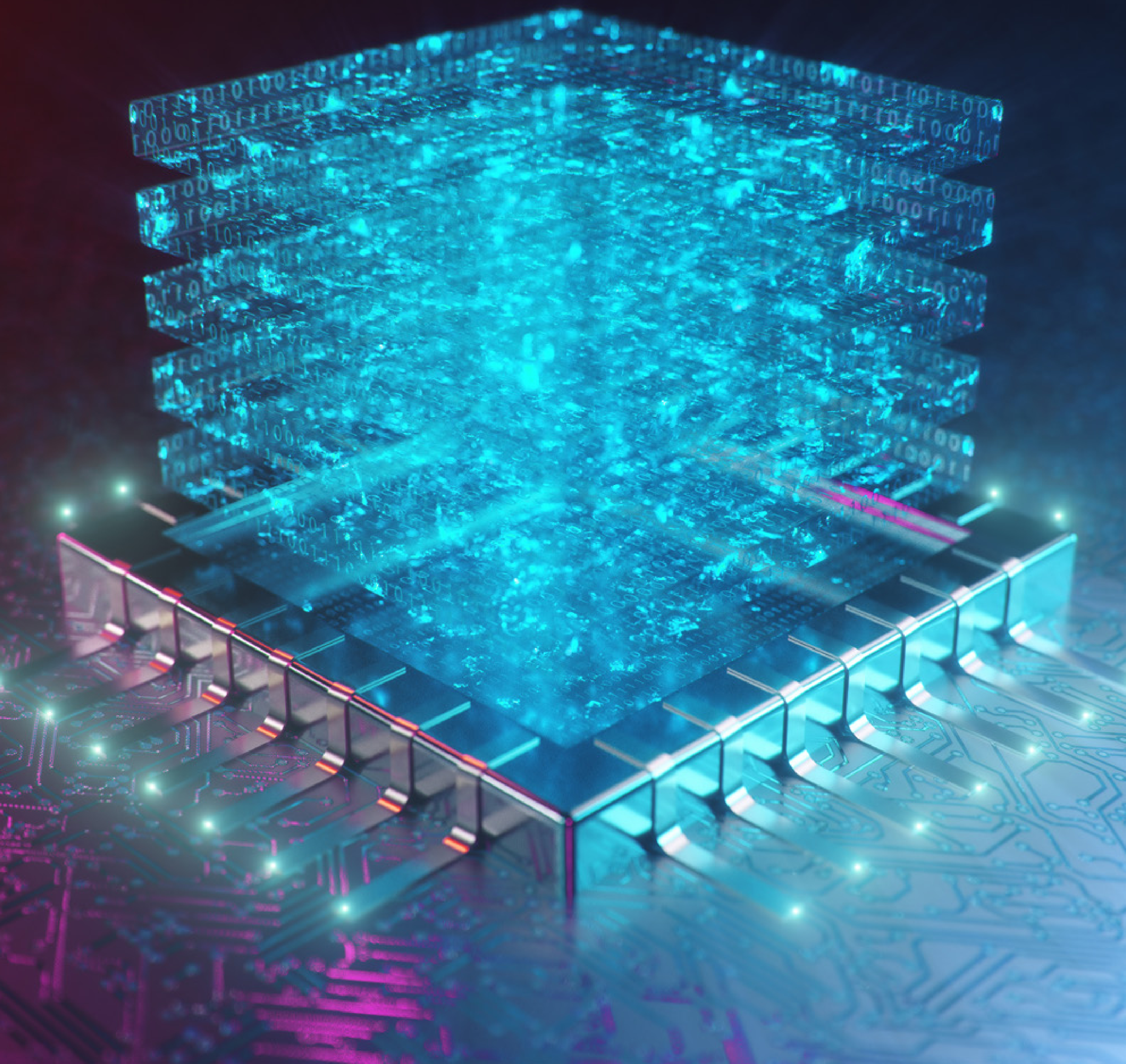
UNIVERSITY OF Southampton

Royal Academy of Engineering

AI TAXONOMY



The development of Artificial intelligence (AI) is a strategic priority for the UK government. This has been accompanied by significant investment in AI capabilities including the creation of the Office for Artificial Intelligence, establishing a National Artificial Intelligence Laboratory, increasing AI skills capabilities and funding for the UKRI Strategic Priority Fund on Trustworthy Autonomous Systems.

