

Welcome to AAAI-MAKE 2023

Andreas Martin, PhD | andreas-martin.ch

AAAI-MAKE Chair

*Lecturer and Senior Researcher in Artificial Intelligence and Information Systems
FHNW University of Applied Sciences and Arts Northwestern Switzerland*



Artificial intelligence is fascinating.

AI
e.g., Robotics



ChatGPT is doing KE!



Create a RDFS ontology as TTL snippet about: A human can have a child relationship with another human. Andreas is a human. Andreas has a child called Aurel. Use the example prefix martin. Create a second TTL snippet where you add the inference of what Aurel is. Finally, explain what you did.



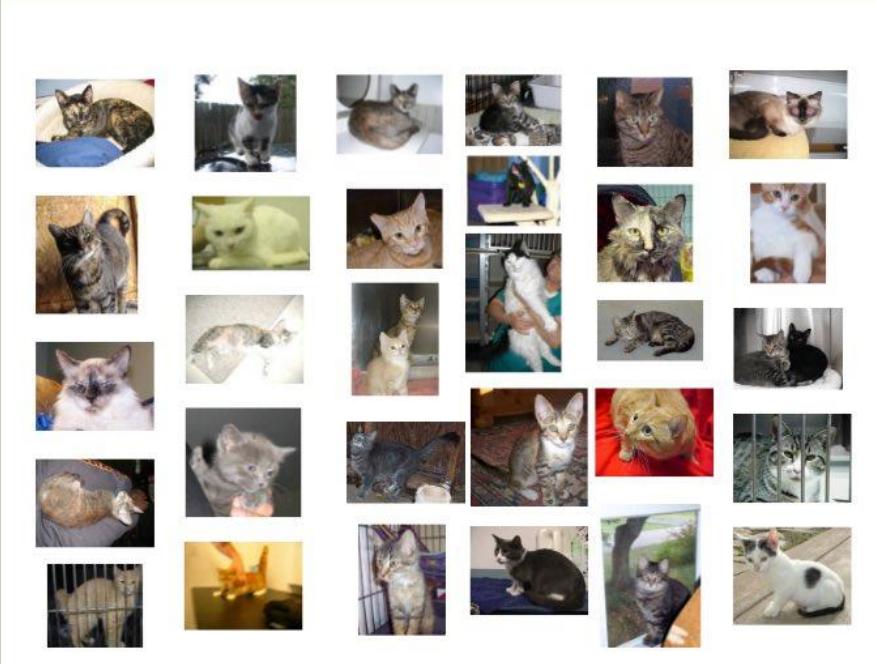
Here is an RDFS ontology written in TTL (Turtle) format that defines a relationship between humans where one human can have a child relationship with another human:



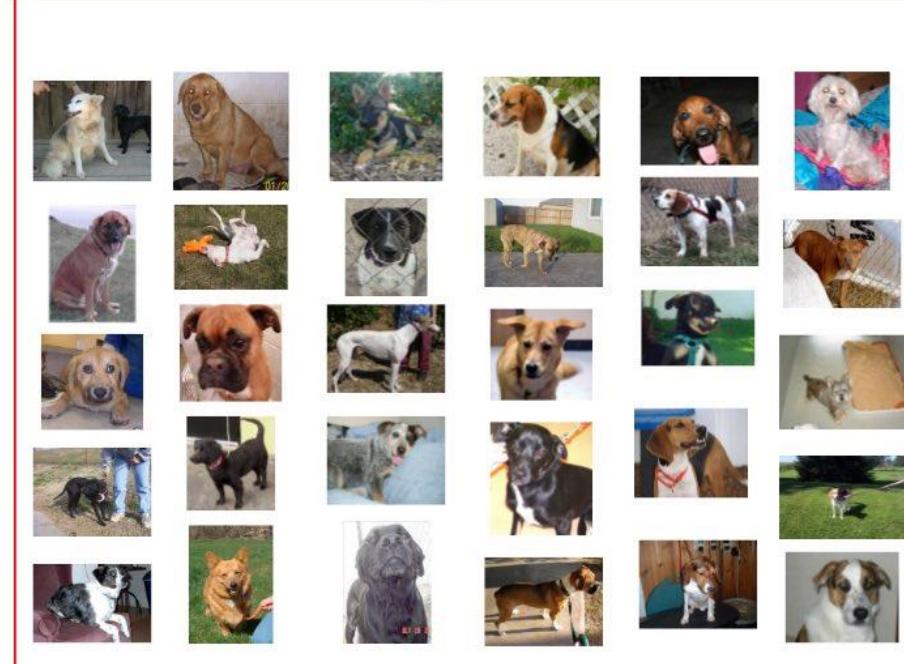
```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .  
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .  
@prefix martin: <http://example.com/martin#> .  
  
martin:Human rdf:type rdfs:Class .  
martin:hasChild rdf:type rdf:Property ;  
    rdfs:domain martin:Human ;  
    rdfs:range martin:Human .  
  
martin:Andreas rdf:type martin:Human .  
martin:Andreas martin:hasChild martin:Aurel .
```

ML needs a vast set of labelled training data!

