Properties of Propositional Logic Encoded into Restricted Boltzmann Machines

Ryan McArdle
4 May 2020
Institute for Artificial Intelligence
University of Georgia
Athens, Georgia, USA
rmcardle@uga.edu

***Abstract*—The current zeitgeist in artificial intelligence is one dominated by the application of artificial neural networks to solve a wide range of problems. However, their proliferation has brought to the forefront many concerns regarding our ability to understand and ultimately trust these networks that we so often employ in our decision making processes. We must develop methods which allow us to faithfully and efficiently audit our neural networks, and it appears that such a method may be feasible within Restricted Boltzmann Machines. A Restricted Boltzmann Machine is a statistical, energy based artificial neural network architecture which represents a joint probability distribution over the data with which it is trained, which can then be used to infer likely values for data that are missing or undefined in the training set. It has been shown that knowledge bases of propositional logic can be associated with a Restricted Boltzmann Machine which, once trained on the knowledge base, can identify with tractable computational complexity the truth value assignments which are models of said knowledge base. We explore here the properties of propositional logic when encoded into Restricted Boltzmann Machines in order to understand the behaviors of this new logic. We show that this method will faithfully recreate each of the explored properties of classical propositional logic in a connectionist network. This method presents a promising path forward for connectionist logic programming and suggests the possibility of representing more descriptive and abstracted logics.**

***Index Terms*—Knowledge representation, Artificial neural networks, Unsupervised learning, Statistical learning, Logic programming.**

I. Introduction

The current zeitgeist in artificial intelligence is one dominated by the application of artificial neural networks (ANNs) to solve a wide range of problems. The amount of data and large-scale parallel processing power widely available in modernity makes the training of these networks quite efficient compared to any attempts of previous decades. However, their proliferation has brought to the forefront many concerns regarding our ability to understand and ultimately trust these networks that we so often employ in our decision making process. These networks, due to their complicated structure modeling a massively high-dimensional space, are quite opaque to human interpretation and audit.

There have been advancements made in recent years to address this issue, however correcting our lack of understanding regarding a deep network's inner workings is still a major concern for the field [1], [2]. In general, it is quite difficult for a human to interpret how a trained ANN processes the provided data in the way that it does, or to construct one that will process data via some intended methodology. This can make it difficult to understand what metrics the ANN might be using in its decision process and, when trained on historical data that has been shown to be discriminatory, the ANN will simply replicate these human metrics that can have major impacts on people's lives [3]. As the field continues to apply ANN methods, we must develop methods which allow us to faithfully and efficiently audit our ANNs to ensure that their operation remains both under our control and within our approval. It appears that such a method may be feasible within Restricted Boltzmann Machines (RBMs) and their deep learning counterpart Deep Belief Networks (DBNs).

II. Background and Related Works

A. Restricted Boltzmann Machines

An RBM is a statistical, energy based ANN architecture which, following an unsupervised training process, represents a joint probability distribution over the training data. This distribution can then be used to infer likely values for data that are missing or undefined in the training set [4].

The graphical structure of a Boltzmann Machine consists of two layers of nodes, a "visible" layer and a "hidden" layer. In the Restricted Boltzmann Machine, nodes contained in one layer can be connected only with the nodes contained in the other layer, and all connections between nodes are bidirectional [4]. This restriction offers a significant advantage with regards to the overall computational complexity of the structure, while still allowing one to create faithful models of the input data [5].

Associated with each RBM is a function known as the energy:

$$E(\boldsymbol{x}, \boldsymbol{h}) = -\sum_{i,j} w_{ij} x_i h_j - \sum_i a_i x_i - \sum_j b_j h_j \qquad (1)$$

where $(\boldsymbol{x}, \boldsymbol{h})$ represents an assignment of values to each node of our machine, a_i and b_j are the biases of visible and hidden

nodes x_i and h_j respectively, and w_{ij} is the connection weight between nodes x_i and h_j. From this energy function, we can determine the joint probability distribution of assignment $(\boldsymbol{x}, \boldsymbol{h})$:

$$p(\boldsymbol{x}, \boldsymbol{h}) = \frac{1}{Z} e^{-\frac{1}{\tau} E(\boldsymbol{x}, \boldsymbol{h})} \tag{2}$$

where

$$Z = \sum_{\boldsymbol{x}, \boldsymbol{h}} e^{-\frac{1}{\tau} E(\boldsymbol{x}, \boldsymbol{h})} \tag{3}$$

represents the partition function over all distributions, with 'temperature' τ [5].

This partition function in general can be quite expensive to compute explicitly. However, given any initial state of the nodes, the minimized energy function of an RBM can be approximated arbitrarily well through a tractable Markov chain Monte Carlo process of alternatively resampling the values of both hidden and visible nodes in an iterative process known as Gibbs Sampling [6]. As real-valued RBMs and the necessary computational methods for minimizing intractible partition functions are outside of the scope of this work, the reader is directed towards [4] and [6] for an introduction to the Gibbs Sampling process and its use in training RBMs.

