

ActInf GuestStream 032.1 ~ Adam Pease, “A Neuro-Symbolic Approach to Language Understanding”

Discussion with Dr. Adam Pease of the SUMO formal-ontology database, supporting software, and ongoing development projects.

<https://www.adampease.org/>

<https://www.ontologyportal.org/>

<https://www.youtube.com/channel/UCVyCle22zz5gZjWvNWCBq9Q>

Presented by Active Inference Institute in 2023

ActInf GuestStream Session 032.1, January 10, 2023

<https://www.youtube.com/watch?v=0pMxBM3ahwQ>

Translated closed captions: <https://youtu.be/bp7wrXv3NTo>

SESSION SPEAKERS

Adam Pease, Daniel Friedman, David Douglass

CONTENTS

Start Time Gist

- 00:12 Adam Pease on Common sense auto-communication
- 03:15 An Ontology for Artificial Intelligence
- 12:54 The ontology of Modern programming
- 15:29 Ontology: The formal semantics of words
- 28:27 Machine Learning for Mathematicians
- 30:44 The Birth of the Neural Network
- 38:31 Conceptual semantics in a synthetic language
- 43:47 Language generative and the synthetic text
- 52:02 What if I Had a Higher Order Logic?
- 57:16 Natural language has a lot of processes and objects

1:06:12 The language of natural text
1:09:49 Applying the Pruning to Lua
1:11:22 Daniel on Probabilistic and Uncertain Reasoning
1:22:57 Cognitive modeling, the complexity of action
1:25:38 Machine Learning and the Cognitive Guardrails
1:38:40 Inference and the symbolic logic of the brain
1:40:55 Generative AI: The Future of Writing
1:44:58 Setting up the theorem prover

TRANSCRIPT

00:12 Daniel:

Hello and welcome. This is ActInf lab Livestream number 32.1. It's January 10, 2023. We're here with Adam Pease, and we will be hearing a presentation followed by a discussion. So Adam, thank you very much for joining today.

Looking forward to this and off to you.

00:31 Adam:

Okay, thanks very much for having me. Looking forward to presenting a few bits of initial research as well as some background on long term ongoing research into understanding language and trying to mimic some just some elements of human thought. Everything I do is open source. So I've got some links for you there.

I also have a pretty active YouTube channel, although it's been a little quiet lately, so there's a lot of material here that I won't be able to go into in depth and so eager to answer questions when we get to the question period. And also would encourage folks to take a look at that channel where there's longer video format presentations about a whole host of topics that we'll probably touch on today. So I have a large ontology. I'll explain in a minute what I mean by an ontology. That's called SUMO.

And I'm also going to coherence some work that Facebook has done called their baby corpus.

01:38 So that's the background for the cartoons that I'm showing you.

The main topic that I want to cover today is the notion of common sense auto communication. My objective is to be able to make statements in human natural language, just in English at the moment, ask questions also in natural language and get answers, but get answers with exploration, something that Modern question answering systems that we hear about each day in the popular press are generally not capable of doing. You might get an answer, but you don't know why, and knowing why and understanding an explanation is very important. So we're not talking about just information retrieval, we're not talking about Google, where you ask a question and you get a list of documents in which there might be an answer or even Google's occasional capabilities for very specific sorts of questions where they give you an answer but they'll never tell you why.

02:42 It's more like you ask, what's the weather Coda?

You get a nice concise link to the weather, but you won't get an explanation for why the weather is what it is.

So what do I mean by biology?

02:56 Daniel:

Are you meaning to share? We see the first slide.

02:59 Adam:

You're still only seeing the first slide. Okay, I'm on slide three.

So let's go back to I'm not sure why that's not showing up, but good.

03:09 Daniel:

It looks good.

03:10 Adam:

Now, common sense auto format, this format then instead of the slide presentation format.

03:14 Daniel:

Thank you. It looks good.

03:15 Adam:

Okay, so here's what I just talked about. So let me give you my definition for what is an ontology, because this is a word that at a certain point in recent history, maybe about a decade or two ago, got overloaded so much now that just about anything is called an ontology. And so for me to distinguish ontology from things we already had words for, like semantic network or schema, I consider an ontology to be essentially a dictionary for computer to read. And that also requires a bit of unpacking. What is a dictionary for computers to read?

Well, for me, it means minimally. We've got to have terms, a set of terms that label things in our world. But most importantly, we have to have a set of definitions that are stated in a computable language. So just stating things in English or French that are definitions for words, well, that's a dictionary. It's not an ontology.

And if we don't have definitions in some sort of mathematical language, a programming language of some sort, then our machines are only going to have the barest abilities to manipulate or do things with those natural language definitions.

04:31 So I think they have to be in a formal language to nail down meanings so that we have shared meaning. We have shared meaning between people that agree to use a particular ontology. And maybe more importantly, we have shared meaning between people and computers so that the same constraints, the same inferences that people would be able to make with these terms, the computer is also able to make because of these mathematically expressed definitions. So that's what I've spent the last 22 years trying to build up.

And this is work that draws on linguistics, philosophy, and computer science, as well as logic as a subdiscipline of probably all of those three.

And one reason I'm doing this. Nowadays, artificial intelligence has become almost synonymous with machine learning. So I need to explain a little bit, well, why am I doing this?

05:33 This is sort of old fashioned AI of writing down lots of rules. Why do we still need to do this?

Well, I'm a big fan of Daniel Kahneman's work. This is just one justification of many. But hopefully, because it's very well articulated in his work, hopefully this can be at least the start of a justification for

some of you who might be unfamiliar with this symbolic approach that used to be the primary approach in artificial intelligence. I think that he makes a good case that humans have two modes of thought. We have an instinctive, intuitive, sensory, pattern matching kind of process that's good for face or voice recognition or making a very snap judgment about what to do in a particular situation that is learned from the presentation of many, many similar situations.

And that sort of mode of thought fits very well with statistically based machine learning, neural networks, deep neural networks and so forth.

06:34 That's an important part of intelligence, and it's led to a lot of important commercial activities. But there is another mode of intelligence that Professor Cunningham, I think illustrates quite clearly. That is a slower, more deliberative, deductive approach where people can explain what they do as opposed to just give a reaction. And it seems like in large parts of the AI community people are starting to recognize that, yes, there is still this set of things that we require our computers to do that they're not doing very well with the current neural approach, numeric approach that we need another mode of tool set for.

And so you're starting to see this keyword neurosymbolic approach of trying to balance statistical machine learning approaches with some sort of explicit representation in a symbolic form.

07:39 And it's this latter bit that I've been doing a lot of work with and I think it's the combination of these two main approaches that's going to lead us to some more exciting capability. That's just my assessment of the state of the field. But there is some emerging evidence that this in fact is the case, that we have a potential with integrating these two approaches to handle a broader range of things that would look more like human intelligence.

Along with this grand goal of doing AI, I've also been applying this work in creating this large dictionary to some far more prosaic stuff in data integration. The bulk of the commercial consulting work that I've been doing really over the past ten years is just looking at a software engineering methodology of trying to capture the formal meaning of things that we use in computer science databases, spreadsheets and so forth.

08:46 Anytime you have a spreadsheet, you usually have a set of columns across the top of your spreadsheet that has some labels. And it's very unusual for people to have any sort of explicit definitions of what those labels actually mean. It's usually just assumed that people, by looking at those labels, if you're in the same community or you can ask for an explanation that gives you a good sense of what's the intended semantics of that particular column based on that particular label.

And what I've found is that it might work for a spreadsheet where you've got a few dozen columns and you're working closely with some longtime colleagues. But as soon as you get to a larger organization with a larger set of symbols, labels where people may be working at different locations maybe they've never even met each other. Where there's interactions occurring over the space of years in large multinational corporations, that people's initial assumptions about how clear their labels are are usually completely wrong.

09:55 And so people wind up using data in dramatically different ways inconsistent with its original intent, precisely because nobody's taken the time to actually define things and Beren. In the rare cases of organizations that do have data dictionaries that define their terms, the intended meaning of their terms, they tend to define them in very informal ways using natural language definitions, which means then that the machines can do very little to help ensure that people's intended specification of semantics

is consistent across a large body of terms.