Cats



Dogs

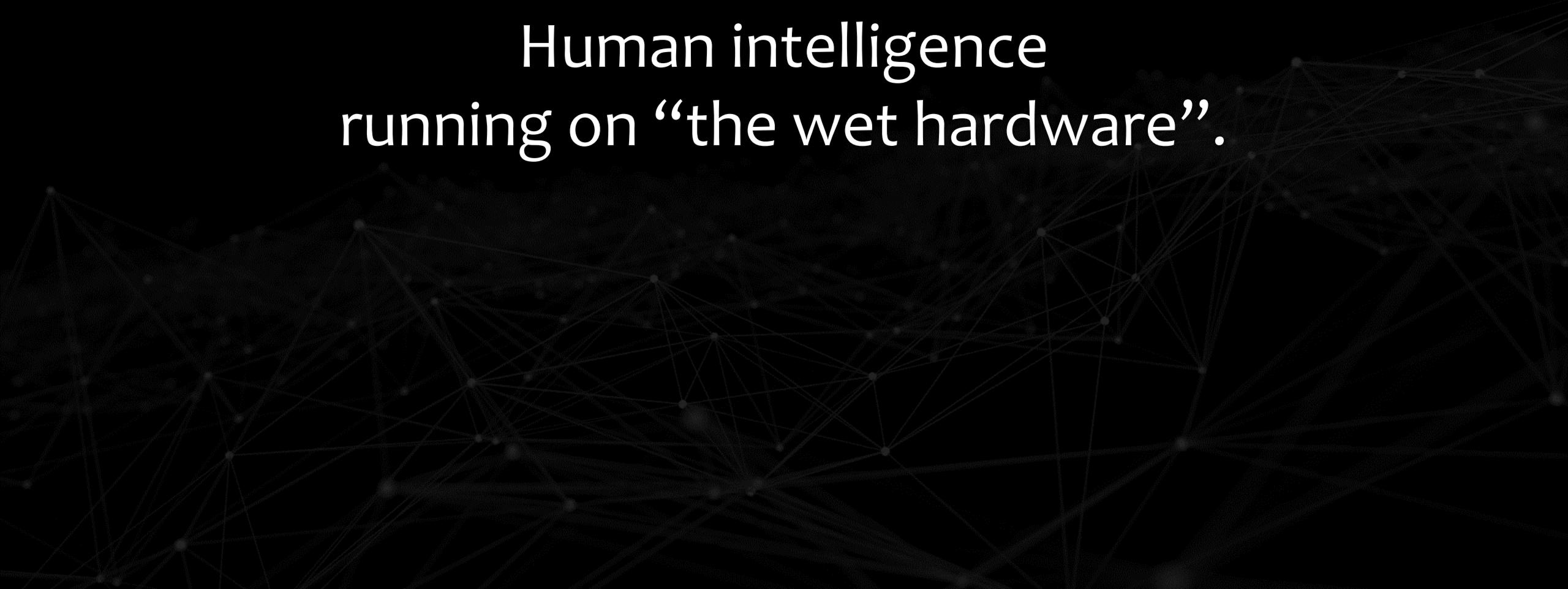


Sample of cats & dogs images from Kaggle Dataset

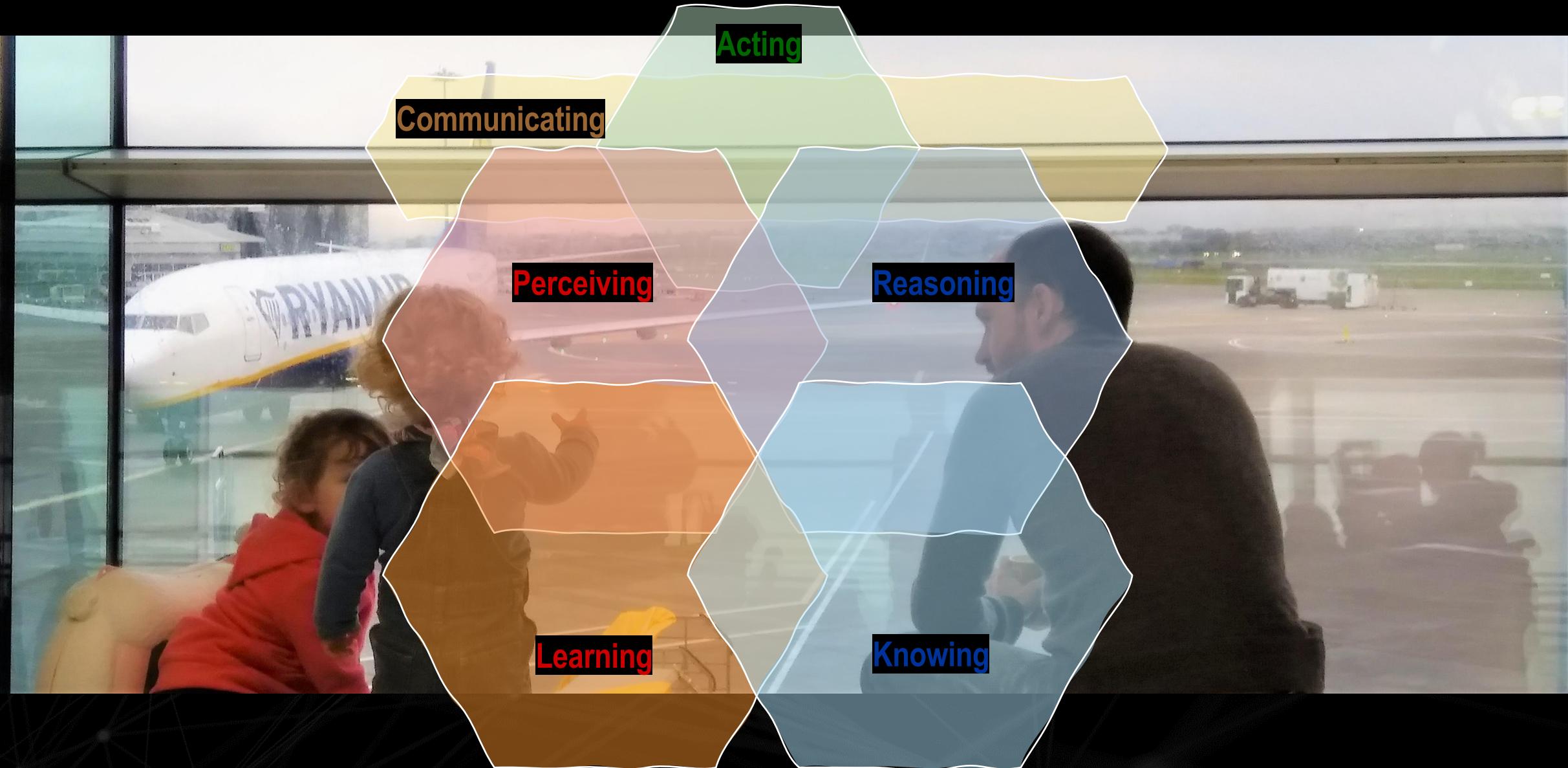
He needs a good example ...