A standard use of the RBM structure interprets visible nodes as corresponding with training data while the hidden layer would express a level of abstraction corresponding to some shared feature of elements of the input data. Once trained, the architecture will represent a joint probability distribution over a potentially incomplete dataset and can then fill in the data set by sampling from the probability distribution [4].

This application has gained attention in recent years when used with the Netflix data set to predict user's movie ratings better than Netflix's own algorithm [7]. This application created visible nodes whose values represented a user's ratings for movies, both known and unknown, and hidden nodes which would represent hidden features shared by movies (inclusion in certain genres, sharing directors or actors, etc.). The algorithm could then predict the user's ratings for unrated movies, filling in the values of missing data points in the input set.

B. Deep Belief Networks

A Deep Belief Net (DBN) is a neural network architecture in which multiple RBMs are stacked onto one another such that the hidden nodes of one RBM act as the visible nodes of the next. While DBNs do suffer from being more computationally complex and their conditional probabilities may be more difficult to compute exactly, they benefit from being able to leverage the multiple levels of RBMs in order to model the data at higher levels of abstraction [8]. This makes this structure particularly effective and robust in classification problems, in which it can leverage ontological classification information which can even be transfered between networks [9], [10]. This ontic approach allows for great modularity in the application of these networks and an ability to benefit from prior training for a range of problems, rather than needing to fully retrain each time the problem is adjusted. It is therefore the author's hope that a proof of concept for a faithful representation of propositional logic in a single RBM layer could prompt work with deeper networks, potentially representing logics with a higher potential for abstraction and expression.

C. Knowledge Representation in RBMs and DBNs

It has been shown in [5] that a knowledge base expressed in propositional logic, when decomposed into a 'Strict Disjunctive Normal Form' (SDNF) in which at most one conjunctive clause holds given an assignment, can be associated with an RBM whose visible nodes correspond to the literals of the knowledge base, whose hidden nodes correspond to the clauses of the SDNF, and whose energy function is determined by the SDNF. The states of the RBM which minimze the energy function are shown to correspond to valuations which will satisfy the knowledge base if it is consistent or provide a maximum satisfiability in the case of weighted logics. It has further been shown that this process can be implemented in reverse in a DBN, isolating a single RBM layer and reverse engineering a logical expression which represents the nodal relationships in said layer and allowing for a process of knowledge extraction from the DBN [8].

The author of [5] points out that while SDNF is more complex and demanding to compute than the normal DNF, which is generally quite expensive, this process is efficient for logical implications. As many knowledge bases are already presented in the form of facts and logical implication rules, there is promise that this method could be tractable for real-world knowledge bases. Further, the number of nodes in the RBM associated with a logical implication grows linearly with the number of literals in the implication, so the entire process, including conversion to SDNF, representation as RBM, and Gibbs Sampling to train the RBM, should be tractably efficient in the case of real-world knowledge bases.

III. Logic Representation in RBMs

We reproduce here the important theorems of [5], which will be the basis for our analysis of the RBM encoded logic.

A. Strict Disjunctive Normal Form

Definition 3.1. *[5]*

- *A "strict DNF" (SDNF) is a DNF where at most one single conjunctive clause is True at a time.*
- *A "full DNF" is a DNF where each variable must appear at least once in every conjunctive clause."*

The author claims that any propositional well-formed formula can be presented as a full DNF which is also an SDNF and further provides a proof of and process for converting a general logical implication into this form.

Theorem 3.2. *[5] A logical implication* $y \leftarrow \bigwedge_{t \in S_T} x_t \wedge \bigwedge_{k \in S_K} \neg x_k$ *where* S_T, S_K *respectively are the sets of positive*

and negative propositions' indices, can be represented as an SDNF having the form:

$$\left(y \wedge \bigwedge_{t \in S_T} x_t \wedge \bigwedge_{k \in S_K} \neg x_k\right) \vee \bigvee_{p \in S_T \cup S_K} \left(\bigwedge_{t \in S_T.\backslash p} x_t \wedge \bigwedge_{k \in S_K.\backslash p} \neg x_k \wedge x'_p\right)$$

where $S.\backslash p$ denotes a set S where p has been removed, and $x'_p \equiv \neg x_p$ if $p \in S_T$ else $x'_p \equiv x_p$.

B. Equivalence of an SDNF and an RBM

In order to represent some WFF ϕ as an RBM, we must first define what will be considered to be an equivalence between the two structures:

Definition 3.3. *[5] A WFF φ is equivalent to a neural network $\mathcal{N}$ if and only if for any truth assignment $\boldsymbol{x}$, $s_\varphi(\boldsymbol{x}) = -AE_{rank}(\boldsymbol{x}) + B$, where $s_\varphi(\boldsymbol{x}) \in \{0, 1\}$ is the truth value of φ given $\boldsymbol{x}$ with True $\equiv 1$ and False $\equiv 0$; $A > 0$ and B are constants; $E_{rank}(\boldsymbol{x}) = \min_{\boldsymbol{h}} E(\boldsymbol{x}, \boldsymbol{h})$ is the energy ranking function of $\mathcal{N}$ minimised over all hidden units.*

This definition of equivalence guarantees that every preferred model of a WFF φ would also minimize the energy of the network $\mathcal{N}$.

