A Position Paper on Statistical Inference Techniques Which Integrate Neural Network and Bayesian Network Models

William H. Hsu
Department of Computer Science and Beckman Institute
University of Illinois at Urbana-Champaign
bhsu@cs.uiuc.edu, http://anncbt.ai.uiuc.edu

Abstract

Some statistical methods which have been shown to have direct neural network analogs are surveyed here; we discuss sampling, optimization, and representation methods which make them feasible when applied in conjunction with, or in place of, neural networks. We present the foremost of these, the Gibbs sampler, *both in its successful role as a convergence heuristic derived from statistical physics and under its probabilistic learning interpretation. We then review various manifestations of Gibbs sampling in Bayesian learning; its relation to "traditional" simulated annealing; specializations and instances such as EM; and its application as a model construction technique for the* Bayesian network *formalism. Next, we examine the ramifications of recent advances in Markov chain Monte Carlo methods for learning by backpropagation. Finally, we consider how the Bayesian network formalism informs the causal reasoning interpretation of some neural networks, and how it prescribes optimizations for efficient random sampling in Bayesian learning applications.*

1. Introduction and Background

1.1. Overview

Among statistical inference methods and their connectionist learning analogs, some correspondences have been understood well enough to support "pure probabilistic" interpretations. That is, the architectures, learning rules, and key properties (e.g., convergence) of many ANN methods have been shown to admit emulation by random sampling algorithms. Better understanding of the probabilistic theory often provides not only the optimizations that lead to faster empirical convergence, but the more significant advance (even in applications) of improved semantic organization. For instance, knowing how to encode symbolic information in an ANN model and how to extract it from the trained model is one benefit of having an *intensional* interpretation, as Pearl terms it, of a connectionist system. [17, 19] Equally important to the generation and usage of a causal model or "domain theory" is the capability to account for the intermediate state of the system, yielding the immediate benefit of incremental learning. Other advantages accrue to integration with symbolic methods, from principled analysis of ANN methods as random sampling. [11]

In this paper, we will discuss two primary issues concerning the integration of probabilistic methods for learning with their connectionist complements (via hybrid systems) or equivalents (via alternative implementations). The former are typified by algorithms for inferring hidden causation — especially by constructing Bayesian or belief networks — and are often semantically better understood than the latter. Our first issue is how to develop a formal, *prescriptive* framework (cf. Valiant's quantitative PAC analysis for bias in inductive learning). [7] The position we defend below is that the "milieu" described by a random sampling problem is a determinant of computational learning complexity, as are the concept and hypothesis "languages" in the PAC framework. Our second issue is the problem, in uncertain reasoning, of statistical learning with unknown data. We identify a specific aspect of unknown data and survey shortcomings in current approaches, especially ANNs. Finally, we consider how the capability to tolerate unknowns might be furthered by better understanding of random sampling analogues of ANN methods.

The foremost of these are the Markov chain Monte Carlo methods: stochastic simulation algorithms which generate Markov chains with known, stationary distributions. [20, 15] The Gibbs sampler, in turn, is one of the best known Markov chain Monte Carlo techniques: it performs well in Bayesian network learning [20, 8] and corresponds very precisely to a auxiliary of simulated annealing (as used in the Boltzmann machine architecture). [14, 20, 11, 1] Research into Gibbs sampling and its maximum entropy relatives, such as EM, has led to the development of generalized

and incremental variants of some now-classical algorithms for Hidden Markov Model (HMM) learning. [15] Random sampling analysis of learning by error backpropagation for feedforward networks is also critical; it permits the development of hybrid systems that have comparable or improved convergence properties. [13]

A chief purpose for studying random sampling algorithms that can emulate neural network learning is to develop efficient, probabilistic methods that retain the data parallelism of ANNs, while achieving a higher level of semantic clarity. The Bayesian (belief) network formalism fits this description well, and has recently been advanced by research into annealing-based learning algorithms for Bayesian networks. [14] Here, we will examine primarily those ANN architectures which admit *dualities* with Bayesian networks. To take full advantage of our ability to interconvert between representations, we apply *network compilers* that map to and from each architecture. A master control mechanism, then, is needed to optimize this opportunistic change of representation for accuracy, semantic clarity (e.g., quality of explanations and causal reasoning traces), and data parallelism.