Human intelligence
running on “the wet hardware”.

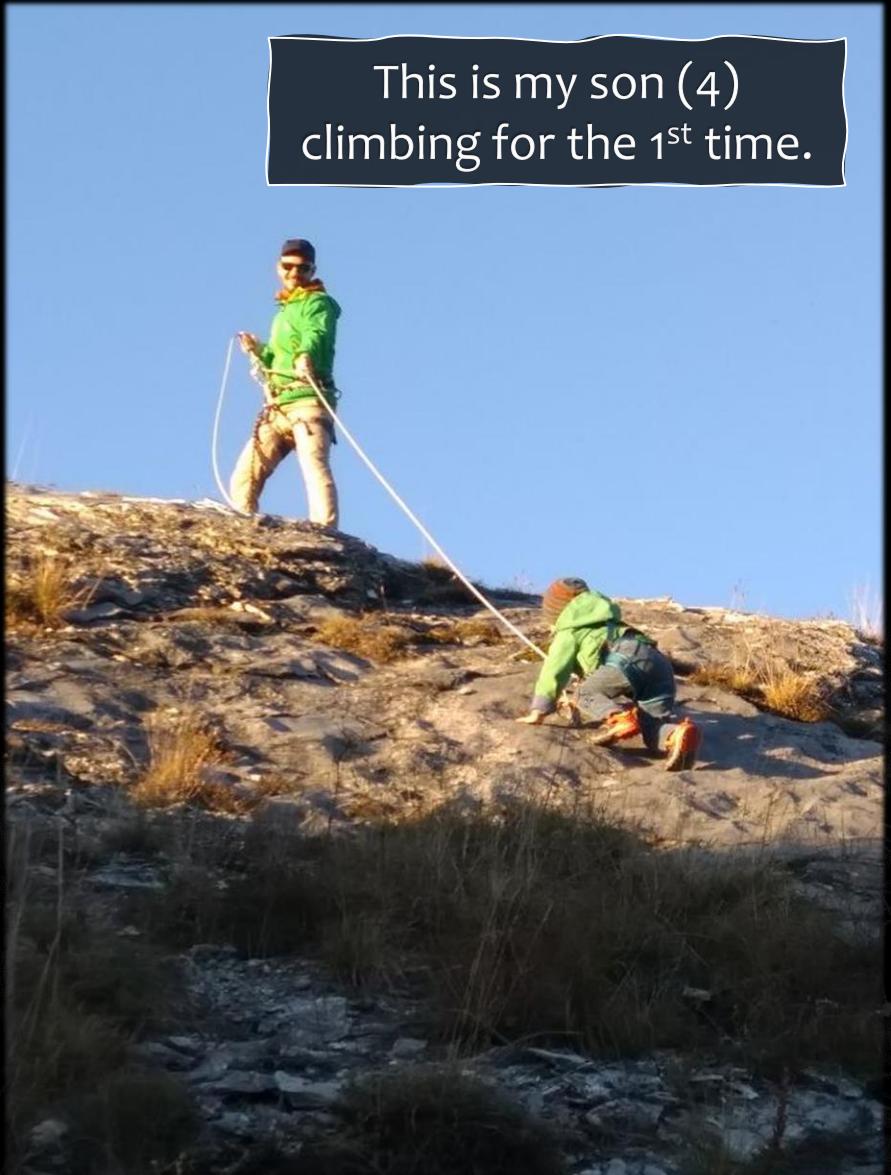


Human Intelligence & Capabilities

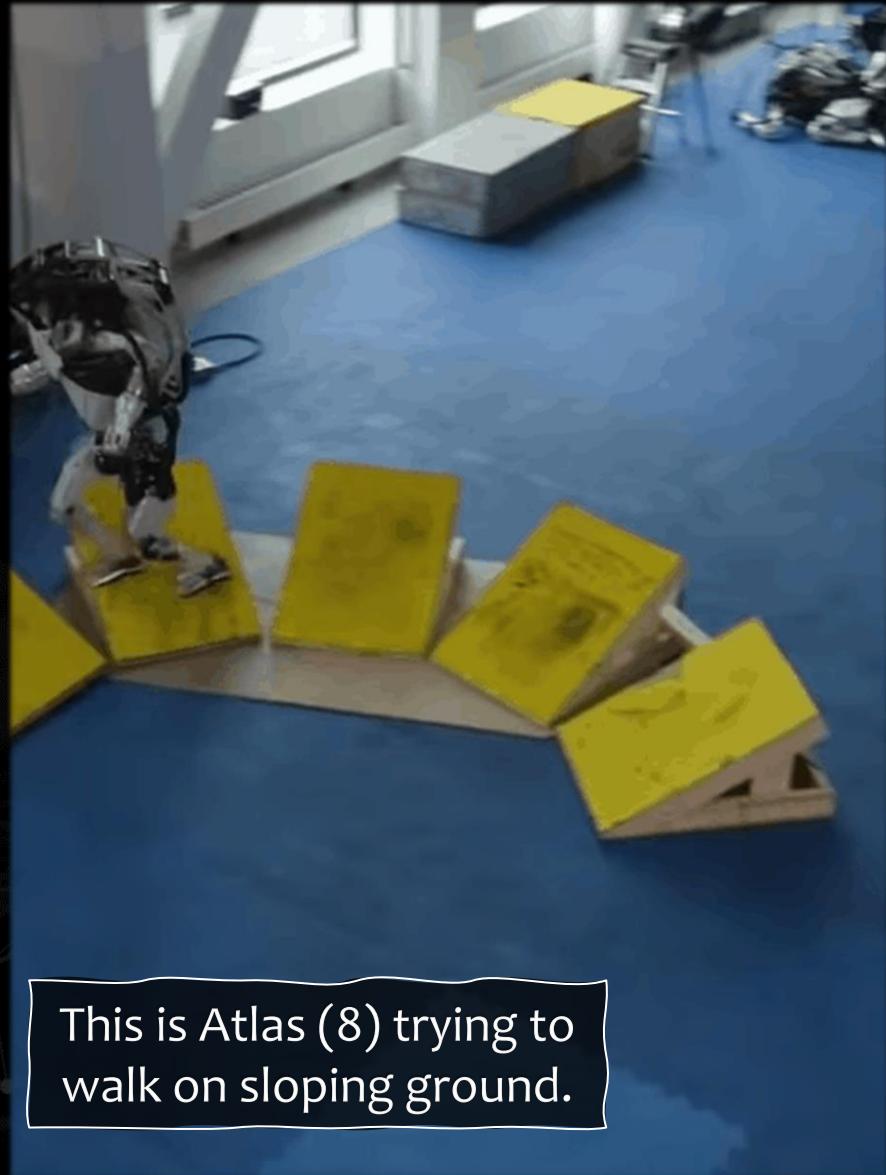