We now show that it is possible to map φ to a Symmetric Connectionist Network (SCN) with an appropriate energy function.

Lemma 3.4. *[5] Any SDNF $\varphi \equiv \bigvee_j (\bigwedge_{t \in S_{T_j}} x_t \wedge \bigwedge_{k \in S_{K_j}} \neg x_k)$ can be mapped onto a SCN with energy function*

$$E = -\sum_j \prod_{t \in S_{T_j}} x_t \prod_{k \in S_{K_j}} (1 - x_k)$$

where S_{T_j} and S_{K_j} are respectively the set of T_j indices of positive literals and the set of K_j indices of negative literals.

We are now prepared to convert any formula φ into an RBM using its SDNF.

Theorem 3.5. *[5] Any SDNF $\varphi \equiv \bigvee_j (\bigwedge_{t \in S_{T_j}} x_t \wedge \bigwedge_{k \in S_{K_j}} \neg x_k)$ can be mapped onto an equivalent RBM with energy function*

$$E = -\sum_j h_j \left(\sum_{t \in S_{T_j}} x_t - \sum_{k \in S_{K_j}} x_k - T_j + \epsilon\right),$$

where $0 < \epsilon < 1$ and S_{T_j} and S_{K_j} are respectively the set of T_j indices of positive literals and the set of K_j indices of negative literals.

In particular, we can now present the core result that the logical implication studied above can be represented by an RBM.

Theorem 3.6. *[5] A logical implication $y \leftarrow \bigwedge_{t \in S_T} x_t \wedge \bigwedge_{k \in S_K} \neg x_k$ can be represented by an RBM with the energy function:*

$$E = -h_y \left(\sum_{t \in S_T} x_t - \sum_{k \in S_K} x_k + y - T - 1 + \epsilon\right) - \sum_{p \in S_T \cup S_K} h_p \left(\sum_{t \in S_T.\backslash p} x_t - \sum_{k \in S_K.\backslash p} x_k + x'_p - |S_T.\backslash p| - \mathbb{I}_{p \in S_K} + \epsilon\right)$$

where $|S_T.\backslash p|$ is the cardinality of the set $S_T.\backslash p$; if $p \in S_T$, then $x'_p = -x_p$ and $\mathbb{I}_{p \in S_K} = 0$, else $x'_p = x_p$ and $\mathbb{I}_{p \in S_K} = 1$

IV. Analysis of RBM Logic Representation

Here we explore some properties of classical propositional logic semantics, namely transitivity, *ex falso quodlibet* (the "principle of explosion"), disjunctive syllogism, resolution, and a modified resolution refutation, to see whether they are faithfully recreated in toy models of the method shown above, which we will henceforth refer to as RBM Logic. We will encode preconditions for each of these properties into RBM Logic, analyze the mathematical behavior of their energy functions, and interpret the logical consequences of the preferred valuations appropriately.

A. Transitivity

The property of transitivity is foundational to making chains of arguments beginning with premises and inferring towards conclusions. With respect to material implication $\rightarrow$, transitivity is defined:

Definition 4.1. *The logical implication $\rightarrow$ is said to be transitive if and only if:*

$$KB \vDash (P \rightarrow Q) \wedge (Q \rightarrow R) \Rightarrow KB \vDash P \rightarrow R$$

for any knowledge base KB and sentences P,Q, and R. This rule can be expressed syntactically as

$$\frac{P \rightarrow Q,\ Q \rightarrow R}{P \rightarrow R}$$

It can be shown that when represented in RBM Logic, the property of transitivity for logical implication does hold.

Theorem 4.2. *When two logical implications are encoded into RBM Logic using Theorem 3.5, the defined RBM and corresponding energy function behave such that the property of transitivity holds.*

Proof. First, we will define our KB to prime the system for transitivity.

$$KB \equiv (P \rightarrow Q) \wedge (Q \rightarrow R) \tag{4}$$

We now must show that, using the RBM energy function, any model of KB is also a model of $P \rightarrow R$. First, we convert (4) to SDNF:

$$KB \equiv (P \wedge Q \wedge R) \vee (\neg P \wedge Q \wedge R) \vee (\neg P \wedge \neg Q) \quad (5)$$

Using Theorem 3.5 and defining $\epsilon = 0.5$ (we will use this value for ϵ implicitly throughout this work), we are able to define an RBM and energy function from (5):

$$E = -h_1(P + Q + R - 2.5) - h_2(-P + Q + R - 1.5) - h_3(-P - Q + 0.5) \quad (6)$$

We note here that we make use of the more general Theorem 3.5 rather than the more complex 3.6, which does deal explicitly with logical implications. This choice has been made primarily for a consistent application of 3.5 throughout, which is more naturally employed in the remainder of our proofs. It is expected that a simple summation of implication energy functions should serve to represent the conjunction of implications in a knowledge base, but further analysis is desired to confirm this.

