

Neural Network Semantics

CALEB SCHULTZ KISBY

Acknowledgments

djafkl;djsal;fkjas fl;jsa df;lkajs f;lkjsa l;fkjs dal;kfj al;sdfj ;lksajf ;lsakj f;lsaj f;lsaj f;lks
jf;lksaj fl;ksaj f;lsakj fl;k js;lkfj dsa
f;as fk;lsajf ;lsakj flk;jas f
sajfl;kas jfl;kj sda;lkfja
sfjsaljkf;lajs f

Preface

sadfkjl;fdjasklf;j as;lkfj as;lkfj kl;asjf ;lkasj d;lkasjd ;lkjf ;lksjd fl;kaj l;fkj as;lfj asd;lkfj
asl;kfj ;alsjf ;laksjf ;laksjf lk;asj f;lkasj f;lajs f;lakjs ;fljsa f;lksa jf;l;lj l;kfj as;lkfj al;ksj f;lkasj
f;lksaj fd;lasj fl;kjas ;lfj sa;lf j
jsadkl;fj asl;dfj a;lksdjf kl;sajd fl;asjd

Caleb Schultz Kisby

Neural Network Semantics

sdfafsfl;asjdfl;lkjf l;sday fl;asjf kl;asjfd ;lasjf ;lksajf ;klsajf l;ksaj f;lksajf;klasj f;klsaj f;klasj
flk;asj fkl;saj fl;ksaj f;lkjs a;flkj sa;lkfj s;klafj ;slakjf ;lasfj

Contents

Introduction	8
Background: Defeasible Reasoning in Artificial Intelligence	14
1 Defeasible Reasoning in Conditional Logic	14
2 Defeasible Reasoning in Modal Logic	16
3 Dynamic Epistemic Logic and Belief Revision	20
4 Defeasible Reasoning in Neural Networks	20
Neural Network Semantics	22
1 Introduction	22
2 Neural Network Models	22
3 Neural Network Semantics	25
4 Dynamic Update in Neural Network Semantics	28
5 Hebbian Learning: A Simple Neural Network Update Policy	29
Soundness: Neural Network Verification	34
1 Introduction	34
2 Properties of Clos, Reach, and Reach^\downarrow	34
3 Soundness for the Base Semantics	37
4 Properties of Hebb and Hebb*	38
5 Soundness for the Logic of Hebbian Learning	42
6 Reflections on Verification and Extraction	42
Completeness: Neural Network Model Building	43
1 Introduction	43
2 Completeness for the Base Semantics	43
3 Reduction Axioms for Iterated Hebbian Update	46
4 Completeness for Iterated Hebbian Update	47
5 Reflections on Interpretability and Alignment	51

Expressivity: Measuring the Modeling Power of Neural Networks	52
1 Introduction	52
2 A Potpourri of Model Classes	52
3 Measuring Expressive Power through Translation	54
4 Expressive Power of the Base Neural Network Semantics	55
5 Expressive Power of Neural Network Update	57
6 Neural Networks and Descriptive Complexity	59
7 Reflections on the Complexity Hierarchy	60
Conclusions	63
Results	63
Open Questions	63
Appendix A Details for the Logic of [best]	63
A.1 Syntax and Semantics	63
A.2 Proof of Soundness	64
A.3 Model Building and Completeness	66
A.4 Building a Finite Model	70
A.5 Dynamic Updates on the Logic of [best]	70
References	71

Chapter 1

Introduction

[In this chapter, I introduce neuro-symbolic AI at a very high level, and build up to the thesis statement (I should state the thesis statement explicitly!)]

References List:

History of neural network semantics. [39] [7] [34] [35] [16] [36] [30] [31] [24] [?] [25]

Social network semantics. [8]

Dynamic logics for learning. [11] [9] [10]

Dynamic epistemic logic; Belief revision. [61] [59] [63] [12] [13] [49] [32] [62] [60] [14]

General neuro-symbolic AI. [5] [53] [15] [27] [19] [22] [6] [23] [38] [18]

Neural network verification. [3]

AI/neural network Alignment.

Neural networks as automata. [41] [57] [43] [42]

Neural network descriptive complexity. [29] [37] [20]

Neural networks & Category theory.

General conditional logic. [33]

General modal logic. [44] [48]

Classic papers in AI. [28] [56] [26] [47] [54] [40] [52] [64] [58] [55]

General TSC/mathematics. [2] [50]

General cognitive science. [46]

Systems and frameworks. [45] [21] [1]

[Incorporate the following text into this chapter, and also the abstract, and so on.]

In the last 15 years, modern artificial intelligence (AI) systems have shown unprecedented success at learning from data with little human guidance. Consider for example large language models such as Llama and GPT [1; 21; 64], which have taken the world by storm with their ability to learn to converse in English merely from unstructured text data they scrape off the web. Or consider AlphaGo [55], which learned to play Go at a human expert level by repeatedly playing against itself. These breakthroughs in machine learning are in large part thanks to the widespread use of neural networks – brain-inspired computational models that are flexible and excel at learning from unstructured data.

But the danger of neural networks is that they come with no safety, reliability, or correctness guarantees. If you play with systems like GPT long enough, you eventually realize that they carry all sorts of misconceptions, make silly logical mistakes, and are quite happy to spew out disinformation [58]. Neural networks also lack transparency, which means diagnosing and correcting these errors is not feasible. (Imagine trying to determine which neurons and connections are responsible for believing that a sailfish is a mammal!) In practice, a computational learner is often a ‘black-box’ whose correct inferences, mistakes, and biases lack interpretation and explanation.

How can we better understand and control this seemingly black-box behavior of neural networks? The answer lies in symbolic (logic) systems, which were commonly used to model reasoning and intelligent behavior prior to the rapid growth of neural network systems. In contrast with neural networks, symbolic systems provide explicit rules for their reasoning in a human-interpretable language. However, this purely logic-based approach was largely abandoned due to logic's inability at the time to model flexible learning or update (known as the *frame problem* in AI [40; 54]).

There is still hope that we might be able to integrate neural networks and symbolic systems while retaining the advantages of both. The field of *neuro-symbolic AI* has emerged in response to this possibility [5; 15; 53]. As a result of this effort, there are now many different proposals for neuro-symbolic systems, including Logic Tensor Networks [6], Distributed Alignment Search [23], DeepProbLog [38], Logic Explained Networks [18], and neural network fibring [22]. But

these systems form a scattered picture; some unifying perspective or theory is needed. In the preface to a recent neuro-symbolic survey book [15], Frank van Harmelen writes:

What are the possible interactions between knowledge and learning? Can reasoning be used as a symbolic prior for learning ... Can symbolic constraints be enforced on data-driven systems to make them safer? Or less biased? Or can, vice versa, learning be used to yield symbolic knowledge? ... **neuro-symbolic systems currently lack a theory that even begins to ask these questions, let alone answer them.**

In my thesis, I will offer a new unifying perspective that sheds light on these questions. The basis for many neuro-symbolic systems is that they encode logical information into neural networks, or conversely, encode neural networks as models in logic [?]. Given these translations, certain neural networks and logic models are able to represent the same information. This suggests that we can think of neural networks in the same way as a logician would think about a model.

The plan is to take this idea as far as it will go: I will develop logics with these *neural network semantics*, whose formulas are interpreted in terms of binary neural networks. First, I will consider *static* conditional and modal logics whose operators are given by the closure (or forward propagation) of signals in the neural network. This closure operator allows these logics to express neural network *inference*, i.e. the input-output behavior of the net. Next, I will give a *dynamic* logic (inspired by Dynamic Epistemic Logic [60; 62; 63]) with an operator for Hebbian learning [28], a simple neural network update policy. Along the way, I will show how foundational questions about neural networks become natural and answerable questions in logic. I'll focus on three questions that are natural to ask about for any logical system: *soundness*, *completeness*, and *expressivity*.

Soundness. What axioms are sound for the semantics? In neural network semantics, this question becomes: What properties can we formally verify for neural network inference? In the dynamic logic setting: What properties can we formally verify for neural network *learning policies*?

Completeness. What are the complete axioms for the semantics? This is equivalent to the question of whether we can build a model that obeys a set of logical constraints Γ . In neural network semantics, completeness asks whether we can build a *neural network* that obeys Γ . And for dynamic logic, this asks whether we can build a neural network that obeys Γ before and after learning takes place. This is a key to the AI Alignment problem, which requires building neural networks with these kinds of guarantees.

Expressivity. What formulas can the semantics model or define? How does the expressive power of two different semantics compare? For neural networks, expressivity is a proxy for what kinds of functions neural networks are capable of representing. Additionally, it provides a metric for comparing the power of neural networks against traditional models in logic. In the dynamic setting: What kinds of *learning policies* are neural networks able to support?

In summary, I will defend the following thesis statement:

Thesis Statement. Neural networks can be treated as a class of models in formal logic (*neural network semantics*). In the process of developing these semantics, foundational questions about neural network inference and learning become natural and answerable questions in logic. In particular:

Soundness	answers	“How can we formally verify that a class of neural networks and its learning policies obey certain properties?”
Completeness	answers	“How can we build a neural network that aligns with constraints, even as the net learns and changes over time?”
Expressivity	answers	“What kinds of functions and learning policies are neural networks capable of representing?”

Related Work. My thesis work builds on existing logics that use neural network semantics, and shares similarities with logics for modeling social networks. Additionally, my approach to modeling learning takes inspiration from work on learning in Dynamic Epistemic Logic (DEL). Here I'll take a moment to situate my thesis in this broader context and clarify what my contribution is in each case.

Logics with Neural Network Semantics. The core idea behind neural network semantics—that neural networks can be treated as models for logic—actually dates back to the very first paper

on neural networks. In McCulloch and Pitts [39], logical formulas are mapped directly to individual neurons in the net. This approach suffers from the well-known “grandmother cell” problem [26]: it is cognitively implausible that an individual neuron could represent a complex concept such as “grandmother”. Instead, concepts in brain networks are distributed across multiple neurons at once.

Neural network semantics is based on a recent reimagining of this approach [7; 36], where logical formulas are mapped to sets of neurons rather than to individual neurons. Early work established the correspondence between inference in a neural network and nonmonotonic conditionals [7; 16; 34; 35]. More recently, [24; 25] proved soundness for inference over fuzzy neural networks. In my thesis work so far [30; 31], I applied ideas from Dynamic Epistemic Logic to model a simple update policy, Hebbian learning, in neural network semantics. The key results of this work are the first ever soundness and completeness theorems for any learning policy on neural networks.

Logics with Social Network Semantics. It has recently come to my attention that a similar approach is being used to model group behavior in social networks. In these social network logics [4; 8; 17], nodes in the graph represent individual agents, and each formula is mapped to the set of agents that adopt a certain social attitude. Agents influence each other, and the spread of their attitudes is modeled much in the same way as forward propagation of a signal in a neural network.

This work shares essentially the same premise and techniques as neural network semantics; I personally view this as a case of parallel discovery. But the two approaches still differ in interesting ways. First, in some sense the two semantics are operating on different “levels”: social networks model interactions between multiple agents, whereas neural networks model interactions between components of the same (single) agent. Second, the two differ in subject matter. Social network semantics focuses on different social links between agents, and how these links change [4]. Neural network semantics, my own work included, instead focuses on inferences and updates inspired by artificial and natural neural networks.

Other Dynamic Logics for Learning. My approach to modeling learning in neural networks takes inspiration from Dynamic Epistemic Logic (DEL) [60; 63]. Perhaps the closest logics

to mine are the logics for plausibility upgrade, in particular conditionalization (Cond), lexicographic (Lex), and conservative (Consr) upgrade [59; 62]. In the Expressivity portion of my dissertation, I will explore whether Hebbian update Hebb* can be simulated by plausibility upgrade, and vice-versa whether Cond, Lex, Consr, and vice-versa whether these plausibility upgrades can be simula.

In my thesis, I mainly focus on the effects of single-step updates. But recent literature on learning in DEL goes beyond this by considering iterated update and convergence to the truth (“learning in the limit”) [9; 10; 11; 14]. The key questions here are: How can we compare the learning power of different iterated update policies? How can we axiomatize important properties of learning? These questions are answered in terms of updates on more classical graphs and plausibility structures. Although in this thesis I don't consider iterated update, I do lay down the groundwork to import neural network learning into this setting.

Chapter 2

Background: Defeasible Reasoning in Artificial Intelligence

The connection between neural networks and formal logic begins with defeasible reasoning (aka nonmonotonic reasoning, or reasoning by default). In standard treatments of logic, the facts you infer are non-revocable, i.e., they cannot be withdrawn in light of new information. But we live in a world of change, partial information, and exceptions—in order to effectively reason, an agent must jump to conclusions about what is “normally” or “plausibly” the case, and be ready to withdraw these inferences. For these reasons, defeasible reasoning is a central to a theoretical understanding of artificially intelligent agents.

Here's a classic example: If you know Tweety is a bird, you should conclude (assuming we're in a “normal” situation) that Tweety flies. But if you then discover that Tweety is a penguin, you must retract that conclusion. The standard material implication fails to model this: If $\text{Tweety} \rightarrow \text{penguin}$, $\text{penguin} \rightarrow \text{bird}$, and $\text{bird} \rightarrow \text{flies}$ we must conclude that Tweety flies.

In this chapter, I will give a tour of many different ways to model defeasible reasoning in formal logic. I will focus on the “preferential” or “plausibility” approach to defeasible reasoning, which branches from the classic papers [33] and [cite Shoham 1988]. First, I will present the standard plausibility semantics for conditional logics (where $\varphi \Rightarrow \psi$ expresses “typically, φ are ψ ”). Then I will discuss many different ways to transfer these semantics to more expressive modal logics. I will present the logic of $[\text{best}]\varphi$ (“the current state is the best one where φ holds”), which forms the backbone of my work connecting neural networks and logic. Finally, I will introduce neural networks, and discuss how they may be seen as models of defeasible reasoning as well. This will set us up for the central plot of my thesis: Developing a neural network semantics for the logic of $[\text{best}]\varphi$.

1 Defeasible Reasoning in Conditional Logic

I will now present the standard way to model nonmonotonic inference in conditional logic, in the KLM tradition [33]. The language is stratified—sentences are conditionals $\varphi \Rightarrow \psi$, where

$\varphi, \psi \in \mathcal{L}_{\text{prop}}$ are propositional formulas connected by $\neg, \wedge, \rightarrow$ in the usual way. Sentences $\varphi \Rightarrow \psi$ cannot be nested within each other, nor within propositional formulas. This odd feature is due to the original conception in [33] that $\varphi \Rightarrow \psi$ specify inference rules, but are not themselves propositions. The intended meaning of $\varphi \Rightarrow \psi$ is “typically (normally), φ are ψ ”, e.g., $\text{bird} \Rightarrow \text{flies}$ reads “typically, birds fly.”

Kraus, Lehmann, and Magidor use the following models to interpret these conditional sentences. I will be moving on pretty quickly to modal logic syntax and semantics, so I won't dwell on these models too long. Let W be an underlying set of worlds (propositional valuations) for $\mathcal{L}_{\text{prop}}$ (not necessarily the set of all worlds for $\mathcal{L}_{\text{prop}}$).

Definition 1.1. A cumulative-ordered model is $\mathcal{M} = \langle \mathcal{S}, l, < \rangle$, where

- \mathcal{S} is a nonempty set of states
- $l: \mathcal{S} \rightarrow \mathcal{P}(W) - \{\emptyset\}$ (a *labelling* of states)
- $<: \mathcal{S} \times \mathcal{S}$ (the *plausibility order*, or *preference relation*)

The plausibility order $<$ is required to be a strict order relation (irreflexive and transitive). $S_1 < S_2$ intuitively means that the agent considers the state $S_1 \in \mathcal{S}$ to be more plausible, or more normal, than $S_2 \in \mathcal{S}$. In order to reason about the most plausible (normal) states, we can look at the $<$ -minimal states. Formally, each cumulative-ordered model determines a function $\text{best}_{<}: \mathcal{S} \rightarrow \mathcal{S}$

$$\text{best}_{<}(S) = \{w \in l(S) \mid \text{For all } u \in l(S), \neg u < w\}$$

We additionally impose the “Smoothness Condition” [33] on $\text{best}_{<}$. This condition says that there are no infinitely descending $<$ -chains, i.e., every nonempty state S has at least one minimal element.

Postulate 1.2. For all cumulative-ordered models \mathcal{M} , states $S \in \mathcal{S}$, and all $w \in W$, if $w \in l(S)$ then either $w \in \text{best}_{<}(S)$, or there is some $v < w$ better than w that is the best, i.e. $v \in \text{best}_{<}(S)$.

Now I can give the KLM interpretation of conditional sentences. For propositional formulas $\varphi \in \mathcal{L}_{\text{prop}}$, $\llbracket \varphi \rrbracket = \{S \in \mathcal{S} \mid w \models \varphi \text{ for all } w \in l(S)\}$, i.e., the set of states where φ is true everywhere. As for conditionals,

$$\mathcal{M} \models \varphi \Rightarrow \psi \text{ iff } \text{best}_{<}(\llbracket \varphi \rrbracket) \subseteq \llbracket \psi \rrbracket$$

That is, in the most plausible (normal) states where φ holds, ψ holds, which was our intended reading. There is a lot more to say about conditional logics like these (expressivity, proof systems, soundness and completeness, their rich history), but I must move on. I will conclude with an example demonstrating that these semantics do in fact resolve our earlier issue with Tweety the penguin.

Example 1.3. [Give an example of these semantics successfully modeling defeasible reasoning. Maybe the Tweety example?]

2 Defeasible Reasoning in Modal Logic

The inability to nest conditionals $\varphi \Rightarrow \psi$ makes conditional logics somewhat flat and inexpressive. Additionally, $\varphi \Rightarrow \psi$ only allows us to refer to the plausibility of the premise φ , and not the antecedent ψ . For example, the following sentences are not expressible in the conditional language above:

- If birds typically fly, then Tweety does.
- The car normally drives, but the check engine light is always on.
- This wasn't done by your typical criminal.
- If this isn't normal, I don't know what is.

We can overcome this by transferring the main ideas of the semantics to a more expressive language—in particular, to modal logics. [is there anything more I need to say here to motivate the reader?]

2.1 A Brief Crash Course in Modal Logic

Let's briefly introduce the basics of modal logic. [cite a standard modal logic text or two!]
A modal logic extends propositional logic with “modal formulas” $\Box\varphi$ and $\Diamond\varphi$ ($\Box\varphi$ is read “it is necessary that φ ”; $\Diamond\varphi$ is read “it is possible that φ .” Standard (normal) modal logics are interpreted using a relational (Kripke) model, which is just an ordinary graph equipped with a valuation of propositions.

Definition 2.1. A relational model is $\mathcal{M} = \langle W, R, V \rangle$, where W is a set of nodes (*worlds*, aka

states), $R: W \times W$ an edge relation (the *accessibility relation*), and $V: \text{propositions} \rightarrow \mathcal{P}(W)$ (the *valuation function*).

Definition 2.2. Let **Rel** be the class of all relational models, and let **Rel**_{s4} be the class of all whose accessibility relation R is reflexive and transitive.

Unlike conditional logic, in modal logics we evaluate a formula *locally*. That is, instead of φ being true or false, we consider the set of worlds where φ is true. We write $\mathcal{M}, w \models \varphi$ to indicate that φ holds at world w . The semantics of propositions and boolean connectives \neg, \wedge are what you might expect. $\diamond\varphi$ is defined as $\neg\Box\neg\varphi$. The key case is for $\Box\varphi$:

$$\mathcal{M}, w \models \Box\varphi \text{ iff for all } u \text{ such that } wRu, \text{ we have } \mathcal{M}, u \models \varphi$$

That is, $\Box\varphi$ holds if φ holds everywhere accessible from the current state (φ is *necessarily* true).

The accessibility relation can have many different interpretations depending on what phenomenon we are trying to model. For example, if R indicates which states are *possibly known* (i.e., *epistemically accessible*), then $\Box\varphi$ takes on the reading “ φ is known (by some agent),” written **box** φ . Similarly, \Box can be cast as belief **B**, obligation **O**, provability **P**, etc. There may be one, or many, modal operators in a modal logic. We may also index modalities \Box_i , indicating a modal attitude for each agent i (in a multi-agent setting), or for different relations R_i within the same agent.

2.2 Defeasible Modal Logics

Unfortunately, this usual treatment of modal logics cannot model defeasible reasoning. [cite Chellas or something] This is because all normal modal logics satisfy the axiom

$$\Box(\varphi \rightarrow \psi) \rightarrow \Box\varphi \rightarrow \Box\psi$$

For concreteness, let's read \Box as belief, and suppose $\Box(\text{bird} \rightarrow \text{flies})$, i.e., for all things we could possibly believe we see from the current state, if we see a bird then it flies. Now say we believe we see a bird ($\Box\text{bird}$). Then the axiom says we *necessarily* believe it flies ($\Box\text{flies}$), leaving no room for revoking our initial conclusion.