Across all of these initiatives, investments are targeting different components of the AI landscape. To maximise the rate of discovery across AI, efforts in all components have to be coordinated and co-optimised.

In order to manage the complexity of coordination across such a vast field, an AI taxonomy that describes the AI superstructure in terms of layers of abstraction is proposed. The aim is that this enables researchers, developers, and policymakers understand where they are intervening within the wider system and encourages consideration of the implications of design decisions at other layers. The taxonomy is designed to provide a common language and awareness of the key, coarse-grain facets of AI to enable clearer communication across the system.

The complexity and interconnectedness of the layers is such that creating clear-cut boundaries is exceedingly difficult and perhaps not even desirable. However, areas clearly belonging to specific facets can generally be identified. This is much akin to being able to identify a blue, green, red spot on a rainbow without being able to clearly tell the exact boundary between them.

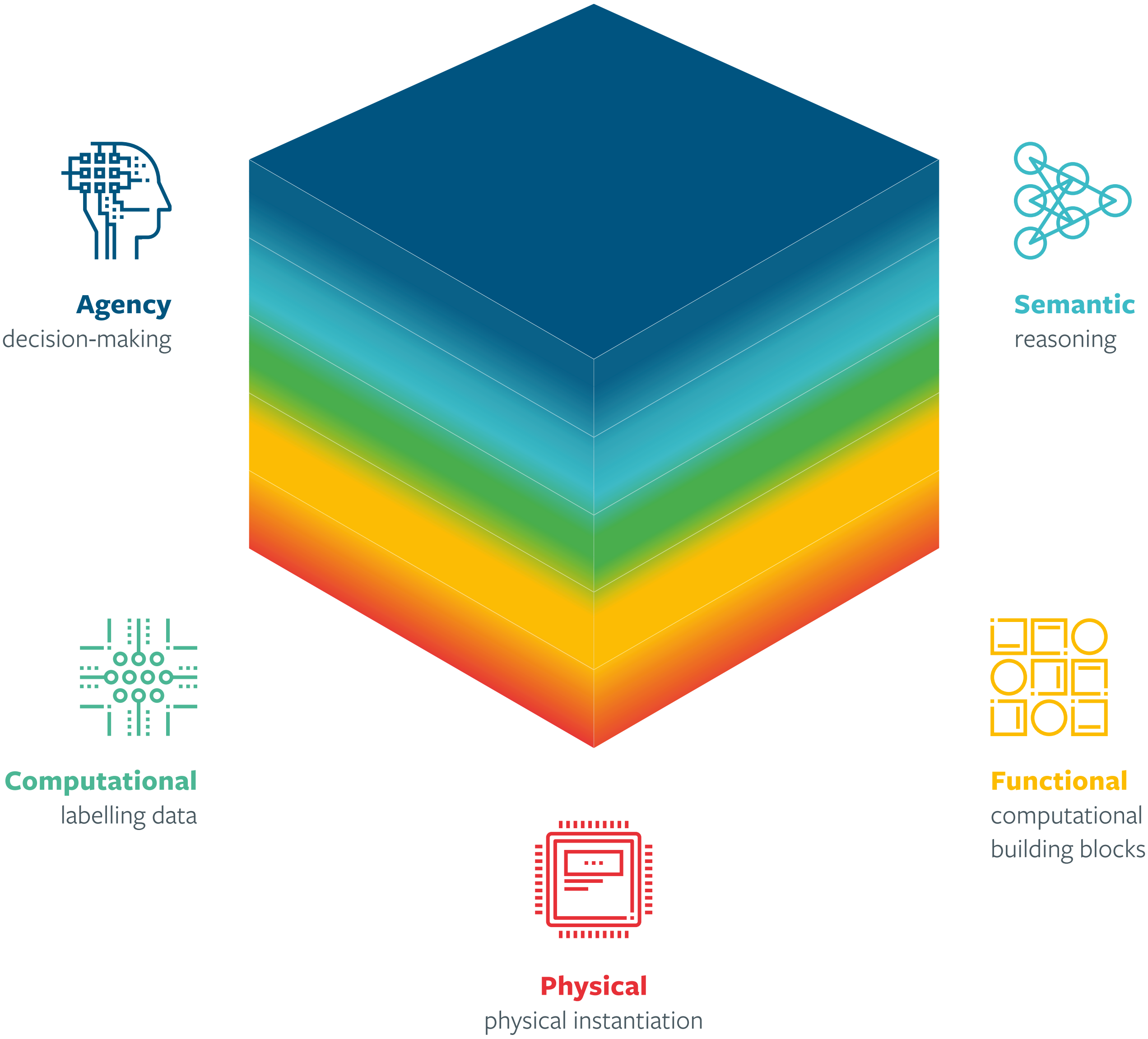
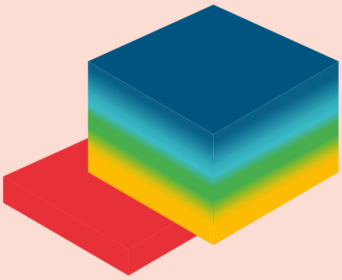