We now consider the truth value assignments $\boldsymbol{x}_i$ which minimize (6). In general, this process will work for real-valued or weighted logics. However this is beyond the scope of this current work and we will simply consider valuations for which 0 and 1 are the possible assignments.

Assignment Energy Functions				
$\boldsymbol{x}_i$	P	Q	R	Energy Function
$\boldsymbol{x}_1$	0	0	0	$2.5h_1 + 1.5h_2 - 0.5h_3$
$\boldsymbol{x}_2$	0	0	1	$1.5h_1 + 0.5h_2 - 0.5h_3$
$\boldsymbol{x}_3$	0	1	0	$1.5h_1 + 0.5h_2 + 0.5h_3$
$\boldsymbol{x}_4$	0	1	1	$0.5h_1 - 0.5h_2 + 0.5h_3$
$\boldsymbol{x}_5$	1	0	0	$1.5h_1 + 2.5h_2 + 0.5h_3$
$\boldsymbol{x}_6$	1	0	1	$0.5h_1 + 1.5h_2 + 0.5h_3$
$\boldsymbol{x}_7$	1	1	0	$0.5h_1 + 1.5h_2 + 1.5h_3$
$\boldsymbol{x}_8$	1	1	1	$-0.5h_1 + 0.5h_2 + 1.5h_3$

TABLE I: Shows truth assignments $\boldsymbol{x}_i$ and simplified energy function (6) for each assignment.

Minimized Energies/Models			
$\boldsymbol{x}_i$	Minimized Energy	Model of KB	Transitivity $(P \to R)$
$\boldsymbol{x}_1$	**-0.5**	**Yes**	**Yes**
$\boldsymbol{x}_2$	**-0.5**	**Yes**	**Yes**
$\boldsymbol{x}_3$	0.0	No	**Yes**
$\boldsymbol{x}_4$	**-0.5**	**Yes**	**Yes**
$\boldsymbol{x}_5$	0.0	No	No
$\boldsymbol{x}_6$	0.0	No	**Yes**
$\boldsymbol{x}_7$	0.0	No	No
$\boldsymbol{x}_8$	**-0.5**	**Yes**	**Yes**

TABLE II: Shows the minimized energy for each assignment $\boldsymbol{x}_i$, whether the assignment is a model of KB and if transitivity holds for that assignment.

In Table I, we express the energy functions of each possible truth value assignment $\boldsymbol{x}_i$. This corresponds to setting each of the visible nodes of the RBM to a fixed value, and we then assign values for each h_j such that the energy function for that valuation is minimized. In practice, the range of each of these functions are a subset of the range of values for the energy function (6) that will be minimized by sampling for both the hidden and visible node values through Gibbs Sampling. Presenting the energy functions for fixed assignments $\boldsymbol{x}_i$ allows us to explore more thoroughly the behaviors of this method with respect to possible valuations, while still being able to identify the models associated with global minima.

Once we identify the set of valuations which minimize to the lowest energy values, the method claims that we have identified the set of models of KB. In order to prove our theorem, we must show that transitivity holds in each of these identified models of KB, i.e. $P \to R$.

Observing Table II, we see that the truth assignments $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, $\boldsymbol{x}_4$, and $\boldsymbol{x}_8$ all have a minimized energy function value of -0.5, the lowest of any assignments (note also that this value is $-\epsilon$). Thus, these assignments would be preferred in the Gibbs Sampling minimization training process for the RBM. We also see that these four assignments are the only assignments which are models of KB, so the minimization process has properly isolated exactly those truth assignments which satisfy KB. Finally, we note that for each of these four assignments transitivity holds, as the sentence $P \to R$ is semantically entailed. We have therefore shown through the RBM energy function method that $KB \vDash (P \to Q) \wedge (Q \to R) \Rightarrow KB \vDash P \to R$. ■

B. Ex Falso Quodlibet

The property *Ex Falso Quodlibet*, also known as the "Principle of Explosion", states that from a contradiction, anything can be derived. Formally:

Definition 4.3 (*Ex Falso Quodlibet*)**.** *The rule of Ex Falso Quodlibet holds in a logic if and only if:*

$$KB \vDash (P \wedge \neg P) \Rightarrow KB \vDash Q$$

for any knowledge base KB and sentences P and Q. This rule can be expressed syntactically as

$$\frac{P,\ \neg P}{Q}$$

Theorem 4.4. *Ex Falso Quodlibet holds in RBM Logic, such that in the event of a contradiction, the system will hold no preference for any valuation and any literal may be posited.*

Proof. We first define a contradictory knowledge base:

$$KB \equiv P \wedge \neg P \wedge R \wedge (Q \vee \neg Q). \quad (7)$$

We include the literal R to explore the systems response to literals of our knowledge base well-founded despite the contradiction. We include the tautology $Q \vee \neg Q$ to explicitly include Q as literal of concern and a visible node in our RBM in order to analyze the systems response to otherwise unfounded literals.