There is a wide variety of different ways to resolve this, to rework the idea of defeasibility into modal logic. This is not the right time or place for a thorough literature review, but I will

tour a representative sample to give you a sense of what can be done. Certain approaches use the cumulative-ordered models defined for conditional logic above; others use relational models, but interpret the relation R to be a plausibility ordering; others still use both. For my purposes, I will define plausibility models as Kripke models, but with an irreflexive plausibility relation $<$ (there is no distinction between epistemic accessibility and plausibility). [cite plausibility models, the word is used by Baltag & Smets]

Definition 2.3. A plausibility model is $\mathcal{M} = \langle W, <, V \rangle$, where

- W is a set of *worlds* or *states*
- $<: W \rightarrow W$ (the *plausibility order*)
- $V: [\text{todo}]$ (the *propositional valuation*)

As with cumulative-ordered models, I require $<$ to be irreflexive, transitive, antisymmetric. In cases where we want to refer to the reflexive extension of $<$, I write $u \leq v$ to mean $u < v$ or $u = v$. As before, each plausibility model determines a $\text{best}_<$ function, whose definition now simplifies to

$$\text{best}_<(S) = \{w \in S \mid \text{For all } u \in S, \neg u < w\}$$

Similarly, we require the $\text{best}_<$ operator to satisfy a similar Smoothness condition:

Postulate 2.4. For all plausibility models \mathcal{M} , sets S , and all $w \in W$, if $w \in S$ then either $w \in \text{best}_<(S)$, or there is some $v < w$ better than w that *is* the best, i.e. $v \in \text{best}_<(S)$.

Definition 2.5. Let **Plaus** be the class of all such plausibility models.

Here are some of the ways we can transfer defeasibility into a modal logic setting:

Boutilier's Modal Treatment.

Baltag & Smets' Safe Belief.

- There are many ways to rework the idea of typicality in modal logic: conditional belief $B^\phi\psi$, regular belief $B\phi$, typicality \cdot, T , “defeasible modalities”

2.3 The Modal Logic of [best]

- Actually the *most* relevant for my purposes will be the logic of $[\text{best}]/\langle \text{best} \rangle$. I should

introduce this logic *here*.

- State all of the theorems I will need for this logic
- This logic happens to be *new*, and I seem to be the first to work out the details (soundness & completeness). This work comprises a good $\frac{1}{4}$ th of my thesis work, but I will put the proofs in the Appendix since they detract from the story here.
- The details (soundness, model building, completeness, expressivity) for this logic of $[\text{best}]\varphi$ can be found in [Appendix A]. [And as far as I'm aware, my work on this logic of $[\text{best}]\varphi$ is new.]

Definition 2.6. Let $\mathcal{L}_{\text{best}}$ be the language whose formulas are given by

$$\varphi, \psi := p \mid \neg\varphi \mid \varphi \wedge \psi \mid \mathbf{A}\varphi \mid \Box\varphi \mid \Box^\downarrow\varphi \mid [\text{best}]\varphi$$

$\top, \perp, \vee, \rightarrow, \leftrightarrow$ and the dual modal operators $\mathbf{E}, \Diamond, \Diamond^\downarrow, \langle \text{best} \rangle$ are defined in the usual way. [I haven't really said what “the usual way” is yet...]

Example 2.7. [A nice example, after giving intended readings of each of these, would be to translate the sentences given at the beginning of this section! Be careful, \Box doesn't necessarily have an epistemic reading!]

- $\mathbf{A}([\text{best}]\text{bird} \rightarrow \text{flies}) \rightarrow (\text{Tweety} \rightarrow \text{flies})$: If birds typically fly, then Tweety does.
- $:$ The car normally drives, but the check engine light is always on.
- $:$ This wasn't done by your typical criminal.
- $\neg[\text{best}]\top \rightarrow \neg\text{box}(\mathbf{E}[\text{best}]\top)$: If this isn't normal, I don't know what is.

Definition 2.8. The full semantics for $\mathcal{L}_{\text{best}}$ is given as follows. For all $\mathcal{M} \in \mathbf{Plaus}$, $w \in W$,

$\mathcal{M}, w \Vdash p$	iff	$w \in V(p)$
$\mathcal{M}, w \Vdash \neg\varphi$	iff	$\mathcal{M}, w \not\Vdash \varphi$
$\mathcal{M}, w \Vdash \varphi \wedge \psi$	iff	$\mathcal{M}, w \Vdash \varphi$ and $\mathcal{M}, w \Vdash \psi$
$\mathcal{M}, w \Vdash \mathbf{A}\varphi$	iff	For all $u \in W$ whatsoever, $\mathcal{M}, u \Vdash \varphi$
$\mathcal{M}, w \Vdash \Box\varphi$	iff	For all u such that $w \leq u$, $\mathcal{M}, u \Vdash \varphi$
$\mathcal{M}, w \Vdash \Box^\downarrow\varphi$	iff	For all u such that $u \leq w$, $\mathcal{M}, u \Vdash \varphi$
$\mathcal{M}, w \Vdash [\text{best}]\varphi$	iff	$w \in \text{best}_R(\llbracket \varphi \rrbracket_{\mathcal{M}})$

where $\llbracket \varphi \rrbracket_{\mathcal{M}} = \{u \mid \mathcal{M}, u \Vdash \varphi\}$.

The semantics for \Box is the totally standard relational one, but using the reflexive extension of the plausibility order \leq as an accessibility relation. [What does this mean for the interpreta-

Axioms for \Box: (Dual) $\Diamond\varphi \leftrightarrow \neg\Box\neg\varphi$ (Distr) $\Box(\varphi \rightarrow \psi) \rightarrow (\Box\varphi \rightarrow \Box\psi)$ (Refl) $\Box\varphi \rightarrow \varphi$ (Trans) $\Box\varphi \rightarrow \Box\Box\varphi$	Axioms for \Box^\downarrow: (Dual) $\Diamond^\downarrow\varphi \leftrightarrow \neg\Box^\downarrow\neg\varphi$ (Distr) $\Box^\downarrow(\varphi \rightarrow \psi) \rightarrow (\Box^\downarrow\varphi \rightarrow \Box^\downarrow\psi)$ (Back) $\varphi \rightarrow \Box\Diamond^\downarrow\varphi$ (Forth) $\varphi \rightarrow \Box^\downarrow\Diamond\varphi$
Axioms for [best]: (Dual) $\langle\text{best}\rangle\varphi \leftrightarrow \neg[\text{best}]\neg\varphi$ (Refl) $[\text{best}]\varphi \rightarrow \varphi$ (Trans) $[\text{best}]\varphi \rightarrow [\text{best}][\text{best}]\varphi$ (Up) $[\text{best}]\varphi \wedge \psi \rightarrow \Box([\text{best}]\varphi \rightarrow \psi)$ (Down) $[\text{best}]\varphi \wedge \psi \rightarrow \Box^\downarrow(\varphi \rightarrow \psi)$	Axioms for \mathbf{A}: (Dual) $\mathbf{E}\varphi \leftrightarrow \neg\mathbf{A}\neg\varphi$ (Distr) $\mathbf{A}(\varphi \rightarrow \psi) \rightarrow (\mathbf{A}\varphi \rightarrow \mathbf{A}\psi)$ (Refl) $\mathbf{A}\varphi \rightarrow \varphi$ (5) $\mathbf{E}\varphi \rightarrow \mathbf{A}(\mathbf{E}\varphi)$ (Interact) $\mathbf{A}\varphi \rightarrow \Box\varphi$
	Rules of Inference: (MP) From $\vdash\varphi \rightarrow \psi$ and $\vdash\varphi$ we can infer $\vdash\psi$ (Nec) From $\vdash\varphi$ we can infer $\vdash\Box\varphi$ for each $\Box \in \{\mathbf{A}, \Box, \Box^\downarrow\}$

Figure 2.1. Axioms and rules of inference for [todo]

tion of $\Box^{??}$] The semantics for \Box^\downarrow comes from temporal logic— \Box^\downarrow looks in the “past”, whereas \Box looks in the “future.” The semantics for [best] is just the modal version of our semantics for conditionals $\alpha \Rightarrow \beta$. [Define $\vdash_{\text{Plaus}}!!!$]

Definition 2.9. The proof system for the base modal logic over $\mathcal{L}_{\text{best}}$ is given as follows: $\vdash\varphi$ iff either φ is one of the axioms: [Todo—maybe I don't need to say it here, I can just point to the Appendix.]

3 Dynamic Epistemic Logic and Belief Revision

[This is really one of the best upshots of modeling something using modal logic! If formulas are nestable, then we can nest things inside and within update operators.] [Introduce Dynamic Epistemic Logic, and various operators Cond, Lex, Consr for belief revision.]

4 Defeasible Reasoning in Neural Networks

Introduce neural networks, for a broad audience (including logicians that know nothing about them). Explain how inference works in a neural network, and how neural networks can be thought of as performing defeasible reasoning. (Hannes explains it pretty well, maybe I should borrow his example.)

Explain learning in a neural network at a high level, both unsupervised (Hebbian learning is representative of unsupervised learning algorithms, mention the relationship with Principal Components Analysis) and supervised (backpropagation is representative here, and the efficient computation of backpropagation is what has made neural networks so successful).

Inference in a neural network is “like” a conditional inference—but this analogy goes further. Many authors have already studied a formal correspondence between the input-output behavior of a neural network and defeasible conditionals. [cite d'Avila Garces, Hannes, Gior-dano, really all the people here who have made the observation before me. This is a good time to break down the history of how it happened.] [Talk about both “soundness” and “completeness” here]

In the rest of this dissertation, I will extend this analogy by giving a neural network interpretation for the more general logic of [best]. My main point in considering a modal language is the same as before: It buys us expressive power over conditionals, and in particular sets us up to express *neural network update* using Dynamic Epistemic Logic. Towards the end of this work, I will extend these neural network semantics for [best] with a dynamic operator for a simple Hebbian update policy over neural networks.

Chapter 3

Neural Network Semantics

1 Introduction

[I wrote the following short paragraph for my thesis proposal. But now I have more space to say more, slowly, about where neural network semantics comes from, what the underlying idea is. It would be nice to introduce it using an example of a neural network in practice—justify why binary, interpreted neural networks are the “right” thing to look at!]

I will now give an overview of the particular neural network semantics [rephrase]I’ve developed during my PhD. First, I will discuss the simplifying assumptions that make it possible to use neural networks as models, and introduce the *closure* (or forward propagation) of a signal in the net. This closure operator allows us to express the inference, or input-output behavior, of the net. I will give a modal logic whose key operator is given by this closure operator. I will then turn to dynamic update in neural networks and introduce iterated Hebbian learning, one of the simplest learning policies over nets. Finally, I will give a dynamic logic whose formulas express the behavior of a neural network before and after Hebbian update.

2 Neural Network Models

A model of neural network semantics is an artificial neural network (ANN), along with a valuation function which interprets propositions as sets of neurons. I will make a few more simplifying assumptions soon, but this is the basic idea.

Definition 2.1. A neural network model is $\mathcal{N} = \langle N, \text{bias}, E, W, A, \eta, V \rangle$, where

- N is a finite nonempty set (the set of *neurons*)
- bias is a fixed node (the *bias* node)
- Each $E \subseteq N \times N$ (the *edge relation*)
- $W: E \rightarrow \mathbb{Q}$ (the *edge weights*)

- $A: \mathbb{Q} \rightarrow \mathbb{Q}$ (the *activation function*)
- $\eta \in \mathbb{Q}, \eta \geq 0$ (the *learning rate*)
- $V: \text{Proposition} \rightarrow \mathcal{P}(N)$ (the *valuation function*)

In general, a *state* is just a possible activation pattern or configuration of the net. In practice, a neural network's nodes take on fuzzy activation values in $[0, 1]$. But we would like to associate each state with a binary set of neurons—either a neuron is active (1) or it is not (0). To do this, I assume that the activation function A is a (nonzero) binary step function ($A: \mathbb{Q} \rightarrow \{0, 1\}$). [Definition: \mathcal{N} has a threshold, $\exists t \in \mathbb{Q}$ with $A(t) = 1$; \mathcal{N} is nondecreasing. These things amount to A being a binary step function.] It turns out this binary constraint is also a common theoretical assumption in work that analyzes neural networks as automata [41; 43; 65]. In their terminology, we say our nets are *saturated*.

- [Todo, put somewhere in this section, optional property]
- **\mathcal{N} is Fully connected:** $\forall m, n \in N$, either $(m, n) \in E$, $(n, m) \in E$, or m and n have exactly the same predecessors and successors.
- In machine learning practice, “fully connected” means that there is an edge from every node in layer l to every node in the *following* layer $l + 1$. But here we mean something much stronger: the graph is fully connected, including “highway edges” that cut between layers, as shown in [DIAGRAM]. (This intuition comes from work on highway networks [56].) This assumption is crucial for our results about iterated Hebbian learning, and we expect that letting it go will not be easy.

Additionally, I assume there is a special bias node that is active in every state. This is purely for ruling out the particular edge case where *no* node is active. Since bias is active in every state, we can assume that no edges go into bias. Putting all this together, the states of the net are defined as follows.

$$\text{State}_{\mathcal{N}} = \{S \mid S \subseteq N \text{ and bias} \in S\}$$

Usually \mathcal{N} is understood from context, and I'll just write *State* without the subscript.

2.1 The Forward Propagation Operator

We can describe the input-output behavior of neural networks in terms of their state. Say we are given an input state A consisting of input-layer nodes, and a possible classification state B consisting of output-layer nodes. Active neurons in A subsequently activate new neurons, which activate yet more neurons, until eventually the state of the net stabilizes. If this final state includes the output B , we say “the net *classifies* A as B ”.

The state at the fixed point of this process is called the *closure* or *forward propagation* of the signal A through the net, $\text{Clos}(A)$. This closure operator is central to my semantics, since it captures the underlying dynamics involved in neural network inference. Formally, each choice of E, W, A specifies a transition function from state $S \in \text{State}$ to the next state. Given an initial state S_0 , this transition function F_{S_0} is given by

$$F_{S_0}(S) = S_0 \cup \left\{ n \mid A \left(\sum_{m \in \text{preds}(n)} W(m, n) \cdot \chi_S(m) \right) = 1 \right\}$$

where $\chi_S(m) = 1$ iff $m \in S$ is the indicator function. In other words, $F_{S_0}(S)$ is the initial state S_0 , along with the set of nodes that are activated by their predecessors in S . Notice that $F_{S_0}(S)$ is extensive: all nodes in the initial state will stay activated in successive states.

My neural network models have one final constraint: This transition function F_{S_0} eventually gives a unique *fixed point* (or stable state) under the input S_0 , i.e. a unique state S such that $F_{S_0}(S) = S$. This guarantees that the closure $\text{Clos}(S)$ exists.

Postulate 2.2. I assume that for all states S_0 , F applied repeatedly to S_0 , i.e.

$$S_0, F_{S_0}(S_0), F_{S_0}(F_{S_0}(S_0)), \dots, F_{S_0}^k(S_0), \dots$$

has a finite fixed point, and moreover that this state is the *only* fixed point under S_0 —that is, it is the only state S such that $F_{S_0}(S) = S$. Let the closure $\text{Clos}: \text{State} \rightarrow \text{State}$ be the function that produces that least fixed point. For concreteness, we can say that there is some $k \in \mathbb{N}$ for which

$$\text{Clos}(S) = F_{S_0}^k(S)$$

Let **Net** be the class of all binary neural network models that satisfy this postulate. Not every neural network has a fixed point. For example, consider the following recurrent neural network:

[DIAGRAM, walk through example]

The neural nets I consider include feed-forward nets, as well as certain controlled forms of recurrence. Characterizing nets that have a unique least fixed point is a big open problem.

An important feature of Clos is that it is nonmonotonic: it is not the case that for all $A, B \in \text{State}$, if $A \subseteq B$ then $\text{Clos}(A) \subseteq \text{Clos}(B)$. Intuitively, this is because our net's weights can be negative, and so $\text{Clos}(B)$ can inhibit the activation of new neurons that would otherwise be activated by $\text{Clos}(A)$. I will come back to this point more formally in Chapter [todo], when we discuss the basic properties of Clos .

2.2 The Graph Reachability Operators

As I mentioned before, a key feature of Clos is that it is not monotonic. Let's consider two closure operators over neural networks that *are* monotonic—graph reachability Reach , and “reverse” graph reachability Reach^\downarrow . $\text{Reach}(S)$ just returns the set of all neurons graph-reachable from S , i.e. $\text{Reach}: \text{State} \rightarrow \text{State}$ is given by $n \in \text{Reach}(S)$ iff there exists $m \in S$ with an E -path from m to n . Similarly, $\text{Reach}^\downarrow(S)$ returns the set of all neurons that graph-reach some node in S , i.e. $\text{Reach}^\downarrow: \text{State} \rightarrow \text{State}$ is given by $n \in \text{Reach}^\downarrow(S)$ iff there exists $m \in S$ with an E -path from n to m .

Reach and Reach^\downarrow are not very interesting operators in their own right, but I consider them in this discussion for two reasons. First, graph reachability is necessary for reasoning about Hebbian learning (see Chapter ?). Second, in Chapter ? I would like to compare the expressive power of neural networks against many classes of models, including relational (graph) models.

3 Neural Network Semantics

[Introduce this all slowly. I will now explain how we can use neural networks as models to interpret formulas in logic. First, I will give Hannes' semantics for conditional logic. Then I will introduce my own semantics for modal logic based on his.]

3.1 Using Conditional Logic

Definition 3.1. Formulas in our conditional language \mathcal{L}_\Rightarrow are given by [todo]

Definition 3.2. The semantics for \mathcal{L}_\Rightarrow is given as follows. [todo]

Definition 3.3. We write $\models \alpha \Rightarrow \beta$ to mean all nets $\mathcal{N} \models \alpha \Rightarrow \beta$, and $\Gamma \models \alpha \Rightarrow \beta$ to mean every model \mathcal{N} of Γ , i.e. $\mathcal{N} \models \gamma \Rightarrow \delta$ for all $\gamma \Rightarrow \delta \in \Gamma$ also models $\alpha \Rightarrow \beta$.

3.2 Using Modal Logic

I can now state the specific logic and neural network semantics that I will consider. Let p, q, \dots be finitely many atomic propositions. These represent fixed states, corresponding to features in the external world that we know ahead of time. Usually these are input and output states, although they could be intermediate “hidden” states if we know these features ahead of time. For example, p might be the set of neurons that represent the color pink. For more complex formulas,

Definition 3.4. Formulas in the base modal language $\mathcal{L}_{\text{best}}$ are given by

$$\varphi, \psi := p \mid \neg \varphi \mid \varphi \wedge \psi \mid \diamond \varphi \mid \diamond^\perp \varphi \mid \langle \text{best} \rangle \varphi$$

$\top, \perp, \vee, \rightarrow, \leftrightarrow$ and the dual modal operators $\text{box}, \square^\perp, [\text{best}]$ are defined in the usual way.

The intended readings for these operators are as follows (\square^\perp is conceptually tricky, I will leave it out of this discussion for now). $\text{box} \varphi$ reads “the agent knows φ ”, and $[\text{best}] \varphi$ reads “bestopically φ ”. It is not immediately clear how these readings are justified; in my dissertation, I will justify these readings by connecting the neural network semantics I give here to more traditional semantics for $\text{box}, \square^\perp$, and $[\text{best}]$.

At last, here are the semantics for $\mathcal{L}_{\text{best}}$. For all $\mathcal{N} \in \mathbf{Net}$, $n \in N$:

$$\begin{array}{lll} \mathcal{N}, n \models p & \text{iff} & n \in V(p) \\ \mathcal{N}, n \models \neg \varphi & \text{iff} & \mathcal{N}, n \not\models \varphi \\ \mathcal{N}, n \models \varphi \wedge \psi & \text{iff} & \mathcal{N}, n \models \varphi \text{ and } \mathcal{N}, n \models \psi \\ \mathcal{N}, n \models \diamond \varphi & \text{iff} & n \in \text{Reach}^\perp(\llbracket \varphi \rrbracket) \\ \mathcal{N}, n \models \diamond^\perp \varphi & \text{iff} & n \in \text{Reach}(\llbracket \varphi \rrbracket) \\ \mathcal{N}, n \models \langle \text{best} \rangle \varphi & \text{iff} & n \in \text{Clos}(\llbracket \varphi \rrbracket) \end{array}$$

where $\llbracket \varphi \rrbracket = \{n \mid \mathcal{N}, n \models \varphi\}$.

[Note that the syntax is backwards/dualed from the syntax for neural network semantics.

But because they're duals of each other, it doesn't matter. (It's a known modal logic trick that

you can do induction on either version/form.)]

Definition 3.5. We write $\mathcal{N} \models \varphi$ (“the net *models* φ ”) to mean $\mathcal{N}, n \Vdash \varphi$ for all $n \in N$; $\models \varphi$ to mean all nets $\mathcal{N} \models \varphi$; and finally, $\Gamma \models \varphi$ to mean every model \mathcal{M} of Γ , i.e. $\mathcal{M} \models \psi$ for all $\psi \in \Gamma$ also models φ .

[talk about $\mathcal{N} \models \varphi$, which in this context means “ φ activates *all* of the neurons in \mathcal{N} ”, or “ φ holds across the entire net.” Also define $\Gamma \models \varphi$.]

[A reader might be confused about the “swaps” from Reach^\downarrow to \diamond , as well as the choice to use diamond operators instead of the normal box ones.] [How can we justify the readings we had above? (at least intuitively)]

[I now have space to say these points in more detail (there's lots to say!)] Although these semantics are based on Leitgeb's [36], they differ in a few key ways. First, his semantics uses conditionals $\varphi \Rightarrow \psi$ to capture neural network inference, whereas mine instead centers on the modal operator $\langle \text{best} \rangle$. Second, I include these additional operators box and \Box^\downarrow that are not mentioned in his work. Finally, Leitgeb battles with the issue of how to correctly interpret negation; I sidestep this issue by using neural networks for interpreting $\langle \text{best} \rangle \varphi$ (where the “action” happens), but not for \neg and \wedge . The bottom line is this: proving completeness for this logic is not necessarily just a matter of importing the proof from [36].