1.2. Gibbs Sampling in Hopfield Networks and Feedforward Network Learning

The Gibbs sampler is a stochastic simulation heuristic which achieves maximum likelihood approximation of an *irreducible* Markov chain (i.e., a model which has only one equivalence class of reachability: all states communicate with one another). [20, 8] The latter property (of the *target* model) is a necessary condition for convergence of the algorithm which uses Gibbs sampling. The purpose is to acquire parameters for (i.e., train) an approximating model for some observable function. This learning problem is very common in pattern recognition applications such as speech and handwriting. Gibbs sampling originates in statistical physics, where it is alternatively referred to as the *heatbath method*.

Gibbs sampling applies both directly to Bayesian networks with full data parallelism (cf. Pearl's "token passing" model), and to generalized Hopfield networks with simulated annealing (i.e., asymmetric Boltzmann machines). [14, 17, 3] The annealing function is enhanced by any Monte Carlo sampler that meets the "Gibbs criterion", and is typically used in conjunction with the Metropolis loop (a canonical sampling method for optimization by annealing). [16, 1] Note that not all properties of discrete Hopfield networks (DHNs) — e.g., the symmetric (bidirectional) weighting of links — are retained in this generalization.

Neal has shown that learning by error backpropagation for feedforward networks can be augmented by an adaptive random sampling approach called the *Hybrid Monte Carlo* algorithm. [13, 15] This results in faster convergence than with traditional annealing, and retains the same benefits of annealing over naive random sampling. The *Hybrid Monte Carlo* algorithm is fully compatible with Gibbs sampling, and is assumed to sample from a Gibbs distribution in the default case.

1.3. Information Theoretic Foundations

EM (Expectation-Maximization, sometimes known as "Estimate-and-Maximize" because of its two phases) is a forward-backward relaxation algorithm that generates a maximum likelihood model from incomplete data (i.e., joint priors). EM has become widely recognized in the connectionist literature as a highly flexible learning method. Like its more general relative, Gibbs sampling, it is often applied in fields that use pure probabilistic learning, such as pattern recognition with Hidden Markov Models (HMMs).

EM is shown to be a manifestation of the maximum entropy principle. [4] Some related incarnations of EM appear in the probability theory literature under the title of the *Baum-Welch algorithm*, which is specifically targeted at HMM learning. Namely, we are given a set Q of (hidden) states and observable alphabet Σ, emitted according to a probabilistic function over stochastic transitions among states. Parameter acquisition algorithms discover, by source modeling of a sequence of symbols from Σ, the transition probabilities a_{ij} among (q_i, q_j) and output emission probabilities $b_{ij}(k)$ for tuples $(q_i, q_j, \sigma \in \Sigma)$. A typical application of a model learned using EM is discussed below.

2. Applications

2.1. Supervised Learning: A Case Study

Why use EM for pattern recognition? First, it has been shown from its inception to be designed precisely for applications with unknown data. This is one of the most difficult aspects of uncertain reasoning (from an intelligent systems perspective) for noisy pattern recognition; but it is extremely common. It manifests both through faults in input acquisition (i.e., unreliable sensors, subjective data) and through limitations of the statistical computation component (e.g., in window-based learning schemes as described below).

The Viterbi algorithm is a dynamic programming method which computes maximum likelihood paths through stochastic models learned using EM variants (especially HMMs with transition probabilities acquired by the Baum-Welch algorithm). This makes it well-suited to "linear" pattern recognition problems (i.e., where one signal is mapped to another, and supervised learning is applied). Examples include prediction of protein secondary fold, a very difficult

problem to attack without extensive domain knowledge or pure random sampling for specific subproblems. [5, 21]

In research on protein folding that compares pure probabilistic methods (simple HMM learning by EM, followed by Viterbi-based matching) to traditional (feedforward with backprop) and knowledge-based (*KBANN*, *EBLANN*) ANN methods, EM has been shown to outperform extant connectionist systems. [5] This is notable, considering that the pure statistical approach is based only on *fixed-width* windows. The best predictor at the time was a hybrid system combining pure statistical, memory-based, and connectionist learning (backprop) with a connectionist front-end for data fusion (also a feedforward net trained with backprop). [21] Even compared to this system, EM achieved comparable cross-validated prediction accuracy. [5]

2.2. Unsupervised HMM Learning

Supervised learning of parameters for HMMs is a popular method for training a pattern recognition system. In spatiotemporal sequence modeling, however, it is often the signal itself that we wish to predict (e.g., sensor output in a high-uncertainty domain, such as biomedical monitoring). The general applicability of seminumerical (i.e., "subsymbolic") systems, including ANNs, to spatiotemporal sequence learning is witnessed by their ability to generate maximum likelihood estimates from incomplete sample data. Our current research examines the ability (or lack thereof) of simple recurrent networks [6] to acquire temporal regularities in observable Markov processes: e.g., duration of runs, periodicities, etc.