We might be able to have a human being keep a data dictionary of 100 terms consistent and well organized over time, over a long period of time. But once you get to thousands or tens of thousands of terms it's well beyond human capacity to keep such a product consistent you need automation. And unless you have definitions that are explicit and in a computable language, the computer can do very little to help keep those definitions consistent.

11:02 So that's one of the big byproducts of all of this work so far. And I use techniques like automated theoremproving that I don't have time to talk about now but would love to talk about it in detail for anyone who's interested to keep these definitions consistent.

So I have a logical approach for the semantics of terms, lexical semantics, the semantics of the words we use. And this extends also to computer science controls like labels for tables or columns or database elements and so forth. And this is useful anywhere. Pretty much anyone has a spreadsheet, anyone has a database. This is something that people don't realize they could make use of.

People seem to accept that they're going to need to have a lot of human communication to explain the meanings of terms. And there's just another way, right?

12:03 And that's what I've developed and that's what I'm basing this work on language understanding on. There's a common belief that oh, this is just too hard, that we should be able to come up with these things easily. There's too much of them, meanings change and so forth.

Once you take the perspective that there is a need for precise semantics for our labels and that this precise semantics is something like definition in the dictionary but in a formal language then it's not just a sort of a graph or a list of relationships or likelihoods or probabilities.

It starts to be hard to imagine then how a machine would just come up with this sort of thing automatically. And of course we have another venue in which we take this sort of harder approach of programming and don't expect machines to just do everything automatically and that's programming.

13:05 We have many, many millions of programmers in the world now. They're expensive. It is a time consuming process to program things but we invest in this because there simply isn't another way to do it.

And we make this enterprise of programming feasible through a couple of points. One is having very rich languages so we can say everything that we want to say. So we don't want to be restricted in the sort of programs we can write. Similarly, we need to make sure we're not restricted in the sorts of definitions that we can write. And the way that we make that practical is through reuse.

You're a Modern programmer. If you deliver some software, you're writing maybe 1% of the program, 1% of the code that's delivered. Most of what you deliver is going to be a set of large libraries and reusable components, operating system, database, web server and any number of other things, java collection classes, NumPy libraries, if you're doing Python.

14:09 The way in which Modern software development is practical is through reuse. And so that's the other thing that I've been working to provide is this thing called sumo, which is a library of distinctions that can be reused that takes this problem that many people would otherwise consider to be completely intractable manual creation of knowledge and makes it eminently practical because you're actually reusing most of your semantics by virtue of using this library.

So this sumo like sumo wrestler suggested upper emerged ontology started in the year 2000. It started as just an upper ontology. So the acronym is a bit of a misnomer, a bit of anachronism, if you will,

right now, that it's really a comprehensive ontology, not just an upper ontology because it's crowned to be about 20,000 terms, 80,000 handwritten statements in an expressive higher order. Mathematical logic includes links to vast fact bases like the Yago system, lots of translations from its authored format into some of the standard representations used in the automated theorem proven community.

15:19 And as I mentioned, it's all open source and online and to every version has been open source. There's nothing that's held back for commercial reasons.

Let me mention WordNet. So another common misconception in ontology is that we have to use words and follow the definitions of words. And as we should all know, words are ambiguous and words are polysimous. The same word can have multiple meanings. So we've created a very clear division, but also a relationship between the senses, the concepts that we have in the world that have formal definitions and the labels that we as humans put on those concepts to communicate them to each other. And this also makes it easy to ensure that our ontology is not tied to any particular one language because any one of us who's multilingual knows an English word and a tagalog word can have the same underlying semantics, the same underlying real world coherence, but we use different labels.

16:31 And so we need to make sure that we can have vocabularies that are mapped to our common single formal ontology that tries to explain and define concepts in our world and not let sort of language and labels and intuitions about those labels creep into our formal definition. So we reuse Princeton's Word Net, which is a project going on for about 30 years now, I think. And it has also projections into multiple languages, polish Word Net toggleordnet for example. There's continuing work on that which I'm very excited because I started the tagalog Word net back in the De La Salle University and so it's great to see that sort of effort continuing.

There's word Nets for Arabic and Chinese and many other languages and so we keep those labels in English, those dictionaries human dictionaries, distinct but related to the formal inventory of mathematically expressed concepts.

17:33 And so back in the early 2000s, we took an effort to go through all 117,000 sunsets in WordNet, all 117,000 linguistically labeled concepts and map them all to Sumo one at a time by hand. It's just an enormous effort where we couldn't have really done it automatically. It required human inspection. We invested that effort and it since paid off for the linguistic work that we do with this ontology.

Why do we need both? Well, the formal definitions are things that just aren't in WordNet or any other dictionary. You might have a word for earlier and a natural language definition for what it means for some events to be earlier than another. But we need a precise mathematical logical definition of what earlier means. And so sumo has this and many, many other axioms that individually are trivial but together add up to creating a certain understanding about the world that without it, machines just don't have.

18:39 We need a machine to know that if one event is earlier than another, that the end point of this first event is before the beginning point of the second event. So we've got a formal mathematical definition of earlier which you just don't have in a dictionary. This allows machine to do logic inference, automated inference, and give you conclusions and explain those conclusions. Not a criticism of WordNet, just they're different products with different needs and we need both. Also, I get a lot of questions about what kind of logic do we need?

Do we really need a very complicated expressive logic? It's very popular nowadays for people to use taxonomies or knowledge graphs. Well, knowledge graphs are essentially semantic networks. It's a

knowledge representation that was created back really in the 1960s. And so it does surprise me that it's kind of what's old is new again in artificial intelligence somehow and that knowledge graphs are a big thing now.

19:47 But it's a knowledge representation technology that leaves out a lot of stuff that we would need to say about the real world in order to define the terms that we use. So natural language definitions aren't enough and we believe that defining our terms or our concepts is good. Which particular language do we choose? And I've had people often say usually really without any backup. Oh, those are the rare cases.

Well, I got frustrated with that. So I did a paper recently, now almost two years old, about trying to quantify how often do we use constructs in language that require more than just a graph of relationships. And so I came up with some statistics. I'm not going to go into this paper in great detail, but if you're interested, I'd love to follow up with you. And I came up with that in looking at two large corporate brown.

20:50 Corpus is kind of old and kind of small by Modern standards, but I also used the Corpus of Contemporary American English, took a random sample of sentences, created an automated system for looking at some various features that require in language a translation into using an expressive logic, such as modal expressions, can, may, should, might, expressions of authorship said wrote Epistemic. Nose believes these are words in language that trigger the need for a more expressive representation. I can give you a lot of detail on why that's directly entailed. This is not just my opinion trying to advocate a more expressive logic. These are very sort of strict things that are well known in linguistics, but at least now it's well quantified that about 50% of statements taken from a random sample of a large corpus do actually require these more expressive semantics.

So I think I've made a strong case that we can't just get away with graphs unless we want to leave out about half of the things that people say to each other.

21:58 And that tells me that we really need this as more expressive logic.

Okay, so that's all the pre gamble that was the library of semantics, of meaning that I already have, and how can I use that now to try to create a system? And this is research, right? And it's relatively experimental even at the moment. The attempt is to try to create a system where the machine can translate automatically from English sentences into a formal, precise semantics using these terms out of the ontology that themselves are precisely defined. So I have at the starting point English statements or questions.

I translate these things to logic. I want to be able to send them to a powerful theorem prover that's at least first order logic plus equality, but preferably higher order logic.

23:00 So I can handle some of these modal expressions. There are a few such provers, and I want it to engage in its process of doing theorem proving, not only with the statements and the question that a particular user has provided at one time, but also this rich background that describes the world. So that the machine is not just doing a sort of simple matching even with these powerful tools, but it's really taking into account background knowledge, world knowledge, in the same way that a person would. And then when it's done, when it's got an answer, I want it to be able to explain what it does. And theorem provers have this capability already. It's built in. They were designed to provide proofs because a lot of the work in theorem proving is for theorem proving in advanced mathematics. So there's a quiet but significant community of mathematicians and computer scientists that are using automated

theorem proving to prove novel theorems in mathematics.