Learning something new & Problem-Solving

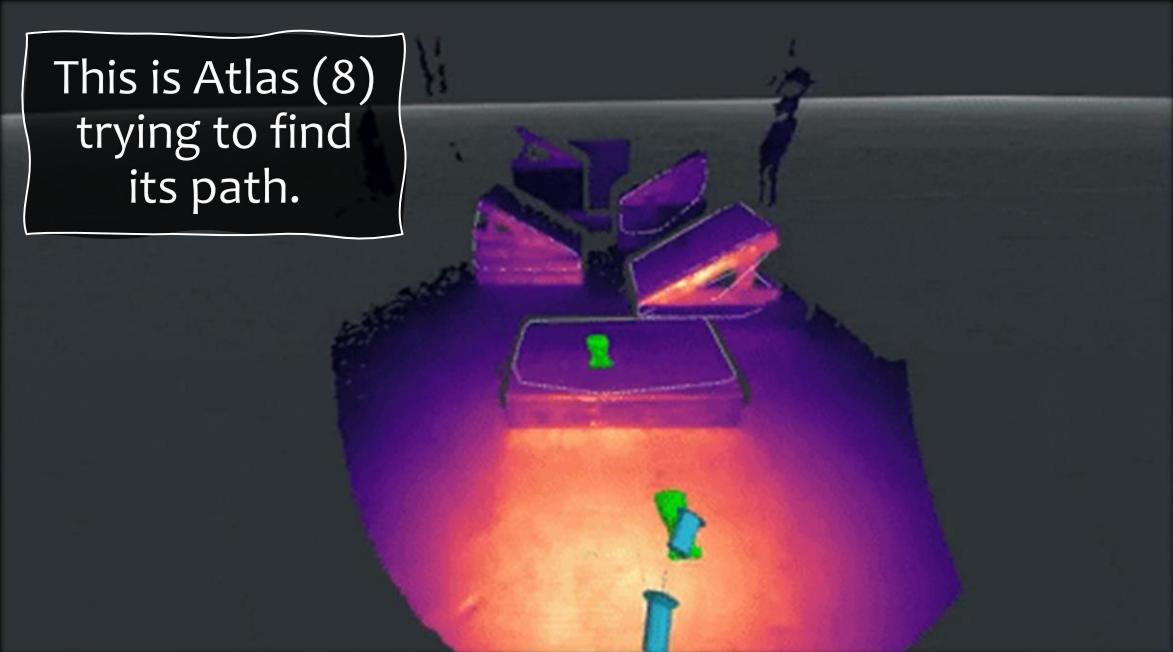
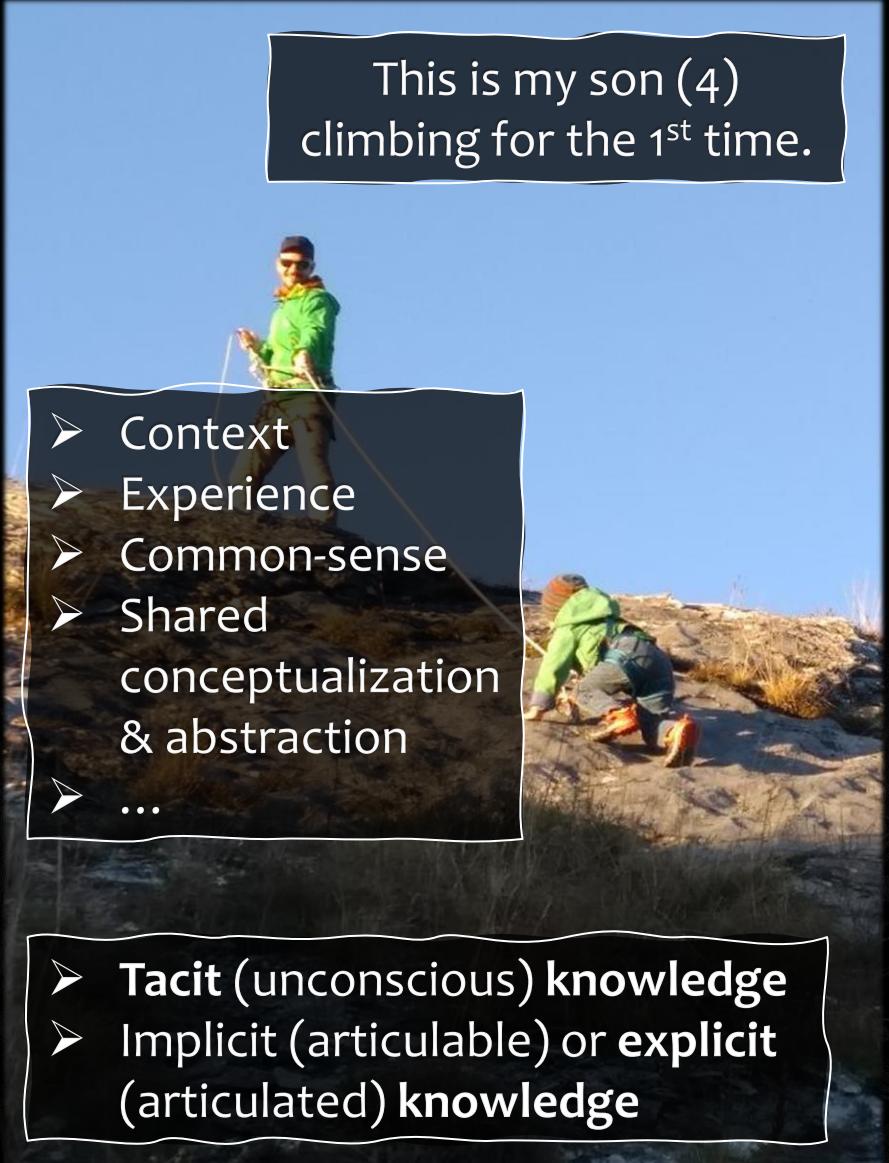
Human vs. AI



