A Logical Degeneration of Vector Symbolic Architectures

Giosuè Zanello

10/07/25

What's XAI?

Explainable AI (XAI) refers to methods and techniques that enable humans to understand, trust, and manage the decisions made by AI systems. It focuses on making the "black box" nature of some AI models more transparent and interpretable, allowing users to see how and why an AI system arrived at a particular outcome.

Why is it urgent nowadays?

- **Increased AI Adoption:** Widespread use in critical sectors (e.g., healthcare, finance) requires transparency for trust.
- **Regulatory Demands:** Laws like the EU AI Act mandate explainability for compliance.
- **Bias and Fairness:** Opaque AI can amplify biases, needing urgent correction (see Grok recently).
- **Safety Risks:** Unexplainable errors in autonomous systems (e.g., self-driving cars) pose immediate dangers.
- **Public Trust:** Lack of understanding erodes confidence, especially in high-stakes decisions.

What’s VSA?

- **Definition:** Vector Symbolic Architecture (VSA) integrates high-dimensional vectors with symbolic reasoning in neuro-symbolic AI, combining neural perception and symbolic logic.
- **Core Concept:** Represents symbols as vectors, using binding (combining) and bundling (adding) for robust, distributed encoding.

How this work

Binding

Definition: Combines two vectors using an operation (e.g., circular convolution) to create a new vector representing their relationship, a relational link

Example: C = bin(A as "Wittgenstein", B as isAuthorOf)

Show C: Wittgenstein isAuthorOf

Proprieties: they also have different mathematical proprieties which I'll list later.

Bundling

Definition: Bundling superimposes multiple vectors to represent a set of attributes, using an operation like vector addition: a cluster of proprieties, not a single relation.

Example: D = bun(A as "Wittgenstein", B as "Philosopher", C as "1898")

D could classificate "Wittgenstein" into "Philosopher born in the late 19th century"

What’s VSA?

- **Definition:** Vector Symbolic Architecture (VSA) integrates high-dimensional vectors with symbolic reasoning in neuro-symbolic AI, combining neural perception and symbolic logic.
- **Core Concept:** Represents symbols as vectors, using binding (combining) and bundling (adding) for robust, distributed encoding.
- **Core Problem (in my opinion):** The sense of representation is lost in linear algebraic operations

"The main idea is a framework that maps the operations of the VSA by symbols and makes them understandable in propositional logic."

Here’s when Ludwig comes in.

Ludwig is a framework that maps distributed operations in a Vector Symbolic Architecture (VSA) to propositional logic rules. It enables transparent tracking, explanation, and verification of symbolic inference within high-dimensional vector spaces, making the system’s reasoning interpretable and its errors auditable.

This occurs by a solid theory based on operation’s logic properties.

We need to create logical operators for those rules:;

Bundling properties

- Aggregation of concepts (not formally composition, as will be seen in the binding)

  R1 := ((A∧B)⇔C)∧(C ⇔ A)∧(C ⇔ B)

- Commutativity of operations

  R2 := (A∧B)⇔(B∧ A)

- Vectorial similarity

  R1 := ((A∧B)→C)∧(C → A)∧(C →B) (reformulating R1)

Binding properties

- Vectorial composition

  R1 := ((A∧B)⇔C)∧(¬(C ⇔ A)∧¬(C ⇔ B))

- Non-commutativity

  R2 := ¬(((A∧B)⇔C)⇔(B∧ A)⇔C) or to try to avoid contradiction

  ¬((A →(B →C))⇔(B →(A →C))), which is structural.

- Symbol deduction

  R3:= ((A∧C)→ B)∧((B∧C)→ A)

Second rule issue

Second rule formalizes asymmetrical inference, rejecting symmetry between A and B.

Negative condition ensures **distinct semantic levels**, <u>not</u> a **logical error**.

Second rule is “structurally false” but “epistemologically true,” securing model distinction. Valid degeneration of binding relies on first (new symbol generation) and third (compositive symbol deduction) rules.

01

Model acquisition (e.g. HRR) and setup: numpy, scipy, networkx, pandas, etc...

02

Vectorial mapping: def LogicMap.

03

Transduction: verification of compliance with logical properties.

04

Verify: pyEDA, Z3 (Microsoft).

Conclusions

- This framework proposes a logical degeneration of VSAs, translating distributed operations into propositional logic; symbols that can be followed
- By collapsing high-dimensional spaces into verifiable symbolic rules, we offer an explainable and auditable approach to neuro-symbolic reasoning.
- Inspired by Wittgenstein’s vision of **logic as the mirror of the world**, we reclaim the epistemic traceability of meaning within AI systems.

“Logic fills the world: the limits of the world are also its limits.”

— Wittgenstein, Tractatus 5.61

Bibliography

- Bechtel, W., & Abrahamsen, A. (2002). Connectionism and the mind: An introduction to parallel processing in networks (2nd ed.). Oxford: Blackwell.
- Besold, T. R., Garcez, A. d., & Bader, S. (2017). Neural-symbolic learning and reasoning: A survey and interpretation. arXiv preprint, arXiv:1711.03902. https://arxiv.org/abs/1711.03902
- Churchland, P. S., & Sejnowski, T. J. (1992). The computational brain. Cambridge, MA: MIT Press.
- Clark, A. (1993). Associative engines: Connectionism, concepts, and representational change. Cambridge, MA: MIT Press.
- Eliasmith, C. (2013). How to build a brain: A neural architecture for biological cognition. Oxford University Press.
- Fodor, J. A., & Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. Cognition, 28(1–2), 3–71. https://doi.org/10.1016/0010-0277(88)90031-5
- Gallant, S. I., & Okaywe, T. W. (2013). Representing objects, relations, and sequences. Neural Computation, 25(8), 2038–2078. https://doi.org/10.1162/NECO_a_00467
- Gayler, R. W. (2003). Vector symbolic architectures answer Jackendoff's Haykin, S. (2009). Neural networks and learning machines (3rd ed.). Upper Saddle River, NJ: Pearson Education.
- Hersche, M., Zeqiri, M., Benini, L., Sebastian, A., & Rahimi, A. (2023). A neuro-vector-symbolic architecture for solving Raven's Progressive Matrices. IBM Research – Zurich & ETH Zürich. Recuperato da https://github.com/IBM/neuro-vector-symbolic-architectures
- Kanerva, P. (2009). Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors. Cognitive Computation, 1(2), 139–159. https://doi.org/10.1007/s12559-009-9009-8
- Marcus, G. (2001). The algebraic mind: Integrating connectionism and cognitive science. Cambridge, MA: MIT Press.
- Neubert, P., Schubert, S., & Protzel, P. (2019). An introduction to hyperdimensional computing for robotics. KI – Künstliche Intelligenz, 33, 195–203. https://doi.org/10.1007/s13218-019-00623-z
- Plate, T. A. (1995). Holographic reduced representations. IEEE Transactions on Neural Networks,
- 6(3), 623–641. https://doi.org/10.1109/72.377968
- Plate, T. A. (2003). Holographic reduced representations: Distributed representation for cognitive
- structures. Stanford: CSLI Publications.
- Rumelhart, D. E., & McClelland, J. L. (1986). Parallel distributed processing: Explorations in the
- microstructure of cognition (Vol. 1). Cambridge, MA: MIT Press.
- Schlegel, K., Neubert, P., & Protzel, P. (2021). A comparison of vector symbolic architectures.