MAPS OF THE LABYRINTH

A SERIES FROM PHANTOM MAZE — AI & LANGUAGE LAB

ESSAY 20

Embeddings: The Map Without a Cartographer

Claudia Marsico · Hernán Inverso

DOI 10.5281/zenodo.23647190

Language can be turned into geometry. Embeddings show how far a machine can go when it steers by relations



Marc Chagall, I and the Village (1911)

Antisthenes, the Socratic who mocked Plato, rejected definitions and wielded his own recipe: to know what a thing is one has to place it among others. The idea has a surprisingly modern ring. A word sits in a network of relations involving similarities, oppositions or habitual associations. Antisthenes's proto-theory of semantic fields echoes in embeddings, which place linguistic units in a mathematical space according to the regularities they learned from their uses. The resemblance ends there, but the

move is similar: instead of asking what something means one has to ask where it falls with respect to the rest.

When we speak of tokens we mean text cut into discrete pieces of a computational vocabulary, which do not necessarily correspond to words, letters, morphemes or units of meaning. But a single chip is still of no use to a neural network, because to compute it needs numbers. The next step therefore consists in turning each token into a vector, that is, an ordered list of numbers that can be imagined as coordinates. With two coordinates we locate a point on a sheet; with three, a point in a room. With three thousand we still have a point, even though we can no longer draw the space it lives in, manageable computationally.

Marc Chagall, I and the Village (1911)

How does each token receive such a list? In a model there is a large table learned during training, the embedding matrix. Each chip of the vocabulary has a row of that table associated with it. When a token appears, the system takes the corresponding row and thereby obtains its first numerical representation. The token identifier — suppose "dog" were a single token and had the identifier 1734 — means nothing by itself. It is merely an address at which to look up its vector. That vector is dense, because it is made up of many numerical values different from zero, it is learned, because nobody decided by hand what its numbers should be, and it is distributed, because a relevant property does not live in a single coordinate, as if the number 82 meant "animal" and 417 "plural". The regularities are spread across many dimensions and acquire meaning as a whole.

The English verb *to embed* means to fix something inside a material that surrounds it, like a stone in concrete. An *embedding* fixes a unit of language inside a space populated by others. The token that in the previous essay was a chip separated from all the others now becomes a position from which proximities, differences and directions can be computed. The linguistic intuition that makes that geometry useful came from the middle of the 20th century. Zellig Harris proposed studying the units of a language by observing their distribution, that is, the contexts in which they appear. Two words that can occupy similar places in many constructions have something in common detectable by distributional analysis without definitions. John Firth condensed the intuition a few years later in the phrase: "you shall know a word by the company it keeps". Use leaves traces. "Dog" appears again and again in surroundings where we find bark, leash, walk, pet, vet; "mortgage", near bank, instalment, interest, housing, loan. If we gather sufficient quantities of text, those regularities begin to form a kind of statistical signature.

In 2013, Tomas Mikolov and his group presented word2vec, which found an efficient way to compress those traces into vectors. In one variant, the training task used a word to predict which ones tend to appear nearby; in the other, it reconstructed a word from those surrounding it. To improve its predictions, without a dictionary or prior classification of terms, training modified again and again the numbers associated with each word. In the end, words used in similar contexts tended to end up represented by vectors that were also similar.

But what does it mean for two lists of hundreds of numbers to be "similar"? Let us imagine each vector as an arrow starting from the origin of that enormous space. If two arrows point roughly in the same direction, they have a high cosine similarity; if they form a larger angle, that similarity decreases. That is how the map is born. Words used in similar ways tend to occupy nearby regions. "Dog" and "cat" may end up close because they appear in similar contexts, but the same can happen with opposite terms. "Cold" and "hot" share a great many companions associated with temperature, water, weather, sensation, winter or summer. The space registers regularities of use first and from the combination of many of them it can distinguish classes of different relations.