3.3 Why Consider this *Modal* Logic?

[The conditional logic came first, and I could have just re-used it in order to save myself a lot of work. But I think it's important to explain my aesthetic choices in moving to the modal logic, rather than just using Hannes' conditional logic. But this is such a technical detail that I should at least advise the reader to skip this section if they're not interested.]

3.4 What Makes these Semantics “Neural”

What makes these semantics “neural” or “connectionistic”? Easy answer: the key operators for inference are implemented in a neural network. Better answer: no single neuron/node holds the information for $\langle \text{best} \rangle \varphi$ How is this different from relational or neighborhood semantics? Is

this a meaningful difference? What does Hannes have to say about it in his dissertation?

Tbh maybe it just means “loosely inspired by real neuron and synapse behavior”, but even then there are probably properties we can write down and check.

4 Dynamic Update in Neural Network Semantics

[I now have space to express these points more slowly, in detail.] The neural network semantics presented so far shows us how we can use neural networks as models for modal logic. Neural network inference can be expressed in this logic using $\langle \text{best} \rangle \varphi$, which denotes the forward propagation of the signal $\llbracket \varphi \rrbracket$ through the net. However, as discussed in the introduction, the mystery about neural networks is how their inference interacts with their *learning*. In this section, I will show how to extend these semantics to model learning and update in a neural net.

As previously mentioned, I formalize neural network update using the methodology of Dynamic Epistemic Logic. Our static operators \diamond , \diamond^\downarrow , and $\langle \text{best} \rangle$ are interpreted by examining the state of the neural net. The DEL trick is to introduce a new “dynamic” operator $[P]$ which *changes* the net in response to some observed formula P . First, we extend the language $\mathcal{L}_{\text{best}}$ to $\mathcal{L}_{\text{best}}^{\text{update}}$, which includes these dynamic operators:

$$\varphi, \psi := p \mid \neg \varphi \mid \varphi \wedge \psi \mid \diamond \varphi \mid \diamond^\downarrow \varphi \mid \langle \text{best} \rangle \varphi \mid [P] \varphi$$

Here, $[P] \varphi$ reads “after the agent observes P , φ is true”.

Let $\text{Update}: \mathbf{Net} \times \text{State} \rightarrow \mathbf{Net}$ be any function which takes a neural network, some state S , and “updates” the net somehow in response to S . We can interpret $[P]$ as performing this update by adding the following line to the semantics:

$$\mathcal{N}, n \Vdash [P] \varphi \quad \text{iff} \quad \text{Update}(\mathcal{N}, \llbracket P \rrbracket), n \Vdash \varphi$$

In other words, in order to evaluate $[P] \varphi$, we simply evaluate φ in the updated net $\text{Update}(\mathcal{N}, \llbracket P \rrbracket)$.

From a DEL perspective, this is a standard move to make. But from a machine learning perspective, there are a couple caveats that I should mention. First, $[P]$ does not model learning in the sense of “iterated update until convergence”, but rather only models a single step of update. Second, we should think of $[P]$ as modeling *unsupervised learning*—the model updates in

response to an input P , but no “expected answer” y is given alongside P . It is an open problem to formalize supervised learning (in this machine learning sense) in DEL in a non-trivial way.

5 Hebbian Learning: A Simple Neural Network Update Policy

[I now have space to express these points more slowly, in detail.] So far, I've discussed learning and update in very general terms. For my thesis, I will model a simple update policy over neural networks: Hebbian learning. The point in starting with Hebbian learning is to get the details right on a simpler example before lifting these ideas to, say, gradient descent through backpropagation [51].

Hebb's classic learning rule [28] states that when two adjacent neurons are simultaneously and persistently active, the connection between them strengthens (“neurons that fire together wire together”). In contrast with backpropagation, Hebbian learning is errorless and unsupervised. Another key feature is that Hebbian update is local — the change in a weight $\Delta W(m, n)$ depends only on the activation of the immediately adjacent neurons. For this reason, the Hebbian family of learning policies is often considered more biologically plausible than backpropagation.

There are many variations of Hebbian learning, but I will only consider the most basic form of Hebb's rule: $\Delta W(m, n) = \eta x_m x_n$, where η is the learning rate and x_m, x_n are the outputs of adjacent neurons m and n . This is the *unstable* variation of Hebb's rule; repeatedly applying the rule will make the weights arbitrarily large. I will not consider stabilizing variants such as Oja's rule [47].

Single-Step Hebbian Update. First, consider what happens in a single step of Hebbian update. Given a net \mathcal{N} and a state S , we first propagate S forward through \mathcal{N} . Any edges that are involved in this propagated activation pattern $\text{Clos}(S)$ simply have their weights strengthened. Formally,

Definition 5.1. Let $\text{Hebb}: \mathbf{Net} \times \text{State} \rightarrow \mathbf{Net}$ be given by

$$\text{Hebb}(\langle N, \text{bias}, E, W, A, \eta, V \rangle, S) = \langle N, \text{bias}, E, W', A, \eta, V \rangle$$

where $W'(m, n) = W(m, n) + \eta \cdot \chi_{\text{Clos}(S)}(m) \cdot \chi_{\text{Clos}(S)}(n)$.

Note that Hebb does not affect the edges, activation function, or evaluation of propositions. This means the resulting net is still binary, and closures $\text{Clos}(S)$ still exist and are unique. Therefore Hebb is well-defined. This also means that Hebb does not affect the Reach or Reach^\downarrow operators.

Proposition 5.2. $\text{Reach}_{\text{Hebb}(\mathcal{N}, A)}(B) = \text{Reach}_{\mathcal{N}}(B)$

Proof. A single step of Hebbian update $\text{Hebb}(\mathcal{N}, A)$ doesn't change the edge relation E of the graph. So if $n \in N$, any path from $m \in B$ to n in $\text{Hebb}(\mathcal{N}, A)$ is the same path in \mathcal{N} . \square

And similarly:

Proposition 5.3. $\text{Reach}_{\text{Hebb}(\mathcal{N}, A)}^\downarrow(B) = \text{Reach}_{\mathcal{N}}^\downarrow(B)$

The following is easy to see [I now have space to explain] (since $\eta \geq 0$).

Proposition 5.4. Let $m, n \in N$. We have:

- $W_{\mathcal{N}}(m, n) \leq W_{\text{Hebb}(\mathcal{N}, S)}(m, n)$
- If either $m \notin \text{Clos}(S)$ or $n \notin \text{Clos}(S)$, then $W_{\text{Hebb}(\mathcal{N}, S)}(m, n) = W_{\mathcal{N}}(m, n)$.

Proof. For the first part, observe:

$$\begin{aligned} W_{\mathcal{N}}(m, n) &\leq W_{\mathcal{N}}(m, n) + \eta && (\text{since } \eta \geq 0) \\ &\leq W_{\mathcal{N}}(m, n) + \eta \cdot \chi_{\text{Clos}(S)}(m) \cdot \chi_{\text{Clos}(S)}(n) && (\text{since for all } S, n, \chi_S(n) \geq 0) \\ &= W_{\text{Hebb}(\mathcal{N}, S)}(m, n) \end{aligned}$$

As for the second part, if either $m \notin \text{Clos}(S)$ or $n \notin \text{Clos}(S)$, then by definition of Hebb,

$$\begin{aligned} W_{\text{Hebb}(\mathcal{N}, S)}(m, n) &= W_{\mathcal{N}}(m, n) + \eta \cdot \chi_{\text{Clos}(S)}(m) \cdot \chi_{\text{Clos}(S)}(n) \\ &= W_{\mathcal{N}}(m, n) + \eta \cdot 0 \\ &= W_{\mathcal{N}}(m, n) + \eta \\ &= W_{\mathcal{N}}(m, n) \end{aligned}$$

\square

Iterated Hebbian Update. In addition to the single-step Hebb operator, in my thesis work I have also modelled *iterated* Hebbian update Hebb^* . The idea is this: what happens when we propagate a signal S through the net, and then *repeatedly* strengthen the weights of the edges that are involved? Recall that our single-step Hebb is unstable; if we repeat Hebb on a single input state S , the net's weights within $\text{Clos}(S)$ will be so high that *any* activation pattern that

makes contact with $\text{Clos}(S)$ will “rip through” it entirely. Repeating Hebb on S further will not change the $\text{Clos}(S)$ -structure, i.e., the update has reached a fixed point. Hebb* returns the net at this fixed point.

Instead of reasoning abstractly about this fixed point, I formalize it by explicitly defining the number of iterations iter needed to reach it. The idea is to set iter to be so high, all updated weights $W'(m, n)$ overpower any negative weights that would otherwise cancel their effect. The following definitions might look like black magic, but they are set up to capture this intuition (I verified in Lean that this is the right choice for iter , see [31]).

Definition 5.5. Let \mathcal{N} be a net, $n \in N$, and let m_1, \dots, m_k list the predecessors of n . The *negative weight score* of n is the sum of all the negative weights of n 's predecessors, i.e.,

$$\text{nws}(n) = \sum_{m \in \text{preds}(n)} \begin{cases} W(m, n) & \text{if } W(m, n) < 0 \\ 0 & \text{otherwise} \end{cases}$$

Definition 5.6. The *minimum negative weight score* is simply

$$\text{mnws} = \min_{n \in N} \text{nws}(n)$$

Proposition 5.7. For all $S \in \text{State}$, $m, n \in N$, we have $\text{mnws} \leq W(m, n) \cdot \chi_S(m)$.

Proof. Let m, n be any nodes in N . We have:

$$\begin{aligned} \text{mnws} &\leq \text{nws}(n) \\ &= \sum_{m \in \text{preds}(n)} \begin{cases} W(m, n) & \text{if } W(m, n) < 0 \\ 0 & \text{otherwise} \end{cases} && \text{(by definition)} \\ &= \sum_{m \in \text{preds}(n)} \begin{cases} W(m, n) \cdot \chi_S(m) & \text{if } W(m, n) < 0 \\ 0 & \text{otherwise} \end{cases} && \begin{aligned} &\text{(since each } W(m, n) < 0 \\ &\text{and } \chi_S(m) \in \{0, 1\}) \end{aligned} \\ &\leq W(m, n) \cdot \chi_S(m) && \begin{aligned} &\text{(the sum of negative terms is } \leq \\ &\text{any particular term)} \end{aligned} \end{aligned}$$

□

Definition 5.8. Recall that the activation function A is nonzero, i.e. there is some $t \in \mathbb{Q}$ such that $A(t) = 1$. We set the number of iterations iter to be exactly

$$\text{iter} = \begin{cases} \left\lceil \frac{t - |N| \cdot \text{mnws}}{\eta} \right\rceil & \text{if } \geq 1 \\ 1 & \text{otherwise} \end{cases}$$

Note that iter will always be a positive integer, and so iterating iter times is well-defined.

This choice for iter may seem opaque, but we will see in Lemma [which] why it guarantees that the updated weights overpower competing edge weights.

Definition 5.9. Let $\text{Hebb}^*: \mathbf{Net} \times \text{State} \rightarrow \mathbf{Net}$ be given by

$$\text{Hebb}^*(\langle N, \text{bias}, E, W, A, \eta, V \rangle, S) = \langle N, \text{bias}, E, W', A, \eta, V \rangle$$

where $W'(m, n) = W(m, n) + \text{iter} \cdot \eta \cdot \chi_{\text{Clos}(S)}(m) \cdot \chi_{\text{Clos}(S)}(n)$.

As with Hebb, Hebb^* does not affect the edges, activation function, or evaluation of propositions. Therefore Hebb^* is well-defined. This also means that Hebb^* does not affect the Reach or Reach^\downarrow operators.

Proposition 5.10. $\text{Reach}_{\text{Hebb}^*(\mathcal{N}, A)}(B) = \text{Reach}_{\mathcal{N}}(B)$

Proposition 5.11. $\text{Reach}_{\text{Hebb}^*(\mathcal{N}, A)}^\downarrow(B) = \text{Reach}_{\mathcal{N}}^\downarrow(B)$

Similar to Proposition [todo], we have the following:

Proposition 5.12. Let $m, n \in N$. We have:

- $W_{\mathcal{N}}(m, n) \leq W_{\text{Hebb}^*(\mathcal{N}, S)}(m, n)$
- If either $m \notin \text{Clos}(S)$ or $n \notin \text{Clos}(S)$, then $W_{\text{Hebb}^*(\mathcal{N}, S)}(m, n) = W_{\mathcal{N}}(m, n)$

Proof. \square

\square

The following fact about Hebb^* is the most important. It is a formal expression of our statement before: Updated weights $W_{\text{Hebb}^*(\mathcal{N}, A)}(B)$ are so high that if m is active in Hebb^* then n must be as well.

Lemma 5.13. (\clubsuit) Let $A, B \in \text{State}, m, n \in N$. If $m \in \text{preds}$, $m, n \in \text{Clos}(A)$, and $m \in \text{Clos}(B)$, then

$$A\left(\sum_{m \in \text{preds}(n)} W_{\text{Hebb}^*(\mathcal{N}, A)}(m, n) \cdot \chi_{\text{Clos}(B)}(m)\right) = 1$$

(Take care to notice the different subscripts for W and χ !)

Proof. A is a binary step function, which in particular means it is binary, has a threshold, some $t \in \mathbb{Q}$ with $A(t) = 1$, and is nondecreasing. Since A is nondecreasing, it's enough for us to show

$$t \leq \sum_{m \in \text{preds}(n)} W_{\text{Hebb}^*(\mathcal{N}, A)}(m, n) \cdot \chi_{\text{Clos}(B)}(m)$$

Well, we have

$$\begin{aligned}
& \sum_{m_i \in \text{preds}(n)} W_{\text{Hebb}^*(\mathcal{N}, A)}(m_i, n) \cdot \chi_{\text{Clos}(B)}(m_i) \\
&= \sum_{m_i \in \text{preds}(n), \text{ and } m_i \neq m} W_{\text{Hebb}^*(\mathcal{N}, A)}(m_i, n) \cdot \chi_{\text{Clos}(B)}(m_i) \\
&\quad + W_{\text{Hebb}^*(\mathcal{N}, A)}(m, n) \cdot \chi_{\text{ClosHebb}^*(\mathcal{N}, A)(B)}(m) \\
&\geq (|N| - 1) \cdot \text{mnws} + W_{\text{Hebb}^*(\mathcal{N}, A)}(m, n) \cdot \chi_{\text{Clos}(B)}(m) \\
&\quad \text{(by Proposition [todo], since we are adding } |N| - 1 \text{ terms)} \\
&= (|N| - 1) \cdot \text{mnws} + W_{\text{Hebb}^*(\mathcal{N}, A)}(m, n) \cdot 1 \\
&\quad \text{(since } m \in \text{Clos}(B) \text{)} \\
&= (|N| - 1) \cdot \text{mnws} + W_{\mathcal{N}}(m, n) + \text{iter} \cdot \chi_{\text{Clos}(A)}(m) \cdot \chi_{\text{Clos}(A)}(n) \\
&\quad \text{(by definition of Hebb}^*\text{)} \\
&= (|N| - 1) \cdot \text{mnws} + W_{\mathcal{N}}(m, n) + \text{iter} \\
&\quad \text{(since } m, n \in \text{Clos}(A) \text{)} \\
&\geq (|N| - 1) \cdot \text{mnws} + \text{mnws} + \text{iter} \cdot \eta \cdot \chi_{\text{Clos}(A)}(m) \cdot \chi_{\text{Clos}(A)}(n) \\
&\quad \text{(the sum of negative weights is } \leq \text{ any particular weight)} \\
&= |N| \cdot \text{mnws} + \text{iter} \cdot \eta \\
&\quad \text{(grouping like terms)}
\end{aligned}$$

So at this point we need to show:

$$t \leq |N| \cdot \text{mnws} + \text{iter} \cdot \eta$$

Rearranging this to solve for iter, it suffices to show:

$$\frac{t - |N| \cdot \text{mnws}}{\eta} \leq \text{iter}$$

But we defined iter to be exactly the integer ceiling of this expression on the left (and 1 if the expression on the left is negative)! □

5.1 Neural Network Semantics for Hebbian Update

[TODO] Give official languages $\mathcal{L}_{\text{Hebb}}$ and $\mathcal{L}_{\text{Hebb}^*}$ for the logics of Hebb and Hebb*, i.e. using operators $[P]_{\text{Hebb}}\varphi$ and $[P]_{\text{Hebb}^*}\varphi$ and their semantics!!! And also just say that the definitions $\mathcal{N} \models_{\text{Hebb}} \varphi$ (same for Hebb*), $\Gamma \models_{\text{Hebb}} \varphi$ (same for Hebb*) are what you'd expect.

Chapter 4

Soundness: Neural Network Verification

1 Introduction

dsafkljsdf;lkja sfl;kj as;lfj asl;fj a;slkjf ;lasj f;lasj fl;kjsa ;flkj as;lkfj
sa;lkfj ;alsfj ;laskjf ;laskj ;lksj afd;lksj f;lkj s;lkfj sa;lkfj s;laf j;lsa fj;ls j

2 Properties of Clos, Reach, and Reach[†]

The following theorem, due to [34], says that we can neatly characterize the algebraic structure of Clos a closure operator. Note that Leitgeb proves this for *inhibition nets*, i.e. weightless neural networks with both excitatory and inhibitory connections. But inhibition nets and our nets $\mathcal{N} \in \mathbf{Net}$ are equivalent with respect to their propagation structure—I prove this result again for \mathbf{Net} as a kind of “sanity check” that my definitions are correct.

Proposition 2.1. (Leitgeb, [34; 35]) For all $S, S_1, \dots, S_k \in \text{State}$,

Inclusion. $S \subseteq \text{Clos}(S)$

Idempotence. $\text{Clos}(\text{Clos}(S)) = \text{Clos}(S)$

Cumulative. If $S_1 \subseteq S_2 \subseteq \text{Clos}(S_1)$, then $\text{Clos}(S_1) = \text{Clos}(S_2)$

Proof. I'll prove each in turn:

Inclusion. By definition, $\text{Clos}(S) = F_S^k(S)$ for some $k \in \mathbb{N}$, where F_S^k is the transition function from Definition [which?]. By induction on this k (the number of iterations needed to construct the closure):

Base Step. $k = 0$, and so $\text{Clos}(S) = S$. So if $n \in S$, then $n \in \text{Clos}(S)$.

Inductive Step. Let $k \geq 0$, and suppose $n \in S$. We have $\text{Clos}(S) = F_S^k(S) = F_S(F_S^{k-1}(S))$.

Expanding this term out, we have

$$F_S(F_S^{k-1}(S)) = S \cup \left\{ n \mid A \left(\sum_{m \in \text{preds}(n)} W(m, n) \cdot \chi_{F_S^{k-1}(S)}(m) \right) = 1 \right\}$$

Since $n \in S$, n is in the left-hand side of this union. And so $n \in \text{Clos}(S)$.

Idempotence. I will prove a stronger claim: For all $k \in \mathbb{N}$,

$$F_S^k(\text{Clos}(S)) = \text{Clos}(S)$$

In other words, applying the transition function F_S any number of times to $\text{Clos}(S)$ has no effect. Since $\text{Clos}(\text{Clos}(S)) = F_S^k(\text{Clos}(S))$ for some $k \in \mathbb{N}$, the Idempotence property follows from this. Let's proceed by induction on k .

Base Step. $k = 0$, and so the goal simplifies to $\text{Clos}(S) = \text{Clos}(S)$, which is true.

Inductive Step. Let $k \geq 0$. We have $F_S^k(\text{Clos}(S)) = F_S(F_S^{k-1}(\text{Clos}(S)))$. Our inductive hypothesis here is that, for $k - 1$, $F_S^{k-1}(\text{Clos}(S)) = \text{Clos}(S)$. So we can substitute that to get $F_S^k(\text{Clos}(S)) = F_S(\text{Clos}(S)) = \text{Clos}(S)$, which is what we wanted to show.

Cumulative. Suppose $S_1 \subseteq S_2 \subseteq \text{Clos}(S_1)$ (see Figure [\[DIAGRAM\]](#)). First, since $\text{Clos}(S_2)$ is a fixed point under S_2 , we have $F_{S_2}(\text{Clos}(S_2)) = \text{Clos}(S_2)$. But I will show that $\text{Clos}(S_1)$ is *also* a fixed point under S_2 , i.e. $F_{S_2}(\text{Clos}(S_1)) = \text{Clos}(S_1)$ (notice the difference in the subscripts). Since we assumed that there is a *unique* fixed point under S_2 , it will follow that these two states must be the same. In other words, $\text{Clos}(S_1) = \text{Clos}(S_2)$.

