AGI from the perspectives of Categorical Logic and Algebraic Geometry

King-Yin Yan[0009−0007−8238−2442]

general.intelligence@gmail.com

Abstract. To "situate" AGI in the context of some current mathematics, so that readers can more easily see whether current mathematical ideas can be fruitfully applied to AGI.

Keywords: AGI · categorical logic · Curry-Howard isomorphism · homotopy type theory · algebraic geometry · topos theory.

1 Goal of this paper

The bottleneck of AGI development is the speed of learning algorithms. The daily cost of training GPT-4 was rumored to be $100M by Sam Altman. To speed up learning, one needs **inductive biases**, according to the **No Free Lunch theorem** [18] [17]. A principled way to introduce inductive bias is by the structure of logic. The reason being that, if humans have discovered the structure of logic in this world, an intelligent program may re-discover the same structure. So our question is: what is the mathematical structure of logic?

2 Results thus far

The conclusions of this paper are mostly *negative*. That is to say, the mathematical structures described here seem unable to offer practical ways to accelerate AGI, unless the reader can discover more ingenious ideas. Nevertheless the author hopes the presentation of these ideas thus far can help the readers on their way.

In each section below, we look at one aspect of the categorical structure of logic and speculate on how it might aid AGI architecture.

2.1 Where is GPT?

GPT [7] can be regarded as a **logic consequence operator** mapping from the space of propositions to itself (as a **set-valued map** [2]), as shown in Fig.1.

Without loss of generality, GPT can be seen as acting on propositions, as probability distributions on tokens are equivalent to probability distributions on sentences (propositions), see Fig. 2.

To what extent can we say that GPT output vectors live in a Curry-Howard type-theoretic space? A neural network (eg. GPT) always maps an input to the

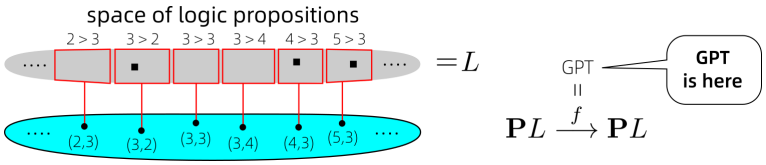

Fig. 1. (Left) A sheaf of propositions over pairs of natural numbers; (Right) The function space where GPT lives. $\mathbf{P}L = 2^L$ is the power set of L.

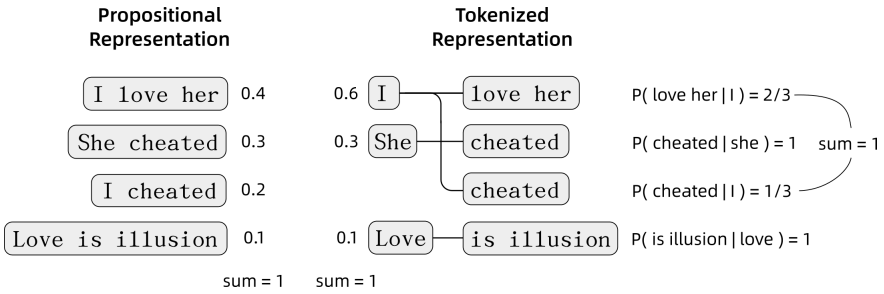

Fig. 2. The equivalence of probability distributions over propositions and over tokens. For simplicity the decomposition for just the leading token is shown.

the same output vector, as it is a *deterministic* map. Thus it seems meaningless to ask what is the meaning of the *neighborhood* of an output vector, if the network never goes there during inference time.

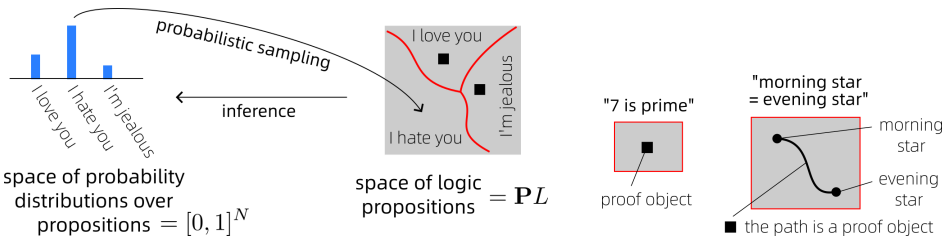

Fig. 3. (Left) How probabilistic inference in GPT results in deterministic vector positions. N = size of vocabulary; (Right) In HoTT, a path is a proof object of an **identity type**.