The surprise of word2vec was that, beyond forming neighbourhoods, some relations repeated throughout the language appeared as roughly regular displacements. Certain plurals with respect to their singulars, some verb tenses, capitals with respect to countries or feminine forms with respect to masculine ones could be found by following similar directions. From that came the operation that made the model famous: take the vector for "king", subtract that of "man", add that of "woman" and end up near "queen". Analogies do not always work, they depend heavily on the corpus and on the mode of training and they do not prove that a universal algebra is hidden behind words, but a relation that nobody had explicitly programmed acquired a geometric form. The statistics of use produced directions.

In the 17th century, Leibniz dreamed of a *characteristica universalis*, an alphabet of concepts in which every idea had a sign assigned to it, so that reasoning would be calculating. Faced with any dispute, instead of arguing it would be enough to sit down and say *calculemus*, let us calculate, and do the sums until we see who is right. *Embeddings* brush against that extravagant dream made of concepts turned into vectors, analogies expressible as arithmetic displacements, although along the way they have lost the rational taxonomy Leibniz expected. They also turn upside down the ideal of an Adamic language in which the name reflects the essence of the thing. The embedding is indifferent to what a dog is and is interested only in what it appears with and in what contexts it moves. Where the Adamic language had essence, the vector space finds neighbourhood.

The geometrisation of language showed itself in latent semantic analysis, developed around 1990. The procedure started from an enormous table recording which terms appeared in which documents and applied algebraic techniques to compress it. That compression made proximities emerge that were not evident in the original table, but word2vec took the principle to a new scale and precision by learning representations through local prediction. GloVe, presented by Stanford researchers in 2014, for its part exploited global co-occurrence statistics. Although the methods differed, they shared the limitation that each word received a single fixed vector. "Bank" had to live always at the same point, whether it was the bank of a river or the institution where money is deposited. The vector ended up being a kind of average of its uses.

And if a word had no single position? ELMo explored in 2018 the idea of building representations that depended on the whole sentence. That same year BERT, already supported by the transformer architecture, generalised and popularised the change. Since then it is worth distinguishing two things.

On entering the model, a token receives a relatively stable initial embedding, taken from the table we described earlier. But that is only the starting point, since as it passes through the layers its vector changes under the effect of context. Thus, the token corresponding to "bank" starts from the same basic lexical embedding, though the input representation also incorporates information about its position and ends up occupying different positions after the architecture incorporates the context. Those later vectors we usually call contextual representations or, in a broader sense, contextual embeddings. Instead of a single point for "bank" there is a position built for that particular occurrence. Attention is one of the central mechanisms that produce that transformation and in that way the map, instead of a printed atlas assigning each word a fixed city, turns out to be a map that is redrawn as the sentence is read.

The same geometric idea later expanded beyond words. Sentence-BERT, in 2019, produced vectors especially useful for representing sentences and whole paragraphs, so that two texts could be compared by their semantic proximity. CLIP, in 2021, took the principle into the territory of images and texts so that both kinds of object could occupy compatible regions of the same space. A photograph of a dog and an appropriate description of the dog could end up close even though they are physically incomparable things, making different modalities coexist within a single learned geometry.

What is this good for? A traditional search relies on matching terms, whereas a vector search can turn questions or documents into vectors and retrieve those that ended up closest, even if they do not use the same vocabulary. Asking about "how to lower the cost of heating a house" can retrieve a passage about "improving thermal insulation", even though the phrases share no terms. To carry out that kind of search at scale, vector databases are used, designed to store and quickly find neighbours within immense collections of vectors. That search is also a central piece of many RAG systems, the acronym for Retrieval-Augmented Generation. Before receiving questions, the system prepares a collection of texts: it divides them into fragments, computes a vector for each and leaves them indexed. When a query arrives, it also computes its vector and looks for which of those fragments ended up closest. Then it places the retrieved passages next to the question and hands them to the language model. The embedding helps decide what information is worth putting in front of the model before it answers. The same logic serves to recommend content similar to what someone has already consumed, to group documents by affinity, to detect atypical cases, to sort large archives or to relate texts and images. What began as a way of representing words ended up converted into a general procedure for turning complex objects into comparable points and measuring between them different forms of similarity.