Let's expand $F_{S_2}(\text{Clos}(S_1))$. By definition of F ,

$$F_{S_2}(\text{Clos}(S_1)) = S_2 \cup \left\{ n \mid A \left(\sum_{m \in \text{preds}(n)} W(m, n) \cdot \chi_{\text{Clos}(S_2)}(m) \right) = 1 \right\}$$

Compare this to $F_{S_1}(\text{Clos}(S_1))$:

$$F_{S_1}(\text{Clos}(S_1)) = S_1 \cup \left\{ n \mid A \left(\sum_{m \in \text{preds}(n)} W(m, n) \cdot \chi_{\text{Clos}(S_2)}(m) \right) = 1 \right\}$$

Putting the two together, we can see that

$$\begin{aligned} F_{S_2}(\text{Clos}(S_1)) &= (F_{S_1}(\text{Clos}(S_1)) - S_1) \cup S_2 \\ &= (\text{Clos}(S_1) - S_1) \cup S_2 && \text{(since } \text{Clos}(S_1) \text{ is fixed under } S_1) \\ &= \text{Clos}(S_1) && \text{(since } S_1 \subseteq S_2 \subseteq \text{Clos}(S_1)) \end{aligned} \quad \square$$

In the terminology of [33], Prop is a *cumulative* closure operator (it satisfies the cumulative property). I mentioned briefly in the previous chapter that Clos is *not* a fully monotonic closure operator—I'll now give the proof. The idea is that negative weights in the net can be used to inhibit incoming signals from a state.

Proposition 2.2. (Leitgeb, [34; 35]) It is not the case that for all $S_1, S_2 \in \text{State}$, if $S_1 \subseteq S_2$, then $\text{Clos}(S_1) \subseteq \text{Clos}(S_2)$.

Proof. Consider the BFNN \mathcal{N} in Figure [DIAGRAM]. Let the activation function A be $A(x) = 1$ iff $x > 0$. We have $S_1 = \{a\} \subseteq \{a, b\} = S_2$, and so the hypothesis holds. But $\text{Clos}(S_1) = \{a, c\} \not\subseteq \{a, b\} = \text{Clos}(S_2)$. (Observe that c does not get activated in $\text{Clos}(S_2)$ because the weights cancel each other out.) \square

Next, let's check that Reach is in fact a monotonic closure operator.

Proposition 2.3. For all $S, A, B \in \text{State}$,

Inclusion. $S \subseteq \text{Reach}(S)$

Idempotent. $\text{Reach}(\text{Reach}(S)) = \text{Reach}(S)$

Monotonic. If $A \subseteq B$ then $\text{Reach}(A) \subseteq \text{Reach}(B)$

Closed under \cup . $\text{Reach}(A \cup B) = \text{Reach}(A) \cup \text{Reach}(B)$

Proof. I'll prove each in turn:

Inclusion. Suppose $n \in S$. We have the trivial path from $n \in S$ to itself.

Idempotent. The (\leftarrow) direction follows from inclusion. As for (\rightarrow) , suppose $n \in \text{Reach}(\text{Reach}(S))$, i.e. there is a path from some $m \in \text{Reach}(S)$ to n . By definition of Reach again, there is a path from some $x \in S$ to m . But we can put these together to obtain a path from $x \in S$ to n .

Monotonic. Suppose $A \subseteq B$, and let $n \in \text{Reach}(A)$. By definition of Reach , we have a path from some $m \in A$ to n . But since $A \subseteq B$, $m \in B$. So we have a path from $m \in B$ to n , i.e. $n \in \text{Reach}(B)$.

Closed under \cup . For the (\rightarrow) direction, suppose $n \in \text{Reach}(A \cup B)$. So there is a path from some $m \in A \cup B$ to n . We have two cases: $m \in A$, in which case we have a path from $m \in A$ to n ; or $m \in B$, in which case we have a path from $m \in B$ to n .

As for the (\leftarrow) direction, suppose $n \in \text{Reach}(A) \cup \text{Reach}(B)$. Similarly, we have two cases, and in either case we have a path from $n \in A \cup B$. So $n \in \text{Reach}(A \cup B)$. \square

Reach^\downarrow is as well, and the proof for this is the same, mutatis mutandis the direction of the path.

Proposition 2.4. For all $S, A, B \in \text{State}$,

Inclusion. $S \subseteq \text{Reach}^\downarrow(S)$

Idempotent. $\text{Reach}^\downarrow(\text{Reach}^\downarrow(S)) = \text{Reach}^\downarrow(S)$

Monotonic. If $A \subseteq B$ then $\text{Reach}^\downarrow(A) \subseteq \text{Reach}^\downarrow(B)$

Closed under \cup . $\text{Reach}^\downarrow(A \cup B) = \text{Reach}^\downarrow(A) \cup \text{Reach}^\downarrow(B)$

[What interaction property do Reach and Reach^\downarrow share?]

3 Soundness for the Base Semantics

[Prove a few key properties for forward propagation, we can read the axioms directly off of these, then the proofs for the axioms' soundness follows]

3.1 Using Conditional Logic

Definition 3.1. The proof system for the conditional logic over $\mathcal{L}_{\Rightarrow}$ is given as follows: $\vdash \varphi$ iff [todo]

Definition 3.2. [Definition of $\Gamma \vdash \alpha \Rightarrow \beta$]

Theorem 3.3. (Hannes Leitgeb, [34; 35]) This proof system is sound; for all $\Gamma \subseteq \mathcal{L}_{\text{best}}$ and $\varphi \in \mathcal{L}_{\text{best}}$, if $\Gamma \vdash \varphi$ then $\Gamma \models \varphi$.

Proof. [todo]

□

3.2 Using Modal Logic

For [best] alone, [34] proves that the properties in Proposition [which?] are complete for Clos over binary, feed-forward nets. We transcribe these into our modal language.

Definition 3.4. The proof system for the base modal logic over $\mathcal{L}_{\text{best}}$ is given as follows: $\vdash \varphi$ iff either φ is one of the axioms:

Axioms for [best].

Dual. $\langle \text{best} \rangle \varphi \leftrightarrow \neg [\text{best}] \neg \varphi$

Cumulative. $((\varphi \rightarrow \psi) \wedge ([\text{best}] \psi \rightarrow \varphi)) \rightarrow ([\text{best}] \varphi \rightarrow \psi)$

Refl. $[\text{best}] \varphi \rightarrow \varphi$

Trans. $[\text{best}]\varphi \rightarrow [\text{best}][\text{best}]\varphi$

Axioms for box .

Dual. $\Diamond\varphi \leftrightarrow \neg\text{box}\neg\varphi$

Distr. $\text{box}(\varphi \rightarrow \psi) \rightarrow (\text{box}\varphi \rightarrow \text{box}\psi)$

Refl. $\text{box}\varphi \rightarrow \varphi$

Trans. $\text{box}\varphi \rightarrow \text{boxbox}\varphi$

Interaction axioms for box and \Box^\downarrow .

Distr. $\Box^\downarrow(\varphi \rightarrow \psi) \rightarrow (\Box^\downarrow\varphi \rightarrow \Box^\downarrow\psi)$

Back. $\varphi \rightarrow \text{box}\Box^\downarrow\varphi$

Forth. $\varphi \rightarrow \Box^\downarrow\Diamond\varphi$

or φ follows from some previously obtained formulas by one of the inference rules.

MP. From $\vdash \varphi \rightarrow \psi$ and $\vdash \varphi$ we can infer $\vdash \psi$

Nec. From $\vdash \varphi$ we can infer $\vdash \Box\varphi$ for $\Box \in \{\text{box}, \Box^\downarrow, [\text{best}]\}$

These axioms are the usual complete axioms for normal modal logic over reflexive and transitive frames [plus some tricks from temporal logic—explain!]. [Figure out if any additional interaction axioms are needed!]

Definition 3.5. If $\Gamma \subseteq \mathcal{L}_{\text{best}}$ is a set of formulas and $\varphi \in \mathcal{L}_{\text{best}}$ a formula, then $\Gamma \vdash \varphi$ whenever there are finitely many $\psi_1, \dots, \psi_k \in \Gamma$ such that $\vdash \psi_1 \wedge \dots \wedge \psi_k \rightarrow \varphi$.

Theorem 3.6. (⚡) These rules and axioms are sound; for all $\Gamma \subseteq \mathcal{L}_{\text{best}}$ and $\varphi \in \mathcal{L}_{\text{best}}$, if $\Gamma \vdash \varphi$ then $\Gamma \models \varphi$.

Proof. [todo]

□


3.3 Example: Verifying a Neural Network's Behavior

4 Properties of Hebb and Hebb*

We have the following algebraic properties for Hebb.

Theorem 4.1. (⚡) [revise properties from FLAIRS paper...]

One worry we might have is that, in each iteration, we always update by $\text{Clos}(S)$ in the *original* net. But it turns out that this $\text{Clos}(S)$ doesn't change with each iteration, i.e.

Proposition 4.2. () $\text{Clos}_{\text{Hebb}(\mathcal{N}, S)}(S) = \text{Clos}_{\mathcal{N}}(S)$

Proof. Let F_S be the state transition function for \mathcal{N} under S , and F'_S be the state transition function for $\text{Hebb}(\mathcal{N}, S)$ under S . First, since $\text{Clos}_{\text{Hebb}(\mathcal{N}, S)}(S)$ is a fixed point under S in $\text{Hebb}(\mathcal{N}, S)$, we have $F'_S(\text{Clos}_{\text{Hebb}(\mathcal{N}, S)}(S)) = \text{Clos}_{\text{Hebb}(\mathcal{N}, S)}(S)$. But I will show that $\text{Clos}_{\mathcal{N}}(S)$ is *also* a fixed point under S in $\text{Hebb}(\mathcal{N}, S)$, i.e. $F'_S(\text{Clos}_{\mathcal{N}}(S)) = \text{Clos}_{\mathcal{N}}(S)$. Since we assumed there is a *unique* fixed point under S in $\text{Hebb}(\mathcal{N}, S)$, it will follow that these two states must be the same. In other words, $\text{Clos}_{\text{Hebb}(\mathcal{N}, S)}(S) = \text{Clos}_{\mathcal{N}}(S)$.

For the (\leftarrow) direction, suppose $n \in \text{Clos}_{\mathcal{N}}(S)$. Since $\text{Clos}_{\mathcal{N}}(S)$ is a fixed point under S in \mathcal{N} , $\text{Clos}_{\mathcal{N}}(S) = F_S(\text{Clos}_{\mathcal{N}}(S))$. By definition of F , either $n \in S$ (in which case we are done), or n is activated by its predecessors m in $\text{Clos}_{\mathcal{N}}(S)$ over \mathcal{N} , i.e.

$$A\left(\sum_{m \in \text{preds}(n)} W_{\mathcal{N}}(m, n) \cdot \chi_{\text{Clos}_{\mathcal{N}}(S)}(m)\right) = 1$$

By the first part of Proposition [\[todo\]](#), each $W_{\mathcal{N}}(m, n) \leq W_{\text{Hebb}(\mathcal{N}, S)}(m, n)$. So the inner sum using the former is \leq the inner sum using the latter. Since A is nondecreasing, we have

$$A\left(\sum_{m \in \text{preds}(n)} W_{\text{Hebb}(\mathcal{N}, S)}(m, n) \cdot \chi_{\text{Clos}_{\mathcal{N}}(S)}(m)\right) = 1$$

But this implies that $n \in F'_S(\text{Clos}_{\mathcal{N}}(S))$.

As for the (\rightarrow) direction, suppose $n \in F'_S(\text{Clos}_{\mathcal{N}}(S))$. By definition of F' , either $n \in S$ (in which case we are done), or n is activated by its predecessors m in $\text{Clos}_{\mathcal{N}}(S)$ over $\text{Hebb}(\mathcal{N}, S)$, i.e.

$$A\left(\sum_{m \in \text{preds}(n)} W_{\text{Hebb}(\mathcal{N}, S)}(m, n) \cdot \chi_{\text{Clos}_{\mathcal{N}}(S)}(m)\right) = 1$$

Suppose for contradiction that $n \notin \text{Clos}_{\mathcal{N}}(S)$. By the second part of Proposition [\[todo\]](#), each $W_{\text{Hebb}(\mathcal{N}, S)}(m, n) = W_{\mathcal{N}}(m, n)$, and so we have

$$A\left(\sum_{m \in \text{preds}(n)} W_{\mathcal{N}}(m, n) \cdot \chi_{\text{Clos}_{\mathcal{N}}(S)}(m)\right) = 1$$

but this implies that $n \in F_S(\text{Clos}_{\mathcal{N}}(S)) = \text{Clos}_{\mathcal{N}}(S)$, which contradicts $n \notin \text{Clos}_{\mathcal{N}}(S)$. \square

and so Hebb^* is equivalent to repeatedly applying Hebb until we reach a fixed point [\[31\]](#). [\[Elaborate on this point, it's said a little too quickly for the reader to internalize it! \(maybe a picture](#)

would help?)]

We have the following algebraic properties for Hebb^* . Before proving these, I'll give some intuition for what these properties say about Hebb^* . [(1) is just used to show (2)] Part (2) expresses a lower bound for $\text{Clos}_{\text{Hebb}^*(\mathcal{N}, A)}(B)$, whereas (3) gives an upper bound within $\text{Clos}(A)$. [There's got to be a better way to explain what these mean... this isn't clear or intuitive at all.]

Proposition 4.3. (\clubsuit) $\text{Clos}_{\text{Hebb}^*(\mathcal{N}, S)}(S) = \text{Clos}_{\mathcal{N}}(S)$

Proof. The proof is the same as the proof for Hebb (see Proposition [todo]). In place of Propositions [todo] and [todo], we instead apply Propositions [todo] and [todo], respectively. \square

Theorem 4.4. (\clubsuit) Let $\mathcal{N} \in \text{Net}$, and suppose \mathcal{N} is fully connected. For all $A, B \in \text{State}$,

$$\text{Clos}_{\text{Hebb}^*(\mathcal{N}, A)}(B) = \text{Clos}(B \cup (\text{Clos}(A) \cap \text{Reach}(\text{Clos}(A) \cap \text{Clos}(B))))$$

Proof. Let F_B be the state transition function for \mathcal{N} under B , and F_B^* be the state transition function for $\text{Hebb}^*(\mathcal{N}, A)$ under B . For notational convenience, let T be the set inside Clos on the right-hand side, i.e.

$$T = B \cup (\text{Clos}(A) \cap \text{Reach}(\text{Clos}(A) \cap \text{Clos}(B)))$$

This proof follows the major plot beats of the proof for Theorem [TODO]. First, since $\text{Clos}_{\text{Hebb}^*(\mathcal{N}, A)}(B)$ is a fixed point under B in $\text{Hebb}^*(\mathcal{N}, A)$, we have $F_B^*(\text{Clos}_{\text{Hebb}^*(\mathcal{N}, A)}(B)) = \text{Clos}_{\text{Hebb}^*(\mathcal{N}, A)}(B)$. But I will show that $\text{Clos}(T)$ is *also* a fixed point under B in $\text{Hebb}(\mathcal{N}, A)$, i.e. $F_B^*(\text{Clos}(T)) = \text{Clos}(T)$. Since we postulated that there is a *unique* fixed point under B in $\text{Hebb}^*(\mathcal{N}, A)$, it will follow that these two states must be the same: $\text{Clos}_{\text{Hebb}^*(\mathcal{N}, A)}(B) = \text{Clos}(T)$.

Let's show that $F_B^*(\text{Clos}(T)) = \text{Clos}(T)$. For the (\leftarrow) direction, suppose $n \in \text{Clos}(T)$. Since $\text{Clos}(T)$ is a fixed point under T in \mathcal{N} , $\text{Clos}(T) = F_T(\text{Clos}(T))$. By definition of F , we have two cases:

Case 1. $n \in T$, i.e. $n \in B \cup (\text{Clos}(A) \cap \text{Reach}(\text{Clos}(A) \cap \text{Clos}(B)))$. If $n \in B$, then we're done by the definition of F_B^* (the next state includes all nodes in B). Otherwise, we have $n \in \text{Clos}(A)$ and a path from some $m \in \text{Clos}(A) \cap \text{Clos}(B)$ to n in \mathcal{N} . Since \mathcal{N} is fully connected, m is in fact a predecessor of n . Moreover, $m \in \text{Clos}(T)$, since $m \in \text{Clos}(A) \cap \text{Clos}(B)$ and by inclusion of Reach and Clos . So we have $m \in \text{preds}(n)$, $m, n \in \text{Clos}(A)$, and $m \in \text{Clos}(T)$. But these are exactly the conditions of Lemma [todo]! This means we

have

$$A\left(\sum_{m \in \text{preds}(n)} W_{\text{Hebb}^*(\mathcal{N}, A)}(m, n) \cdot \chi_{\text{Clos}(T)}(m)\right) = 1$$

which implies that $n \in F_B^*(\text{Clos}(T))$.

Case 2. n is activated by its predecessors m in $\text{Clos}(T)$ over \mathcal{N} , i.e.

$$A\left(\sum_{m \in \text{preds}(n)} W_{\mathcal{N}}(m, n) \cdot \chi_{\text{Clos}(T)}(m)\right) = 1$$

By the first part of Proposition [todo], each $W_{\mathcal{N}}(m, n) \leq W_{\text{Hebb}^*(\mathcal{N}, S)}(m, n)$. So the inner sum using the former is \leq the inner sum using the latter. Since A is nondecreasing, we have

$$A\left(\sum_{m \in \text{preds}(n)} W_{\text{Hebb}^*(\mathcal{N}, A)}(m, n) \cdot \chi_{\text{Clos}(T)}(m)\right) = 1$$

But this immediately implies that $n \in F_B^*(\text{Clos}(T))$.

As for the (\rightarrow) direction, suppose $n \in F_B^*(\text{Clos}(T))$. By definition of F^* , we have two cases:

Case 1. $n \in B$. So $n \in B \cup (\text{Clos}(A) \cap \text{Reach}(\text{Clos}(A) \cap \text{Clos}(B))) = T$. By inclusion of Clos , we have $n \in \text{Clos}(T)$.

Case 2. n is activated by its predecessors m in $\text{Clos}(T)$ over $\text{Hebb}^*(\mathcal{N}, A)$, i.e.

$$A\left(\sum_{m \in \text{preds}(n)} W_{\text{Hebb}^*(\mathcal{N}, A)}(m, n) \cdot \chi_{\text{Clos}(T)}(m)\right) = 1$$

Suppose for contradiction that $n \notin \text{Clos}(T)$. By the second part of Proposition [todo], each $W_{\text{Hebb}^*(\mathcal{N}, A)}(m, n) = W_{\mathcal{N}}(m, n)$, and so we have

$$A\left(\sum_{m \in \text{preds}(n)} W_{\mathcal{N}}(m, n) \cdot \chi_{\text{Clos}(T)}(m)\right) = 1$$

but this implies that $n \in F_T(\text{Clos}(T)) = \text{Clos}(T)$, which contradicts $n \notin \text{Clos}(T)$. So we must have $n \in \text{Clos}(T)$. □

Corollary 4.5. If $\text{Clos}(A) \cap \text{Clos}(B) = \emptyset$, then $\text{Clos}_{\text{Hebb}^*(\mathcal{N}, A)}(B) = \text{Clos}(B)$.

Proof. Suppose $\text{Clos}(A) \cap \text{Clos}(B) = \emptyset$. We have

$$\begin{aligned} \text{Clos}_{\text{Hebb}^*(\mathcal{N}, A)}(B) &= \text{Clos}(B \cup (\text{Clos}(A) \cap \text{Reach}(\text{Clos}(A) \cap \text{Clos}(B)))) \\ &= \text{Clos}(B \cup (\text{Clos}(A) \cap \text{Reach}(\emptyset))) \\ &= \text{Clos}(B \cup \emptyset) \\ &= \text{Clos}(B) \end{aligned}$$

□

5 Soundness for the Logic of Hebbian Learning

[Prove soundness of axioms for both single-step and iterated Hebbian update!]

5.1 Example: Verifying a Neural Networks Behavior After Learning

do example using single-step Hebbian learning, since iterated is a bit more abstract...

6 Reflections on Verification and Extraction

[Here's where I can discuss things like property verification vs model building (alignment), extraction, “valuation search”, and Thomas Icards' method]

Chapter 5

Completeness: Neural Network Model Building

1 Introduction

2 Completeness for the Base Semantics

2.1 Using Conditional Logic

- This section is prior work; I will basically follow Hannes [34; 35].
- The crux of completeness is *neural network model building*—we need to construct a neural network \mathcal{N} for any given set of conditional constraints Γ .
- We do this by way of KLM *plausibility models* [33]. KLM proved that for every consistent set of conditionals $\Gamma \in \mathcal{L}^\Rightarrow$ over this cumulative logic [todo], there is a finite cumulative-ordered model $\mathcal{M} = \langle W, R, V \rangle$ such that $\mathcal{M} \models \Gamma$, i.e., $\mathcal{M} \models \varphi \Rightarrow \psi$ for all $\varphi \Rightarrow \psi \in \Gamma$. I will take this fact for granted, but for more details see Appendix [todo]. (I might also have to briefly explain what a plausibility model is and talk about the best function + the semantics for $\varphi \Rightarrow \psi$!)
- So the plan is: Given this finite plausibility model, we will construct an equivalent finite weighted neural network \mathcal{N} . Hannes does this using the following NAND construction.

The NAND Construction. Suppose we have plausibility model $\mathcal{M} = \langle W, R, V \rangle$ and we want to construct an equivalent neural network \mathcal{N} . In [34], Hannes first does this for *inhibition nets*, i.e., nets with inhibitory edges that block excitatory edges. (He handles weighted nets later.) I will first consider his construction, and then modify it for weighted nets.