Figure 1. Framework for five interacting layers of abstraction within the AI superstruce

LEVEL 1
PHYSICAL



The physical layer handles the physical instantiation of the system.

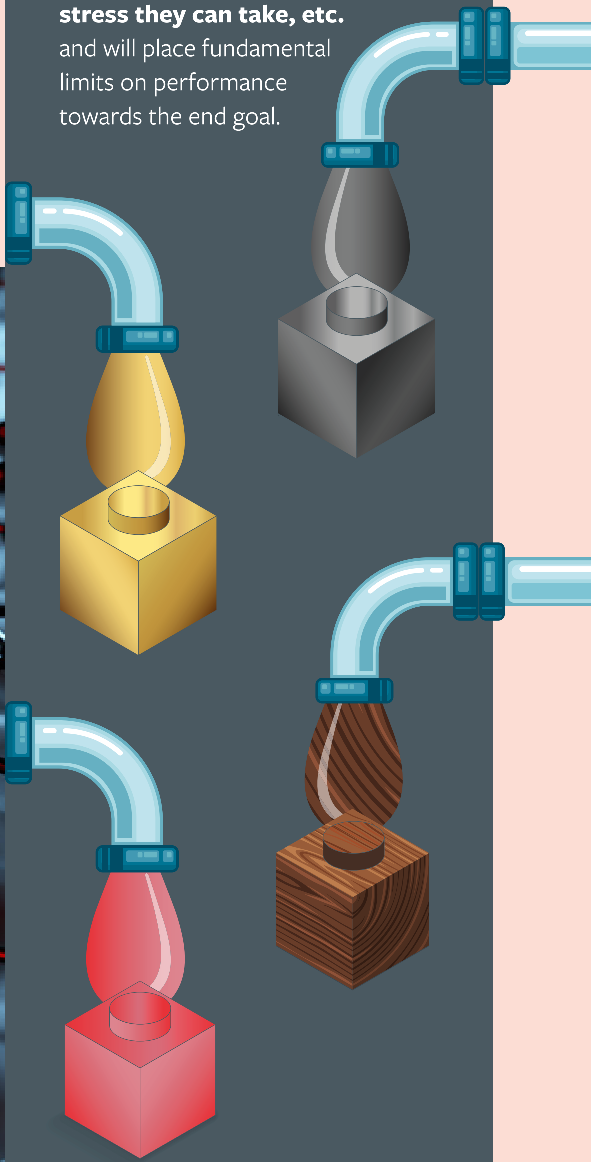
Matters pertaining to the physics, functional materials and fabrication processes that enable the AI system to compute are all included here. It is responsible for the mapping between physical quantities (voltage, current, charge) and abstract mathematical quantities (abstract numbers) used at higher levels. Innovation in materials and devices affects computational efficiency, which can be optimised for different applications. This level is represented by a community whose native language is that of physics and chemistry. As a rule, the coarsest terms being used when discussing at level 1 are device components such as transistors, memristors, quantum diodes etc.