Recent research into delay-based ANN emulation of the Viterbi and EM algorithms has demonstrated that the Viterbi algorithm can be adequately modeled by the *Viterbi network*, a time-delay neural net with direct correspondence between output units and HMM states. In addition, some EM implementations (specifically, the Baum-Welch algorithm) can be emulated by *Alpha networks* (named after the iterative relaxation parameter in the forward pass). [4] This integration effort between the "pure probabilistic" and connectionist interpretations of has acted as a catalyst to expose their information theoretic underpinnings (namely, Kullback-Leibler divergence, a "cross-entropy" or mutual information measure).

2.3. Gibbs Sampling: a Brief Survey

Gibbs sampling is a general heuristic, derived from statistical mechanics, that encompasses a very broad family of Markov chain Monte Carlo algorithms, with a common simulation constraint. It applies to problems where the input consists of joint priors for a data vector ("multi-dimensional parameter") θ, whose elements are random variables. These random variables correspond to elements of a discrete Hopfield or Bayesian network (not all DHNs are Bayesian networks, but we shall consider this distinction later). The desired output is a sequence denoting a Markov chain (over network states — i.e., conditional probabilities for activations *or* weights). We wish the simulation of this Markov chain (i.e., random perturbations) to be amenable to annealing methods, with faster convergence.

Gibbs sampling selects $\theta_j^{(t+1)}$ as follows:

$$P(\theta_j^{(t+1)} | \Theta_{<j}^{(t+1)}, \Theta_{>j}^{(t)}) \quad (1)$$

where $\Theta_{<j}^{(t)} = \{\theta_i^{(t)} | i < j\}$, $\Theta_{>j}^{(t)} = \{\theta_i^{(t)} | i > j\}$, and $\theta_j^{(t)}$ is the observable distribution of the jth random variable at time increment t (sampled from the given joint prior probabilities) under the stationary Markov chain (i.e., state transition model) $Q(\theta)$.

Gibbs sampling is not always feasible for Bayesian optimization of neural networks, because conditional probabilities for some θ_j cannot always be sampled from Q for arbitrary groups of parameters. [16] This is referred to in the study of intelligent systems (specifically, Bayesian networks) as *locality*. [17] In ANNs, locality is *typically* not respected; often, the statistical character of the conditionals is highly complex and multimodal. Gibbs sampling, however, has the desirable property of being fully compatible with simulated annealing. This synergy makes it extensible to parallel (distributed) relaxation. It is used in two classes of annealing algorithms: in supervised learning by stochastic backpropagation, and to achieve convergence in the pattern recognition phase of "associative memory" systems (especially Hopfield networks). Thus, the efficacy of Gibbs sampling depends less on the application (learning versus "recall") than on the distributed nature of the inferential problem (*modularity* in the Bayesian networks literature). [16, 17]

The stochastic modeling requirements are simply the existence of a stationary distribution for the sampled process (i.e., positive recurrence and aperiodicity). For ANNs, the process describes conditional probabilities of network component states (*weights* for feedforward network learning; *neurons* for Boltzmann machines), given (possibly incomplete) observed data. Even this sufficient condition may be relaxed to the necessary condition of irreducibility. [20]

In a review of typical learning problems served by Gibbs sampling, York gives a proof of sufficiency for irreducibility, leading to the result that a Markov process induced by Gibbs sampling yields maximum likelihood estimates under this condition. [20] This survey also discusses an interesting application of the sampler to construction of a knowledge-based system — relating the learning characteristic of the Gibbs sampler to *fusion and propagation* in Pearl's framework of fusion, propagation, and structuring. [20, 17]

3. Theoretical Foundations

3.1. Gibbs Sampling and Annealing Methods

Gibbs sampling can be viewed as a variation on the "traditional" stochastic search algorithm of Metropolis *et al* as applied to simulated annealing. [16, 9] The best known application of Gibbs sampling is also one of the best known annealing-based ANN methods — namely, the *Boltzmann machines* of Hinton *et al*. Boltzmann machines are discrete Hopfield networks that use simulated annealing with Gibbs sampling as the ordering mechanism for simulation. This distinguishes the Boltzmann machine from general Hopfield networks (which use arbitrary Monte Carlo methods).