24:06 And because they're in mathematics, they have a very strict standard of what constitutes a real work. You have to show a proof, just like almost all of us probably experienced in high school geometry of proving. One triangle is the same as another through maybe the side angle, side process. Machines can now do this and much more significant and complicated and sophisticated sorts of proofs and provide detailed breakdown of how it reached its conclusion. And I take this as to being something that's required.

If you ask another person, what's the answer to question X? Okay, they give you Y as an answer, but then you want to know, what's the details? How did you come to that conclusion? It's not just enough to say, Well, I told you so, so believe me, sometimes you might get away with that, but not all the time, especially not for anything that's complicated. And so our machines should be able to do that as well. So I've done a number of forays into this grand goal over the course of my career.

25:09 I started out with something called Controller English to Logic Translation, where it had a hand built grammar of a small subset of English that was designed to be unambiguous. It was all written in prologue and I was able to create expressions using Sumo terms in its logic, which was mostly first order logic back then in the early two thousands. And it was, you know, somewhat somewhat successful. But it was a fairly restricted subset of English.

And so the parser would break and say syntax error all too often. And that was kind of frustrating. It was sort of like what one experiences today with something like Alexa or Google where you ask a question and says, I'm sorry, I don't understand. And so people wind up self governing their utterances to adapt to the machine. And people were able to do that with my Kelt system.

But it's still kind of frustrating. I'd like to be able to handle text that was written not for that particular system and not dumbed down on the fly by a human, but rather any arbitrary sort of text.

26:20 So then I started working with Stanford's core NLP system and using dependency parse transformations and trying to turn dependency parses into logic. I did have some success with that, but I found that even as good as machine learning based statistical parsers are, there's still a lot of cases where they break down. In fact, they give a lot of metrics that look very encouraging, usually on a per word or per symbol basis, and they'll say, or we're 97% correct on.

And unlabeled attachment score is one that's used in the dependency parsing community. And that sounds like a very high percentage, but your average sentence in English is depending upon, you know, who you talk to, how you collect the statistics, about 24 words. Well, if you look at 97% raised to the 24th power, then you get a problem and you get actually less than 50% of your sentences are going to be parsed correctly.

27:32 And again, that's unlabeled attachment score for dependency parses. That merely means that two words are found to have the same dependency relation without a commitment to what exactly that relation is.

Well, you need the labeled attachment score, which of course is lower. And you needed to be correct for all of these words to have really a coherent understanding of a sentence if you're going to do any sort of automatic translation. So, although the systems are good and they've certainly gotten better from 2015, I mean, Stanford Stanza is yet another few percentage points better than core NLP was in terms of its dependency parsing, but it's still not reliable enough, not consistent, not stable enough for me to use that as a basis for translation. And then there was all the problem of, well, how do I actually take

that output and reliably create logic from it? And that meant at that time I was doing a handbuilt rule base and that just didn't work in the limit.

28:32 It was encouraging. I think there's potential if you had enough people working on it, a bit more research. But it wasn't the way in which I thought I could reach success. So I'm doing something different now. So I'm doing what I guess is more prevalent now in machine learning that instead of doing a pipeline of machine learning operations trying to do everything all at once with extraordinarily large corporate where I have trying to brain a system to do auto formalization of language to go straight from language to logic without any intermediate approach like dependency parsing.

And this is based on some of the success with my colleague Joseph Urban who's been doing this sort of work for mathematics. So one of the challenges of using theorem provers, automated theorem provers for mathematics is that as formal as mathematics is, it's not really quite formal enough in a mathematics textbook. There's still a lot of text and there's still a lot of things that aren't themselves precisely defined.

29:38 And there's also a lot of stuff there in mathematics that's not in a common and computable format. So for a long time there's been this problem of how do we take mathematical textbooks and turn them into extraordinarily detailed formalizations that an automated theorem prover can handle in first order logic plus equality or some other kind of mathematical logic.

So Joseph has had considerable success at this and we decided to team up on trying to do this even harder problem of just looking at unrestricted text, not just mathematics. And they've been using Google's Neural machine translation system, which is built on top of TensorFlow. We're at the moment using kind of a relatively old version of that. We tried a newer version. It turned out to actually be slower for our purposes, but we need to do a lot more expectation and make this work more popular so people tell us where we're going wrong and maybe there's some better machine learning libraries that we could use for this.

30:41 So this is the architectures. Now I want to tell you a bit more about how we actually come up with this set of natural language logic pairs. Because like other work of this sort, the libraries of pairs, of your input output pairs that you're trying to train has to be really big orders of magnitude bigger than any sort of pipeline approach because you're trying to tackle the whole problem and to end. And there's a lot more variability. Your statistical significance for the appearance of any given feature correlated with any other given feature is really low.

He has a very long tail in language of things that are not seen too often. So your corpus has to be very big. How do we get that? Corpus certainly not feasible to translate a lot of English by hand Hinton logic. So how do we get this thing to begin with?

So we were aware of this project done at Facebook called the Baby Corpus, in part because Tomash Micheloff has returned to his home country of the Czech Republic and now works with Joseph Urban at CTU Prague.

31:53 So we had benefit of getting some of his perspective on this work. It was basically an effort to try to teach machines how to do simple inferences and we'll look at some of those simple inferences in a moment. And in order to do that, they tried to generative a very large synthetic data set of simple natural language sentences and simple natural language inferences that they wanted the machines to be able to train up to handle.

Ultimately they used really only a few sort of rules or patterns. I think Tomash has said there's

something like 20 different patterns. I've looked at the code, it's written in lieu. It wouldn't be hard to rewrite it in a sort of more common language. I think it's not particularly complicated code, not a huge code base, a lot of simple patterns, a great idea, but I think we can do better at it given the resources that Sumo provides for generation as well as understanding and anchoring meaning.

32:59 So they had a lot of their knowledge representation, such as it is, was things like, you know, you have a bedroom, a thing, the location actors existing in this world. It sort of harks back to the blocks world of Patrick Henry Winston, I think it was Winston Hope. I've got that reference right now. Back in the 70s there was this notion that we could do there's a system called Shirley, I think it was, that did operations in a world of blocks, so a constrained world. There were actors, there were a few things that could be manipulated and the system was able to understand very simple utterances.

Well, they've kind of turned this on its head, created a knowledge representation for a certain world and then tried to generate a lot of sentences from it to create this synthetic model of things that a machine should be able to do coherence about. So here's some of the content of their questions. So Free is either in the school or the park Mary went back to.

34:02 The office. Is Mary in the office?

So the first sentence is just a confusion element of the corpus. It's really if Mary went back to the office, yes, she's in the office. Bill is either in the kitchen or the park. Another confusion sentence Fred moved to the cinema. Is Fred in the park?

Well, no, he's not in the park, but park has Beren seen as a token earlier and so they wanted to make sure that the automated learning system isn't just trying to find proximity, which is a good thing to include in their data set. So lots of stuff like this and more complicated things. Also I've just presented really some of the most simple examples but all generated from a very simple knowledge model, concepts that weren't really defined and a massive presentation of mind numbingly tiny variations and a wide variety of sentences. So we started out thinking well, let's just cover the stuff that's in the baby corpus so we understand what they were doing.

35:08 And the big advantage that we thought we could bring to this is we don't have to learn logic in a way they had a simple corpus that was trying to accomplish something that's very hard, which was to teach machines just through massive examples how to do logical inferences or a certain subset of logical inferences.

We have an easier problem if we use a theorem prover. We don't have to teach the system to do logical inference, we already have that. We just need to get the knowledge representation from English into logic and then let the theorem provers do their thing. So again, very simple world. They had objects, locations, actors, states, ActInf lab, few other decorations and so we tried to figure out really what they meant in a way that was hard because they didn't have a distinction between instances and classes. So we can talk about in sumo relationship that in an action one thing is an object transferred from one location to another.

36:16 They have this action of being gettable which is really a modal. So we had to figure out what did they mean? They didn't really mean a modal, they meant something that can be transferred and then there has to be some kind of transferred examples. We wound up interpreting a lot of their symbols not as classes.