This is my son (4)
climbing for the 1st time.



This is Atlas (8) trying to
walk on sloping ground.



- Learned/trained model
- Current sensor data
- Some formal knowledge (code)
- ...

➤ **(Mainly) tacit/non-explainable model**

ChatGPT = MAKE-Style?

ChatGPT is doing KE! ... Really?



Create a RDFS ontology as TTL snippet about: A human can have a child relationship with another human. Andreas is a human. Andreas has a child called Aurel. Use the example prefix martin. Create a second TTL snippet where you add the inference of what Aurel is. Finally, explain what you did.



Here is an RDFS ontology written in TTL (Turtle) format that defines a relationship between humans where one human can have a child relationship with another human:

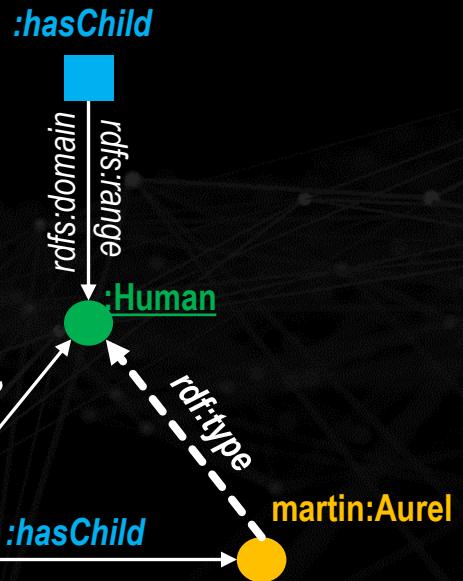


```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .  
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .  
@prefix martin: <http://example.com/martin#> .  
  
martin:Human rdf:type rdfs:Class .  
martin:hasChild rdf:type rdf:Property ;  
    rdfs:domain martin:Human ;  
    rdfs:range martin:Human .  
  
martin:Andreas rdf:type martin:Human .  
martin:Andreas martin:hasChild martin:Aurel .
```

Explicit knowledge in ChatGPT?



Create a RDFS ontology as TTL snippet about: A human can have a child relationship with another human. Andreas is a human. Andreas has a child called Aurel. Use the example prefix martin. Create a second TTL snippet where you add the inference of what Aurel is.



RDF Schema → «Commonsense» (Explicit) **Knowledge**

- A **Human** «A» can *have a child* relationship of a **Human** «B».

RDF → Specific (Explicit) **Knowledge**

- Andreas is of **type Human**
- Andreas *has a child* called Aurel

Querying → Questioning

- What is **Aurel**?

RDF(S) Inferencing → Logic **Reasoning**

- Aurel is of **type Human**

Here is a second TTL snippet where you add the inference of what Aurel is:

```
martin:Aurel rdf:type martin:Human .
```

In this snippet, I have added that martin:Aurel is also an instance of the class martin:Human. This means that Aurel is a Human.

No explicit knowledge? No reasoning?

```
less
Copy code

@prefix martin: <http://example.com/martin#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

martin:Human rdf:type rdfs:Class .
martin:hasChild rdf:type rdf:Property ;
    rdfs:domain martin:Human ;
    rdfs:range martin:Human .

martin:Andreas rdf:type martin:Human .

martin:Aurel rdf:type martin:Human ;
    martin:hasChild martin:Andreas .

{ martin:Aurel martin:hasChild ?child } => { ?child rdf:type martin:Human }.
```

In this snippet, we've added an RDFS rule that states that if "Aurel" has a child, then that child must be of the "Human" class. This is an inference we can make based on the information we've provided.

Overall, we've created an RDFS ontology that represents the relationship between "Andreas" and "Aurel" as a parent and child who are both individuals of the "Human" class. We've also added an inference that any child of "Aurel" must also be a member of the "Human" class.