Fig.3(Left) illustrates why even despite probabilistic sampling, the outputs of GPT seem to follow fixed trajectories (as soon as learning has finished). Nevertheless, if we artificially "perturb" the input, the output probability distribution will change *smoothly*, say from favoring token A to favoring token B, as neural networks are always *differentiable* functions. One can define the **boundary** between tokens A and B as where their probabilities reverse in magnitude. This

forms a Voronoi-like tessellation of the output space that can be regarded as Curry-Howard type-spaces.

2.2 Homotopy type theory (HoTT)

HoTT is a step along the Curry-Howard tradition where propositions = types = spaces, and such spaces are given **homotopy** structure. An example is "morning star = evening star", illustrated in Fig.3(Right). But the enterprise does not end here: higher homotopy types give rise to a system of ∞-groupoids. From my shallow understanding of this subject, this seems to suggest that proofs have their own proofs, so that an entire **inference trace** can be recorded as homotopy paths. Doing so may be useful from the perspective of **Truth Maintenance Systems** of classical AI.

From the previous section we may argue that the output vectors of a neural network can be regarded as **proof objects** in their respective type-spaces. However we can also argue that such type-spaces as implemented by neural networks are *unlikely* to have complex internal structures, because one proof object must vary *smoothly* into another proof object. To be able to process HoTT information, we may need neural networks with **fractal structure** (this can be implemented using recursion + scaling), but current neural architectures seem to lack it.

2.3 Commutativity of $\wedge$ and $\vee$

Permutation symmetry is the easiest to recognize and implement [20] [14]. It is well-known the Transformer [16] is **equivariant** to permutations of inputs. This may be seen as evidence that Transformer **tokens** are proposition-like entities, with the caveat that we may be confusing the propositional level with the sub-propositional level of atomic concepts. An easy-to-remember example is: $I \heartsuit U \neq U \heartsuit I$, but $I \heartsuit U \wedge U \heartsuit I = U \heartsuit I \wedge I \heartsuit U$.

2.4 $\forall$ and $\exists$ as adjunctions

It seems difficult to translate this structure into a structural modification of neural networks. From our experience in logic-based AI, logic rules are usually implicitly $\forall$-quantified, and $\exists$ is usually implicit by the **Closed-World Assumption**.

The following two conditions concern the well-behavior of quantification, as described in [10] and on nLab [3] [4]:
The **Beck-Chevalley condition** says that substitution of free variables commutes with quantification.
The **Frobenius condition** corresponds in logic to saying that $\exists x.(\phi \wedge \psi)$ is equivalent to $(\exists x.\phi) \wedge \psi$ if x is not free in ψ.

Both conditions are "self-evident" from the logic perspective, but it remains to be seen how they can be applied to neural networks.

2.5 Predicates as fibration

The relevant mental picture here is Fig.1(Left). Current neural networks seem to operate in the space L above the base space and are unaware of the predicate-fibration structure. Knowledge graphs have an obvious **first-order structure** as they are made of nodes and links. One can embed nodes into a metric space D and form the Cartesian product $D \times D$, then a link $a\,R\,b$ is just a point sitting vertically above this domain, and the relation R is a point-set or its **cover**. This setup may increase efficiency if we know *a priori* that the dataset is first-order.

More interesting is the case of **higher-order logic** (HOL), which means we can have quantified rules *over* relations, which suggests we should embed rules in the same manner as we embedded first-order objects. This seems to require the use of **set-valued maps** [2].

2.6 Iteration of $\vdash$ and Looped Transformers

This idea is easy to implement, and it also comes from an obvious feature of logic: we know that inferences in logic are *repeated* applications of the *same* set of rules of a knowledge-base K, $\Gamma \vdash_K \vdash_K \ldots \vdash_K \Delta$. But current Transformer architectures (which can be > 100 layers deep) do not re-use their layers at a fine scale. Perhaps the recent research in **Looped Transformers** [19] [9] can offer improvements in this direction.