Example

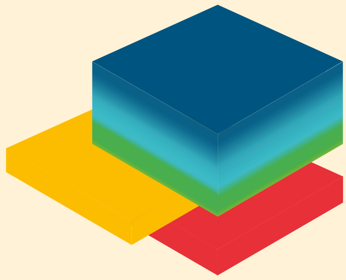
Nanotechnology and cutting-edge manufacturing techniques can be employed for engineering AI systems at the atomic and fabrication level, in order to push the envelope of what is possible to achieve with circuit design techniques. This is very similar to engineering carbon fibre car bodies for pushing the limits of what is possible to achieve using the same car engine. It can be understood as making every atom in the physical system contribute as much as possible in the process of computation.

LEGO ANALOGY

Engineering the material properties of Lego pieces. This may have substantial ramifications in terms of **joinery accuracy, how much stress they can take, etc.** and will place fundamental limits on performance towards the end goal.



LEVEL 2 FUNCTIONAL

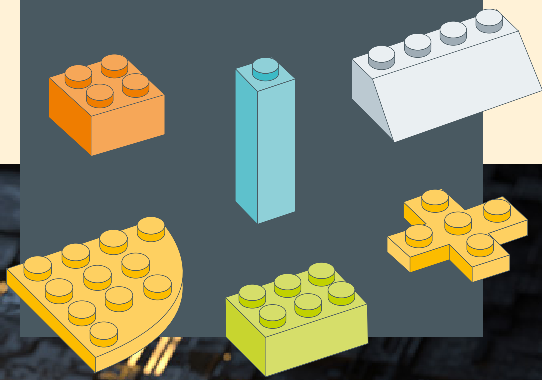


The functional layer provides the computational building blocks.

This layer includes the implementation of fundamental mathematical building blocks, for example operators such as addition, multiplication, logic functions, look-up tables, and variables including numbers, characters, generic mathematical set elements. In contrast to Layer 1, where the mathematical behaviours we tackle are given by the underlying physics and materials properties of the system, these blocks are 100% artificial; specified and engineered to suit our purposes. In Level 2 the discourse typically does not use terms finer than individual devices or terms coarser than simple digital or analogue blocks such as gates, filters or artificial neurons.