It is important to note that when Gibbs sampling is applied as part of the *Hybrid Monte Carlo* algorithm for feedforward network training, that it serves a *learning* function. By contrast, training — in the machine learning sense — has already occurred at the stage where Gibbs sampling is applied in Boltzmann machines. Simulated annealing is used to avoid local minima during the convergence phase, where a pattern is presented and the network is stochastically updated until a stable attractor is reached. The use of constraint satisfaction networks as associative memories in the above method is generalizable over Bayesian and neural networks.

3.2. Constraint Satisfaction in Bayesian Networks

Apolloni and DeFalco discuss a biologically plausible specialization of parallel Boltzmann machines: their generalization to asymmetry. [3, 12] This stipulation turns out to be extremely useful for the study of probabilistic semantics of neural networks, because such constraint satisfaction networks (with normalizations of the weight to probabilities) dualize with causal, or Bayesian, networks. A simple Bayesian network is shown in Figure 1; its *sparse, bipartite Boltzmann machine dual* in Figure 2. Our only additional caveat is an additional semantic issue: namely, that the use of Bayesian networks as true *causal reasoning*, *evidential reasoning*, or *belief revision* systems depends on the degree of match between random variables and well-defined propositions about the system being modeled (events, predictions, etc.). Merely observing that a subset of Boltzmann machines can be interpreted as Bayesian networks does not automatically respect this property of the probabilistic model!

3.3. Hybrid Stochastic Methods for Feedforward Networks

The *Hybrid Monte Carlo* algorithm was investigated by Neal as as method for overcoming several shortcomings of traditional error backpropagation. [13] This new method for stochastic training augments backpropagation, and is similar to the traditional Metropolis loop (simulated annealing). The common idea is to perturb weights and to use weight decay in order to lower susceptibility to local minima, at a cost of slower convergence. The Hybrid Monte Carlo algorithm differs from the Metropolis algorithm by using a full Hamiltonian energy measure ("kinetic" as well as "potential" energy). The purpose of this augmentation is to account for effect of gradient on the *total* energy.

The new algorithm is amenable to the same weight perturbation and decay framework, and the resulting system has better empirical performance than that of pure backpropagation and Gaussian approximation of learning by backpropagation. [10] In principle, annealing with Hybrid Monte Carlo simulation facilitates arbitrarily close approximation (as opposed to requiring approximator components to be drawn from a family of Gaussian conditional distributions, cf. MacKay). [10] The main benefit of the general Bayesian framework is its avoidance of overfitting through regularized weight decay. Finally, the further improvement given by Gibbs sampling demonstrates the benefits of a "normalized" information theoretic measure of "free energy" (versus the traditional entropy measure, both absolute and relative). [13, 20] This supports development of a much-needed, higher-order metric for comparing *across* feedforward architectures with different learning rules and heuristics.

Neal's results strongly underscore the point that Markov chain Monte Carlo approaches are applicable to pure feedforward architectures, and not just a small subset of ANNs such as constraint satisfaction (e.g., Hopfield) networks. Moreover, augmentation of backpropagation learning by classical stochastic methods such as the Metropolis algorithm is shown to be an oversimplified approach, leaving much work (including theoretical developments) to be done in the area.

4. Conclusions and Ramifications

4.1. Gibbs Sampling for Bayesian Learning

We have surveyed the Gibbs sampler and shown how it is one of two optimizing improvements that can be applied in many annealing applications, as well as in learning situations for Bayesian networks. We have also discussed the orthogonal improvement of adding a momentum term to annealing by random sampling, that is justified in the statistical thermodynamics interpretation. These improvements together generate the Hybrid Monte Carlo family of algorithms. We have seen how Hybrid Monte Carlo is used to enhance performance of backprop learning. Both the Gibbs sampler and the Metropolis algorithm augmented with kinetic energy can be applied to other annealing problems,

including fusion and propagation for Bayesian networks (as a subset of stochastic convergence for constraint networks).

