [85] J. Pearl, "Fusion, propagation, and structuring in belief networks," *Artif. Intell.*, vol. 29, no. 3, pp. 241–288, Sep. 1986.
 - [86] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Natl. Acad. Sci. USA*, vol. 79, no. 8, pp. 2554–2558, Apr. 1982.
 - [87] G. E. Hinton and T. J. Sejnowski, "Learning and relearning in Boltzmann machines," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Cambridge: MIT Press, 1986, pp. 282–317.
 - [88] L. Xu, "Machine learning and causal analyses for modeling financial and economic data," *Appl. Inform.*, vol. 5, no. 1, pp. 11, Dec. 2018.
 - [89] P. Spirtes, C. Glymour, and R. Scheines, *Causation, Prediction, and Search*. New York, USA: Springer, 1993.
 - [90] P. Spirtes, C. Glymour, and R. Scheines, *Causation, Prediction, and Search*. 2nd ed. Cambridge, USA: MIT Press, 2000.
 - [91] P. Judea, "An introduction to causal inference," *Int. J. Biostat.*, vol. 6, no. 2, pp. 7, Feb. 2010.
 - [92] P. Spirtes and K. Zhang, "Causal discovery and inference: concepts and recent methodological advances," *Appl. Inform.*, vol. 3, pp. 3, Dec. 2016.
 - [93] S. Wright, "The method of path coefficients," *Ann. Math. Stat.*, vol. 5, no. 3, pp. 161–215, Sep. 1934.
 - [94] G. W. Imbens and D. B. Rubin, *Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction*. New York, USA: Cambridge University Press, 2015.
 - [95] R. B. Kline, *Principles and Practice of Structural Equation Modeling*. New York, USA: Guilford Publications, 2016.
 - [96] J. Peters, D. Janzing, and B. Schölkopf, "Causal inference on discrete data using additive noise models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2436–2450, Dec. 2011.
 - [97] S. Shimizu, P. O. Hoyer, A. Hyvärinen, and A. Kerminen, "A linear non-Gaussian acyclic model for causal discovery," *J. Mach. Learn. Res.*, vol. 7, pp. 2003–2030, Oct. 2006.
 - [98] K. Zhang and Hyvärinen, "On the identifiability of the post-nonlinear causal model," in *Proc. 25th Conf. Uncertainty in Artificial Intelligence*, Montreal, Canada, 2009, pp. 647–655.
 - [99] O. Goudet, D. Kalainathan, P. Caillou, I. Guyon, D. Lopez-Paz, and M. Sebag, "Causal generative neural networks," arXiv preprint arXiv: 1711.08936, 2017.
 - [100] L. Xu and J. Pearl, "Structuring causal tree models with continuous variables," in *Proc. 3rd Annu. Conf. Uncertainty in Artificial Intelligence*, Seattle, USA, pp. 170–179, 1987.
 - [101] B. Efron, *The Jackknife, the Bootstrap and Other Resampling Plans*. Philadelphia, USA: SIAM, 1982.
 - [102] A. N. Gomez, M. Y. Ren, R. Urtasun, and R. B. Grosse, "The reversible residual network: Backpropagation without storing activations," in *Proc. 31st Conf. Neural Information Processing Systems*, Long Beach, USA, 2017, pp. 2214–2224.
 - [103] J. H. Jacobsen, A. Smeulders, and E. Oyallon, "i-RevNet: Deep invertible networks," in *Proc. 2018 Int. Conf. Learning*

Representations, Vancouver, Canada, 2018.

- [104] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans. Syst. Sci. Cybern.*, vol. 4, no. 2, pp. 100–107, Jul. 1968.
- [105] J. Pearl, *Heuristics: Intelligent Search Strategies for Computer Problem Solving*. Reading, USA: Addison-Wesley Pub. Co., Inc., 1984.



Lei Xu (F'01) Emeritus Professor of Computer Science and Engineering, Chinese University of Hong Kong (CUHK); Zhiyuan Chair Professor of Computer Science and Engineering Department, Chief Scientist of AI Research Institute, Chief Scientist of Brain Sci & Tech Research Centre, Shanghai Jiao Tong University (SJTU); Director of Neural Computation Research Centre in Brain and Intelligence Science-Technology Institute, Zhang Jiang National Lab. Completed Ph.D thesis at Tsinghua Univ by the

end of 1986, Get PhD certificate from Tsinghua Univ in March 1987, joined Peking Univ as postdoc in 1987 and promoted exceptionally to associate professor in 1988. Worked as postdoc and visiting scientist in Finland, Canada and USA (including Prof. A. Yuille team in Harvard and Prof. M. Jordan team in MIT) during 1989–93. Joined CUHK as senior lecturer since 1993, professor since 1996, and chair professor during 2002–16. Joined SJTU as

Zhiyuan chair professor since the summer of 2016.

Given over dozens keynote /invited lectures at various international conferences. Served as EIC and associate editors of several academic journals, e.g., including Neural Networks (1994–2016), Neurocomputing (1995–2017), IEEE Tr. Neural Networks (1994–98). Taken various roles in academic societies, e.g., members of INNS Governing Board (2001–03), INNS award committee (2002–03), Fellow committee of IEEE Computational Intelligence society (2006–07), EURASC scientific committee (2014–17), and APNNA Past president (1995–96); also, general cochair, PC cochair, honorary chairs, international advisory committee chairs, as well as program/organizing /advisory committee members on major world conferences on Neural Networks; additionally, a nominator for the prestigious Kyoto prize (2004, 2008, 2012, 2016, 2020) and for the LUI Che Woo Prize (2016, 2017, 2018, 2019).

Published more than 400 papers, and internationally known with well-cited contributions on RHT, RPCL, classifier combination, mixture models, Lmsr, nonlinear PCA, BYY harmony and bidirectional learning, with more than 13500 citations (over 8200 by top-10 papers with 2773 for top-1 and 298 for 10th) according to Google Scholar versus more than 5500 citations (over 3900 by top-10 papers with 1319 for top-1 and 119 for 10th) according to Web of Science. Received several national and international academic awards, including 1993 National Nature Science Award, 1995 Leadership Award from International Neural Networks Society (INNS) and 2006 APNNA Outstanding Achievement Award. Elected to Fellow of IEEE in 2001; Fellow of intl. Association for Pattern Recognition in 2002 and of European Academy of Sciences (EURASC) in 2003.