2.7 Modal logic

Grothendieck topology is re-captured as a modal operator.

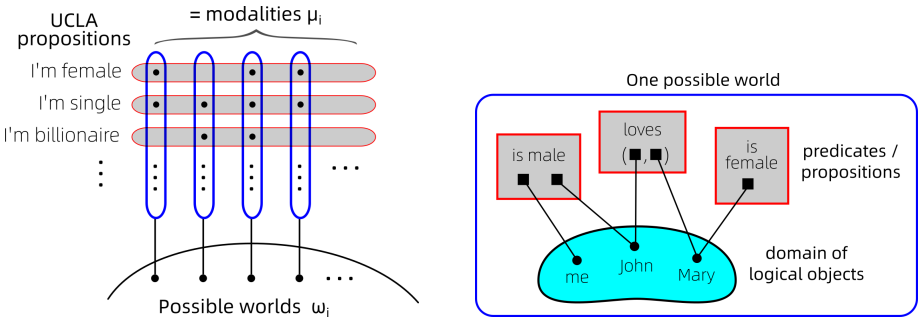

Fig. 4. (Left) The set underneath are indexes to possible worlds, for example $\omega_i = \{1, 2, 3...\}$. Each "stalk" represents a possible world, together they form a fibration over the base space.

2.8 Algebraic geometry and topos theory

The fundamental duality in algebraic geometry is:

$$\left\{\begin{array}{c}\text{spaces, or}\\ \text{varieties}\end{array}\right\} \longleftrightarrow \left\{\begin{array}{c}\text{commutative}\\ \mathbf{k}\text{-algebras}\end{array}\right\} \tag{1}$$

within this correspondence, "points" in geometry are identified with **prime ideals**.

An approach suggested by Yuri Manin is to turn logic into an **algebra**, such as the Boolean ring (but this can only handle propositional logic). Varieties defined by such Boolean polynomials [13] live in the space $\mathbb{Z}_2^n$, the **discrete hypercube**.

One of the interesting discoveries in categorical logic is that every topos admits an **internal language**. This is a simple consequence of Curry-Howard: since a type-space corresponds to a logic proposition, and categorical logic interprets type-spaces as objects in a category, thus every category (satisfying extra conditions) can be interpreted as having an "internal" logic. The converse of this correspondence is the **classifying topos** of a logic theory $\mathbb{T}$:

$$\underset{\text{classifying topos}}{\mathcal{E}_{\mathbb{T}}} \quad \overset{\text{internal language}}{\rightleftharpoons} \quad \underset{\text{theory}}{\mathbb{T}} \tag{2}$$

Olivia Caramello [8] developed an idea where toposes play the central role of "bridges" that transfer information between theories (AGI can be seen as the **common-sense theory** of our physical world). She showed that for any geometric theory $\mathbb{T}$, interpreted in the Grothendieck topos $\mathcal{E}_{\mathbb{T}}$, there is a **universal** model U such that any model of $\mathbb{T}$ up to isomorphism is a pullback of U along a geometric morphism. This means that the **classifying topos** of $\mathbb{T}$ is the **representing object** in a **Yoneda lemma**[1]. A diagram in her book is reproduced here with simplifications in Fig.5(Left).

In a similar vein, Ingo Blechschmidt's PhD thesis [5] and his IHES presentation 9 years ago [6] brings the categorical idea of internal language back to its classical setting in algebraic geometry. The situation is as depicted in Fig.5(Right). In this setup, the internal logic comes from the ("big" and "small") **Zariski topology** of the base space, or **site**. The basic idea is that the topology of **open sets** is a **Heyting algebra** that can be interpreted as **intuitionistic logic**[2]. The external view of "sheaves of objects" is simplified to "plain objects" in the internal view, where such objects can be rings (such as $\mathcal{O}_{\text{spec}X}$), modules, etc. The ring $\mathcal{O}_{\text{spec}X}$ contains the polynomials that define the algebraic variety X, and the internal logic can be used to reason about such polynomials.

[1] The Yoneda lemma can be understood as saying some object in a category is able to "represent" the entire category. A archetypal example is **Cayley's theorem** in group theory, that says that every finite group is isomorphic to a subgroup of the symmetric group $\mathfrak{S}_n$. Here the symmetric group is the **representing object** capable of representing all finite groups.