We now express (7) in SDNF:

$$KB \equiv (P \wedge \neg P \wedge R \wedge Q) \vee (P \wedge \neg P \wedge R \wedge \neg Q) \quad (8)$$

and define our energy function:

$$E = h_1(P - P + R + Q - 2.5) - h_2(P - P + R - Q - 1.5).$$

As we can see, each clause of the SDNF in which the contradiction holds has the expression $P - P$ included. These terms will consistently cancel each other out, and our energy function can therefore be simplified to:

$$E = -h_1(R + Q - 2.5) - h_2(R - Q - 1.5). \tag{9}$$

We now consider the truth value assignments $\boldsymbol{x}_i$ which minimize (9). Note that because P and $-P$ have been removed from our energy function, the only relevant variable assignments are on Q and R.

Assignment Energy Functions				
$\boldsymbol{x}_i$	Q	R	Energy Function	Minimized Energy
$\boldsymbol{x}_1$	0	0	$2.5h_1 + 1.5h_2$	0.0
$\boldsymbol{x}_2$	0	1	$1.5h_1 + 0.5h_2$	0.0
$\boldsymbol{x}_3$	1	0	$1.5h_1 + 2.5h_2$	0.0
$\boldsymbol{x}_4$	1	1	$0.5h_1 + 1.5h_2$	0.0

TABLE III: Shows truth assignments $\boldsymbol{x}_i$, simplified energy function (9), and the minimized energy value for each assignment.

Observing Table III, we note that any valuations over Q and R will provide a minimized energy value of 0.0, i.e. no preference will be given by the RBM method to any valuations and any literal may be posited. ■

C. Disjunctive Syllogism

An important rule of inference used in classical logic is that of Disjunctive Syllogism.

Definition 4.5 (Disjunctive Syllogism)**.** *The rule of Disjunctive Syllogism holds in a logic if and only if*

$$KB \vDash (P \vee Q) \wedge \neg P \Rightarrow KB \vDash Q$$

for any knowledge base KB and sentences P and Q. This rule can be expressed syntactically as

$$\frac{(P \vee Q),\ \neg P}{Q}.$$

Theorem 4.6. *Disjunctive Syllogism holds in RBM Logic.*

Proof. We begin by defining a knowledge base

$$KB \equiv (P \vee Q) \wedge \neg P$$

and converting it into SDNF

$$KB \equiv (P \wedge \neg P) \vee (Q \wedge \neg P). \tag{10}$$

Using Theorem 3.5, we create an energy function from (10)

$$E = -h_1(P - P - 0.5) - h_2(Q - P - 0.5)$$

and cancel out $P - P$ to get

$$E = -h_1(-0.5) - h_2(Q - P - 0.5). \tag{11}$$

Assignment Energy Functions			
$\boldsymbol{x}_i$	P	Q	Energy Function
$\boldsymbol{x}_1$	0	0	$0.5h_1 + 0.5h_2$
$\boldsymbol{x}_2$	0	1	$0.5h_1 - 0.5h_2$
$\boldsymbol{x}_3$	1	0	$0.5h_1 + 1.5h_2$
$\boldsymbol{x}_4$	1	1	$0.5h_1 + 0.5h_2$

TABLE IV: Shows truth assignments $\boldsymbol{x}_i$ and simplified energy function (11) for each assignment.

Minimized Energies/Models			
$\boldsymbol{x}_i$	Minimized Energy	Model of KB	Disjunctive Syllogism (Q)
$\boldsymbol{x}_1$	0.0	No	No
$\boldsymbol{x}_2$	**-0.5**	**Yes**	**Yes**
$\boldsymbol{x}_3$	0.0	No	No
$\boldsymbol{x}_4$	0.0	No	**Yes**

TABLE V: Shows the minimized energy for each assignment $\boldsymbol{x}_i$, whether the assignment is a model of KB, and if Disjunctive Syllogism holds for that assignment.

We now consider possible truth value assignments $\boldsymbol{x}_i$ and identify the valuations which minimize (11).

Observing Table V, we see that $\boldsymbol{x}_2$ is the only valuation with minimial energy, and it is both a model of KB and assigns Q a value of *True*, i.e. Disjunctive Syllogism holds. ■

D. Resolution

One of the central tools of logic programming is the method of resolution, which is a process of eliminating complementary literals from conjoined disjunctive clauses through Disjunctive Syllogism.

Definition 4.7. *The generalized resolution rule can be stated as*

$$\frac{l_1 \vee \ldots \vee l_k, \qquad m_1 \vee \ldots \vee m_n}{\begin{array}{r} l_1 \vee \ldots \vee l_{i-1} \vee l_{i+1} \vee \ldots \vee l_k \vee m_1 \vee \ldots \\ \vee\, m_{j-1} \vee m_{j+1} \vee \ldots \vee m_n \end{array}}, \tag{12}$$

where l_i and m_j are complementary literals, i.e. $l_i \equiv \neg m_j$ [11].