So the notion of and apple is a class, but the apple that gets transferred in a particular sentence is an instance. So this difference between existential and universal quantification so we'd create a term like

apple, one is a kind of apple is an instance of an apple and that's a gettable thing and it's located in the particular area, et cetera. I don't have a lot of time to go into our knowledge representation, but if you want a full course on sumo, I can do that at some point. There's a lot of videos online also, so maybe I'll skip through a little bit of this we wanted to make sure we covered all of their stuff. They referred to things like chocolate and milk and garden and container and box.

And we actually had all this in sumo.

37:17 So getting back to this notion of, well, if you had to define all this stuff from scratch, you'd be tempted, just like the Facebook guys did, to just use the labels and assume everything, everybody means the same thing. But we didn't have to do that. We already had these terms precisely defined and sumo and we're able just to reuse them. So we did a big table, this is just a portion of that table, all the different things that they referred to that we already had or in some a few cases didn't have, and we needed to add them.

But adding the new concepts and their definitions is easy because the new concepts had definitions that were based on, in turn, other terms we already had. So we just had to combine them in the right ways. We didn't take for granted that we were just using the first meaning of things. For example, they had the word pajamas in some of their database. While pajamas have a couple of meanings in the dictionary, I wasn't all that familiar with the second meaning, but it does exist, and so we want to make sure we're using the right meaning.

38:21 So we also did mappings to word net of the terms we defined so that we had disambiguation potential force, our natural language. There were a number of things because of this small and artificial world, they had inferences like if you're hungry, then essentially you go to the kitchen and eat something that's in the kitchen that doesn't really hold in the real world.

Obviously we're not in this fantasy constrained world that hunger immediately entails. You go to the kitchen and eat an apple. So we had to create some kind of some bogus rules in Sumo that were very specific just to this corpus to try to replicate the inferences that they were doing. So here's the rule that gives the right answer for this corpus. It gets a little complicated to state it, but we can state it in sumo that if you have a cognitive agent that's hungry and that's at a particular time, then there's a translocation event that is desired by that agent and that translocation has a destination of the kitchen where there's a getting of an apple.

39:33 A rule that doesn't belong to be in sumo on any permanent basis, but is what's used to cover the semantics of this particular corpus. And we looked at things like being bored and tired, which they had. They all say, if you're bored and tired, you go to sleep. Okay. I wish that that were true for all of us hardworking engineers that we had that option.

We don't at all times, but that was what covered the semantics of the corpus. So you had to do similar things for Bored and Tired. They don't really have weights and measures and units in the baby ontology. Sumo of course, has a detailed ontology of weights and measures of units, all the system inclination measures and we had to figure out how we were going to translate these hacks that were in this corpus of things have a numerical size, one, two or three, something like that. How are we going to represent that?

So we figured out a similar hack for covering that.

40:33 All right, so we did this exploration to make sure that we could cover one of these examples worlds. But what we really want to do is generate a much larger corpus with a more interesting, more

varied Seth of things that are referred to in the real world. So we wanted to do a synthetic text generation as our first step so that we COVID kind of seed the system long term. We know that we need to go beyond that synthetic test.

We're not there yet. We're just still in the process of creating this very large set of synthetic sentences because we can start with an idea or a knowledge representation about what we want to express in the world and generate syntactically and semantically valid logic expressions from that. While at the same time, because we've created the state of the world artificially in a program, we can then also create a sentence that models that state of the world. So we go kind of both directions, both to logic and language from a very ad hoc sort of expression in some Java code about what's true in the real world. 41:44 And that allows us to do a lot of variation, but it's never going to cover the full expressivity of natural language.

But we think it will cover enough that it gives us a head start where we can then start saying, well, where does our automatic language to logic system break down? What are the sentences? Where it fails? Where it generates something that's not even syntactically correct? Maybe.

Or violates constraints that we know from Sumo about how the world works and then try to add those as new patterns to our generation system so we can find one sentence that causes a problem. We look at the core issue of what causes the problem and look at maybe the things that are variable the sentence. So we might have a sentence that's problematic, that has some subject and an object where the subjects and the objects have a wide varieties that could things that could be substituted in that we can turn into sort of a template to try to cover another pattern, if you will, in natural language.

42:46 And one of our possibilities, I think, will be to look at framenet that is exactly such a large inventory of these things. I've wanted to do a projection of framenet into sumo for several decades now. Did some early work with the team there, and we published some work about that in the mid two thousands, but we didn't do a comprehensive inventory of all of their different frames. And so I think there's a good project there to do a more detailed exploration of all the frames in framenet and translating them to Sumo because the frames in framenet don't have a precise logical semantics. Just one of the challenges I think that people have found with using that resource. It's not anchored in the same way that it needs to be, I think, to be used more productively there's. I think an interesting project there that would have a very practical outcome.

43:47 But for now I'm just going to talk about the synthetic text because we're not even into that phase yet. The synthetic text is the tough stuff. We just started this work sort of midfall so that kind of gives you a perspective on where we are. This is going to be a long term thing. So I tried to collect all of the things that I thought I could detail both linguistically and in Sumo.

So things like propositional attitudes, modals tenses names and roles and basic forms of transitive, intransitive ditransitive verbs in the subject verb object, basic grammar of English, try to say what some of these things are. So I'm going to give you an example. It's a real example of stuff we can generate and it looks kind of funny. And I'll tell you why it looks funny and what we can do to fix it. And it's other examples like it.

But I'm giving you words and all of this extremely early research even though I think it's very promising, I'm going to tell you what's wrong so far and the examples show stuff that look kind of funny.

44:57 So an internet user traps a crustacean. That's a pretty funny sentence. It's syntactically, valid and

I think you can imagine an interpretation of that sentence that's legitimate in the real world. Okay, I'm an internet user and I go down to the beach and I catch a crab.

You can imagine that actually happening, but I'm not sure you'd ever find that sentence uttered by a person. We think we can do better constraining. Ultimately our language generative through using large language models. It should be in principle fairly easy to do so that we generate stuff that's just a little bit more reasonable, little more common, a little more expected. But this is what we've got right now because we're just creating these templates and we substitute in all of the legitimate possibilities where legitimate is just does it meet syntactic and type constraints specified by SUMO.

And so it's possible for an internet user to trap across station.

45:58 So we generate this sort of thing amongst many other possibilities and at the same time we generate the logic expression that you see on the left. I'm sorry, I'm throwing code at you and I don't have the time here to go into the full syntax of this expression at the moment, this is just first order logic. So it's an existential quantifier saying there exists an internet user, a trapping or an ambush is the term that's used in Sumo. Again, remember, labels and definitions are different.

So we used a particular software identifier of ambush but its definition fits with what you would expect of trapping. Okay? And just the word that we happen to be using in English is trapping. The software programming language Identifier we're using in the Sumo library is ambush. So you shouldn't expect those to be identical because we have different labels for things that we may have used consistently in terms of their formal semantics, but their labels may vary.

47:00 So we have an agent that does something, the agent is the internet user. The object of the action is the crustacean and the object of the action is something and that something is a crustacean. So very simple sentence, very simple logic, formalization and so we generate tons of sentences like this for different objects and different agents and different actions. This is your most simple subject verb object sentence and we have lots of different objects that can play roles, different agents that can play roles and different processes that are already defined in Sumo. So we just go through the entire inventory and we also probabilistically generate them.

We look at each of these symbols in Sumo has a connection to Word net and WordNet has word frequencies for each of those concepts. So by looking into WordNet we can see which concepts, formally defined concepts in Sumo are more likely to occur in free text individually.

48:06 And so we do prioritize our generation on the basis of trying to make it more likely using a randomizer in Java and then a frequency table. We try to make it more likely that we generate sentences with things that are more commonly seen. But as we can see, even though internet user may be fairly common in natural language text, you wouldn't likely see it paired with trapping and or a crustacean.

And so it's these pairwise selectional restrictions that we don't have right now, which is why you get these funny combinations.

We can generate time, so we can have the internet user trapstation or trap expectation or will trap or is trapping and its formal temporal logic expression. One thing you often find in simpler knowledge expressions is verbs used as predicates without tense.