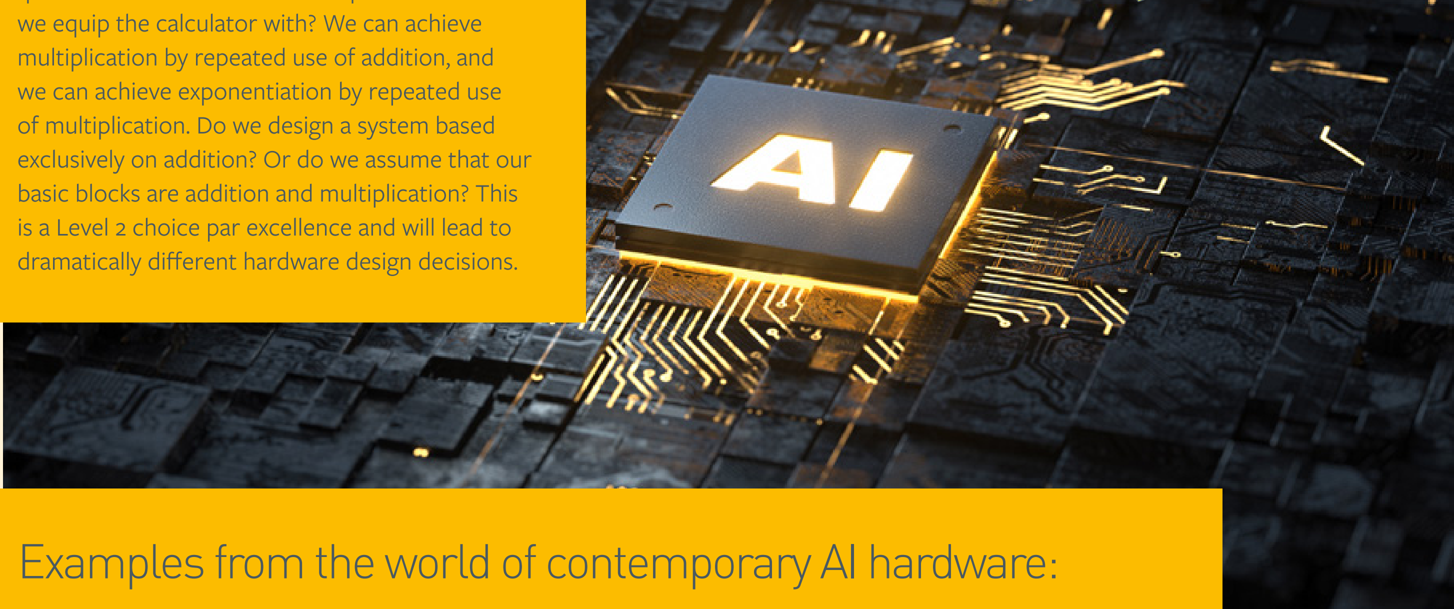
LEGO ANALOGY

It's like creating pieces of different shapes and capabilities, e.g. classic Lego bricks, L-shapes, pieces of different thickness, sloped pieces etc. **This makes many more possible designs feasible and/or practically accessible.**



Simple example:

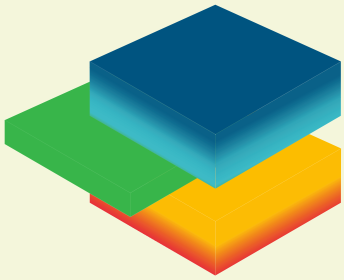
Suppose we want to make a calculator. Our spec states that our calculator must be able to perform addition, multiplication and exponentiation; nothing else is required by the customer. The question is what "fundamental operations" should we equip the calculator with? We can achieve multiplication by repeated use of addition, and we can achieve exponentiation by repeated use of multiplication. Do we design a system based exclusively on addition? Or do we assume that our basic blocks are addition and multiplication? This is a Level 2 choice par excellence and will lead to dramatically different hardware design decisions.



Examples from the world of contemporary AI hardware:

Neurons are typically modelled as simple "integrate (sum) and fire" units. Building complex artificial neurons consisting of multiple compartments (as opposed to being just single units) creates a much more capable computational unit that can be used to achieve better performance or functionality at the Layer above. In logic design this would be equivalent to designing a gate library: do we design our system using only NAND gates or do we also have AND, OR, XOR, etc. at our disposal? Any design made with any logic gates can be translated into an implementation using only NAND gates, but the efficiency of these implementations could differ substantially.