Theorem 4.8. *The generalized rule of resolution holds in RBM Logic for resolvents of the form $(P \vee Q) \wedge (\neg P \vee R)$. That is:*

$$KB \vDash (P \vee Q) \wedge (\neg P \vee R) \Rightarrow KB \vDash (Q \vee R) \tag{13}$$

Proof. For compactness and simplicity's sake, we only show this for short resolvents. Long cases should hold just as well.

We begin by defining a knowledge base on which to test the validity of the resolution rule

$$KB \equiv (P \vee Q) \wedge (\neg P \vee R),$$

and express it in SDNF:

$$\begin{aligned} KB \equiv (\neg P \wedge Q \wedge \neg R) \wedge (\neg P \wedge Q \wedge R) \wedge \\ (P \wedge \neg Q \wedge R) \wedge (P \wedge Q \wedge R). \end{aligned} \tag{14}$$

Using Theorem 3.5, we define an energy function to represent (14):

$$E = -h_1(-P + Q - R - 0.5) - h_2(-P + Q + R - 1.5) - h_3(P - Q + R - 1.5) - h_4(P + Q + R - 2.5) \quad (15)$$

and identify the valuations $\boldsymbol{x}_i$ which minimize (15).

Assignment Energy Functions				
$\boldsymbol{x}_i$	P	Q	R	Energy Function
$\boldsymbol{x}_1$	0	0	0	$0.5h_1 + 1.5h_2 + 1.5h_3 + 2.5h_4$
$\boldsymbol{x}_2$	0	0	1	$1.5h_1 + 0.5h_2 + 0.5h_3 + 1.5h_4$
$\boldsymbol{x}_3$	0	1	0	$-0.5h_1 + 0.5h_2 + 2.5h_3 + 1.5h_4$
$\boldsymbol{x}_4$	0	1	1	$0.5h_1 - 0.5h_2 + 1.5h_3 + 0.5h_4$
$\boldsymbol{x}_5$	1	0	0	$1.5h_1 + 2.5h_2 + 0.5h_3 + 1.5h_4$
$\boldsymbol{x}_6$	1	0	1	$2.5h_1 + 1.5h_2 - 0.5h_3 + 0.5h_4$
$\boldsymbol{x}_7$	1	1	0	$0.5h_1 + 1.5h_2 + 1.5h_3 + 0.5h_4$
$\boldsymbol{x}_8$	1	1	1	$1.5h_1 + 0.5h_2 + 0.5h_3 - 0.5h_4$

TABLE VI: Shows truth assignments $\boldsymbol{x}_i$ and simplified energy function (15) for each assignment .

Minimized Energies/Models			
$\boldsymbol{x}_i$	Minimized Energy	Model of KB	Resolution $(Q \vee R)$
$\boldsymbol{x}_1$	0.0	No	No
$\boldsymbol{x}_2$	0.0	No	**Yes**
$\boldsymbol{x}_3$	**-0.5**	**Yes**	**Yes**
$\boldsymbol{x}_4$	**-0.5**	**Yes**	**Yes**
$\boldsymbol{x}_5$	0.0	No	No
$\boldsymbol{x}_6$	**-0.5**	**Yes**	**Yes**
$\boldsymbol{x}_7$	0.0	No	**Yes**
$\boldsymbol{x}_8$	**-0.5**	**Yes**	**Yes**

TABLE VII: Shows the minimized energy for each assignment, as well as whether the assignment is a model of KB and if resolution holds for that assignment.

Observing Table VII, we see that $\boldsymbol{x}_3$, $\boldsymbol{x}_4$, $\boldsymbol{x}_6$, and $\boldsymbol{x}_8$ are the preferred valuations with minimized energy value -0.5. We also note that each of these valuations is a model of KB and that the generalized resolution rule holds for each one. ■

E. Resolution Refutation

A standard method of checking for entailment or consistency in logic programming is resolution refutation. This process takes a knowledge base KB and a query Q, creates a new knowledge base $KB' \equiv KB \cup \{\neg Q\}$ and repeatedly applies resolution to the sentences of the new knowledge base KB'. If the empty clause is derived through this process, then KB' is shown to be inconsistent, and thus $KB \vDash Q$ is proven. Because this refutation can be made for any knowledge base and query, resolution is considered a refutation-complete inference technique [12].

We now consider a similar process in the RBM Logic. We first define our idea of resolution refutation within this method.

Definition 4.9. *Resolution refutation in the RBM Logic will be defined as the process of adding the query Q to the knowledge base KB, then creating and minimizing the resulting energy function based upon $KB \cup \{Q\}$.*

Notice that this does differ from the standard resolution refutation process in that Q is added to KB, rather than $\neg Q$. This convention of definition allows the following resulting theorem to be more intuitive.