49:07 So you might have in a knowledge graph internet user a relation trapping or traps of some sort and then crustacean, right? This would be a more typical expression in a graph and that doesn't really capture what the entire sentence is saying. It says that there is a temporal qualification to what is

happening in the sentence is happening now.

And so there's a begin function of this activity and an end function of this activity and now is between those two things. You have a dictic as we say in linguistics, something that's being referred to that gives semantics to the sentence that's only meaningful in the context of the speaker and the listener here, there now, et cetera. There's a lot of these things and we've got to be able to handle that in order to capture the semantics of the sentence. So it's not enough to say there's a trapping relation between the internet user and the crustacean. We really have to go to a temporal modal logic beyond first order logic with equality to capture this information and capture it in a way that's going to work not just now for this one isolated case, but for all possible cases of these sort of temporal expressions.

50:20 It's also common for people that don't have a background in modal logic to sort of assume that there's some kind of a fudge for this in a graph representation. There isn't, right? This is why logicians work on this stuff. It's hard. Getting it right has taken many, many decades of effort and so you can fudge it, but essentially somewhere down the line you're going to get completely wrong conclusions, nonsensical conclusions, because you don't have the same strength of semantics in your representation. That's why we need this difficult and expressive logic.

We have negation. Negation is another big problem for graphs, right? You've got to be able to negate an arbitrary conjunction or disjunction of other statements. To do that, you've got to have a construct that is beyond a particular relation, that is sort of a contextualized relation. It can handle a whole expression.

Logic already gives you this and getting negation right is really a necessity in logic or in any other representation you try to create.

51:24 Sometimes people try to say, well, I can do that in my graph or I can do that in my description logic. And well, maybe you can for this one point case. But essentially, if you're doing this in your ad hoc representation, that's not a formal expressive logic, you're going to wind up syntactically recreating what logicians have already done and done many years prior and you're going to sort of discover these things on a point by point basis instead of just getting them right from the start. So encourage people to use a representation that can handle the full complexity of the semantics of English and only a higher order logic will give you that.

So here's a case where the internet user doesn't trap a crustacean on negating, this entire complex existentially quantified expression that includes a temporal Coda logic expression within it. What if I have an agent? Patrick desires that the internet user trap across station pretty common to see in news texts. Kevin McCarthy said he believes that the next vote for speaker will be successful.

52:33 Bolsonaro desires that he be the leader of the country again. These are very common expressions in natural language. They're not infrequent and they're difficult and they require this sort of expressive logic and that's the sort of thing that we're handling. We're able to generate this stuff and have a representation that handles it and then maybe, if we're lucky, have a therapeutic that can handle it. Higher order logic.

Therapeutic is still difficult and somewhat experimental, and so that's a parallel line of research. But we're not sort of hand waving away this problem. We're not imagining it doesn't exist or imagining that we can really handle all this complexity and language without this additional set of computational support that we believe is, in fact, really required. So here just making the logic and the sentence more and more difficult. The Plumber knows that Patrick desires that the internet user traps and crustacean

pretty artificial sounding sentence, admittedly but not all such constructs for sentences are that rare.

53:44 You do get a news report, the New York Times reported that the speaker desires that this bill pass without further amendment. That would not be a particularly unusual sentence and yet it follows exactly this pattern.

So one of the things that we need to do and if anyone is at all intrigued by this work and thinking well, how can I contribute? Which would be wonderful, here is the great way to do that one thing that we need that needs to be added to sumo and yet is also fairly well constrained in terms of the complexity of the underlying axioms that need to be expressed. So it's possible to make these fairly formulaic and in fact the use that we make of them inside my Java code that generates logic language pairs is very formulaic and we'll look for only axioms that are of a very particular restricted force in order to do its restriction of which sentences get generated.

54:54 We have this notion of capability, a capability relation in sumo so it says that some kind of thing is capable of performing some kind of role in some kind of action. So ears are capable of hearing, guns are capable of shooting and we have to talk about things that might happen in order to restrict the sentences that we generate.

We'd like to be able to say that John can't be eating a boat dock, we might generate something like that now but we want to be able to restrict eating to have an object transferred or a resource that is only food, ideally food for that particular animal. So we want Termites to be able to eat boat docks, but we don't want Bob the Plumber to be able to eat a boat dock. And we have a lot of processes and we have a lot of objects. In sumo. And so creating an inventory of these different capability expressions would eliminate, I think, a lot of the kind of funny sounding sentences we generate now.

56:00 That said, I think it's not catastrophic that we have some of these funny sentences because remember that we're ultimately trying to go from organically occurring natural language text in the wild in order into arbitrary logic expressions. So the fact that we have other sentences that kind of have a similar form but don't really make a lot of common sense may not be a big issue because as long as we have enough examples of things, that are reasonable, that do cover the vocabulary we care about, that do have reasonable selectional restrictions that will still enable us to translate the good stuff that occurs the sensible sentences into sensible logic expressions and the fact that we have a corpus of translations where the translation is reasonable but the core sentence is kind of bogus. May just be background noise. We may be able to get away with that. But even if that proves not to be a problem it still would be better for us to have in Sumo these capability expressions just to add to our knowledge of how the world works.

57:05 So if anybody is intrigued by this, I'd love to help you get started because I think it would be not too hard to do and would be of great value to the project.

So we have a lot, as I've said, I've said a couple of times now, we have a lot of processes, we have a lot of objects. What do I mean by a lot? Well, here's just a segment of the natural language comments on a portion of the hierarchy, starting from the top. Here are all the process types. In SUMO, we have 1300 of them at last count and here are a number of them.

So in a lot of sentences we don't restrict what processes can hold for a given sentence. And we'd like to have again these selectional restrictions based on capability expressions for the type of objects that can participate in the types of processes, so that we don't get silly stuff.

For each of these processes we have mappings to verbs and word net. So that's what allows us to cycle

to iterate through all the process types and still generate a sentence, not using the labels in Sumo, but rather using the labels, the legitimate words in the word net dictionary.

58:18 So for example, for Ambush it's mapped to a particular numbered sense set a collection of synonyms, collection of synonymous words in WordNet that linguistically identifies a sense in English. And so we have ambush that's mapped to since that blah blah blah 926, it has a certain definition, has a certain set of synonyms ambush, god, ambush, lying in wait and a trap. And that's why we saw trapping earlier.

The internet user traps a crustacean that was exactly this equivalent symbol sunset. For ambush we have lots of case rules. So these are the ways in which entities can participate in events. There are some simple systems, including some simple systems in linguistics. If you look at, I think Verbelen has some of this set of restrictions.

So agent, patient, instrument, resource, these are very common and you'll also see agent, zero agent one, sometimes as labels in semiformal linguistics, talking about the roles that entities play in activity.

59:36 And so occasionally I've gotten this question, which I think is funny, is like, well, why isn't it enough to have just agent, patient and instrument? Well, yes, that does cover, at a high level, maybe exhaustively the roles that things can play. I mean, James Pustovsky also has this set of qualia relations. This is a common, I think, approach in linguistics to try to explain language through a small number of covering predicates or abstractions. And while that might be fine for abstract linguistics in practice, there's a lot more that we can say about the world, about how things interact in events that happen.

And so we've come up with 67 of them. And I think this is, you know, many orders of magnitude too small, but it's a good start. It's the stuff that we've found that's been useful over the years. So invading virus catalyst broker enjoys reagent all of these relations. Many of them are subrelations of things like agent and patient and instrument and resource, but they're more specific.

1:00:43 They entail different additional knowledge when they're used. And it's knowledge which fits with our knowledge of the real world. We have lots of objects. So here's just the corpuscular objects, things that are not substances, that have Parr, and we have 930 of them. A lot of them are content bearing objects or things that might have writing or symbols on them that are common in our world, as well as organic objects like organisms.

There's a lot of these, and they all play different roles in different activities. Here are some of the roles that human beings or agents can play. You can have a certain profession. You can be a plumber or a programmer, and you can switch freely from one to the other. This past weekend and made my toilet breaks and I have to be a plumber.

Today I'm a programmer. Maybe tonight I will be teaching a course, and so I'll be a lecturer.

1:01:45 These different roles are free and flexible and large, and they also are temporarily qualified, right? You can play different roles at different times. So it's not enough to say, as WordNet unfortunately does, as a linguistic product, that a subtype of human is plumber.