LEVEL 3 COMPUTATIONAL

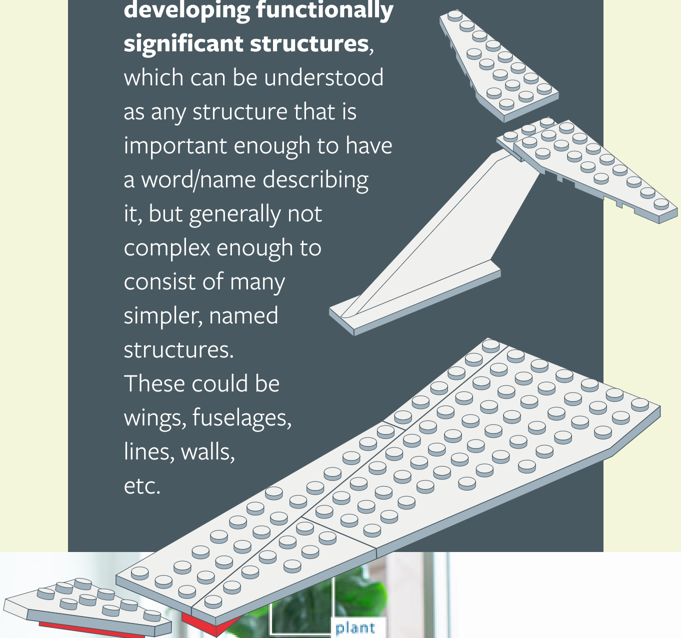


The computational layer is fundamentally responsible for labelling data.

It takes real world spatial data and temporal sequence data and collapses a lot of the continuity and uncertainty into manageable, easily manipulable symbols that define how it should be classified/labelled. The transition from a mass of low-level (sensor-level) unreliable data to a highly refined, stable label is critical: it flattens the uncertainty into as little as a single number: classification accuracy (example: we know the displayed character is a ‘k’ with 98% accuracy). As such, these symbols can be thought of as stable representations: they allow classification of an input, such as a car or a banana with great tolerance to pose, colour, illumination, etc. The specific weight and connectivity configurations can be actively engineered to achieve specific higher-level functionality. Data labelling is not the only task performed in this Layer. Representational transformations and other vector-level operations are also handled at this Layer, yet data labelling currently remains dominant. Level 3 rarely concerns itself with notions finer than the simple circuit blocks at the top of Level 2 or with notions coarser than neural network modules, microprocessors or memories.

LEGO ANALOGY

This would be similar to **developing functionally significant structures**, which can be understood as any structure that is important enough to have a word/name describing it, but generally not complex enough to consist of many simpler, named structures.
These could be wings, fuselages, lines, walls, etc.



Example

Artificial neural networks and their architectures can be understood to operate at this level, effectively sifting often multidimensional and noisy inputs and producing a set of reasonably stable responses (typical example of an image; from an ocean of noisy pixels we get a class of object, e.g. “child” or “sofa”). Different network architectures will be more conducive to carrying the labelling tasks on different types of inputs: Combinational, feedforward networks are good for e.g. images, where all information required for classification is assumed to be concurrently available. Long-short-term memory (LSTM) networks incorporate recurrence and are broadly applicable to time-varying, or sequential data such as speech.

So, how are these blocks different from the basic units in Level 2 or Level 4? We note that a good way to discern which level we are at is the language of discourse: each layer tends to have its own vocabulary and terms. Simultaneously, however, it is important to recognise that the borders between layers are blurred and can move depending on expediency. Yet amidst this complex landscape, it is possible to clearly state that some e.g. Layer 3 blocks certainly do not belong to either Layer 2 or Layer 4. For example, a car made of Lego blocks is certainly not on par with a single Lego piece (a Layer 2 structure) because its behaviour can be described comprehensively at a semantic level where reference to individual (L2) Lego blocks is almost completely unnecessary. **Overall, the separation between layers is perhaps best understood as a chocolate bar: the entire chocolate is one connected object, but there are natural lines along which it can be usefully divided into smaller parts and also lines along which it would be very unnatural to make the division.** A point in the middle of a piece quite clearly belongs to that piece, but borderline points may be shifted around depending on expediency.

LEVEL 4 SEMANTIC

The semantic Layer is responsible for reasoning.