Here's the inhibition net construction: First, take $N = W$ (so \mathcal{N} is still finite), $V = V$, let the excitatory edges be exactly $E = R$, and create a fresh node bias. Next, create an edge from bias to every n that is not E -minimal (in other words, if n has any predecessors at all, then bias is one of them). Then for each node n and its predecessors bias = m_0, m_1, \dots, m_r , connect inhibition

edges as follows.

[DIAGRAM]

That is, each node m_i is inhibited by m_{i-1} (bias = m_0 inhibited by m_r). This has the following effect: if all m_i activate, they each inhibit each other, and so n does not activate. If only *some* m_i activate, then there is some m_i that is uninhibited, and so n activates. And finally, since bias is always active we cannot have *no* m_i active. In other words, $n \in F_{S_0}(S)$ iff $n \in S_0$, or *not all non-bias predecessors m are in S* . (Since bias is always active, this results in a NAND-like output.)

We can simulate this effect with weighted neural networks. Create an edge from bias to every n that is not E -minimal. Then pick $W(m, n) = \frac{1}{|\text{preds}| + 1}$ (the extra +1 accounts for the bias). Finally, pick $A(x) = 1$ iff $x < 1$. For now, the choice of learning rate η is arbitrary (see Section [todo]). Take a moment to check that $n \in F_{S_0}(S)$ iff $n \in S_0$, or at least one non-bias predecessor $m \notin S$.

[DIAGRAM]

What is the relationship between this neural network's fixed points $\text{Clos}(S)$ and the plausibility model's minimal states $\text{best}_R(S)$? It turns out that for this NAND construction, Clos is precisely the *dual* of best_R :

Lemma 2.1. Let $\mathcal{M} = \langle W, R, V \rangle$ be a plausibility model, and \mathcal{N} be given by the NAND construction above. For all $S \in \text{State}$, $\text{Clos}(S) = (\text{best}_R(S^c))^c$.

Proof. Once again, I will take advantage of the fact that fixed points of the transition function F_S are unique. First, since $\text{Clos}(S)$ is a fixed point under S , we have $F_S(\text{Clos}(S)) = \text{Clos}(S)$. But I will show that $(\text{best}_R(S^c))^c$ is *also* a fixed point under S , i.e. $F_S((\text{best}_R(S^c))^c) = (\text{best}_R(S^c))^c$. Since we assumed that there is a *unique* fixed point under S , it will follow that these two sets must be the same. In other words, $\text{Clos}(S) = (\text{best}_R(S^c))^c$.

For the (\rightarrow) direction, suppose $n \in F_S((\text{best}_R(S^c))^c)$. By construction of F_S , we have two cases:

Case 1. $n \in S$. In this case, we trivially have $n \notin \text{best}_R(S^c)$, since n is not even in S^c . And so $n \in (\text{best}_R(S^c))^c$.

Case 2. At least one non-bias predecessor $m \notin (\text{best}_R(S^c))^c$. In this case, we have $m \in \text{best}_R(S^c)$. But m is R -better than n (that is, mRn), which implies that n cannot be a

best S^c -element: $n \notin \text{best}_R(S^c)$. So $n \in (\text{best}_R(S^c))^c$.

As for the (\leftarrow) direction, suppose contrapositively that $n \notin F_S((\text{best}_R(S^c))^c)$. By construction, this means that $n \notin S$ and all predecessors m of n (including bias of course) are in $(\text{best}_R(S^c))^c$. We already have $n \in S^c$; from here I'd like to show that n is the *best* S^c -element. Suppose for contradiction that $n \notin \text{best}_R(S^c)$. By the smoothness condition (see Appendix [todo]) there must be some $m \in S^c, mRn$ that is the best, i.e., $m \in \text{best}_R(S^c)$. Note that we always have $\text{bias} \in S$, and in particular this means $\text{bias} \notin \text{best}_R(S^c)$ (by best-inclusion, since $\text{best}_R(S^c) \subseteq S^c$). So m cannot be the bias node. Complementing, we see that $m \notin (\text{best}_R(S^c))^c$. In other words, some non-bias predecessor of n is not in $(\text{best}_R(S^c))^c$. By construction of F_S , this means $n \in F_S((\text{best}_R(S^c))^c)$, which contradicts our initial hypothesis. \square

The following lemma says that the constructed net \mathcal{N} is in fact equivalent to \mathcal{M} .

Lemma 2.2. Let $\mathcal{M} = \langle W, R, V \rangle$ be a plausibility model, and \mathcal{N} be given by the NAND construction above. For all conditional terms $\alpha \in [\text{I need a symbol for this...}]$, $\llbracket \alpha \rrbracket_{\mathcal{N}} = \llbracket \alpha \rrbracket_{\mathcal{M}}^c$ (the complement of $\llbracket \alpha \rrbracket_{\mathcal{M}}$!)

Proof. We proceed by induction on α .

[TODO] \square

Lemma 2.3. Let $\mathcal{M} = \langle W, R, V \rangle$ be a plausibility model, and \mathcal{N} be given by the NAND construction above. For all conditional formulas $\alpha \Rightarrow \beta \in \mathcal{L}^{\Rightarrow}$, where $\alpha, \beta \in [\text{todo}]$,

$$\mathcal{N} \models \alpha \Rightarrow \beta \text{ iff } \mathcal{M} \models \alpha \Rightarrow \beta$$

Proof. Combining the previous two lemmas, we have

$$\begin{aligned} \mathcal{N} \models \alpha \Rightarrow \beta & \text{ iff } \llbracket \beta \rrbracket_{\mathcal{N}} \subseteq \text{Clos}(\llbracket \alpha \rrbracket_{\mathcal{N}}) && \text{(by definition)} \\ & \text{ iff } \llbracket \beta \rrbracket_{\mathcal{M}}^c \subseteq \text{Clos}(\llbracket \alpha \rrbracket_{\mathcal{M}}^c) && \text{(by Lemma [todo])} \\ & \text{ iff } \llbracket \beta \rrbracket_{\mathcal{M}}^c \subseteq \text{best}_R(\llbracket \alpha \rrbracket_{\mathcal{M}})^c && \text{(by Lemma [todo])} \\ & \text{ iff } \text{best}_R(\llbracket \alpha \rrbracket_{\mathcal{M}}) \subseteq \llbracket \beta \rrbracket_{\mathcal{M}} && \text{(flipping } \subseteq \text{ and complementing both sides)} \\ & \text{ iff } \mathcal{M} \models \alpha \Rightarrow \beta && \text{(by definition)} \end{aligned}$$

\square

Theorem 2.4. (Model Building for $\mathcal{L}^{\Rightarrow}$) For all consistent $\Gamma \subseteq \mathcal{L}^{\Rightarrow}$, there is finite \mathcal{N} such that $\mathcal{N} \models \Gamma$.

Proof. \square

\square

Corollary 2.5. (Completeness for \mathcal{L}^\Rightarrow) For all consistent $\Gamma \subseteq \mathcal{L}^\Rightarrow$ and all conditionals $\alpha \Rightarrow \beta \in \mathcal{L}^\Rightarrow$,


$$\text{if } \Gamma \models \alpha \Rightarrow \beta \text{ then } \Gamma \vdash \alpha \Rightarrow \beta$$

Proof. □

□

2.2 Using Modal Logic


- Need to also have modal logic semantics for plausibility models. The big thing here is that I have to prove we can still build a finite plausibility model in this setting!!!—completeness on the plausibility model end is going to be the hard part, and will involve temporal logic tricks.
- We will get a lot more mileage out of Lemma [todo]!

Lemma 2.6. () Let $\mathcal{M} = \langle W, R, V \rangle$ be a plausibility model, and \mathcal{N} be given by the NAND construction above. For all modal formulas $\varphi \in \mathcal{L}$ and all $w \in W = N$,

$$\mathcal{N}, w \Vdash \varphi \text{ iff } \mathcal{M}, w \Vdash \varphi$$


Proof. □

□

Theorem 2.7. () (Model Building for \mathcal{L}) For all consistent $\Gamma \subseteq \mathcal{L}$, there is finite \mathcal{N} such that $\mathcal{N} \models \Gamma$.

Proof. □

□

Corollary 2.8. () (Completeness for \mathcal{L}) For all consistent $\Gamma \subseteq \mathcal{L}$ and all formulas $\varphi \in \mathcal{L}$,

$$\text{if } \Gamma \models \varphi \text{ then } \Gamma \vdash \varphi$$

Proof. □

□

2.3 Example: Building a Neural Network

3 Reduction Axioms for Iterated Hebbian Update

- In the previous section, I gave sound axioms for Hebb*. It turns out those axioms are nearly complete! I'll show this here, and give the complete set of “reduction axioms”

- I already proved soundness, but I have to do it again for these new axioms!

Definition 3.1. [Reduction axioms for Hebb*]

Theorem 3.2. (☞) These reduction axioms are sound; for all $\Gamma \subseteq \mathcal{L}^*$ and $\varphi \in \mathcal{L}^*$, if $\Gamma \vdash \varphi$ then $\Gamma \models \varphi$.

Proof. [todo]

□

Proof. [todo]

□

4 Completeness for Iterated Hebbian Update

Definition 4.1. [Term rewriting translation system]

Proposition 4.2. (☞) Each $\text{tr}(\varphi)$ is update-operator free [todo, fix this statement!]

Proof. [todo]

□

Proposition 4.3. (☞) Each $\text{tr}(\varphi)$ actually terminates [todo, fix this statement!]

Proof. [todo]

□

Proposition 4.4. (☞) $\vdash \varphi \leftrightarrow \text{tr}(\varphi)$ [todo, fix this statement, which \vdash is this?]

Proof. [todo]

□

Theorem 4.5. (☞ Model Building for \mathcal{L}^*) For all consistent $\Gamma \subseteq \mathcal{L}^*$, there is finite \mathcal{N} such that $\mathcal{N} \models \Gamma$.

Proof. [todo]

□

Theorem 4.6. (☞ Completeness for \mathcal{L}^*) For all consistent $\Gamma \subseteq \mathcal{L}^*$ and all formulas $\varphi \in \mathcal{L}^*$,

if $\Gamma \models \varphi$ then $\Gamma \vdash \varphi$

Theorem 4.7. (☞) For all $\Gamma^* \subseteq \mathcal{L}^*$, there is \mathcal{N} such that $\mathcal{N} \models \Gamma^*$.

Proof. Let $\Gamma^* \subseteq \mathcal{L}^*$. As outlined in the paper, our plan is to define rewrite rules based on our reduction axioms that “translate away” all of the dynamic formulas $\langle \varphi \rangle_{\text{Hebb}} \psi$ in Γ^* , resulting in $\Gamma^{\text{tr}} \subseteq \mathcal{L}$. By our assumption, we have a net $\mathcal{N} \models \Gamma^{\text{tr}}$, and we show that this very same net $\mathcal{N} \models \Gamma^*$.

It's easy to see intuitively how this translation should go. For example, given the formula

$$\langle p \rangle_{\text{Hebb}} (\langle p \rangle_{\text{Hebb}} \langle \mathbf{B} \rangle \wedge \diamond) \in \Gamma^*$$

we would recursively apply our reduction axioms, pushing $\langle p \rangle_{\text{Hebb}}$ further into the expression until we can eliminate the propositional cases $\langle p \rangle_{\text{Hebb}} q$.

We define the term-rewriting system that does the translation $\tau(\varphi)$ for all φ as follows.

- $\tau(p) = p$
- $\tau(\neg \varphi) = \neg \tau(\varphi)$
- $\tau(\varphi \wedge \psi) = \tau(\varphi) \wedge \tau(\psi)$
- $\tau(\mathbf{K}\varphi) = \text{box}(\tau(\varphi))$
- $tr(\langle \varphi \rangle_{\text{Hebb}} p) = tr(p)$
- $tr(\langle \varphi \rangle_{\text{Hebb}} \neg \psi) = tr(\neg \langle \varphi \rangle_{\text{Hebb}} \psi)$
- $tr(\langle \varphi \rangle_{\text{Hebb}} (\psi \wedge \rho)) = tr(\langle \varphi \rangle_{\text{Hebb}} \psi \wedge \langle \varphi \rangle_{\text{Hebb}} \rho)$
- $tr(\langle \varphi \rangle_{\text{Hebb}} \diamond) = tr(\diamond)$
- $tr(\langle \varphi \rangle_{\text{Hebb}} \langle \mathbf{B} \rangle) = tr(\langle \mathbf{B} \rangle)$
- $tr(\langle \varphi \rangle_{\text{Hebb}} \langle \psi \rangle_{\text{Hebb}} \rho) = tr(\langle \varphi \rangle_{\text{Hebb}} (tr(\langle \psi \rangle_{\text{Hebb}} \rho)))$

Formally, the term-rewriting system takes a formula φ and recursively applies these equational rules to φ (from left-to-right). We just need to check that

1. For all ψ , $tr(\psi)$ is update-operator-free
2. This term rewriting actually terminates

The work involved in showing termination is long and tedious. The usual approach is to define a measure on formulas $c(\varphi)$ that *decreases* with each application of our reduction axioms (from left-to-right). In particular, we need c to satisfy

- If ψ is a subexpression of φ , $c(\varphi) > c(\psi)$
- $c(\langle \varphi \rangle_{\text{Hebb}} p) > c(p)$
- $c(\langle \varphi \rangle_{\text{Hebb}} \neg \psi) > c(\neg \langle \varphi \rangle_{\text{Hebb}} \psi)$
- $c(\langle \varphi \rangle_{\text{Hebb}} (\psi \wedge \rho)) > c(\langle \varphi \rangle_{\text{Hebb}} \psi \wedge \langle \varphi \rangle_{\text{Hebb}} \rho)$

- $c(\langle \varphi \rangle_{\text{Hebb}} \diamond) > c(\diamond)$
- $c(\langle \varphi \rangle_{\text{Hebb}} \langle \mathbf{B} \rangle) > c(\langle \mathbf{B} \rangle)$
- $c(\langle \varphi \rangle_{\text{Hebb}} \langle \psi \rangle_{\text{Hebb}} \rho) > c(\langle \varphi \rangle_{\text{Hebb}} (tr(\langle \psi \rangle_{\text{Hebb}} \rho)))$

But coming up with a measure c that works is tricky, and is dependent on the specific reduction axioms. For the gritty details involved in coming up with this measure, as well as proving termination for the term rewriting system, see [12].

From here, we assume we have this measure c . We now have two things left to show:

Proposition. (\clubsuit) For all $\varphi \in \Gamma^*$, we have $\vdash \varphi \leftrightarrow \tau(\varphi)$

Proof. By induction on $c(\varphi)$.

Base Step. If φ is a proposition p , then we (trivially) have $\vdash p \leftrightarrow p$.

Inductive Step. We consider each possible inductive case, and suppose the claim holds for formulas ψ with smaller $c(\psi)$. The $\neg\varphi$, $\varphi \wedge \psi$, \mathbf{K} , and \mathbf{B} cases all follow from applying the translation, and then applying inductive hypothesis on the subexpression that results from this.

Here are the rest of the cases. Notice that we apply the inductive hypothesis to terms whose c -cost is smaller (this is why we needed the decreasing properties of c before).

$\langle \varphi \rangle_{\text{Hebb}} p$ case. We have

$$tr(\langle \varphi \rangle_{\text{Hebb}} p) = tr(p) = p$$

and so we need to show that

$$\vdash \langle \varphi \rangle_{\text{Hebb}} p \leftrightarrow p$$

but this holds by our propositional reduction axiom.

$\langle \varphi \rangle_{\text{Hebb}} \neg\psi$ case. We have:

$$\begin{aligned} & \vdash \langle \varphi \rangle_{\text{Hebb}} \neg\psi \\ & \leftrightarrow \neg \langle \varphi \rangle_{\text{Hebb}} \psi \quad (\text{by the reduction axiom}) \\ & \leftrightarrow tr(\neg \langle \varphi \rangle_{\text{Hebb}} \psi) \quad (\text{inductive hypothesis}) \\ & = tr(\langle \varphi \rangle_{\text{Hebb}} \neg\psi) \quad (\text{by our translation}) \end{aligned}$$

$\langle \varphi \rangle_{\text{Hebb}} \psi \wedge \rho$ case.. We have:

$$\begin{aligned} & \vdash \langle \varphi \rangle_{\text{Hebb}} (\psi \wedge \rho) \\ & \leftrightarrow \langle \varphi \rangle_{\text{Hebb}} \psi \wedge \langle \varphi \rangle_{\text{Hebb}} \rho \quad (\text{by the reduction axiom}) \\ & \leftrightarrow tr(\langle \varphi \rangle_{\text{Hebb}} \psi \wedge \langle \varphi \rangle_{\text{Hebb}} \rho) \quad (\text{inductive hypothesis}) \\ & = tr(\langle \varphi \rangle_{\text{Hebb}} (\psi \wedge \rho)) \quad (\text{by our translation}) \end{aligned}$$

$\langle \varphi \rangle_{\text{Hebb}} \mathbf{K}$ case.. We have:

$$\begin{aligned} & \vdash \langle \varphi \rangle_{\text{Hebb}} \mathbf{K} \\ & \quad \leftrightarrow \mathbf{K} && \text{(by the reduction axiom)} \\ & \quad \leftrightarrow tr(\mathbf{K}) && \text{(inductive hypothesis)} \\ & \quad = tr(\langle \varphi \rangle_{\text{Hebb}} \mathbf{K}) && \text{(by our translation)} \end{aligned}$$

$\langle \varphi \rangle_{\text{Hebb}} \mathbf{B}$ case.. We have:

$$\begin{aligned} & \vdash \langle \varphi \rangle_{\text{Hebb}} \mathbf{B} \\ & \quad \leftrightarrow \langle \mathbf{B} \rangle && \text{(by the reduction axiom)} \\ & \quad \leftrightarrow tr(\langle \mathbf{B} \rangle) && \text{(inductive hypothesis)} \\ & \quad = tr(\langle \varphi \rangle_{\text{Hebb}} \mathbf{B}) && \text{(by our translation)} \end{aligned}$$

$\langle \varphi \rangle_{\text{Hebb}} \langle \psi \rangle_{\text{Hebb}} \rho$ case.. This case is more interesting. First, notice our translation for this case:

$$tr(\langle \varphi \rangle_{\text{Hebb}} \langle \psi \rangle_{\text{Hebb}} \rho) = tr(\langle \varphi \rangle_{\text{Hebb}} tr(\langle \psi \rangle_{\text{Hebb}} \rho))$$

That is, we translate the inner expression first, then translate the outer expression. This inner $tr(\langle \psi \rangle_{\text{Hebb}} \rho)$ is equivalent to some update-operator-free formula χ :

$$\vdash \chi \leftrightarrow tr(\langle \psi \rangle_{\text{Hebb}} \rho) \leftrightarrow \langle \psi \rangle_{\text{Hebb}} \rho \quad (4.1)$$

(This last equivalence follows from our inductive hypothesis, which we can apply because $\langle \psi \rangle_{\text{Hebb}} \rho$ is a subexpression of $\langle \varphi \rangle_{\text{Hebb}} \langle \psi \rangle_{\text{Hebb}} \rho$.)

What about $tr(\langle \varphi \rangle_{\text{Hebb}} \chi)$? Well, since χ is update-operator-free, this reduces to our previous inductive cases. So we have

$$\vdash tr(\langle \varphi \rangle_{\text{Hebb}} \chi) \leftrightarrow \langle \varphi \rangle_{\text{Hebb}} \chi \quad (4.2)$$

Putting this all together, we have:

$$\begin{aligned} & \vdash \langle \varphi \rangle_{\text{Hebb}} \langle \psi \rangle_{\text{Hebb}} \rho \\ & \quad \leftrightarrow \langle \varphi \rangle_{\text{Hebb}} \chi && \text{(by (4.1))} \\ & \quad \leftrightarrow tr(\langle \varphi \rangle_{\text{Hebb}} \chi) && \text{(by (4.2))} \\ & \quad \leftrightarrow tr(\langle \varphi \rangle_{\text{Hebb}} (tr(\langle \psi \rangle_{\text{Hebb}} \rho))) && \text{(by (4.1))} \\ & \quad \leftrightarrow tr(\langle \varphi \rangle_{\text{Hebb}} \langle \psi \rangle_{\text{Hebb}} \rho) && \text{(by our translation)} \end{aligned} \quad \square \quad \square$$

Theorem 4.8. (Completeness) The logic of Hebbian Learning is completely axiomatized by the base axioms [ref!], along with the above reduction axioms. That is, for all consistent $\Gamma^* \subseteq \mathcal{L}^*$, if $\Gamma^* \models \varphi$ then $\Gamma^* \vdash \varphi$.