Well, you can be a plumber one time and not be a plumber another time, and that doesn't mean you cease to be a human. It's a bad relation to have formally, although it's a reasonable relation to have linguistically, because you can do the substitution test that linguists do of substituting, say, a superclass for a subclass. And it works in a lot of sentences, but it doesn't work formally, doesn't work in logic. So we need to separate out these rules and say that they're transient, temporally qualified rules that people

can have. So we can generate against bogus stuff like Robert Kills a boat dock.

It doesn't really make any sense? It makes sense syntactically, but we need these capability restrictions to handle them. And a lot of times we do have restrictions, like the rule below here.

1:02:49 But I can't afford to do theorem proving while I'm doing generation. If I need to generate a 1 TB corpus of language logic pairs, I can't prove the truth of every sentence.

This would take many years just to do one run of the generation system. I've got to find a way to optimize that. And the way to optimize it is just to look at capability expressions, even though the following rule that I'm showing up here says that a killing has to have an object that's an organism and a boat doc is not an organism. So this would be trapped if I could do theoremproving for everything I generate. But I just don't have time.

So I need to use this more restrictive form that I can turn into a sort of a lookup table and make run much faster. Okay? So I need this following expression that if there's a capability of killing the killing event, only the patient has to be an organism. If it's the patient or the object of the killing event, it must be an organism.

1:03:50 I also had the question of how much is too much?

Right now I am able to generate, say, a terabyte corpus in the span of a couple of hours and then train on it. Training takes about 20 hours on a multi GPU, fairly beefy server. How much is too much? I don't have a sense of that. Hohwy much is it too little?

We're getting good fidelity. We break things up in the standard train test split and we get good metrics. We get a perplexity score that's down around 1.11.2. That's one reasonable measure for the fidelity of your deep neural network training approach. But is that good enough?

I don't know yet. We just need to do a lot more experiments. So we have about 100 names that we use. We have a corpus of common human names. We have about these 300 roles.

We have lots of different propositional attitudes, each of which could be negated.

1:04:55 We have modals like can, may, should that are also formally expressed in our modal logic. Those in turn, could be negated. We have 400 different sorts of roles like plumber and teacher, 1300 processes. Those in fast could be negated.

And we have past, present, future, and then the progressive tenses for each of those. All of those possibilities, we've got 930 direct corpuscular objects. And then we've got substances which I haven't listed on here. We've got lots of substances like water or gasoline, 67 case rules. And again, 930 of those same corpuscular objects that can be indirect objects.

And then I've also added a bunch of politeness expressions like can I may, I should do the following so a lot of these modal expressions can become politeness. So please. May I? Et cetera. So we can handle stuff in chatbot or conversational, kind of stuff that's very commonly seen.

So I did an inventory of stuff that a lot of the chat bots use for their sentence patterns to make sure we cover that.

1:05:59 But a lot of these combinations are nonsense, so I've got to somehow Parr it down. If not if it's not a problem in terms of creating a lot of junk that can otherwise get ignored when I'm translating a sentence, at the very least, it slows down training. And so as I make the corpus larger, I have to pare down the nonsense just so that my training phase is possible, because I don't have the resources of OpenAI to spend \$3 million a month on running their chat GPT for the world, right? I've got to run on one server that my friend and Prague has access to.

So there's lots more that I can still do. Units and measures. I've only begun starting to do that. I've got a little bit of that. So I can say Bob puts three birds in his wagon or £3 of steak in his jar, something of that sort, because we've got a whole set of I think there are about 60 different system international units and Sumo has them all.

1:07:04 And then, of course, your values can be within all sorts of different ranges. We've got logical combinations of sentences that I could generate. So Bob goes to the store and Bob buys cookies. We can temporarily qualify all of these things. I've got a little bit of that already.

So Bob goes to the store on Tuesday, bob goes to the store on January 27 of 2003. We have a bit of an issue that a lot of these case rules require different prepositions, so I don't have those working correctly. I need to actually do a whole set of patterns that are a little fragment of framenet, if you will, for different action types and which particular prepositions are historically used. So you get in the car, you get on the boat, you go to the station. I don't handle all of those correctly.

Things we can do, like with more than one agent right now. Agents are always singular in my examples. We can certainly have Jack and Jill go up the hill. And not just Jack. I have now just implemented quotes for said so we can handle this common form for news reports or novels or what have you.

1:08:12 Lots more things that can be done to make this a more comprehensive corpus. Long before I run out of stuff to do, I can cover a lot larger portion or subset of English.

So here's just some detail on how we run the model. I just got a script. It's not too hard, it just takes a long time. So culture work. Once I get done exhausting the set of things I can think of that I can do, then we need to start seeing where it fails on natural text.

A big thing to do is to use either word to VEC or some large language model, maybe as a way to tear down the number of nonsensical combinations along with my preferred approach of having a lot more capability. Sentences which will then add to the understanding, the competence of this core model regardless of whether we're trying to do language understanding or some other kind of project. And I really want to do a comprehensive test on the baby corpus to just see if I can do better.

1:09:13 If I'm not learning and teaching a machine learning system how to do inference, but using a system that already knows how to do inference, I think I should be able to get better scores. But that remains an empirical question.

If I can do that, I think maybe then people will pay more attention to this work. So it would be a nice demo. Okay, that's it. So thanks for learning about this early work as well as the more mature model that it's based on that's already being applied commercially and would love it if I got somebody on this call so excited that you wanted to do some work together.

1:09:48 Daniel:

Excellent. Thank you for the presentation, Adam. You can unshare or leave it up, and we will begin with Dave providing some initial comments and questions.

1:09:59 Dave:

Yes. One idea.

(Can you hear me now?)

1:10:02 Adam:
(I can.)

1:10:03 Dave:

Great. One idea that occurred to me several points during your presentation was if you put part of the Pruning inside the model by creating persistent agents with goals and beliefs - Bob isn't just a transient pattern of activations, but Bob is something that stays around for a while. And perhaps Bob's desires evolve more slowly than Bob's beliefs, including Bob's beliefs about Carol and Carol's relevance, and whether Carol believes correct things about what he's interested in right now. Have you gotten around to something as charming as that?

1:10:51 Adam:

Well, that's what the "baby" folks did, right? They've got this model world model, and they generate sentences and variations of those sentences based on that world model. And we could certainly re-implement the simple system that they've done in Lua in a more complicated blocks world, if you will, that uses all the entities and objects that are in sumo. That's something I'd like to do.

1:11:22 Daniel:

(Okay, reconnecting here.)

1:16:35 Adam:

(We've got Daniel back.)

(Okay, maybe he's having some back Internet issues.)

1:16:45 Daniel:

(Yes, I had to move to a backup force for a strange reason. But let us continue the conversation with no video.) And yes, the last thing you added was the capacities and the opportunity to contribute there. Please feel free to pick up there, or I can pick up there and take it in a different direction.

1:17:11 Adam:

Yeah, Dave and I was just talking a little bit further about maybe some other work that's going on in psychology and ontologies of emotions, and I was just commenting how SUMO already has extensive ontology of emotions but of course it has ontologies of many other things. So anybody that wanted to study a representation of emotion for use in psychotherapy or any other reason would kind of get for free all of this other stuff. Defining model of the rest, which would, I think make that work, have a greater impact over time and be more readily adoptable in different concepts because it has this link to a whole host of other semantically defined entities.

1:17:57 Dave:

Yeah, one of the things that would make it a lot easier too to participate across groups is if it were easier to actually run Sigma. Several of us have tried to install and run it. I don't know that anybody in the active inference group has ever succeeded for the appropriate environment. You have some really

nice videos and I've tried to follow those. In every case it broke down with some warning like if you keep going, your Windows system may self destruct.

Why don't you stop now? So it's pretty scary. We even tried to go to a vendor, a provider of cloud services systems.

1:18:42 Daniel:

Like Sumo et al. That provide rigor in how those different states get translated and mapped to each other like short text to long text, long text to short text, short text to mouse movement.

All those different mappings are implausible in multiple ways without a formal semantics the kind that you're describing.