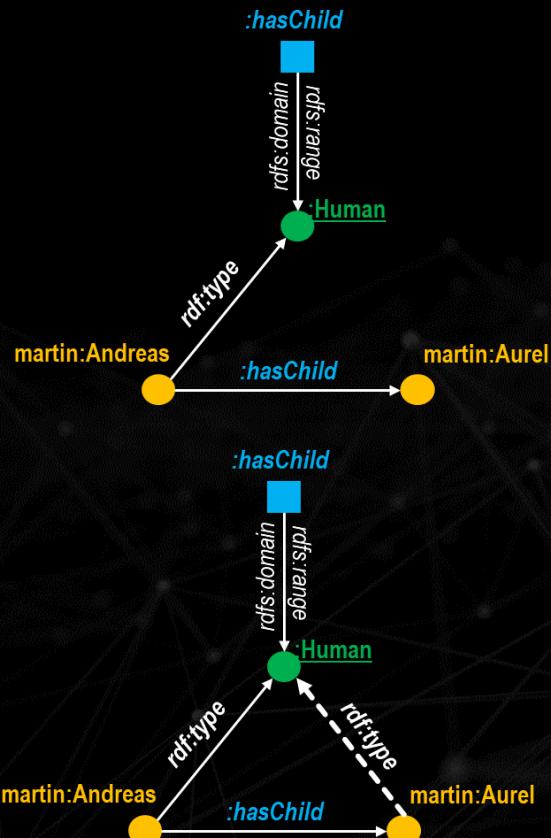
That is wrong.

N3 logic not needed.
And too specific.

Parroting something together.

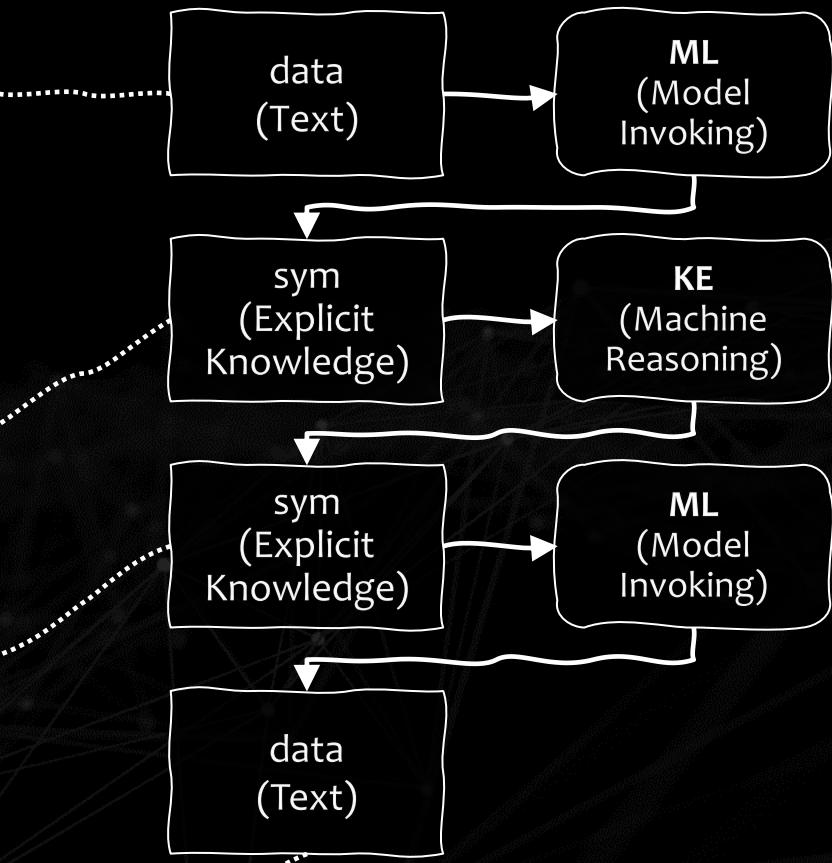
Typical MAKE Boxology Pattern:

Create a RDFS ontology as TTL snippet about: A human can have a child relationship with another human. Andreas is a human. Andreas has a child called Aurel. Use the example prefix martin. Create a second TTL snippet where you add the inference of what Aurel is.



```
martin:Human rdf:type rdfs:Class .  
martin:hasChild rdf:type rdf:Property ;  
    rdfs:domain martin:Human ;  
    rdfs:range martin:Human .  
  
martin:Andreas rdf:type martin:Human .  
martin:Andreas martin:hasChild martin:Aurel .
```

```
martin:Aurel rdf:type martin:Human .
```



In this snippet, I have added that `martin:Aurel` is also an instance of the class `martin:Human`. This means that Aurel is a Human.

Tacit model

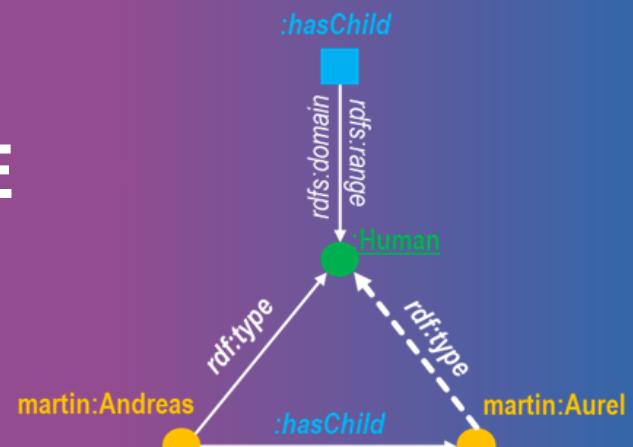
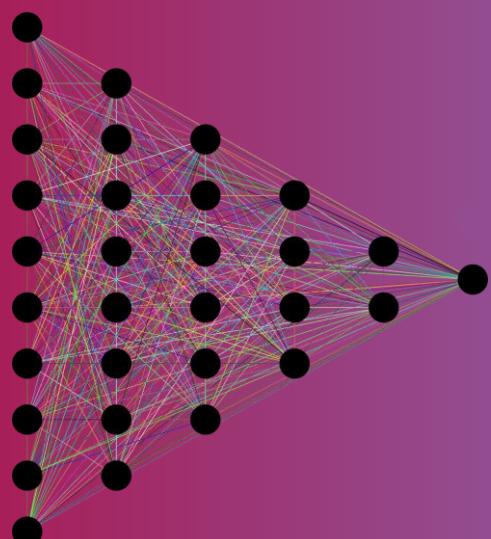
So, there we have it: MaKE!