Proof. Since our language \mathcal{L}^* has negation, completeness follows from model building in the usual way; this proof is entirely standard. Suppose contrapositively that $\Gamma^* \not\models \varphi$. It follows that $\Gamma^* \vdash \neg\varphi$. So $\Gamma^* \cup \{\neg\varphi\}$ is consistent, and by Theorem [todo—need to modify the construction to make sure the net is fully connected. Quote: “But remember that our nets are also fully connected! So we need to modify the model construction from [34] by introducing a zero weight edge between every pair of previously unconnected nodes. Note that this change does not affect the $\text{\textit{Prop}}$ -structure of the net.”], we have $\mathcal{N} \in \text{Net}$ such that $\mathcal{N} \models \Gamma^* \cup \{\neg\varphi\}$. But then $\mathcal{N} \models \Gamma^*$ yet $\mathcal{N} \not\models \varphi$, which is what we wanted to show. \square

4.1 Example: Building a Neural Network with Learning Constraints

5 Reflections on Interpretability and Alignment

Chapter 6

Expressivity: Measuring the Modeling Power of Neural Networks

1 Introduction

2 A Potpourri of Model Classes

- In this section, I will introduce a number of models in the literature that I will compare neural networks against.
- To make the comparison fair, I need to generalize the language here. All semantics will be over the multi-modal language. You can think of the \Box_i 's as different modalities for different accessibility relations / transition functions (e.g. knowledge vs belief, also done in FOL), or alternately as different per agent in a multi-agent setting.
- Give this language. As an example, mention that our language above of $\text{box}, \Box^\downarrow$ and $[\text{best}]$ is an instance of this language, where the operators $\text{box}, \Box^\downarrow$ and $[\text{best}]$ have additional interaction axioms.
- I is some fixed set of indices.

2.1 Relational Models

A relational model is $\mathcal{M} = \langle W, \{R\}_{i \in I}, V \rangle$, where

- W is some finite set of worlds (or states)
- Each $R_i \subseteq W \times W$ (the accessibility relations)
- $V: \text{Proposition} \rightarrow \mathcal{P}(W)$ (the valuation function)

Define **Rel** to be the class of all such models, and define **Rel**_{S4} to be the class of all such models where R is additionally reflexive and transitive. The semantics for both classes is given by:

$$\begin{aligned} \mathcal{M}, w \models p & \quad \text{iff } w \in V(p) \\ \mathcal{M}, w \models \neg \varphi & \quad \text{iff } \mathcal{M}, w \not\models \varphi \\ \mathcal{M}, w \models \varphi \wedge \psi & \quad \text{iff } \mathcal{M}, w \models \varphi \text{ and } \mathcal{M}, w \models \psi \\ \mathcal{M}, w \models \Box_i \varphi & \quad \text{iff for all } u \text{ with } w R_i u, \mathcal{M}, u \models \varphi \end{aligned}$$

[Mention axioms, soundness, completeness (refer to the appendix!)]

2.2 Plausibility Models

A plausibility model, first introduced in [33], is $\mathcal{M} = \langle W, \{R\}_{i \in \mathbf{I}}, V \rangle$, i.e. the models themselves are just relational models. As before, I assume that W is finite, and as with \mathbf{Rel}_{S4} , each R_i is reflexive and transitive. The key difference is that we interpret $\Box_i \varphi$ to hold in the best (or most plausible) states satisfying φ . Formally, let $\text{best}_{R_i}(S) = \{w \in S \mid \text{for all } u \in S, \neg u R_i w\}$ (the R_i -minimal states over S). We additionally impose the following “smoothness condition” [33] on best_{R_i} :

Postulate 2.1. For all models \mathcal{M} , $i \in \mathbf{I}$, sets S , and all $w \in W$, if $w \in S$ then either $w \in \text{best}_{R_i}(S)$, or there is some $v R_i w$ better than w that is the best, i.e. $v \in \text{best}_{R_i}(S)$.

The new semantics for \Box_i is

$$\mathcal{M}, w \models \Box_i \varphi \text{ iff } w \in \text{best}_{R_i}(\llbracket \varphi \rrbracket)$$

where $\llbracket \varphi \rrbracket = \{u \mid \mathcal{M}, u \models \varphi\}$. In practice, plausibility semantics coexist alongside relational semantics, so I allow some $\Box_i \varphi$ to be given relational semantics instead. Let **Plaus** be the class of all such models. Since we include relational operators, note that $\mathbf{Rel}_{S4} \subseteq \mathbf{Plaus}$.

Any plausibility operator \Box_i picks out a corresponding conditional: $\Box_i \varphi \rightarrow \psi$ reads “the best φ are ψ ,” which in the KLM tradition is the semantics for the conditional $\varphi \Rightarrow \psi$.

[Mention axioms, soundness, completeness (refer to the appendix!)]

2.3 Social Network Models

- Introduce social network models, an example with a [DIAGRAM] would be nice!
- In these social network logics [4; 8; 17], nodes in the graph represent individual agents, and each formula is mapped to the set of agents that adopt a certain social attitude. Agents influence each other, and the spread of their attitudes is modeled much in the same way as forward propagation of a signal in a neural network.
- Give a concrete social network logic: Social majority. Make sure to emphasize that social majority is one of many (!) choices, and is a relatively simple choice to model.

- Both kinds of models use fundamentally the same approach (“This work shares essentially the same premise and techniques as neural network semantics”): distributed information over several connected nodes, modal operator interpreted as the fixed-point of some diffusion
- But the two approaches still differ in interesting ways. First, in some sense the two semantics are operating on different “levels”: social networks model interactions between multiple agents, whereas neural networks model interactions between components of the same (single) agent. Second, the two differ in subject matter. Social network semantics focuses on different social links between agents, and how these links change [4]. Neural network semantics, my own work included, instead focuses on inferences and updates inspired by artificial and natural neural networks.

2.4 Neighborhood Models

A neighborhood model is $\mathcal{M} = \langle W, \{N_i\}_{i \in I}, V \rangle$, where W and V are as before and each $N_i: W \rightarrow \mathcal{P}(\mathcal{P}(W))$ is an accessibility *function*. The intuition is that N_i maps each state w to the “formulas” (sets of states) that hold at w . Define **Nbhd** to be the class of all neighborhood models.

Moreover, the *core* of N is $\cap N(x) = \bigcap_{X \in N(w)} X$. As with **Rel**, let **Nbhd**_{S4} be the class of all neighborhood models that are additionally reflexive ($\forall w, w \in \cap N(w)$) and transitive ($\forall w$, if $X \in N(w)$ then $\{v \mid X \in N(v)\} \in N(w)$).

The semantics for both classes is the same as the previous classes, except the \Box_i case is now:

$$\mathcal{M}, w \Vdash \Box_i \varphi \text{ iff } \llbracket \varphi \rrbracket \in N_i(w)$$

where again $\llbracket \varphi \rrbracket = \{u \mid \mathcal{M}, u \Vdash \varphi\}$.

3 Measuring Expressive Power through Translation

- To compare the modeling power of neural networks with other logic models, I need to pick a measure of expressivity.
- I will consider logics as languages paired with a class of models $(\mathcal{L}, \mathcal{C})$ (also known

as *institutions* in Institution Theory [Cite Institution theory, elaborate a bit more!]). I'll use this to compare the expressive power of neural networks (using both Modal and Conditional logics) against other models over those languages

- I will measure expressivity through *translations* between two logics. Definition: There is a translation (aka *infomorphism*) from $(\mathcal{L}_1, \mathcal{C}_1)$ into $(\mathcal{L}_2, \mathcal{C}_2)$ if there exist $f: \mathcal{C}_2 \rightarrow \mathcal{C}_1$, $\tau: \mathcal{L}_1 \rightarrow \mathcal{L}_2$ such that for all $\varphi \in \mathcal{L}_1, \mathcal{M} \in \mathcal{C}_2$

$$f(\mathcal{M}) \models \varphi \text{ iff } \mathcal{M} \models \tau(\varphi)$$

- Call the translation *strict* if there is not a translation in the converse direction.
- [Cite infomorphism, either from the book “Information Flow: The Logic of Distributed Systems” or alternatively the book “Categories, Allegories”]
- [DIAGRAM]
- “flipped/contravariant”, notice that we construct the model backwards from \mathcal{C}_2 to \mathcal{C}_1 .
- Proposition: If there is a translation from $(\mathcal{L}_1, \mathcal{C}_1)$ to $(\mathcal{L}_2, \mathcal{C}_2)$, this means that $(\mathcal{L}_1, \mathcal{C}_1)$ is at least as *general* (i.e. requires fewer axioms, i.e. $\text{Th}(\mathcal{L}_1, \mathcal{C}_1) \subseteq \text{Th}(\mathcal{L}_2, \mathcal{C}_2)$)
- This also means, in order to show that there is *no* translation, all we need to do is show $\text{Th}(\mathcal{L}_1, \mathcal{C}_1) \not\subseteq \text{Th}(\mathcal{L}_2, \mathcal{C}_2)$, i.e. find an axiom $\varphi \in \text{Th}(\mathcal{L}_1, \mathcal{C}_1)$ such that $\varphi \notin \text{Th}(\mathcal{L}_2, \mathcal{C}_2)$.
- Give an example that's a sort of tutorial for comparing the modeling power of classes of modal logic: **Rel**_{S4} and **Nbhd**_{S4}. Show that there is a translation from (Modal, **Rel**_{S4}) into (Modal, **Nbhd**_{S4}), but also show that there is *not* one the other way around. (This example teaches us how to give a translation, but also how to show one doesn't exist!)

4 Expressive Power of the Base Neural Network Semantics

- Let's now do model translations to get at the expressivity of neural networks (over Modal and Conditional logic). Here's a hierarchy of the models above, to start:
- To make the comparison with neural networks fair, I will only consider the reflexive and transitive variants **Rel**_{S4}, **Nbhd**_{S4} of relational and neighborhood models.
- [DIAGRAM]
- The translations from **Nbhd**_{S4} to **Plaus** and from **Plaus** to **Rel**_{S4} are folklore. Instead of giving these translations, I will instead translate from **Net** to **Rel**_{S4} and from **Nbhd**_{S4} to

Net in order to explicitly show how to translations involving neural networks (neural network model constructions). The equivalence between **Plaus** and **Net** is already known, but the backwards direction has never been proven with an explicit model construction. So although the results in this section are already known, the proofs I give here are totally new.

- First, show translation from **Net** to **Rel**_{S4}, and show there is no translation the other way around (axiom in **Rel**_{S4} that is not an axiom in **Net**)
- Next, show translation from **Nbhd**_{S4} to **Net**, and show there is no translation the other way around (axiom in **Net** that is not an axiom in **Nbhd**_{S4})
- Next, from the model-building construction in the completeness chapter, we have a translation from **Net** to **Plaus**.
- Explain that completeness implies that *in principle* we have a translation the other way (**Plaus** to **Net**), but it doesn't actually give the explicit model building procedure! Here is where I will give my own.
- Make a note here about the social majority logic above: There is a strict translation from **Net** to the social majority logic (**Net** is more general, in the sense that it requires fewer axioms).
- Give a brief note here on the expressive power of the conditional logic **Net** vs the modal logic **Net**.

Proposition 4.1. (\hookrightarrow) There is a strict translation from (Modal, **Net**) to (Modal, **Rel**_{S4}).

Proof. \square

\square

Proposition 4.2. (\hookrightarrow) There is a strict translation from (Modal, **Nbhd**_{S4}) to (Modal, **Net**).

Proof. \square

\square

Proposition 4.3. There is a translation from (Modal, **Net**) to (Modal, **Plaus**).

Proof. [This is just a corollary of completeness: Mention that we just use the NAND construction from the Completeness proof.] \square

Theorem 4.4. (\hookrightarrow) There is a translation from (Modal, **Plaus**) to (Modal, **Net**)

Proof. [This is the hard part!! It deserves to be a theorem, imo.] \square

\square

Theorem 4.5. (\clubsuit) There is a strict translation from (Modal, **Net**) to the social majority logic [give it a name]

Proof. \square

Theorem 4.6. (\clubsuit) There is a strict translation from (Modal, **Net**) to (Conditional, **Net**). [does it go this way, or the other way??]

Proof. \square

5 Expressive Power of Neural Network Update

- Reference all work in dynamic logic where some operator is shown to be not translatable into another! (Read these, they may help with the proof.) List so far: [4]
- [My thinking here is a mess. I need to sort out exactly what questions about update I would like to answer (very general? very specific? what do translations even look like in this context? what's helpful to ask? what's even answerable?)]

Definition 5.1. Let $\mathcal{L}_1, \mathcal{L}_2$ be languages with a single dynamic operator and $\mathcal{C}_1, \mathcal{C}_2$ be model classes. There is a *dynamic translation* from $(\mathcal{L}_1, \mathcal{C}_1)$ into $(\mathcal{L}_2, \mathcal{C}_2)$ if there exist $f: \mathcal{C}_2 \rightarrow \mathcal{C}_1$, $\tau: \mathcal{L}_1 \rightarrow \mathcal{L}_2$, and

$$F: (\mathcal{C}_2 \rightarrow \mathcal{L}_2 \rightarrow \mathcal{C}_2) \rightarrow (\mathcal{C}_1 \rightarrow \mathcal{L}_1 \rightarrow \mathcal{C}_1)$$

i.e. “update to update”, such that for all $\varphi \in \mathcal{L}_1$, $\mathcal{M} \in \mathcal{C}_2$, and $\heartsuit: \mathcal{C}_2 \rightarrow \mathcal{L}_2 \rightarrow \mathcal{C}_2$,

$$f(\mathcal{M}), F(\heartsuit) \models \varphi \text{ iff } \mathcal{M}, \heartsuit \models \tau(\varphi)$$

As before, we call the translation *strict* if there is not a translation in the converse direction.

The following theorem says that any update over neural networks corresponds to an update over plausibility models (and conversely). Actually, this is just a straightforward consequence of the translations between $(\mathcal{L}_{\text{best}}, \mathbf{Plaus})$ and $(\mathcal{L}_{\text{best}}, \mathbf{Net})$.

Theorem 5.2. There is a dynamic translation from $(\mathcal{L}_{\text{Update}}, \mathbf{Plaus})$ into $(\mathcal{L}_{\text{Update}}, \mathbf{Net})$, and conversely.

Proof. I will show the (\rightarrow) direction, since the converse is similar. Let $\tau: \mathcal{L}_{\text{Update}} \rightarrow \mathcal{L}_{\text{Update}}$ just be the identity $\tau(\varphi) = \varphi$. Since we have a translation from $(\mathcal{L}_{\text{best}}, \mathbf{Plaus})$ into $(\mathcal{L}_{\text{best}}, \mathbf{Net})$ and

conversely, by [CBS—todo, I need to prove this] this gives us a $\mathcal{L}_{\text{best}}$ -truth-preserving bijection $f: \mathbf{Net} \rightarrow \mathbf{Plaus}$. Since f is bijective, it has an inverse $f^{-1}: \mathbf{Plaus} \rightarrow \mathbf{Net}$. We now need a function

$$F: (\mathbf{Net} \rightarrow \mathcal{L}_{\text{Update}} \rightarrow \mathbf{Net}) \rightarrow (\mathbf{Plaus} \rightarrow \mathcal{L}_{\text{Update}} \rightarrow \mathbf{Plaus})$$

In other words, F takes as input an arbitrary neural network update $\heartsuit: \mathbf{Net} \rightarrow \mathcal{L}_{\text{Update}} \rightarrow \mathbf{Net}$ and simulates it as an update over plausibility models. We define F as follows: $F(\heartsuit, \mathcal{M}, \varphi) = \clubsuit$, where $\clubsuit: \mathbf{Plaus} \rightarrow \mathcal{L}_{\text{Update}} \rightarrow \mathbf{Plaus}$ is given by

$$\clubsuit(\mathcal{M}, \varphi) = f(\heartsuit(f^{-1}(\mathcal{M}), \varphi))$$

\clubsuit simulates \heartsuit by first converting \mathcal{M} to a neural network, then it applies the update \heartsuit with φ , and then converts the result back into an equivalent plausibility model. Visually:

[Draw diagram!!!]

Now we just need to check that this is a dynamic translation, i.e.

$$f(\mathcal{M}), \clubsuit \models \varphi \text{ iff } \mathcal{M}, \heartsuit \models \varphi$$

Let's proceed by induction on $\varphi \in \mathcal{L}_{\text{Update}}$. The propositional and boolean cases are easy. The \mathbf{A} , \mathbf{box} , \square^\perp , and $[\text{best}]$ cases are all similar: None of these cases depend on the choice of update, so we just apply our inductive hypothesis along with the fact that (τ, f) is a translation from $(\mathcal{L}_{\text{best}}, \mathbf{Plaus})$ into $(\mathcal{L}_{\text{best}}, \mathbf{Net})$. Finally, consider the update case:

[P] Case. We have

$$\begin{aligned} f(\mathcal{M}), \clubsuit \models [P]\varphi & \text{ iff } \clubsuit(f(\mathcal{M}), P), \clubsuit \models \varphi && \text{(by the semantics for } [P]) \\ & \text{ iff } f(\heartsuit(f^{-1}(f(\mathcal{M})), \varphi)), \clubsuit \models \varphi && \text{(by definition of } \clubsuit) \\ & \text{ iff } f(\heartsuit(\mathcal{M}, \varphi)), \clubsuit \models \varphi && \text{(since } f^{-1} \text{ is the inverse of } f) \\ & \text{ iff } \heartsuit(\mathcal{M}, \varphi), \heartsuit \models \varphi && \text{(by inductive hypothesis)} \\ & \text{ iff } \mathcal{M}, \heartsuit \models [P]\varphi && \text{(by the semantics for } [P]) \end{aligned}$$

This concludes the proof. □

My original goal was to explore what “classical” updates correspond to neural network updates such as Hebb*, and conversely what neural updates correspond to plausibility updates such as Cond, Lex, and Consr. This theorem doesn't seem to clarify that; it's constructive (we *do* in fact build the corresponding updates), but by translating back and forth at the static level we don't define the update in the original updates “native environment.” [Elaborate/clarify this point]. For the remainder of this section, I will put in the extra work to see what Hebb*, Cond, Lex, and Consr “look like” on the other side.

Definition 5.3. [Define the plausibility update that simulates Hebb*]

Definition 5.4. [Define the neural update that simulates Cond]

Definition 5.5. [Define the plausibility update that simulates Lex]

Definition 5.6. [Define the plausibility update that simulates Consr]

6 Neural Networks and Descriptive Complexity

- References: [29], [37], [20], ... (find more, including other Finite Model Theory books, Fagin's original result, and more recent results that may be relevant in descriptive complexity theory)
- Introduction to descriptive complexity, Fagin/Immerman style (base it on the book Elements of Finite Model Theory by Libkin, since I personally find this the easiest to follow.)
- Descriptive complexity largely takes the shared class of models for granted, translations are largely syntactic (translations from one language to another)
- The way to measuring expressive power is through definability. Define problems (boolean queries where inputs are models over a certain class. Then give definition of definability in this setting (we have no free variables, so we don't have to worry about tuples of satisfying elements, etc.):

$$\text{Def}(\mathcal{L}, \mathcal{C}) = \{P \subseteq \mathcal{C} \mid \text{There is some } \varphi \in \mathcal{L} \text{ such that } \mathcal{M} \in \mathcal{C}, \mathcal{M} \in P \text{ iff } \mathcal{M} \models \varphi\}$$

- Give a standard example: First-order logic can define strictly more than Modal logic.
- Note that problems P are defined over a *specific* class of models! If we want to compare different model classes, we need a notion of *translating* P to the other class. If $P \in \mathcal{C}_1$ and we want to translate it to \mathcal{C}_2 :

$$f^{-1}(P) = \{\mathcal{M} \in \mathcal{C}_2 \mid f(\mathcal{M}) \in P\}$$

- [DIAGRAM] A picture showing how f^{-1} maps back to \mathcal{C}_2 would be nice here!
- How do we connect the ideas of translation-expressivity and definability-expressivity?

The key result is this:

Theorem 6.1. Suppose there is a translation f, τ from $(\mathcal{L}_1, \mathcal{C}_1)$ to $(\mathcal{L}_2, \mathcal{C}_2)$. For all problems $P \subseteq \mathcal{C}_1$, if $P \in \text{Def}(\mathcal{L}_1, \mathcal{C}_1)$ then $f^{-1}(P) \in \text{Def}(\mathcal{L}_2, \mathcal{C}_2)$.

Proof. Suppose P is defined by $\varphi \in \mathcal{L}_1$. I will show that $f^{-1}(P)$ (that is, P translated over to \mathcal{C}_2) is defined by $\tau(\varphi)$ (that is, φ translated over to \mathcal{L}_2).

Let $\mathcal{M} \in \mathcal{C}_2$ be any model. Our goal is to show that $\mathcal{M} \in f^{-1}(P)$ iff $\mathcal{M} \models \tau(\varphi)$. Well,

$$\begin{aligned} \mathcal{M} \in f^{-1}(P) & \text{ iff } f(\mathcal{M}) \in P \quad (\text{by definition of } f^{-1}) \\ & \text{ iff } f(\mathcal{M}) \models \varphi \quad (\text{since } \varphi \text{ defines } P, \text{ and } f(\mathcal{M}) \in \mathcal{C}_1) \\ & \text{ iff } \mathcal{M} \models \tau(\varphi) \quad (\text{by translation } f, \tau) \end{aligned}$$

□

- I may also need the converse fact, which says that if there is *not* a translation, the definability sets cannot be equal.
- Revisit our example from before: Show that $\text{Def}(\text{Modal}, \mathbf{Nbhd}) \subset \text{Def}(\text{Modal}, \mathbf{Rel})$. But also walk through what this means: What property is definable in $(\text{Modal}, \mathbf{Rel})$ but not $(\text{Modal}, \mathbf{Nbhd})$? (This has me a bit stumped and confused...)
- As a consequence of this connection, we can embed the neural network translations above into the descriptive complexity hierarchy
- Corollaries for each of the inclusions, this time stated for $\text{Def}(\text{Modal}, \mathbf{Net})$ (using strict subset).
- **[BIG DIAGRAM]**, situating the results above into the descriptive complexity hierarchy (these are all Modal or Conditional, sub-FOL. It is currently an open problem to find sound and complete neural network semantics for FOL, but once we have this it is possible to do this for the classes above Modal.)

