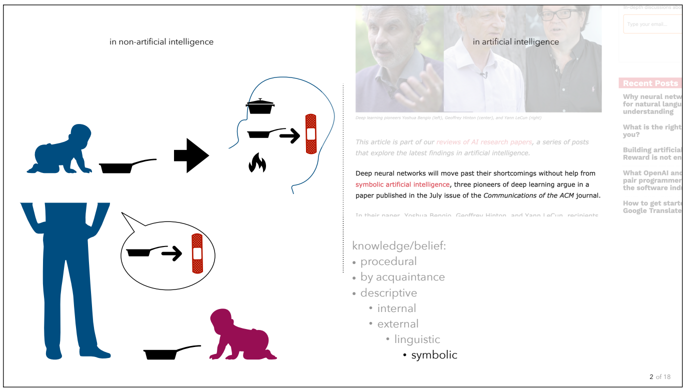
The purpose of this talk is to introduce two things. First of all myself. I've worked here for a while, but I've recently started as assistant professor, so I thought I'd take this opportunity to set out the sort of things I plan to work on.

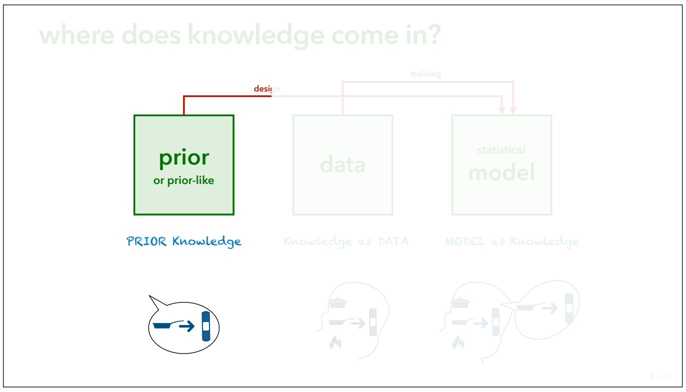
Second, the learning and reasoning group. This is one half of the recently split up KR&R group, of which I am a member. This group will focus on the interplay of machine learning and (symbolic) reasoning. This talk is my view on this intersection. They are very much my specific opinions, and other member of the group may say very different things.



When we produce non-artificial intelligence (also known as children), combining knowledge and learning is the most natural thing in the world. A child may learn through experience that touching a hot pan hurts, but a concerned parent will try to limit such personal experience as much as possible. We do this by distilling our own experiences into knowledge representations (in this case the phrase "touching a hot pan will hurt") and hoping that the child heeds our warnings.

So why then, when it comes to artificial intelligence do large parts of the learning community seem to reject the help of such symbolic prior knowledge? Why do we insist on learning everything from scratch?

Note that I'm casting a slightly wider net with the definition of knowledge than the common definition of a "justified true belief", since the definition doesn't allow us to distinguish between the beliefs that are knowledge and those that aren't before we use them.



I'd like to discuss today what knowledge in general can do for us, and what symbolic knowledge specifically can (and can't) do.

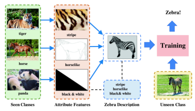
Here is a simple way to structure the roles knowledge might play in learning processes into three different categories. The simplest way to understand them is by analogies to the way children make use of knowledge when they learn.

- The first is knowledge as a prior. This could be a prior probability distribution, but it could also be something that behaves like a prior like a regularization loss term, or a particular weight initialization. When we teach children, we often tell them knowledge directly that we hope may be useful as prior information. For instance "if you touch a hot frying pan, it'll hurt". It's up to the child to decide whether this knowledge applies in a particular setting (is this a pan, it it hot, and if so, does the rule apply in this setting)
- Knowledge can also be the input to a learning mechanism. From many different instances of specific knowledge, we may for example infer more general rules, or certain consequences. for instance, a child that is told that a hot frying pan hurts and that a hot pot hurts, may infer that it is likely that all hot cookware hurts, regardless of shape.
- Finally, the output of a learning process may also be considered knowledge. This could be as simple as "this email is spam", but we are increasingly capable of learning rich structured outputs. This corresponds to when our child learns to speak and is able to confer the knowledge it has learned to other people, for instance through language.

All three are goals of the L&R group (and of myself), but the one I'd like to focus on today is the first: using prior knowledge to help us make models better *before* they start learning.

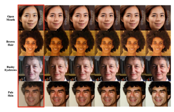
The benefits of *prior* knowledge

Out-of-distribution learning
Low-shot learning
Interpolation



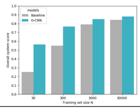
Tian, Y., Zhang, W., Zhang, Q., Cheng, J., Hao, P., & Lu, G. (2018, December). **Co-consistent Regularization with Discriminative Feature for Zero-Shot Learning.** In *International Conference on Neural Information Processing* (pp. 33-45). Springer, Cham.