Theorem 4.10. *Given a knowledge base KB and a query Q, the RBM Logic will prefer no valuations if $KB \cup \{Q\}$ is inconsistent and will prefer models in which $Q \equiv$ True if $KB \cup \{Q\}$ is consistent.*

Proof.
Claim 1: The RBM Logic will prefer no models if $KB \cup \{Q\}$ is inconsistent

Subproof. We define a simple knowledge base:

$$KB \equiv P \wedge (P \rightarrow Q)$$

and inconsistent query:

$$\neg Q.$$

We add our query to our knowledge base and get

$$KB' \equiv P \wedge (P \rightarrow Q) \wedge \neg Q,$$

which we then convert into SDNF

$$KB' \equiv (P \wedge \neg P \wedge \neg Q) \vee (P \wedge Q \wedge \neg Q). \quad (16)$$

We note here that each of our conjunctive clauses contains an explicit contradiction, i.e. $P \wedge \neg P$ and $Q \wedge \neg Q$. As such, none of the clauses in (16) can actually be satisfied, as is to be expected in the inconsistent case. Further, the SDNF of KB' amounts to conjoining the query into each of our conjunctive clauses in the SDNF of KB.

We now use (16) to define our energy function:

$$E = -h_1(P - P - Q - 1 + 0.5) - h_2(P + Q - Q - 2 + 0.5),$$

from which we cancel out contradictions and simplify to:

$$E = -h_1(-Q - 0.5) - h_2(P - 1.5). \quad (17)$$

We now consider the possible valuations $\boldsymbol{x}_i$ and identify those which minimize the energy function (17).

Assignment Energy Functions			
$\boldsymbol{x}_i$	P	Q	Energy Function
$\boldsymbol{x}_1$	0	0	$0.5h_1 + 1.5h_2$
$\boldsymbol{x}_2$	0	1	$1.5h_1 + 1.5h_2$
$\boldsymbol{x}_3$	1	0	$0.5h_1 + 0.5h_2$
$\boldsymbol{x}_4$	1	1	$1.5h_1 + 0.5h_2$

TABLE VIII: Shows truth assignments $\boldsymbol{x}_i$ and simplified energy function (17) for each assignment .

As we expected, there are no assignments which would serve as a model of $KB \cup \{Q\}$. We can also see from Table IX that all assignments have the same minimized energy, and as such none are selected as preferred assignments. □

Claim 2: The RBM Logic will prefer valuations $\boldsymbol{x}_i$ in which $Q \equiv_{\boldsymbol{x}_i}$ *True* if $KB \cup \{Q\}$ is consistent.

Minimized Energies/Models			
$\boldsymbol{x}_i$	Minimized Energy	Model of KB	Model of $KB \cup \{\neg Q\}$
$\boldsymbol{x}_1$	0.0	No	No
$\boldsymbol{x}_2$	0.0	No	No
$\boldsymbol{x}_3$	0.0	No	No
$\boldsymbol{x}_4$	0.0	**Yes**	No

TABLE IX: Shows the minimized energy for each assignment $\boldsymbol{x}_i$, whether the assignment is a model of KB, and whether the assignment is a model of $KB \cup \{\neg Q\}$.

Subproof. We use the same knowledge base as before, but instead now offer Q as our query. Therefore,

$$KB' \equiv P \wedge (P \rightarrow Q) \wedge Q,$$

and when converted into SDNF:

$$KB' \equiv (P \wedge \neg P \wedge Q) \vee (P \wedge Q \wedge Q). \tag{18}$$

We now define our energy function to represent (18)

$$E = -h_1(P - P + Q - 2 + 0.5) \\ - h_2(P + Q + Q - 3 + 0.5)$$

and simplify it to

$$E = -h_1(Q - 1.5) - h_2(P + 2Q - 2.5). \tag{19}$$

We now consider possible evaluations $\boldsymbol{x}_i$ and identify those which minimize energy function (19).

Assignment Energy Functions			
$\boldsymbol{x}_i$	P	Q	Energy Function
$\boldsymbol{x}_1$	0	0	$1.5h_1 + 2.5h_2$
$\boldsymbol{x}_2$	0	1	$0.5h_1 + 0.5h_2$
$\boldsymbol{x}_3$	1	0	$1.5h_1 + 1.5h_2$
$\boldsymbol{x}_4$	1	1	$0.5h_1 - 0.5h_2$

TABLE X: Shows truth assignments $\boldsymbol{x}_i$ and simplified energy function (19) for each assignment.

Minimized Energies/Models			
$\boldsymbol{x}_i$	Minimized Energy	Model of KB	Model of $KB \cup \{Q\}$
$\boldsymbol{x}_1$	0.0	No	No
$\boldsymbol{x}_2$	0.0	No	No
$\boldsymbol{x}_3$	0.0	No	No
$\boldsymbol{x}_4$	**-0.5**	**Yes**	**Yes**

TABLE XI: Shows the minimized energy for each assignment $\boldsymbol{x}_i$, whether the assignment is a model of KB, and whether it is a model of $KB \cup \{Q\}$.

We see then that $\boldsymbol{x}_4$ is the valuation with minimum energy and the only valuation which serves as a model for $KB \cup \{Q\}$. □

■

V. Results

We have therefore seen that each of the studied properties either hold exactly as they would be expected to hold in standard propositional logic, or (as in the case of resolution refutation) hold in a slightly modified but similar fashion.