4.2. Causal Reasoning and Constraint Satisfaction

Another crucial research objective that arises from the interpretation of asymmetric parallel Boltzmann machines as Bayesian networks is the causal reasoning interpretation of this class of ANNs. First, this theory lends a much higher level of proximity to symbolic intelligent systems and the probabilistic basis of inductive learning. This improved understanding leads, in turn, to advances in application of Bayesian learning methods to such traditionally symbolic endeavors as case-based problem solving and analogical reasoning by prototypes. [11] Finally, we expect the Bayesian network interpretation to yield insights into incremental and reinforcement learning for knowledge-based systems in uncertain domains: problems to which ANNs have already been applied in force, but rarely with remarkable success.

4.3. Network Efficiently Representable Functions

A quantitative characterization that has been envisioned and recently sought after, also without extensive success, is that of **N***etwork* **E***fficiently* **R***epresentable* **F***unctions* or *NERFs*. This term, employed by Russell and Norvig, refers to the class of functions in a manifold which correspond to those, in a dual manifold of neural networks, that meet certain (unspecified) complexity restrictions. [2, 18] Our current research into spatiotemporal pattern recognition considers possible metrics for characterizing *NERF*s with respect to some temporal regularity (especially duration or periodicity). A potentially rich field of study in the theory of ANNs, hinging on the development of improved probabilistic and information theoretic semantics for certain ANN models, is this quantitative theory of *NERF*s.

4.4. Current and Future Work

Our current research investigates the ramifications of the Bayesian network/asymmetric Boltzmann machine duality in terms of: practical application of Gibbs sampling for supervised learning; extension of the known duality to temporal ("persistent") augmentations of Bayesian networks as compared to recurrent networks; and extension of the full Hybrid Monte Carlo approach (including the momentum-based model) to recurrent networks, in a probabilistically sound framework. We are focusing on the capability to acquire spatiotemporal regularities in both Bayesian and simple recurrent networks, with adaptivity ranging from simple persistence to more complex dynamical systems modeling. We believe that a characterization of NERFs for these capabilities is both a motivation for and a gauge of progress, in developing a theory for methods that integrate Bayesian and neural networks.

References

[1] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski. A learning algorithm for boltzmann machines. *Cognitive Science*, 9:147–169, 1985.

[2] S.-I. Amari. Information geometry of the EM and *em* algorithms for neural networks. *Neural Networks*, 8(9):1379–1408, 1995.

[3] B. Apolloni and D. de Falco. Learning by asymmetric parallel boltzmann machines. *Neural Comp.*, 3:402–408, 1991.

[4] H. A. Bourlard and N. Morgan. *Connectionist Speech Recognition*. Kluwer, Boston, MA, 1994.

[5] A. L. Delcher, S. Kasif, H. R. Goldberg, and W. H. Hsu. Probabilistic prediction of protein secondary structure using causal networks. In *Proceedings of the 11th National Conference on AI (AAAI-93)*, volume 11, pages 316–321, 1993.

[6] J. L. Elman. Finding structure in time. *Cognitive Science*, 14:179–211, 1990.

[7] D. Haussler. Quantifying inductive bias: AI learning algorithms and valiant's learning framework. *Artficial Intelligence*, 36:177–221, 1988.

[8] D. Heckerman. A tutorial on learning bayesian networks. Technical Report 95-06, Microsoft, 1995.

[9] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):832–834, 1983.

[10] D. J. C. MacKay. A practical bayesian framework for backpropagation networks. *Neural Comp.*, 4:448–472, 1992.

[11] P. Myllymäki. Mapping bayesian networks to boltzmann machines. In *Proceedings of Applied Decision Technologies 1995*, pages 269–280, 1995.

[12] R. M. Neal. Asymmetric parallel boltzmann machines are belief networks. *Neural Comp.*, 4:832–834, 1992.

[13] R. M. Neal. Bayesian training of backpropagation networks by the hybrid monte carlo method. Technical Report CRG-TR-92-1, University of Toronto, 1992.

[14] R. M. Neal. Connectionist learning of bayesian networks. *Artficial Intelligence*, 56:71–113, 1992.

[15] R. M. Neal. Probabilistic inference using monte carlo methods. Technical Report CRG-TR-93-1, University of Toronto, 1993.

[16] R. M. Neal. *Bayesian Learning for Neural Networks*. Springer-Verlag, New York, NY, 1996.