Disentanglement



Nie, W., Karras, T., Garg, A., Debnath, S., Patney, A., Patel, A., & Anandkumar, A. (2020, November). **Semi-supervised StyleGAN for disentanglement learning.** In *International Conference on Machine Learning* (pp. 7360-7369). PMLR.

Data-efficient learning



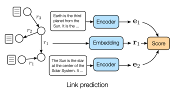
Winkels, M. **Group-Convolutions: Overcoming the data challenge in medical image analysis.** MSc thesis 2019

Here are three of the places where knowledge might help.

The benefits of *symbolic* prior knowledge

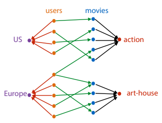
Out-of-distribution learning
Low-shot learning
Interpolation

Disentanglement



Daza, D., Cochez, M., & Groth, P. (2021, April). **Inductive Entity Representations from Text via Link Prediction.** In *Proceedings of the Web Conference 2021* (pp. 798-808).

Data-efficient learning



Wilcke, X. et al. (2017). **The knowledge graph as the default data model for learning on heterogeneous knowledge.** *Data Science*, 1(1-2), 39-57.

Downside: highly use case specific.

The *downsides* of symbolic prior knowledge

- The platypus problem
- The rhinoceros problem
- The chair problem
- The spork problem

Let's look at four examples of how we use symbolic knowledge in everyday life that show the downsides of relying too much on it.

The platypus problem

No mammals lay eggs. Only birds have bills.

This doesn't mean that these rules are *useless*, just that there are occasional exceptions. More importantly, there will be occasional exceptions that we cannot account for a-priori.

We will observe them in the wild, and we will need to decide on the fly whether to trust our knowledge, or our eyes.

With a little creativity, I believe you can come up with potential counterexamples to any rule. This was a famous point of disagreement between Russell and Wittgenstein when they first met. The latter asserted that there was no such thing as a "truly knowable empirical fact". Russel suggested the statement "There is no Rhinoceros in this room." Apparently Russell even suggested looking under the desks. Wittgenstein's point appears to have been that it was merely very unlikely that was a rhinoceros in the room but not fully impossible.

I'm on Wittgenstein's side. We don't need to go so far as to image microscopic or invisible rhinoceros. With a little creativity, we can, for instance, imagine the possibility that one of the people present had a rhinoceros keychain. That would be a coincidence, but certainly not impossible.

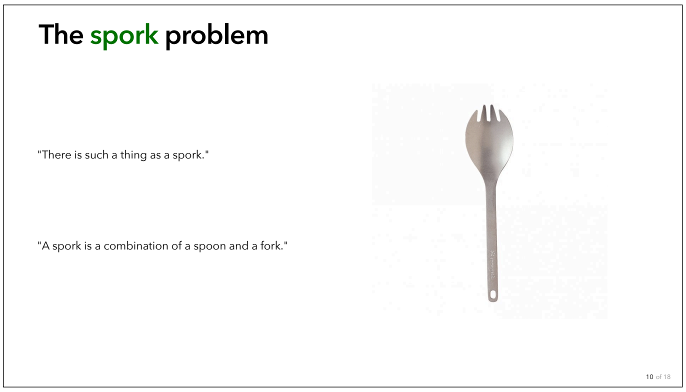
You may argue that this is cheating. Russell was surely referring to *actual* rhinoceros. But for our purposes, at least, this is an important point. If we are talking about small probabilities, we must consider the possibility that the original statement was poorly phrased, or ambiguous. It's truth depends on our interpretation and the context in which we apply it.

When we start using knowledge in the real world this distinction very quickly stops being academic. The way we frame symbolic knowledge is usually extremely dependent on the context in which we use it. A statement of knowledge is almost always tailored to the specific context in which it is going to be used.

Another problem is that there are certain concepts that are simply difficult to define in simple terms. We all know when something is a chair, but when you start making rules, like "it must have legs", "you can sit on it" or so on, it becomes very easy to come up with counterexamples. Things that break the rules and are very clearly chairs, or things that satisfy all the rules and are very clearly not.

Wittgenstein used games as the prime example of this type of concept, and called them *family-likeness terms* (Familienänlichkeiten).

If it's so difficult to define precisely what makes something a chair, a soup or a game, why is it that we use these concepts so easily? Probably more easily than we do concepts with very precise definitions, like "right-of-way", "finite-state-machine" or "submission deadline"? I think the answer is that we use *learning*. We see two or three examples of a chair and we get the general idea. As we go through life we see more examples and counter-examples and we refine our internal representations.