7 Reflections on the Complexity Hierarchy

[This is where I give “the chart”, and make higher-level connections to descriptive complexity of neural networks, neural nets as automata, FLaNN work, and give my own personal long-term vision for complexity theory. (I can also discuss *dynamic* complexity here, which is imo underrated in complexity work)]

[Integrate the following into this chapter!]

The Graph-Reachability Construction. I will show here how the more general Clos operator can be used to simulate Reach and Reach^\downarrow . Suppose we are given a graph $\langle N, E \rangle$ and a valuation function V . How can we build a neural network whose closure Clos is graph-reachability Reach? We want to build a net:

$$\mathcal{N} = \langle N, \text{bias}, E, W, A, \eta, V \rangle$$

[what to do about bias and η ?] For the weights, pick

$$W(m, n) = \begin{cases} 1 & \text{if } mEn \\ 0 & \text{otherwise} \end{cases}$$

Then pick the activation function $A(x) = 1$ iff $x > 0$. Recall that $n \in F_{S_0}(S)$ iff $n \in S_0$ or is activated by its predecessors in S . In this case, $n \in F_{S_0}(S)$ whenever $n \in S_0$ or at least one E -predecessor m of n is in S . I call this the *graph-reachability construction* because the closure $\text{Clos}(S)$ produces exactly those nodes graph-reachable from S :

Proposition 7.1. For all states $S \in \text{State}$, $\text{Clos}(S) = \text{Reach}(S)$.

Proof. First, the (\subseteq) direction. Let $n \in \text{Clos}(S) = F_S^k(S)$ for some $k \in \mathbb{N}$. By induction on k .

Base Step. $n \in F_S^0(S) = S$. So there is a trivial E_i -path (length=0) from $n \in S$ to itself.

Inductive Step. Let $k \geq 0$. We have $n \in F_S^k(S) = F_S(F_S^{k-1}(S))$. By construction of F_S , we have two cases: Either $n \in F_S^{k-1}(S)$ or at least one E -predecessor x of n is in $F_S^{k-1}(S)$. In the first case, our inductive hypothesis gives a path from some $m \in S$ to n . In the second case, our inductive hypothesis gives a path from some $m \in S$ to x . But since xEn , we can extend this path to be from m to n .

As for the (\supseteq) direction, suppose there is an E -path from some $m \in S$ to n . We proceed by induction on the length of that path.

Base Step. The path is trivial, i.e. has length 0. So $n \in S$. But $S = F_i^0(S) \subseteq \text{Clos}_i(S)$, and so $n \in \text{Clos}_i(S)$.

Inductive Step. Say the path is of length $l \geq 0$. Let x be some immediate E_i -predecessor of n . By the inductive hypothesis, $x \in \text{Clos}_i(S)$, and so $x \in F_i^k(S)$ for some natural k . But since x is an E_i -predecessor of n , by construction of F_i , $n \in F_i(F_i^k(S)) = F_i^{k+1}(S)$. Since $\text{Clos}_i(S)$ is a closure, it includes $F_i^{k+1}(S)$. So $n \in \text{Clos}_i(S)$. \square

As for Reach^\downarrow , we first *reverse* the edges E , and then do the graph-reachability construction.

In other words, let $mE'n$ iff nEm ,

$$W(m, n) = \begin{cases} 1 & \text{if } mE'n \\ 0 & \text{otherwise} \end{cases}$$

and pick A, η, V the same as above. For this construction, the closure $\text{Clos}(S)$ produces exactly those nodes which reach some node in S . The proof is similar to the proof for Reach . [Check that that's actually true!!]

Proposition 7.2. For all states $S \in \text{State}$, $\text{Clos}(S) = \text{Reach}^\downarrow(S)$.

The Social Majority Construction. [Introduce the idea of social networks here if I haven't already, and mention the “social majority” propagation/diffusion (tell it slowly, like a story). I will show that our neural networks can simulate this simple social majority operator (make the social majority operator a bit more formal).]

As before, we want to build a neural network \mathcal{N} where the graph $\langle N, E_i \rangle$, bias, and evaluation V are given. This time, pick $W_i(m, n) = \frac{1}{|\text{preds}(n)|}$, and then pick $A_i(x) = 1$ iff $x \geq \frac{1}{2}$. Visually, for each node n and its predecessors m_1, \dots, m_r we have

[DIAGRAM!]

This gives us $n \in F_{S_0}(S)$ if $n \in S_0$ or if the majority (more than half) of E -predecessors are in S . In this case, the closure Clos can be interpreted as the diffusion of an opinion or attitude through a social network. This is one of the choices that [8] consider for modelling influence in social networks. [this paragraph is a bit terse now that I'm writing a longer version of this.]

Chapter 7

Conclusions

[I like the way Levin Hornischer wrote his: A summary, followed by a list of results, followed by a list of open questions]

Results

1.

Open Questions

1.

Appendix A

Details for the Logic of [best]

A.1 Syntax and Semantics

[Emphasize that this appendix is all novel work!!! To me it's an annoying technical detail, but it deserves to be written up as a paper in its own right.]

I will prove soundness and completeness for the language $\mathcal{L}_{\text{best}}$:

$$\varphi, \psi := p \mid \neg\varphi \mid \varphi \wedge \psi \mid \mathbf{A}\varphi \mid \Box\varphi \mid \Box^\downarrow\varphi \mid [\text{best}]\varphi$$

Actually, the model construction I give works for the language without \mathbf{A} ; adding $\mathbf{A}\varphi$ to the logic of \Box alone is a known result in modal logic. [Cite this! Who proved this first?]

[I think it's okay to repeat syntax + semantics here, since otherwise the reader would have to flip to the beginning...]

[Talk about the expressive power of this language: show how this logic of $\mathbf{A}, \Box, \Box^\downarrow, [\text{best}]$ can express many of the other ways of expressing defeasible reasoning—and mention which ones it cannot express.]

Axioms for \Box: (Dual) $\Diamond\varphi \leftrightarrow \neg\Box\neg\varphi$ (Distr) $\Box(\varphi \rightarrow \psi) \rightarrow (\Box\varphi \rightarrow \Box\psi)$ (Refl) $\Box\varphi \rightarrow \varphi$ (Trans) $\Box\varphi \rightarrow \Box\Box\varphi$	Axioms for \Box^\downarrow: (Dual) $\Diamond^\downarrow\varphi \leftrightarrow \neg\Box^\downarrow\neg\varphi$ (Distr) $\Box^\downarrow(\varphi \rightarrow \psi) \rightarrow (\Box^\downarrow\varphi \rightarrow \Box^\downarrow\psi)$ (Back) $\varphi \rightarrow \Box\Diamond^\downarrow\varphi$ (Forth) $\varphi \rightarrow \Box^\downarrow\Diamond\varphi$
Axioms for [best]: (Dual) $\langle\text{best}\rangle\varphi \leftrightarrow \neg[\text{best}]\neg\varphi$ (Refl) $[\text{best}]\varphi \rightarrow \varphi$ (Trans) $[\text{best}]\varphi \rightarrow [\text{best}][\text{best}]\varphi$ (Smooth) $\varphi \wedge \neg[\text{best}]\varphi \rightarrow \Diamond^\downarrow[\text{best}]\varphi$ (Up) $[\text{best}]\varphi \wedge \psi \rightarrow \Box([\text{best}]\varphi \rightarrow \psi)$ (Down) $[\text{best}]\varphi \wedge \psi \rightarrow \Box^\downarrow(\varphi \rightarrow \psi)$	Axioms for \mathbf{A}: (Dual) $\mathbf{E}\varphi \leftrightarrow \neg\mathbf{A}\neg\varphi$ (Distr) $\mathbf{A}(\varphi \rightarrow \psi) \rightarrow (\mathbf{A}\varphi \rightarrow \mathbf{A}\psi)$ (Refl) $\mathbf{A}\varphi \rightarrow \varphi$ (5) $\mathbf{E}\varphi \rightarrow \mathbf{A}(\mathbf{E}\varphi)$ (Interact) $\mathbf{A}\varphi \rightarrow \Box\varphi$
	Rules of Inference: (MP) From $\vdash\varphi \rightarrow \psi$ and $\vdash\varphi$ we can infer $\vdash\psi$ (Nec) From $\vdash\varphi$ we can infer $\vdash\Box\varphi$ for each $\Box \in \{\mathbf{A}, \Box, \Box^\downarrow\}$

Figure A.2.1. Axioms and rules of inference for [todo]

A.2 Proof of Soundness

[First, give Inclusion & Idempotence properties for best_\prec ! And of course we have the Smoothness Condition from before. (recall it)]

The proof system for the modal language is exactly the same as for the neural semantics (see Section []). I'll repeat it here for reference:

Definition A.2.1. The proof system for the base modal logic over $\mathcal{L}_{\text{best}}$ is given as follows: $\vdash\varphi$ iff either φ is valid in propositional logic, or φ is one of the axioms listed in Figure A.2.1, or φ follows from some previously obtained formulas by one of the inference rules.

The axioms for \Box and \mathbf{A} form “*MLU*” (modal logic with the universal quantifier), and this is the standard complete axiomatization of this logic [Cite! Johan van Benthem mentions it in *Modal Logic for Open Minds*]. The logic of \Box is just *S4*, and \mathbf{A} is just *S5* with an additional interaction axiom $\mathbf{A}\varphi \rightarrow \Box\varphi$ stating that if φ holds everywhere, then it holds everywhere above the current world.

The axioms for \Box^\downarrow come from temporal logic—[talk about Future and Past modalities, our **(Back)** and **(Forth)** axioms are exactly the ones needed.] In fact, the proof of completeness for \Box^\downarrow will require some tricks from this temporal logic. [cite the book on modal temporal logic... it's been a while since I've read it].

The axioms for $[\text{best}]$ are, to my knowledge, totally new! $[\text{best}]$ is a non-normal modal operator, in the sense that it doesn't satisfy a **(Distr)** axiom or have a valid **(Nec)** rule. Instead, $[\text{best}]$ satisfies the axioms **(Smooth)**, **(Up)**, and **(Down)**. **(Smooth)** is an explicit statement of the Smoothness Condition—the fact that we can express it in our logic is an interesting feature of this defeasible logic (I will discuss this more later). **(Up)** says that if the current state w is a best- φ state, then every state above w that is a best- φ state must in fact be w (so any ψ that holds at w must hold at this new state as well). **(Down)** says that if the current state w is a best- φ state, then every state below w where φ holds at all must in fact be w (so any ψ that holds at w must hold at this new state as well).

Definition A.2.2. If $\Gamma \subseteq \mathcal{L}_{\text{best}}$ is a set of formulas and $\varphi \in \mathcal{L}_{\text{best}}$ a formula, then $\Gamma \vdash \varphi$ whenever there are finitely many $\psi_1, \dots, \psi_k \in \Gamma$ such that $\vdash \psi_1 \wedge \dots \wedge \psi_k \rightarrow \varphi$.

Definition A.2.3. A set $\Gamma \subseteq \mathcal{L}_{\text{best}}$ is *consistent* iff $\Gamma \not\vdash \perp$. Γ is *maximally consistent* if Γ is consistent and for all $\varphi \in \mathcal{L}_{\text{best}}$ either $\varphi \in \Gamma$ or $\varphi \notin \Gamma$.

Theorem A.2.4. (\clubsuit Soundness for $\mathcal{L}_{\text{best}}$ over \models_{Plaus}) These rules and axioms are sound for plausibility models; for all consistent $\Gamma \subseteq \mathcal{L}_{\text{best}}$ and $\varphi \in \mathcal{L}_{\text{best}}$, if $\Gamma \vdash \varphi$ then $\Gamma \models_{\text{Plaus}} \varphi$.

Proof. Suppose $\Gamma \vdash \varphi$. That is, there are finitely many $\psi_1, \dots, \psi_k \in \Gamma$ such that $\vdash \psi_1 \wedge \dots \wedge \psi_k \rightarrow \varphi$, which in turn means (by **(MP)**) that if $\vdash \psi_1, \dots, \vdash \psi_k$, then $\vdash \varphi$. Now let $\mathcal{M} \models \Gamma$. In particular, this means $\mathcal{M} \models \psi_1, \dots, \psi_k$. I now need to show that $\mathcal{M} \models \varphi$. Since $\vdash \varphi$, φ is itself an axiom or follows from previously obtained formulas by the inference rules. In order to prove $\mathcal{M} \models \varphi$, it's enough to show that the axioms and rules of inference are valid (hold for all $\mathcal{M} \in \mathbf{Plaus}$ at all states $w \in W$ whatsoever). The propositional axioms, \Box axioms, \Box^\downarrow axioms, **A** axioms, **(MP)** and **(Nec)** are known to be sound [cite multiple for each], and our plausibility models don't change the semantics of $\Box, \Box^\downarrow, \mathbf{A}$, so I can safely skip these. As for the $[\text{best}]$ axioms, let $\mathcal{M} \in \mathbf{Plaus}, w \in W$.

(Dual) for $[\text{best}]$. This holds by definition of $\langle \text{best} \rangle \varphi$.

(Ref) for [best]. Suppose $\mathcal{M}, w \Vdash [\text{best}]\varphi$. So $w \in \text{best}_<(\llbracket \varphi \rrbracket_{\mathcal{M}})$. By Inclusion of $\text{best}_<$, $w \in \llbracket \varphi \rrbracket_{\mathcal{M}}$. And so $\mathcal{M}, w \Vdash \varphi$.

(Trans) for [best]. Suppose $\mathcal{M}, w \Vdash [\text{best}]\varphi$. So $w \in \text{best}_<(\llbracket \varphi \rrbracket_{\mathcal{M}})$. By Idempotence of $\text{best}_<$, $w \in \text{best}_<(\text{best}_<(\llbracket \varphi \rrbracket_{\mathcal{M}}))$. And so $\mathcal{M}, w \Vdash [\text{best}][\text{best}]\varphi$.

(Smooth) for [best]. Suppose $\mathcal{M}, w \Vdash \varphi$ and $\mathcal{M}, w \Vdash \neg[\text{best}]\varphi$. By the semantics, this means $w \in \llbracket \varphi \rrbracket_{\mathcal{M}}$, but $w \notin \text{best}_<(\llbracket \varphi \rrbracket_{\mathcal{M}})$. But then the Smoothness Condition says there must be some $v < w$ better than w that is the best, i.e., $v \in \text{best}_<(\llbracket \varphi \rrbracket_{\mathcal{M}})$. By the semantics for [best], $\mathcal{M}, v \Vdash [\text{best}]\varphi$. But since v was picked arbitrarily, and $v \leq w$, we have $\mathcal{M}, w \Vdash \Diamond^{\downarrow}[\text{best}]\varphi$.

(Up) for [best]. Suppose $\mathcal{M}, w \Vdash [\text{best}]\varphi$ and $\mathcal{M}, w \Vdash \psi$. By the semantics, $w \in \text{best}_<(\llbracket \varphi \rrbracket)$ (which also means $w \in \llbracket \varphi \rrbracket$). Now let $u \in W$ with $w \leq u$, and suppose $\mathcal{M}, u \Vdash [\text{best}]\varphi$, i.e., $u \in \text{best}_<(\llbracket \varphi \rrbracket)$. Since $u \in \text{best}_<(\llbracket \varphi \rrbracket)$ and $w \in \llbracket \varphi \rrbracket$, definition of $\text{best}_<$ we have $\neg w < u$. Putting $w \leq u$ and $\neg w < u$ together gives us $w = u$, which implies $\mathcal{M}, u \Vdash \psi$. Since u was picked arbitrarily, $\mathcal{M}, w \Vdash \Box([\text{best}]\varphi \rightarrow \psi)$.

(Down) for [best]. Suppose $\mathcal{M}, w \Vdash [\text{best}]\varphi$ and $\mathcal{M}, w \Vdash \psi$. By the semantics, $w \in \text{best}_<(\llbracket \varphi \rrbracket)$ (which also means $w \in \llbracket \varphi \rrbracket$). Now let $u \in W$ with $u \leq w$, and suppose $\mathcal{M}, u \Vdash \varphi$. Since $w \in \text{best}_<(\llbracket \varphi \rrbracket)$ and $u \in \llbracket \varphi \rrbracket$, by definition of $\text{best}_<$ we have $\neg u < w$. Putting $u \leq w$ and $\neg u < w$ together gives us $u = w$, which implies $\mathcal{M}, u \Vdash \psi$. Since u was picked arbitrarily, $\mathcal{M}, w \Vdash \Box^{\downarrow}(\varphi \rightarrow \psi)$. \square

A.3 Model Building and Completeness

Lemma A.3.1. (Lindenbaum's Lemma [\[cite!\]](#)) We can extend any consistent set Γ to a maximally consistent set $\Delta \supseteq \Gamma$.

Proposition A.3.2. Let Σ, Δ be maximally consistent. The following are equivalent:

1. $\Box\varphi \in \Sigma$ implies $\varphi \in \Delta$
2. $\Box^{\downarrow}\varphi \in \Delta$ implies $\varphi \in \Sigma$

Proof. Suppose (1) holds, and let $\Box^{\downarrow}\varphi \in \Delta$. For contradiction, suppose $\varphi \notin \Sigma$. Since Σ is maximally consistent, $\neg\varphi \in \Sigma$. Applying the **(Back)** axiom, we get $\Box \Diamond^{\downarrow}\neg\varphi \in \Sigma$, i.e. $\Box\neg\Box^{\downarrow}\varphi \in$

Σ . By (1), $\neg \Box \downarrow \varphi \in \Delta$, i.e. $\Box \downarrow \varphi \notin \Delta$. But this contradicts $\Box \downarrow \varphi \in \Delta$!

Now suppose (2) holds, and suppose $\Box \varphi \in \Sigma$. For contradiction, suppose $\varphi \notin \Delta$. Since Δ is maximally consistent, $\neg \varphi \in \Delta$. Applying the **(Forth)** axiom, we get $\Box \downarrow \neg \varphi \in \Delta$, i.e. $\Box \downarrow \neg \Box \varphi \in \Delta$. By (2), $\neg \Box \varphi \in \Sigma$, i.e. $\Box \varphi \notin \Sigma$. But this contradicts $\Box \varphi \in \Sigma$! \square

Proposition A.3.3. Let Δ be consistent, and suppose $\Diamond \downarrow [\text{best}] \varphi \in \Delta$. Then the set

$$\Delta' = \{\psi \mid \Box \downarrow \psi \in \Delta\} \cup \{[\text{best}] \varphi\}$$

is consistent.

Proof. Suppose for contradiction that Δ' is inconsistent. Then $\Delta' \vdash \neg [\text{best}] \varphi$. By definition of \vdash , there must be finitely many $\psi_1, \dots, \psi_n \in \Delta'$ such that $\vdash (\psi_1 \wedge \dots \wedge \psi_n) \rightarrow \neg [\text{best}] \varphi$. By the **(Nec)** rule for $\Box \downarrow$, $\vdash \Box \downarrow ((\psi_1 \wedge \dots \wedge \psi_n) \rightarrow \neg [\text{best}] \varphi)$. Then by **(Distr)** for $\Box \downarrow$, $\vdash \Box \downarrow (\psi_1 \wedge \dots \wedge \psi_n) \rightarrow \Box \downarrow \neg [\text{best}] \varphi$. Now, by construction of Δ' , $\Box \downarrow \psi_1, \dots, \Box \downarrow \psi_n \in \Delta$. So $\Box \downarrow \psi_1 \wedge \dots \wedge \Box \downarrow \psi_n \in \Delta$. Now, with a bit of work with the **(Distr)** axiom (see [\[cite modal logic book or notes\]](#)), we have

$$\vdash (\psi_1 \wedge \dots \wedge \psi_n) \rightarrow \neg [\text{best}] \varphi \text{ implies } \vdash \Box \downarrow \psi_1 \wedge \dots \wedge \Box \downarrow \psi_n \rightarrow \Box \downarrow \neg [\text{best}] \varphi$$

which gives us $\Box \downarrow \neg [\text{best}] \varphi \in \Delta$. By **(Dual)**, we have $\neg \Diamond \downarrow [\text{best}] \varphi \in \Delta$, which contradicts our hypothesis $\Diamond \downarrow [\text{best}] \varphi \in \Delta$ and the fact that Δ is consistent. \square

The canonical model for this logic is almost entirely standard—we define $<^c$ in the usual way for an accessibility relation, except we make it irreflexive.

Definition A.3.4. The *canonical model* for this logic over $\mathcal{L}_{\text{best}}$ is a plausibility model $\mathcal{M}^c = \langle W^c, R^c, <^c, V^c \rangle$, where

- $W^c = \{\Delta \mid \Delta \text{ is maximally consistent over } \mathcal{L}_{\text{best}, \mathbf{A}}\}$
- $\Delta_1 <^c \Delta_2$ iff $\Delta_1 \neq \Delta_2$ and for all $\varphi \in \mathcal{L}_{\text{best}}$, if $\Box \varphi \in \Delta_1$ then $\varphi \in \Delta_2$.
- $\Delta \in V^c(p)$ iff $p \in \Delta$

Note that the $W^c, <^c, V^c$ lines are all part of the standard canonical model construction for modal logic—the only new change is that we ensure the accessibility relation $<^c$ is irreflexive.