[17] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan-Kaufmann, San Mateo, CA, 1988.

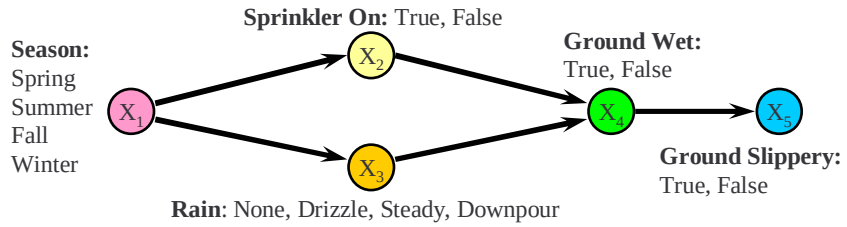
[18] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*, chapter 19. Prentice-Hall, Englewood Cliffs, NJ, 1995.

[19] S. Thrun. *Explanation-Based Neural Network Learning*. Kluwer Academic Publishers, Norwell, MA, 1996.

[20] J. York. Use of the gibbs sampler in expert systems. *Artficial Intelligence*, 56:115–130, 1992.

[21] X. Zhang, J. P. Mesirov, and D. L. Waltz. A hybrid system for protein secondary structure prediction. *Journal of Molecular Biology*, 1993.

Figure 1. A Simple Bayesian Network



A bipartite Hopfield network is generated by the dualization, or *network compilation*, process. This network is called an *asymmetric Boltzmann machine*, because $w_{ij} = w_{ji}$ need not necessarily hold, and because simulated annealing and Gibbs sampling are applied for convergence.

Left-hand side units in the ANN dual all represent prior probabilities; right-hand side units represent either priors for source vertices (i.e., those with no parents), or conditionals for vertices based on their parents (by the principle of ***d***-separation).

This example is adapted from one used by Pearl; the network compiler is based on a duality researched by Myllymäki.

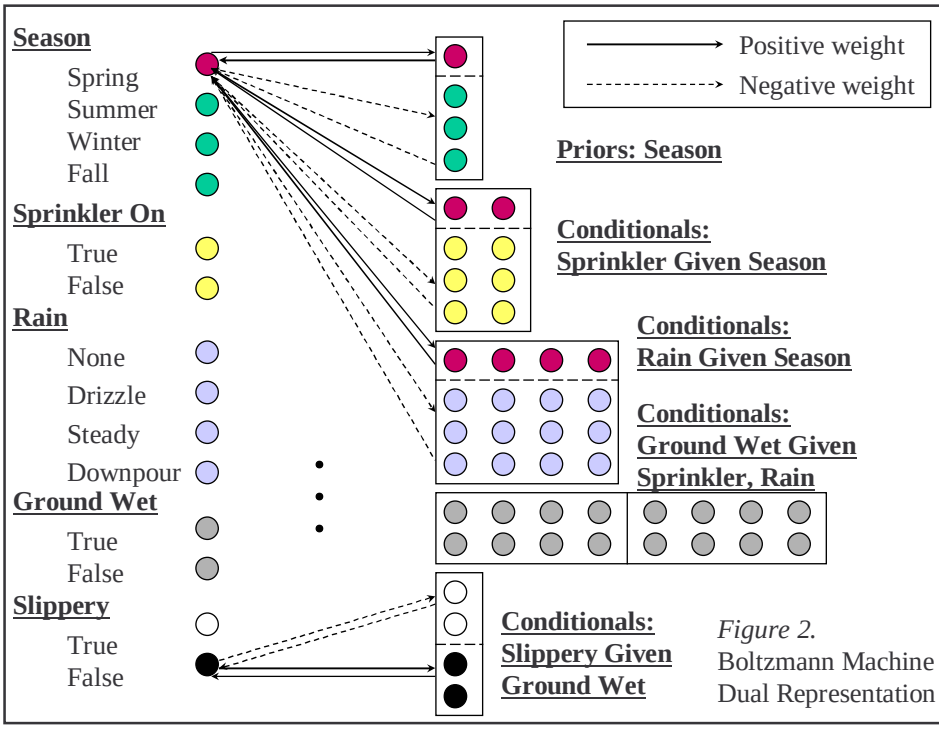


Figure 2. Boltzmann Machine Dual Representation