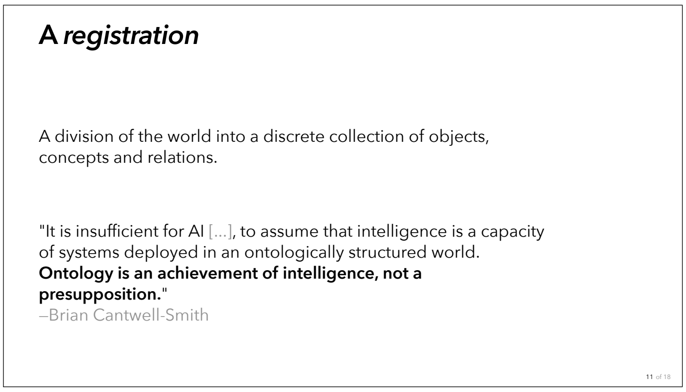


Finally, and most importantly, there's the spork problem. Imagine that you don't know what a spork is. I can tell you that there is such a thing. Even though you don't know what it is, or anything about it, you have no problem processing the information that such a thing exists. As we speak, you are creating space in your head for the concept of a spork and perhaps making some educated guesses about what it might be.

Then, as I tell that it's a combination of a spoon and a fork, you start to fill in the blanks. You now know its approximate shape and size, and you know what it's for. There are a few ways one might combine a spoon and a fork, so you still don't know exactly what it looks like, but you can already narrow it down to a small and finite number of possibilities.

Then I show you a picture and your idea of a spork is complete. Now, whenever you come across one in the rest of your life, you can recognize it. Even though you'll probably never come across one that looks exactly like this.

On the fly, with zero effort, based on almost no knowledge, you have created a new concept and tied it into the rest of your internal semantic network.



I believe all of these issues emerge from one single problem in the way symbolic knowledge is used on all neurosymbolic approaches being studied today.

The problem of registration. This is a phrase coined by philosopher Brian Cantwell-Smith. An intelligence's registration is the way it takes its collection of raw, continuous input signal, and organizes them into a (mostly) discrete picture of the world. In short, the way it maps observations to symbols.

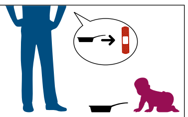
The point that Cantwell-Smith makes is that building a registration, including the vocabulary of symbols must be part of a true intelligence. An agent must be allowed to build its own registration, it's own collection of symbols, introducing new ones as the need arises.

If we take our registration, our ontologies, and limit the agent to that particular registration of the world, it can never be truly intelligent, and one or all of the four problems we saw before will emerge.

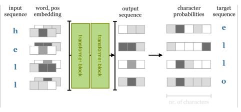
That doesn't mean we can't use our own knowledge to help intelligent agents emerge, only that our knowledge can't form the internal registration of the agent. It must be outside of the agent, and the agent must be able to accept or reject it as it chooses.

How do we allow an algorithm to develop its own registration, while guiding it with the symbolic knowledge we have?

A simple option: externally



Transformer model



Data augmentation

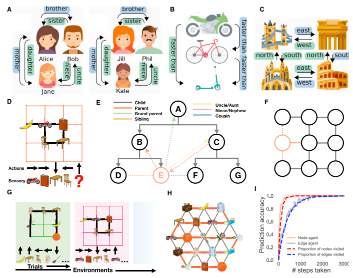
$\forall_x$ Human(x) $\rightarrow$ Mortal(x)
Human(Socrates)
—
Mortal(Socrates)

"All humans are mortal. There is a human called Socrates. Socrates is mortal."

"There once was a person called Scorates. Since people are mortal, so was Socrates."
(sic)

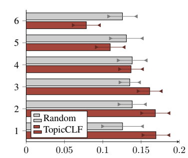
"There once was a human called Socrates. He was mortal. This stands to reason, since all men are mortal"

More complex: internally

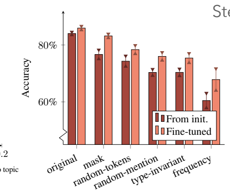


Whittington et al. (2020). **The Tolman-Eichenbaum machine: Unifying space and relational memory […].** *Cell, 183*(5), 1249-1263.

First results: Probing the representations of named entities in Transformer-based Language Models

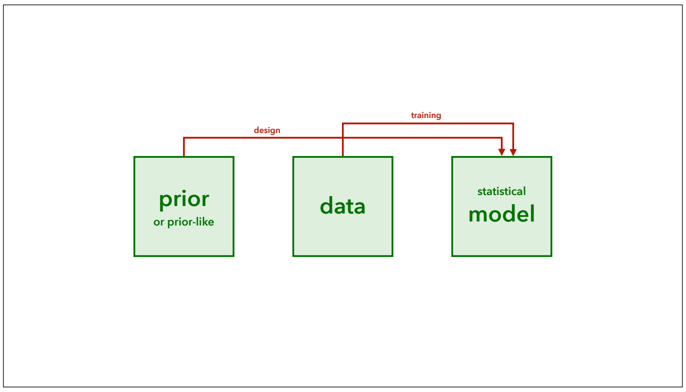


(a) Randomly initialized (untrained) model compared to topic classifier trained from initialization.



Stefan Schouten

Piek Vossen


training
design
prior
or prior-like
data
statistical
model