1:19:08 Adam:

Yeah, I agree. Maybe I could say a word about uncertain reasoning, because this is an area that I think a lot of people that are used to being in a probabilistic representation often have some misconceptions about what's possible in logic. Usually. Based on the fact that the logic that most people are exposed to academically is going to be first order logic where you do have just truth values.

Binary truth values, right? It's either true or false. And people want to be able to represent things that are uncertain or likely or unlikely and therefore assume that logic can't handle that and while it can, it's just you have to go beyond first order logic. And so that's exactly why we're in the realm of higher order logic and sumo and have modal expressions. We want to be able to say and can say in the logic that something is likely or that one thing is more likely than another or that because of such and such event something has become more likely.

1:20:17 So we have to make a projection that includes some temporal reasoning as well. So logic is maybe you could call it a non classical logic, right? I find that a little uncomfortable because most non classical logics are things like four valued logics or things that don't just have true and false. Well, modal logics do still just have true and false but they are able to describe an entire situation with all its possibilities and conditionals things like possible worlds which allow us to capture some of these same intuitions that a probabilistic representation like Bayesian reasoning can. I think there's to my mind a clear advantage of the symbolic representation.

It is possible, of course, to have large libraries of empirical data, to have realistic probability assessments and say that such and such an event is 73% likely in this context.

1:21:26 But a lot of times when I see people using probabilistic or Bayesian representations, they're sort of using them. Mathematics is precise, the source data is often not precise. People are making a guess, I guess, that it's probably 70% likely that the following will happen. Then they encode that in their system and I think those things are better represented as symbolic preferences.

This is more likely than that, but we don't know that it's 73% likely versus 56% likely. An additional problem I think often not maybe not in your case, but in certainly in many cases with people that are doing numeric probabilities is that the numeric probabilities force a total order essentially. And you've got these appearance of precision where 73% is more likely than 56%. But in reality, if that comes from anything less than statistically rigorous empirical data, then it's just sort of a confidence factor like was used in a lot of old style expert systems.

1:22:37 That doesn't give you reliable results.

So none of those criticisms may actually apply to your situation. I don't have enough detail about what you're doing, but I ask you to consider those points.

1:22:51 Daniel:

Awesome. I'll give a thought on that and then Dave happy to hear from it. So for any given model architecture symbolically represented like a symbolic regression or a Bayesian graph constraining on that model structure which in the variational inference world we just call Q .

It's just the variational distribution, the one that we are getting to choose, the family that we're parameterizing it from. So it's like our designer symbolic model and why the data come in. So then free energy, which is the imperative for inclination, whether it's the real time variation of free energy happening right now or it's the planned or anticipated expected free energy in either of those cases, for a given generative model, given Q distribution and given set of data, you might have a very high confidence in something that was very inadequate.

1:23:58 And so in that way it's like there are distributions of scattered plots that can confuse a least squares regression or you get some sort of misleading model outcome and it may even be invisible. You may be able to look at diagnostics of the linear regression and tell that some sort of error or aberration is happening.

But you also may not like a kind of unknown unknowns question. And in the case of cognitive modeling, these state spaces can be very large and they're often trained with sparse data. So this problem of overfitting and conditional and dynamics relationships so that something can be like true and trained upon for 100 video frames and then something chance, and then there's a nonlinear change in what can happen next or how things unfold through time. So that's kind of like the complexity of action. And perhaps could you speak to how action is represented or treated or how the concept of action selection and planning and strategy and implementation impact?

1:25:11 How did these features develop in sumo through time?

1:25:21 Adam:

I think there was more to that question. It sounds like Daniel maybe having some connectivity issues.

1:25:26 Daniel:

That was it just how did the concept of action evolve through time?

1:25:32 Adam:

How does the concept yeah, let's see. There was a lot packed into your question.

I'm not sure I grasped most of it, but let me just maybe make I'll give myself a related question, which is how do we handle the fact that sometimes in machine learning systems, the mathematics has such linearities that we get nonsensical conclusions? I think that's a fairly common problem. So I think that the interaction between these two models of the world, the statistically machine learning derived model and the symbolic model, is really important so that our machines don't go off the rails, as it were. I'm using a term called cognitive guardrails, and I think this applies well to people. People have instinctive action to many things, and we have at the same time a generative cognitive mind that constrains those reactions.

1:26:36 We might react emotionally and react badly to something that sort of triggers us in some way, but then we have another process in our head that says, well, we're in a social situation. And so, no, I don't want to punch this person that's offended me, even though I might be tempted to do so. If we have a self driving car that sees an object in front of it, my initial reaction might be to plow through it, but the Costa to doing so might be very high. So maybe I want to swerve or brake gradually. And having this interaction between these two models, I think can address some of this problem that we see with nonlinearities of Modern machine learning systems that do work really well in a lot of common cases and then sometimes just veer off into infinity, as it were, and do something completely ridiculous.

1:27:28 Daniel:

Yes, I appreciate that response. I think it's exactly like you described with overfitting to the meaning and going off the rails from a cognitive perspective. There's like the known unknown variants as.

1:27:44 Adam:

And example especially there's University of Wyoming, I think did a nice paper maybe quite a while back now, probably at least five years back, that would say, yes, it's a cat or a dog because it was just training on sort of a coincidental feature of pixels that maybe approximated the fur of the animal. And so it couldn't recognize it recognized a test pattern as being a cat, and it couldn't recognize a simple cartoon drawing as a cat because it never saw anything like that and wasn't looking at the essential features of the world.

And so machines are really great at finding correlations, but that doesn't mean they're the right correlations. And I think that's at the root of a lot of things that we call overtraining or overfitting, but we never know, really, whether it's fit correctly or overfit.

1:28:45 Sure there's some numeric measures that at times can tell you. Okay, we're bouncing out of this local minimum and our fidelity has gone down numerically. But we don't know what that really means in respect to the common sense, larger world that is represented by something that's much larger than any given test set that we might be training on.

So I think that having a sort of completely different representation of the world can help guard against some of this. And so I thought of using machine learning and symbolic systems together where the symbolic system addresses problems in the input data because faulty input data, erroneous input data is always a big problem. How do we know we're training on the right stuff? How do we know that if we're training on a system to diagnose check fraud, for example, or credit card fraud, that we don't have some bogus data that's crept in here there that might have a spurious correlation?

1:29:50 We've got, say, zeroed out Fields or a field of 99.

That's really a don't care field. But it's in the data, and there's no easy way for us to discover it. If we've got terabytes of data, wouldn't it be nice to have a common sense system that could act like an intern, a human intern working at warp speed, looking over all of our input data to make sure we've not included any inputs that are at odds with what we know about the real world? And then on the output side, making a few inferences in our symbolic system could help the overall computational system not make really dumb conclusions like, oh, we don't know what that thing is, so let's run over. It might be the conclusion that a statistical system has, but we'd like to have our combined agent be able to trap things like that in the same way that a person wouldn't presumably punch somebody else who's offended him

because they know that that's both illegal and immoral.

1:30:56 Thanks.

1:30:57 Daniel:

Dave, do you want to add or ask something?

1:31:01 Dave:

Yeah. A couple of the other approaches that sometimes simplify exactly the same kind of problems that Daniel and Adam have both been cutting through. One is fuzzy logic.

Sometimes a fuzzy solution and a probabilistic evolution agree numerically, but in other situations they don't. And in many of those cases, the fuzzy answer turns out to be way back when Lotfi Zadeh defining and reacting to situations. And it wasn't until he started calling the logic fuzzy that people were beating on him. He was just coming up with the great results. Another approach that's been impressing me is David Deutsch and Chiara Marletto's work in Constructor Theory, where they start by asking what can be done and what can't be done, and what does it mean to do something, what does it mean to be able to construct?

1:32:07 They generalize the notion of catalyst, for instance, as a general concept in systems theory, and they've just identified a lot of things that people spend a lot of brain power on as just simply being nonsensical. "You can't construct that. Therefore, you're wasting your time. If you try to work out how it could be done, would be done with the consequences. If it could be done, it just can't be.

So go on to something interesting."