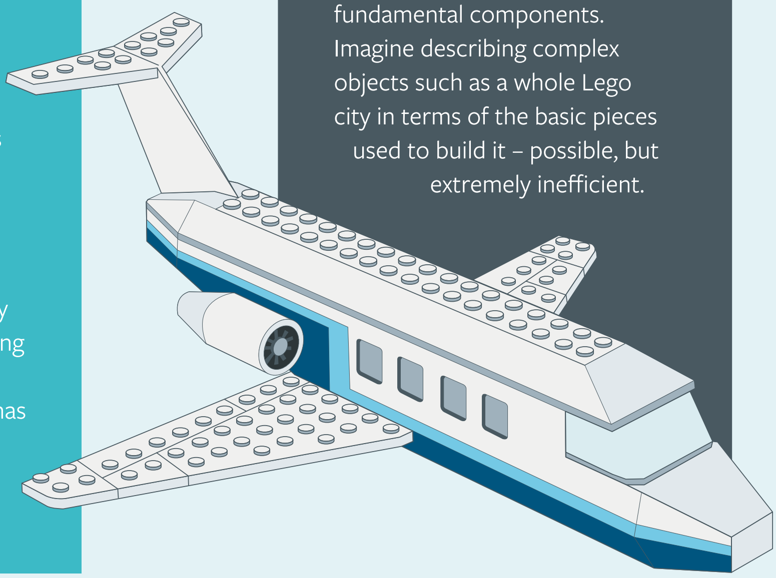
This layer provides the mechanisms for systems to manipulate already labelled data and reasonably hypothesise how it should act. Rules can be expressed at the symbolic abstraction level and therefore tend to be human-readable and explainable. This layer generally handles objects at least as complex as the microprocessors and neural networks at the top of Level 3 and no more complex than grammar parsers or inference engines.

LEGO ANALOGY

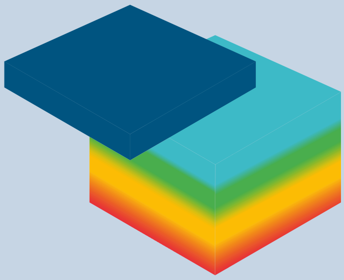
This is like **modularly using ready-made functional components** (wings, fuselages empennages, etc.) **in order to build complex objects** that are hard to describe in more basic terms. At Level 4 one would describe a Lego aeroplane as consisting of 2x wings, a fuselage and an empennage with few or no other details necessary; any further details are relegated to the layer below. This is a key point because it allows objects of in principle arbitrary complexity to be expressed as relatively small and manageable collections of relatively self-contained "sub-objects" as opposed to always having to refer to the fundamental components. Imagine describing complex objects such as a whole Lego city in terms of the basic pieces used to build it – possible, but extremely inefficient.

Example

Current systems attempt to learn concepts from the ground up (see enough examples of raw data so that the general pattern emerges without 'thinking'). That manifests itself as, for instance, trying to have a system play so many games of chess or go, that any board configuration in subsequent games "is familiar" to the system and it knows how to react, or reasonably guess. Instead, we can equip the system with the ability to create new representations by combining older ones in a "constructive" manner: by applying rules on data, as opposed to passively receiving examples until a substantial percentage of all possible combinations has been sampled. This allows new learning to be powered by previous learning; such recycling of knowledge immensely accelerates the evolution of our system.