Explicit knowledge

Machine Learning &
Model Invoking

Knowledge Engineering
& Machine Reasoning

MaKE



AAAI Spring Symposium

AAAI-MAKE



AAAI Spring Symposium

AAAI-MAKE

AAAI-
MAKE



AAAI-MAKE @ Stanford University, Palo Alto, California, USA

- Chairs: Knut Hinkelmann & Andreas Martin
- Organizers: Hans-Georg Fill, Aurona Gerber, Doug Lenat, Reinhard Stolle, Frank van Harmelen
- Since 2019: 155 papers

- AAAI **2019** Spring Symposium on **Combining Machine Learning with Knowledge Engineering**
- AAAI **2020** Spring Symposium on **Combining Machine Learning and Knowledge Engineering in Practice**
- AAAI **2021** Spring Symposium on **Combining Machine Learning and Knowledge Engineering**
- AAAI **2022** Spring Symposium on **Machine Learning and Knowledge Engineering for Hybrid Intelligence**
- AAAI **2023** Spring Symposium on **Challenges Requiring the Combination of Machine Learning and Knowledge Engineering**

Since 2019, 155 publications



2023 Spring
(shortened timeframe given)



2022 Spring



2021 Spring
(with some extended papers from 2020)



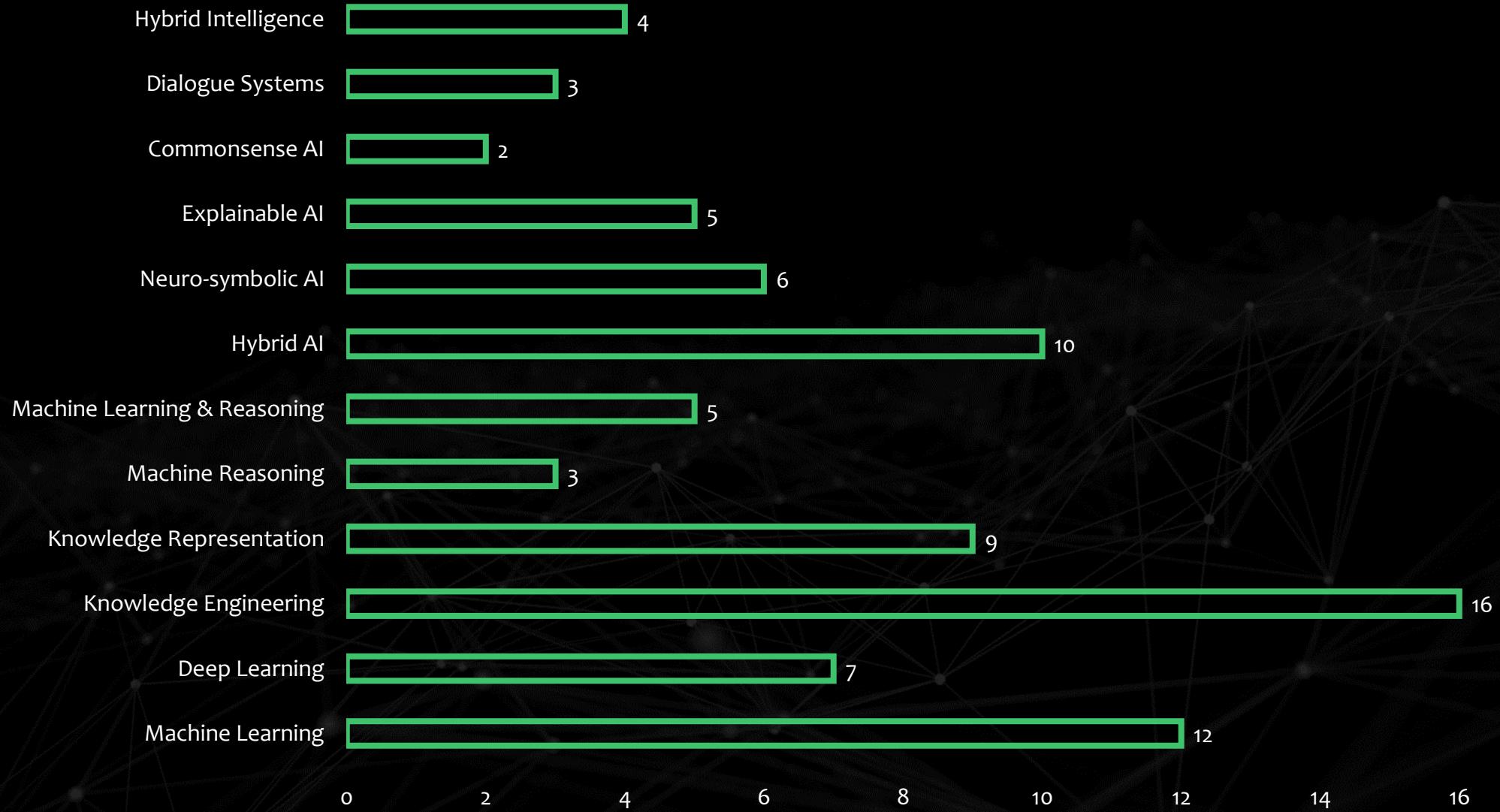
2020 Spring
(cancelled due to COVID-19; papers published)