Let us return to the strangest point. No cartographer drew this map or decided in advance what each axis should mean. Latitude and longitude are a human decision, with a zero meridian that ended up at an observatory in London like a flag of naval power in the 19th century. In an embedding space nobody began by saying that a certain dimension would represent gender, another temporality, another authority and another the relation between an animal and its sound. Training focused on predicting contexts and adjusted the numbers until it did so ever better. The geometry appeared as a

result. We can explore it by training probes that detect grammatical or semantic information and even intervene on some patterns, but the map works better than we can gloss it.

We began with Antisthenes abandoning essence for relation, changing the question what a dog is for where it falls. In that the embedding is a kind of son of his. The filial link explodes when Antisthenes affirms that every name refers to an existing thing, to the point of the paradox that error is impossible: whoever says something, says something that is, and whoever says something that is, says the truth. The embedding cut its moorings to the world. "Dog" receives its position because the word appeared millions of times near bark, leash and vet. Words are compared by their neighbourhood and the result is a theory of meaning that works without the world intervening. We designed a technology that can dispense with the question of correspondence. The map represents the traces that some terms left when appearing next to others. It bears remembering because we are indeed interested in the world and in things. The embedding shows how far an intelligence can go when it orients itself by relations without starting from an essential definition and it also makes plain the cost of that freedom, since it can learn to move with astonishing precision among our words without its criteria of relevance therefore coinciding with ours. So this map may be excellent for travelling through language and yet not tell us where the borders that matter in the world are.

◊

HOW TO CITE

Marsico, Claudia, and Hernán Inverso. 2026. "Embeddings: The Map Without a Cartographer." *Maps of the Labyrinth*, essay 20. Phantom Maze — AI & Language Lab. https://doi.org/10.5281/zenodo.23647190

Maps of the Labyrinth is a series from Phantom Maze, AI & Language Lab. The collection lives at **phantommaze.cloud**; new essays also appear at phantommazelab.substack.com. Subscription is free.

Some of the framing ideas in this piece are developed in *The Egoless Phantom: Mapping the AI Labyrinth*, by Hernán Inverso and Claudia Marsico (Phantom Maze Press, 2026; ISBN 978-987-3729-16-4).

PHANTOM MAZE

Read this essay online · All essays · The book

References. Scott Deerwester et al., "Indexing by Latent Semantic Analysis," *Journal of the American Society for Information Science*, vol. 41, no. 6, 1990, pp. 391–407; Jacob Devlin et al., "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *arXiv*, 11 Oct. 2018, arXiv:1810.04805; J. R. Firth, "A Synopsis of Linguistic Theory, 1930–1955," *Studies in Linguistic Analysis*, special volume of the Philological Society, Basil Blackwell, 1957, pp. 1–32; Zellig S. Harris, "Distributional Structure," *Word*, vol. 10, nos. 2–3, 1954, pp. 146–62; Gottfried Wilhelm Leibniz, *Dissertatio de arte combinatoria*, Fick and Seubold, 1666; Tomas Mikolov et al., "Efficient Estimation of Word Representations in Vector Space," *arXiv*, 16 Jan. 2013, arXiv:1301.3781; Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig, "Linguistic Regularities in Continuous Space Word Representations," *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics, 2013,

pp. 746–51; Jeffrey Pennington et al., "GloVe: Global Vectors for Word Representation," *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, 2014, pp. 1532–43; Matthew E. Peters et al., "Deep Contextualized Word Representations," *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, Association for Computational Linguistics, 2018, pp. 2227–37; Alec Radford et al., "Learning Transferable Visual Models from Natural Language Supervision," *Proceedings of the 38th International Conference on Machine Learning*, edited by Marina Meila and Tong Zhang, vol. 139, Proceedings of Machine Learning Research, 2021, pp. 8748–63; Nils Reimers and Iryna Gurevych, "Sentence-BERT: Sentence Embeddings Using Siamese BERT-Networks," *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Association for Computational Linguistics, 2019, pp. 3982–92.