In the case of contradictions, one may note that the inclusion of a complementary pair of literals in a single conjunctive clause of the SDNF leads to the associated hidden node never activating. In the energy function term for such an h_j, the canceling out of a positive literal x_i, which initially contributed to an increased $|T_j|$, force the coefficient of h_j to remain negative regardless of any possible binary assignment $\boldsymbol{x}_i$. As such, h_j must always receive a value of 0 in order to minimize the energy function. Since this value never varies, its assignment will have no influence on the value of the energy function, and no valuation can be preferred by this term. In the case of a necessarily contradictory knowledge base, every potential model for KB must include a pair of these contradictory literals, and the result is that every term associated with some h_j will contain the complementary pair, forcing $h_j = 0$ for all j. In this situation, no truth assignment will be preferred, and none could be considered a viable model.

The above discussion of contradictions suggests that a change to the value T_j, or rather, the addition of another term to the energy function, could be made to adjust for the issue caused by contradictory literals. Further investigation into this suggestion is warranted, but it seems that adding 1 to the coefficients of energy terms containing contradictory literals would completely negate the effects of contradictions in each of the conjunctive SDNF clauses and result in a system similar to the paraconsistent three-valued logics.

Further analysis remains for weighted or real-valued logics, but it seems that this system may be able to more robustly handle the possibility of contradictions when the value of assignments is not strictly limited to $\{0, 1\}$. We note that in the above examples, valuations would have minimized energy values of either $-\epsilon$ or 0.0 for models and non-models respectively. It is suspected that when preferences can be given to the facts and rules of KB or when the nodes of the RBMs can be assigned values between 0 and 1, rather than the classical binary assignments, a wider range of minimized values may appear.

VI. Summary

Formal logic in a rudimentary form may be the oldest attempt at modeling intelligence undertaken by humans, and it has taken significant strides throughout the 20th century. However, the potential to employ massively-parallelized connectionist techniques for modeling intelligence and decision making processes is a power that should not be understated. An efficient marriage of these two methodologies would be monumental in the study of intelligent systems.

The RBM Logic presented in [5] is not yet well studied

and the computational efficiency of such a logical system may still prove to be a challenge, but we have seen here that it is quite possible to model many desirable properties of a formal logic in a system based upon current technologies in order to make logical inferences and check for entailment or unstaisfiability. The method presents a promising path forward for connectionist logic programing, however it remains to be seen whether this methodology could be manipulated to represent more expressive logics or if this methodolgy is simply restricted to a grounded or propositional logic.

In future work, we intend to analyze RBM Logic in real-valued and weighted versions of propositional logic to investigate the same properties explored here, and also to implement the methods presented in order to take a knowledge base, convert it to the necessary form, create and train the associated RBM, and empirically study the overall efficiency and viability of this method as compared to other logical inference methods.

REFERENCES

[1] R. Guidotti, A. Monreale, S. Ruggieri, F. Turin, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models.," *ACM COMPUTING SURVEYS*, vol. 51, no. 5, n.d.

[2] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should i trust you?"," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*, 2016.

[3] A. Caliskan, J. Bryson, and A. Narayanan, "Semantics derived automatically from language corpora contain human-like biases," *Science*, vol. 356, pp. 183–186, 4 2017.

[4] A. Fischer and C. Igel, "An introduction to restricted boltzmann machines," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications* (L. Alvarez, M. Mejail, L. Gomez, and J. Jacobo, eds.), (Berlin, Heidelberg), pp. 14–36, Springer Berlin Heidelberg, 2012.

[5] S. Tran, "Propositional knowledge representation in restricted boltzmann machines," *ArXiv*, 05 2017.

[6] G. Casella and E. I. George, "Explaining the gibbs sampler," *The American Statistician*, vol. 46, no. 3, pp. 167–174, 1992.

[7] R. Salakhutdinov, A. Mnih, and G. Hinton, "Restricted boltzmann machines for collaborative filtering," in *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, (New York, NY, USA), p. 791–798, Association for Computing Machinery, 2007.

[8] S. Tran and A. S. d'Avila Garcez, "Deep logic networks: Inserting and extracting knowledge from deep belief networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, pp. 246–258, February 2018.

[9] H. Larochelle and Y. Bengio, "Classification using discriminative restricted boltzmann machines," in *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, (New York, NY, USA), p. 536–543, Association for Computing Machinery, 2008.

[10] H. Wang, D. Dou, and D. Lowd, "Ontology-based deep restricted boltzmann machine," in *Database and Expert Systems Applications* (S. Hartmann and H. Ma, eds.), (Cham), pp. 431–445, Springer International Publishing, 2016.

[11] S. J. Russell and P. Norvig, *Artificial Intelligence a Modern Approach*. Prentice Hall, 3 ed., 2010.

[12] R. J. Brachman and H. J. Levesque, *Knowledge representation and reasoning*. Morgan Kaufmann, 2004.