Proposition A.3.5. The canonical model \mathcal{M}^c is in fact a plausibility model, i.e. $\mathcal{M}^c \in \mathbf{Plaus}$.

Proof. I need to show is that $<^c$ is irreflexive, transitive, and smooth:


$<^c$ is irreflexive. This is almost by definition; let $\Delta \in W^c$, and suppose for contradiction that $\Delta <^c \Delta$. So $\Delta \neq \Delta$, which is a contradiction.

$<^c$ is transitive. Suppose $\Delta_1 <^c \Delta_2$ and $\Delta_2 <^c \Delta_3$. By definition, for all φ , $\Box\varphi \in \Delta_1$ implies $\varphi \in \Delta_2$ and for all φ , $\Box\varphi \in \Delta_2$ implies $\varphi \in \Delta_3$. To show $\Delta_1 <^c \Delta_3$, let $\varphi \in \mathcal{L}_{\text{best}}$ and suppose $\Box\varphi \in \Delta_1$. By **(Trans)** for \Box , $\Box\Box\varphi \in \Delta_1$. By hypothesis, this means $\Box\varphi \in \Delta_2$, and so $\varphi \in \Delta_3$ (and we are done).

$<^c$ is smooth. Let $S \subseteq W^c$ be any set, and suppose $\Delta \in S$ but $\Delta \notin \text{best}_{<^c}(S)$. I would like to show that there is some $\Delta' \in W^c$ better than Δ ($\Delta' <^c \Delta$), that is the best, ($\Delta' \in \text{best}_{<^c}(S)$). Consider $\Delta' = \{\Box\varphi \mid \varphi \in \}$

[How does this follow from **(Up)** and **(Down)**??]

□

Lemma A.3.6. ( Truth Lemma) We have, for all $\Delta \in W^c$, $\varphi \in \mathcal{L}_{\text{best}}$,

$$\mathcal{M}, \Delta \Vdash \varphi \text{ iff } \varphi \in \Delta$$

Proof. By induction on φ . The propositional and boolean cases are straightforward. The $\Box\varphi$ case is the standard one from modal logic, and follows from the usual lemmas about maximally consistent sets (using the **(Dual)** rule and **(Distr)** axiom for \Box) [cite]. Similarly, the $\mathbf{A}\varphi$ and \Box^\downarrow cases are already known, and the introduction of $[\text{best}]\varphi$ do not affect them. I'll skip to the most relevant case:

[best] Case. For the (\rightarrow) direction, suppose $\mathcal{M}^c, \Delta \Vdash [\text{best}]\varphi$ but for contradiction $[\text{best}]\varphi \notin \Delta$. Observe:

Δ . Observe:

$$\begin{aligned} \mathcal{M}^c, \Delta \Vdash [\text{best}]\varphi &\rightarrow \Delta \in \text{best}_{<^c}(\llbracket \varphi \rrbracket_{\mathcal{M}^c}) && \text{(by the semantics)} \\ &\rightarrow \Delta \in \llbracket \varphi \rrbracket_{\mathcal{M}^c} \text{ and for all } \Delta', && \text{(by definition of } \text{best}_{<^c}) \\ &\quad \text{if } \Delta' \in \llbracket \varphi \rrbracket_{\mathcal{M}^c} \text{ then } \neg\Delta' <^c \Delta \\ &\rightarrow \varphi \in \Delta \text{ and for all } \Delta', && \text{(by inductive hypothesis)} \\ &\quad \text{if } \varphi \in \Delta' \text{ then } \neg\Delta' <^c \Delta \end{aligned}$$

Now consider $\Delta' = \{\psi \mid \Box^\downarrow\psi \in \Delta\} \cup \{[\text{best}]\varphi\}$. Since Δ is maximal, $\neg[\text{best}]\varphi \in \Delta$. So we have $\varphi \wedge \neg[\text{best}]\varphi \in \Delta$, and by **(Smooth)**, $\Diamond^\downarrow[\text{best}]\varphi \in \Delta$. This fact allows us to apply Proposition A.3.3, which says Δ' is consistent.

So we can extend Δ' to a maximally consistent set $\Delta^{\max} \supseteq \Delta'$. Since $[\text{best}]\varphi \in \Delta^{\max}$, by **(Refl)** for $[\text{best}]$, $\varphi \in \Delta^{\max}$. So by the last line of implications above, $\neg\Delta^{\max} <^c \Delta$. Now observe that by construction, for all formulas ψ , $\Box^\downarrow\psi \in \Delta$ implies $\psi \in \Delta^{\max}$. By Proposition A.3.2, this means for all ψ , $\Box\psi \in \Delta^{\max}$ implies $\psi \in \Delta$. But this is precisely the definition of $\Delta^{\max} \leq^c \Delta$! Putting $\neg\Delta^{\max} <^c \Delta$ and $\Delta^{\max} \leq^c \Delta$ together, we must have $\Delta^{\max} = \Delta$. But this gives us $[\text{best}]\varphi \in \Delta$, which contradicts our hypothesis $[\text{best}]\varphi \notin \Delta$.

(so $[\text{best}]\varphi \in \Delta$ must be true).

As for the (\leftarrow) direction, suppose $[\text{best}]\varphi \in \Delta$. Applying **(Ref)** gives us $\varphi \in \Delta$, which by our inductive hypothesis means $\Delta \in \llbracket \varphi \rrbracket$. I will now show that Δ is a *best* such point in $\llbracket \varphi \rrbracket$. Suppose not, say there is some $\Delta' \in \llbracket \varphi \rrbracket$ with $\Delta' <^c \Delta$. (Note by inductive hypothesis, $\varphi \in \Delta'$) By definition of $<^c$ we have:

$$\Delta' \neq \Delta \text{ and for all } \psi \in \mathcal{L}_{\text{best}}, \text{ if } \Box\psi \in \Delta' \text{ then } \psi \in \Delta$$

Note that by Proposition A.3.2, this also means that

$$(*) \quad \text{For all } \psi \in \mathcal{L}_{\text{best}}, \text{ if } \Box^\downarrow\psi \in \Delta \text{ then } \psi \in \Delta'$$


I will now show that $\Delta = \Delta'$, which contradicts $\Delta' <^c \Delta$. For (\subseteq) let $\psi \in \Delta$. We have

$$\begin{aligned} \psi \in \Delta & \text{ implies } [\text{best}]\varphi \wedge \psi \in \Delta \quad (\text{since } [\text{best}]\varphi \in \Delta') \\ & \text{ implies } \Box^\downarrow(\varphi \rightarrow \psi) \in \Delta \quad (\text{by } \mathbf{(Down)}) \\ & \text{ implies } \varphi \rightarrow \psi \in \Delta' \quad (\text{by } (*) \text{ above}) \\ & \text{ implies } \psi \in \Delta' \quad (\text{since } \varphi \in \Delta') \end{aligned}$$


As for (\supseteq) , let $\psi \in \Delta'$. Observe that $[\text{best}]\varphi \rightarrow [\text{best}]\varphi \wedge [\text{best}]\varphi \rightarrow \Box^\downarrow(\varphi \rightarrow [\text{best}]\varphi)$ (this is just **(Down)**, but substituting $\psi := [\text{best}]\varphi$). Since $[\text{best}]\varphi \in \Delta$, $\Box^\downarrow(\varphi \rightarrow [\text{best}]\varphi) \in \Delta$, and then by $(*)$ $\varphi \rightarrow [\text{best}]\varphi \in \Delta'$. From all this we get $[\text{best}]\varphi \in \Delta'$. From here we have:

$$\begin{aligned} \psi \in \Delta' & \text{ implies } [\text{best}]\varphi \wedge \psi \in \Delta' \quad (\text{since } [\text{best}]\varphi \in \Delta') \\ & \text{ implies } \Box([\text{best}]\varphi \rightarrow \psi) \in \Delta' \quad (\text{by } \mathbf{(Up)}) \\ & \text{ implies } [\text{best}]\varphi \rightarrow \psi \in \Delta \quad (\text{by definition of } <^c) \\ & \text{ implies } \psi \in \Delta \quad (\text{since } [\text{best}]\varphi \in \Delta) \end{aligned}$$

From this contradiction, we conclude that Δ is a best φ -world, and so by the semantics we have $\mathcal{M}^c, \Delta \Vdash [\text{best}]\varphi$. □

Theorem A.3.7. ( Model Building for $\mathcal{L}_{\text{best}}$ over \models_{Plaus}) For all consistent $\Gamma \subseteq \mathcal{L}_{\text{best}}$, there is some $\mathcal{M} \in \mathbf{Plaus}$ and state $w \in W$ such that $\mathcal{M}, w \models \Gamma$.

Proof. This proof is standard for modal logics. Let Γ be consistent. Take the model to be the canonical model \mathcal{M}^c , and extend Γ to maximally consistent set $\Delta \supseteq \Gamma$. Since Δ is maximally consistent, $\Delta \in W^c$. Since every $\varphi \in \Gamma$ is in Δ , by our truth lemma for all $\varphi \in \Gamma$, $\mathcal{M}^c, \Delta \Vdash \varphi$. So $\mathcal{M}^c, \Delta \Vdash \Gamma$, and we are done. □

Corollary A.3.8. ( Completeness for $\mathcal{L}_{\text{best}}$ over \models_{Plaus}) For all consistent $\Gamma \subseteq \mathcal{L}_{\text{best}}$ and all formulas $\varphi \in \mathcal{L}_{\text{best}}$,

$$\text{if } \Gamma \models \varphi \text{ then } \Gamma \vdash \varphi$$

Proof. Since the language $\mathcal{L}_{\text{best}}$ has negation, completeness follows from model building in the usual way; this proof is entirely standard. Suppose contrapositively that $\Gamma \not\models \varphi$. It follows that $\Gamma \vdash \neg\varphi$. So $\Gamma \cup \{\neg\varphi\}$ is consistent, and by Theorem A.3.7 we have $\mathcal{M} \in \mathbf{Plaus}$ and $w \in W$ such that $\mathcal{M}, w \models \Gamma \cup \{\neg\varphi\}$. But then $\mathcal{M}, w \models \Gamma$ yet $\mathcal{M}, w \not\models \varphi$, which is what we wanted to show. \square

A.4 Building a Finite Model

[Do the filtration construction here!!!]

A.5 Dynamic Updates on the Logic of [best]

[Show how the [best] operator cleans up the reduction axioms for Lex and Consr upgrades.]

References

- [1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat et al. GPT-4 technical report. *ArXiv preprint arXiv:2303.08774*, 2023.
- [2] Alfred V. Aho, Michael R Garey, and Jeffrey D. Ullman. The transitive reduction of a directed graph. *SIAM Journal on Computing*, 1(2):131–137, 1972.
- [3] Aws Albarghouthi et al. Introduction to neural network verification. *Foundations and Trends® in Programming Languages*, 7(1–2):1–157, 2021.
- [4] Edoardo Baccini, Zoé Christoff, and Rineke Verbrugge. Dynamic logics of diffusion and link changes on social networks. *Studia Logica*, pages 1–71, 2024.
- [5] Sebastian Bader and Pascal Hitzler. Dimensions of neural-symbolic integration – A structured survey. In *We Will Show Them! Essays in Honour of Dov Gabbay, Volume 1*, pages 167–194. College Publications, 2005.
- [6] Samy Badreddine, Artur d'Avila Garcez, Luciano Serafini, and Michael Spranger. Logic Tensor Networks. *Artificial Intelligence*, 303:103649, 2022.
- [7] Christian Balkenius and Peter Gärdenfors. Nonmonotonic inferences in neural networks. In *KR*, pages 32–39. Morgan Kaufmann, 1991.
- [8] Alexandru Baltag, Zoé Christoff, Rasmus K Rendsvig, and Sonja Smets. Dynamic epistemic logics of diffusion and prediction in social networks. *Studia Logica*, 107:489–531, 2019.
- [9] Alexandru Baltag, Nina Gierasimczuk, Aybüke Özgün, Ana Lucia Vargas Sandoval, and Sonja Smets. A dynamic logic for learning theory. *Journal of Logical and Algebraic Methods in Programming*, 109:100485, 2019.
- [10] Alexandru Baltag, Nina Gierasimczuk, and Sonja Smets. Truth-tracking by belief revision. *Studia Logica*, 107:917–947, 2019.
- [11] Alexandru Baltag, Dazhu Li, and Mina Young Pedersen. On the right path: A modal logic for supervised learning. In *International Workshop on Logic, Rationality and Interaction*, pages 1–14. Springer, 2019.
- [12] Alexandru Baltag, Lawrence S Moss, and Sławomir Solecki. Logics for epistemic actions: completeness, decidability, expressivity. *Logics*, 1(2):97–147, 2023.
- [13] Alexandru Baltag, Lawrence S Moss, and Sławomir Solecki. The logic of public announcements, common knowledge, and private suspicions. In *Proceedings of the 7th conference on Theoretical aspects of rationality and knowledge*, pages 43–56. 1998.
- [14] Alexandru Baltag and Sonja Smets. Group belief dynamics under iterated revision: Fixed points and cycles of joint upgrades. In *Proceedings of the 12th Conference on Theoretical Aspects of Rationality and Knowledge*, pages 41–50. 2009.

- [15] Tarek R Besold, Artur d'Avila Garcez, Sebastian Bader, Howard Bowman, Pedro Domingos, Pascal Hitzler, Kai-Uwe Kühnberger, Luis C Lamb, Priscila Machado Vieira Lima, Leo de Penning et al. Neural-symbolic learning and reasoning: A survey and interpretation. In *Neuro-Symbolic Artificial Intelligence: The State of the Art*, pages 1–51. IOS press, 2021.
- [16] Reinhard Blutner. Nonmonotonic inferences and neural networks. *Synthese*, 142:143–174, 2004.
- [17] Zoé Christoff and Jens Ulrik Hansen. A logic for diffusion in social networks. *Journal of Applied Logic*, 13(1):48–77, 2015.
- [18] Gabriele Ciravegna, Pietro Barbiero, Francesco Giannini, Marco Gori, Pietro Lió, Marco Maggini, and Stefano Melacci. Logic Explained Networks. *Artificial Intelligence*, 314:103822, 2023.
- [19] Artur d'Avila Garcez, Krysia Broda, and Dov M Gabbay. Symbolic knowledge extraction from trained neural networks: A sound approach. *Artificial Intelligence*, 125(1-2):155–207, 2001.
- [20] Walter Dean. Computational complexity theory. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, 2021.
- [21] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan et al. The Llama 3 herd of models. *ArXiv preprint arXiv:2407.21783*, 2024.
- [22] Artur SD'Avila Garcez, Luis C Lamb, and Dov M Gabbay. *Neural-Symbolic Cognitive Reasoning*. Springer Science & Business Media, 2008.
- [23] Atticus Geiger, Zhengxuan Wu, Christopher Potts, Thomas Icard, and Noah Goodman. Finding alignments between interpretable causal variables and distributed neural representations. In *Causal Learning and Reasoning*, pages 160–187. PMLR, 2024.
- [24] Laura Giordano, Valentina Gliozzi, and Daniele Theseider Dupré. A conditional, a fuzzy and a probabilistic interpretation of self-organizing maps. *Journal of Logic and Computation*, 32(2):178–205, 2022.
- [25] Laura Giordano and Daniele Theseider Dupré. Weighted defeasible knowledge bases and a multipreference semantics for a deep neural network model. In *Logics in Artificial Intelligence: 17th European Conference, JELIA 2021, Virtual Event, May 17–20, 2021, Proceedings 17*, pages 225–242. Springer, 2021.
- [26] Charles G Gross. Genealogy of the “grandmother cell”. *The Neuroscientist*, 8(5):512–518, 2002.
- [27] Frankvan Harmelen. Preface: The 3rd AI wave is coming, and it needs a theory. In *Neuro-Symbolic Artificial Intelligence: The State of the Art*, page 0. IOS Press BV, 2022.
- [28] Donald Hebb. *The Organization of Behavior*. Psychology Press, apr 1949.
- [29] Neil Immerman. *Descriptive Complexity*. Springer Science & Business Media, 1998.
- [30] Caleb Kisby, Saúl Blanco, and Lawrence Moss. The logic of Hebbian learning. In *The International FLAIRS Conference Proceedings*, volume 35. 2022.
- [31] Caleb Schultz Kisby, Saúl A Blanco, and Lawrence S Moss. What do Hebbian learners learn? Reduction axioms for iterated Hebbian learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*,

- volume 38, pages 14894–14901. 2024.
- [32] Dexter Kozen and Rohit Parikh. An elementary proof of the completeness of PDL. *Theoretical Computer Science*, 14(1):113–118, 1981.
 - [33] Sarit Kraus, Daniel Lehmann, and Menachem Magidor. Nonmonotonic reasoning, preferential models and cumulative logics. *Artificial intelligence*, 44(1-2):167–207, 1990.
 - [34] Hannes Leitgeb. Nonmonotonic reasoning by inhibition nets. *Artificial Intelligence*, 128(1-2):161–201, 2001.
 - [35] Hannes Leitgeb. Nonmonotonic reasoning by inhibition nets II. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 11(supp02):105–135, 2003.
 - [36] Hannes Leitgeb. Neural network models of conditionals. In *Introduction to Formal Philosophy*, pages 147–176. Springer, 2018.
 - [37] Leonid Libkin. *Elements of Finite Model Theory*, volume 41. Springer, 2004.
 - [38] Robin Manhaeve, Sebastijan Dumančić, Angelika Kimmig, Thomas Demeester, and Luc De Raedt. Neural probabilistic logic programming in DeepProbLog. *Artificial Intelligence*, 298:103504, 2021.
 - [39] Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4):115–133, dec 1943.
 - [40] Drew McDermott. A critique of pure reason. *Computational intelligence*, 3(3):151–160, 1987.
 - [41] William Merrill. Sequential neural networks as automata. *ArXiv preprint arXiv:1906.01615*, 2019.
 - [42] William Merrill and Ashish Sabharwal. The expressive power of transformers with chain of thought. *ArXiv preprint arXiv:2310.07923*, 2023.
 - [43] William Merrill, Gail Weiss, Yoav Goldberg, Roy Schwartz, Noah A Smith, and Eran Yahav. A formal hierarchy of RNN architectures. *ArXiv preprint arXiv:2004.08500*, 2020.
 - [44] Lawrence S Moss. Finite models constructed from canonical formulas. *Journal of Philosophical Logic*, 36:605–640, 2007.
 - [45] Leonardo de Moura and Sebastian Ullrich. The Lean 4 theorem prover and programming language. In *Automated Deduction—CADE 28: 28th International Conference on Automated Deduction, Virtual Event, July 12–15, 2021, Proceedings 28*, pages 625–635. Springer, 2021.
 - [46] Gregory Murphy. *The Big Book of Concepts*. MIT press, 2004.
 - [47] Erkki Oja. Simplified neuron model as a principal component analyzer. *Journal of mathematical biology*, 15:267–273, 1982.
 - [48] Eric Pacuit. *Neighborhood Semantics for Modal Logic*. Springer, 2017.
 - [49] Jan A. Plaza. Logics of public communications. *Synthese*, 158:165–179, 2007.
 - [50] George Polya. *Mathematics and Plausible Reasoning: Induction and Analogy in Mathematics*, volume 2. Princeton University Press, 1954.
 - [51] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error

- propagation. *Biometrika*, 71(599-607):6, 1986.
- [52] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.
- [53] Md Kamruzzaman Sarker, Lu Zhou, Aaron Eberhart, and Pascal Hitzler. Neuro-Symbolic Artificial Intelligence: Current Trends. *AI Communications*, 34, 2022 2022.
- [54] Murray Shanahan. The frame problem. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, 2016.
- [55] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton et al. Mastering the game of Go without human knowledge. *Nature*, 550(7676):354–359, 2017.
- [56] Rupesh K Srivastava, Klaus Greff, and Jürgen Schmidhuber. Training very deep networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28, page 0. Curran Associates, Inc., 2015.
- [57] Lena Strobl, William Merrill, Gail Weiss, David Chiang, and Dana Angluin. What formal languages can transformers express? A survey. *Transactions of the Association for Computational Linguistics*, 12:543–561, 2024.
- [58] Alex Tamkin, Miles Brundage, Jack Clark, and Deep Ganguli. Understanding the capabilities, limitations, and societal impact of Large Language Models. *ArXiv preprint arXiv:2102.02503*, 2021.
- [59] Johan Van Benthem. Dynamic logic for belief revision. *Journal of applied non-classical logics*, 17(2):129–155, 2007.
- [60] Johan Van Benthem. *Logical Dynamics of Information and Interaction*. Cambridge University Press, 2011.
- [61] Johan Van Benthem and Fenrong Liu. Dynamic logic of preference upgrade. *Journal of Applied Non-Classical Logics*, 17(2):157–182, 2007.
- [62] Johan Van Benthem and Sonja Smets. Dynamic logics of belief change. In H. Van Ditmarsch, J. Halpern, W. van der Hoek, and B. Kooi, editors, *Handbook of Epistemic Logic*, pages 313–393. College Publications, London, UK, 2015.
- [63] Hans Van Ditmarsch, Wiebe van Der Hoek, and Barteld Kooi. *Dynamic Epistemic Logic*, volume 337. Springer, 2007.
- [64] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.
- [65] Gail Weiss, Yoav Goldberg, and Eran Yahav. On the practical computational power of finite precision RNNs for language recognition. *ArXiv preprint arXiv:1805.04908*, 2018.