LEVEL 5 AGENCY

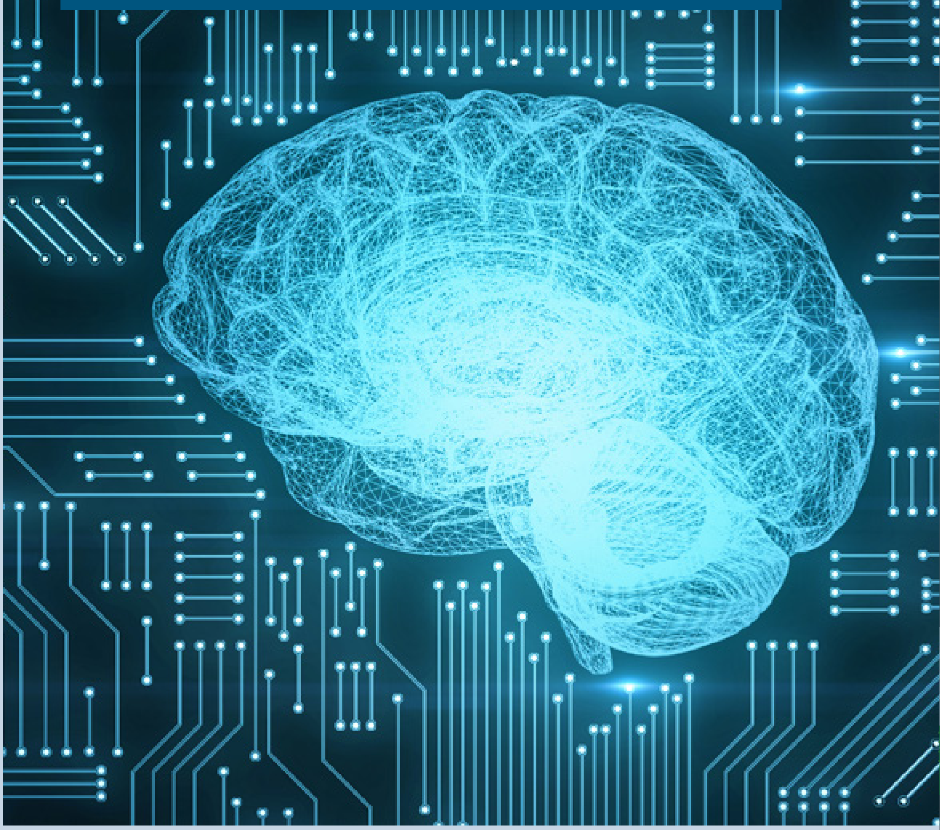


The agency layer is responsible for decision making.

This layer concerns itself with the systems' motivations, as well as how it evaluates the operational environment and how it draws on existing knowledge in order to formulate and put into motion plans of action. This spans the interval from autonomously deciding to "stop everything and go recharge" to ethical and moral decision making. This layer handles anything above the complexity ceiling of Layer 4 and typically involving top-level decision-making.

Example

An example of agency layer research and development is designing and equipping autonomous systems with a set of "desires" to drive their decision making and mechanisms for elaborating those into concrete goals. The agency layer interacts strongly with the next (semantic) layer to turn a selected goal/action into a plan and subsequently executing it. This layer can be said to include agent-agent interactions: e.g. in swarm robotics each member of the swarm (agent) must be able to take into account both its surroundings and the situation of its peer agents when deciding which action to select.



LEGO ANALOGY

This is equivalent to **deciding to build a Lego structure for some specific purpose** and then evaluating the result with respect to that purpose. The purpose may be as esoteric as for example building a Lego tower for the satisfaction of seeing it stand tall. In most artificial systems we would typically expect a human to do the purpose-giving decision making and the agent to execute the plan and then perform the evaluation. We note that handing over the authority for choosing a purpose completely independently to the machine turns it into a completely free entity, which has significant ethical implications.

Authors

Professor Themis Prodromakis FRSC, FinstP, FBCS, FIET
RAEng Chair in Emerging Technologies
Centre for Electronics Frontiers, ECS
University of Southampton, UK

Dr Alex Serb
Centre for Electronics Frontiers, ECS
University of Southampton, UK

Acknowledgements

Dr Alexandra Smyth RAEng
Senior Programme Manager – Positive Response
Royal Academy of Engineering, London, UK

Professor Steve Furber CBE FRS FREng
ICL Professor of Computer Engineering
Department of Computer Science,
The University of Manchester, UK

Steven Meers
Head of AI lab DSTL

Professor Dame Wendy Hall DBE FRS FREng
Regius Professor of Computer Science
Director, Web Science Institute
University of Southampton, UK