1:32:44 Adam:

Well, I'm not familiar with the latter work, but I am a bit familiar with fuzzy logic. And maybe I should just mention that this has been another thing where people that are familiar with fuzzy logic wonder how you could do this in a non fuzzy logic. And the challenge then comes down to, again, people that aren't familiar with logics beyond first order. A lot of things we might say about, like, "John is tall." Okay, and maybe John how do we compute whether John is tall?

We have this problem of well, in a traditional logic system, we would have the problem that, okay, if he's above 6ft, he's tall. If he's below 6ft, he's tall. Therefore, if he's 5ft eleven and seven eight inches, he's not tall. And of course, we all know from a common sense point standpoint, that's ridiculous. And so we'd want to use fuzzy logic.

But in fact, what we should be able to say is that he's tall with respect to a particular group.

1:33:51 And if we can make these more complex statements, generally involving sets of things and conditionals that are beyond first order logic, we can express these things perfectly well in this more expressive logic. We don't always have to go to fuzzy logic, because one of the challenges that I think we have with fuzzy logic is, again, a lot of the people that did work in this sort of world back almost 30. Years ago came up with things that were functioning well empirically as sort of hacks because they didn't have data about what is the actual statistical value curve of Tallness. It was just somebody's guess about what the curve of tallness membership should be, for example.

So I like the notion of fuzzy logic intuitively. I'm not sure that I care for it so much on a theoretical basis and on a practical basis, there's simply another way to do it.

1:34:59 Daniel:

Very interesting. Dave, want to add anything?

1:35:03 Dave:

Yeah. Now, if we have time, this is a much deeper question about the way the sumo rules are written is, do we have time to go in a lot of depth in this venue?

Not necessarily breadth, not necessarily a big answer, but okay, I'll just ask this reading through the rule, a number of rules have very specifically asked in the first participant of a relation if the following holds, then the following logic applies. If a different set of conditions on the recipient of action, indirect object or direct object of a process hold, then you can apply the following logic and I look at that and I say wait a minute, these different rules are splitting off very different kind of real world situations within the same Sumo class.

1:36:20 Well there's got to be two things have to be true in order to do that, in order to find these differentiating rules within a single class. First, you call those by the same name in whatever natural languages you were thinking in when you wrote the rules. And second, the formal literally the abstract algebra of these various cases have to all be analogous, they all have to be isomorphic.

And my name for that situation is you're covering analogical cases in the same Sumo class and in that sense the Sumo class itself is and analogy, it's an analogy among these various cases which in different domains of applause. Am I just not thinking discerningly enough about what a Sumo class represents or can a Sumo class actually be a higher analogy?

1:37:29 Adam:

It's hard for me to know because there's no analogical reasoning happening so I'm not sure you're talking about analogies and there's no natural language happening once you're in the formal system. So I'm not sure why you mentioned natural language. I think we'd have to look at a very specific example and then we could run that example on a theorem proof and you'd see exactly how things behave. It's not something about classes per se. You just mentioned how the rules work in a logic.

You've got preconditions for the antecedent of a rule and you've got the coherence of a rule and yes indeed the symbols in that formal system have to match. That's how logic works, right?

1:38:10 Dave:

I'll get some specific examples of an email to you.

1:38:13 Adam:

Okay, sounds great in our last minutes.

1:38:17 Daniel:

Here and also apologies for the internet disrupt.

This is part of our long term journey learning about this and applying it active inference lab. So we're going to keep this thread alive and be able to bring some more informed areas of intersection and application. But one that I wanted to mention was to return to your type one, type two thinking and the

way that there's a faster reflexive or habitual statistical type one inference and then a slower type two, what you referred to as deliberate and deductive that involves symbolic logic. And in chapter five of the active coherence textbook there's a schematic of a Dopaminergic neural circuit in the central and peripheral nervous system where there's a Dopamine related parameter computationally that's like a dial in the model that is controller the balance between replicating actions according to the prior like just the habit.

1:39:36 So it's some distribution of capacities, affordance and then if there's three options, it's zero 8.1.1.

So then under the most habitual setting, like what we might even call nonagentic or subconscious, then that vector of affordances is recapitulated in action selection. And then in a different dopaminergic regime there is increasingly fine scale and potentially overfit application of the free energy. And that can reflect like a symbolically informed, deep generative model of the world which captures semantics of the world like nested timescales and temporal priority effects. So it's interesting that this area, while deep hierarchical generative models often are described as if they have semantic value. However, that is usually just sort of specified or only explored in a few limited cases.

1:40:42 So integrating and working at that mapping between the top down symbolic and the bottom up data driven is going to be a really important, interesting area. So as a closing round or question, what do you see in the ecosystem and or what directions, if you didn't state them earlier, you're working on this year?

1:41:10 Adam:

Well, yeah, the generative AI stuff certainly got huge amount of press. It's fascinating as an intellectual exercise, certainly. And it's fascinating to have systems that can complete sentences and write stories and always have an answer to a question.

But as we're seeing with a lot of newer things that are being written as follow ups to all the excitement that people are showing how these systems will give you an answer, but it's often a complete hallucination. Right. So one recently of saying, well, you know, Hillary Clinton was the 43rd president. Well, it came up with that somehow by foraging together various texts that other people have written, but it doesn't make sense together. It's not true.

I think these systems are interesting in the way that a parrot can have a very limited conversation.

1:42:14 You can teach your Parr how to say Polly ant a cracker and say thank you and you give it a cracker. That doesn't mean it actually understands anything about what it's doing. It's just a stimulus response. Now, first there is a philosophical question if the stimulus response pairing is complicated enough.

Well, isn't our human beings also just collections of stimulus and response? That may be true, but I don't think we're at that level of complexity yet for our machines. And I think some of the things that we can get these generative models to do that are so obviously nonsensical human are good evidence that they're certainly not sentient and they're not even smart and they're certainly not very reliable. If we have very challenging or high criticality tasks to worry about, I think they'll be useful in the same way that Google search is useful. It doesn't always get the right answer, it will universally get many wrong answers, but it's very powerful and useful tool when you have a human being available to filter out all the nonsense and just use that as a filter for what they'd otherwise have to do.

1:43:26 It's not feasible for a human being to answer question by reading all the books in the library. But if Google can give you ten books and ten pages in those ten books where your answer may be found, it's done you a great service, even though it's given you Winnie Lapoo along with the Handbook of Analytical or Organic Chemistry or something where your actual answer is found. So in that same way, Chat GBT might be very useful in breaking your writer's block to help write the summary for a proposal or a student essay, but you certainly don't want to have it write the whole essay for you and hand it in because you'll probably fail. So I'm not quite as worried about high school or college essays as some commentators have been. I think it'll be a useful partner, and that's about all. And we've got to understand these limitations and ideally pair it with a symbolic system that really knows something about the real world so that they can work together.

1:44:25 Daniel:

Awesome. Dave first, and then Adam, what are your closing thoughts or reflections?

1:44:38 Dave:

I've enjoyed this very much. We've got lots of work to plow forward and do. I'm glad there's going to be comma work that we can become affordances for one another on. Thank you, gentlemen.

1:44:58 Adam:

Well, thank you, Dave, for the good questions, good comments, and Daniel as well, and for setting this up. I guess I would encourage what I've maybe encouraged in the past is a lot of the things that we've talked about in these several conversations we've had are general, right? We're saying, what's an overarching direction for the field or something? And I think that if we got down into some real specific examples and actually doing some representation together, then a lot of these more philosophical notions would go away and we'd get down to doing something that has practical utility. That's why I also encourage actually having, if people do, say, some of the sumo exercises together, because I think, Dave, some of the questions that you were having over what is a rule? How is it working? I can answer all those questions and give you some very precise experience that would give you a real, actual understanding of how logic works and how sumo works.

1:46:04 If we took the time together to go through those exercises, do some work with the theorem prover, and it would clear up, I think, a lot of maybe misunderstandings or misconceptions and give us a real path to do something concrete together. Because although the talking is great fun, it's the doing and writing a practical computational system that hopefully we all want to get to.

1:46:27 Dave:

As the Jesuits say, "Let us descend into the particular."

1:46:33 Adam:

Yes. That's great. I love it.

1:46:36 Daniel:

Great! - Ending the stream.