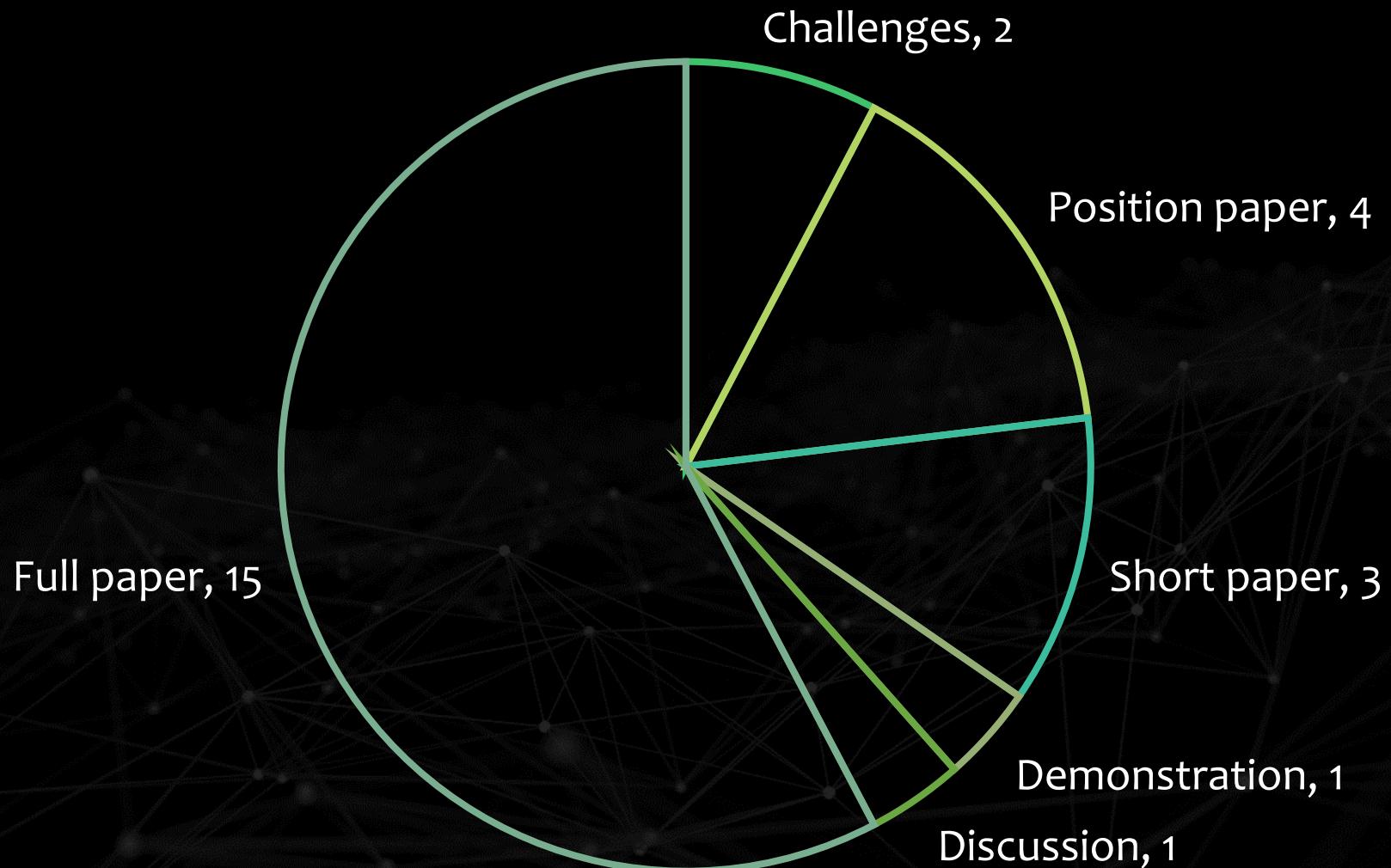
2019 Spring



Thematic spectrum in 2023



Type of contribution in 2023



A next MAKE challenge



- Challenge papers (with datasets or ontologies) describing future **societal, environmental, economic, or scientific challenges** that require **hybrid (artificial) intelligence**.
- In a follow-up event, solutions to these challenges shall be presented.
- Challenge Pre-Registration:
 - aaai-make.info/next

Now: A once in a lifetime experience!

This [MAKE] seems to be too important!

Is this going to be held at another time/place/venue?

 Edward Feigenbaum
An mail@aaai-make.info

   Antworten  Allen antworten  Weiterleiten  ...

Sa 07.03.2020 20:01

AAAI-MAKE 2020: Combining Machine Learning and Knowledge Engineering in Practice

Alas, the cancellation. But this seems to be too important not to take place somewhere, sometime.

Edward Feigenbaum
Kumagai Professor of Computer Science Emeritus
Stanford University

A once in a lifetime experience: Two AI legends holding a joint Keynote



Edward Albert Feigenbaum

- Often called the "father of expert systems."
- Winner of the Turing Award, the highest honor in computer science.
- Professor Emeritus of Computer Science at Stanford University.
- AAAI and ACM fellow
- AAAI-MAKE supporter



Douglas Bruce Lenat

- The founder of the Cyc project and Cycorp.
- Winner of the IJCAI Computers and Thought Award
- AAAI, AAAS, and Cognitive Science Society fellow
- AAAI-MAKE organizer