[2] The **negation** (complement) of an open set has to be another open set, leaving the boundary out, thus $\neg\neg A = A$ is not generally valid, as opposed to classical logic.

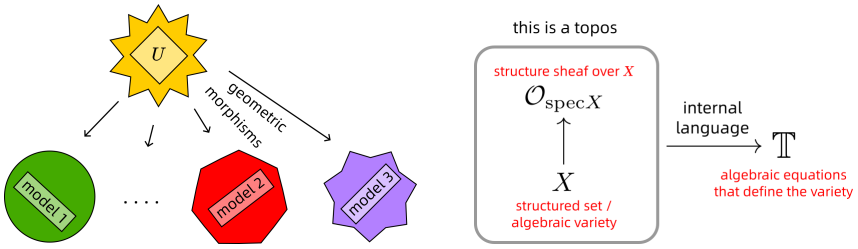

Fig. 5. (Left) The universal model U sits inside its classifying topos (darker color); (Right) The classical formulation of algebraic geometry

Andrei Rodin in his book [15] argues that logic is an axiomatic **abstraction** of the objective world, or as Lawvere [12] [11] puts it, we should "*concentrate the essence of practice to guide practice*" (in the Foreword to [1]). This serves as a nice closing remark.

References

[1] Adámek, Rosický, and Vitale. *Algebraic theories – a categorical introduction to general algebra*. 2011.
[2] J.P. Aubin and H. Frankowska. *Set-valued Analysis*. Modern Birkhauser classics. Springer, 1990. ISBN: 9780817634780. URL: https://books.google.com.hk/books?id=g_OUevAowq4C.
[3] nLab authors. *Beck-Chevalley condition*. URL: https://ncatlab.org/nlab/show/Beck-Chevalley+condition.
[4] nLab authors. *Frobenius reciprocity*. URL: https://ncatlab.org/nlab/show/Frobenius+reciprocity.
[5] Ingo Blechschmidt. "Using the internal language of toposes in algebraic geometry". PhD thesis. University Augsburg, 2017. URL: arXiv:2111.03685v1.
[6] Ingo Blechschmidt. *Using the internal language of toposes in algebraic geometry, presentation in IHES, video on YouTube*. 2015. URL: https://www.youtube.com/watch?v=7S8--bIKaWQ.
[7] Tom B. Brown et al. *Language Models are Few-Shot Learners*. 2020. arXiv: 2005.14165 [cs.CL].
[8] Caramello. *Theories, sites, toposes – relating and studying mathematical theories through topos-theoretic 'bridges'*. 2018.
[9] Angeliki Giannou et al. *Looped Transformers as Programmable Computers*. 2023. arXiv: 2301.13196 [cs.LG].
[10] Bart Jacobs. *Categorical logic and type theory*. Elsevier, 1999.
[11] Lawvere and Rosebrugh. *Sets for mathematics*. Cambridge, 2003.
[12] Lawvere and Schanuel. *Conceptual mathematics*. Cambridge, 1997, 2009.
[13] Samuel Lundqvist. *Boolean ideals and their varieties*. 2015. arXiv: 1211.3398 [math.AC].

[14] Qi et al. “Pointnet++: Deep hierarchical feature learning on point sets in a metric space”. In: *Advances in Neural Information Processing Systems* (2017), pp. 5105–5114.
[15] Andrei Rodin. *Axiomatic method and category theory*. 2014.
[16] Vaswani et al. “Attention is all you need”. In: (2017). https://arxiv.org/abs/1706.03762.
[17] Wikipedia. *No free lunch theorem*. https://en.wikipedia.org/wiki/No_free_lunch_theorem. URL: https://en.wikipedia.org/wiki/No_free_lunch_theorem.
[18] Wolpert and Macready. “No free lunch theorems for optimization”. In: *IEEE transactions on evolutionary computation 1* 67 (1997).
[19] Liu Yang et al. *Looped Transformers are Better at Learning Learning Algorithms*. 2024. arXiv: 2311.12424 [cs.LG].
[20] Zaheer et al. “Deep sets”. In: *Advances in Neural Information Processing Systems* 30 (2017), pp. 3391–3401.