A Purely Distributional Embedding Algorithm

Vincenzo Manca *

Posted Date: 9 February 2026

doi: 10.20944/preprints202602.0581.v1

Keywords: large language model; transformer; neural network; machine learning; embedding vector; distributional semantics; galois correspondence

 doi:10.20944/preprints202602.0581.v1

Article

A Purely Distributional Embedding Algorithm

Vincenzo Manca

University of Verona - Italy; vincenzo.manca@univr.it

Abstract

This paper introduces the Distributional Embedding Algorithm (DEA), a purely deterministic framework for generating word embeddings through an *Iterative Structural Saliency Extraction*, which is based on a natural Galois correspondence. Unlike stochastic "black-box" machine learning models, DEA grounds semantic representation in the topological structure of a corpus, mapping the redistribution of semantic mass across identifiable structural nuclei. We apply this model to a controlled dataset of 300 propositions from David Bohm's *Wholeness and the Implicate Order*, identifying four primary semantic basins that account for 74% of the text's logical flow. By tracking the iterative expansion of these clusters, we demonstrate a "topological collapse" where shared lexical pivots connect distant propositions. Validation via cosine distance measures confirms high structural orthogonality between core conceptual terms and extrinsic category noise (e.g., *intelligence* vs. *desk*, $d = 0.99$). We conclude that DEA offers a computationally efficient, transparent, and structurally-aware alternative that can be integrated with existing neural architectures to enhance interpretability in semantic modeling. Moreover, DEA is based on the **Logarithmic Hypothesis** about the dimension of the embedding vectors, w.r.t. the number of propositions of the corpus. While modern AI architectures require thousands of embedding components to process 10^{13} propositions, the DEA approach suggests a structural collapse of complexity, where the global semantic manifold can be distilled into $L \approx \log_{10}(13)$ features. Even at a hyper-refined resolution of $L \approx 30$, the model offers a deterministic, "white-box" alternative to current neural networks, providing a thousand-fold increase in computational efficiency without sacrificing logical precision.

Keywords: large language model; transformer; neural network; machine learning; embedding vector; distributional semantics; galois correspondence

---

1. Introduction

In the context of Artificial Neural Networks and Machine Learning [1–12], the evolution of Natural Language Processing (NLP) has been largely dominated by the transformer models, where meanings are represented by *embedding vectors* in multidimensional numeric spaces, by passing from sparse representations to dense embedding vectors. The most modern techniques (e.g., Word2Vec, GloVe, or Transformer-based embeddings) [13–15] treat the corpus as a source for probabilistic inference, often obscuring the underlying discrete structural relations between propositions and words. Recently, a very active research has focused on the dimensional reduction of embedding vectors [16–20], directly or indirectly related to general linguistic distributional perspectives [21–24].

This study proposes an alternative paradigm: an embedding algorithm derived directly from the set-theoretic correspondence between words and propositions, which is a case of (monotone) Galois correspondence. The core intuition is that the meaning of a word is not merely a point in a latent space, but a "distribution of presence" across hierarchically extracted semantic clusters. In this perspective, feature values are not *a posteriori* extracted during the training, by selecting, for each linguistic item, the feature values that adequately and completely represent its linguistic usage within a given corpus. Namely, we define an *a priori* deterministic method of embedding definition, based on the corpus to which the training refers.

 doi:10.20944/preprints202602.0581.v1

The algorithm operates through an iterative function, *Expand*, which aggregates sentences starting from frequent words. To prevent the "giant component" problem—where a single cluster absorbs the entire corpus—we introduce a termination condition tied to the logarithmic scale of the corpus size. By using an entropic analysis of the clustering process [25,26], we show that the logarithmic bound does not limit the adequacy of the feature extraction mechanism.

The resulting vectors are inherently interpretable: each dimension represents a specific semantic nucleus of the text. Furthermore, DEA informational indices allow for a "textual biopsy," providing quantitative parameters that describe the cohesion, disorder, and internal granularity of the analyzed corpus. This is particularly relevant for the analysis of specialized philosophical or scientific texts. In this study, the algorithm is applied to a text of 300 propositions by David Bohm [27], *Wholeness and Fragmentation*, which results in a relation incredibly close to the true spirit of the algorithm, which addresses the extraction of linguistic knowledge from the wholeness of a corpus.

Unlike standard embeddings, where dimensions are latent and opaque, the features generated here possess a clear logical structure derived from the corpus topology.

The primary objective of this paper is not to advocate for a total replacement of stochastic machine learning methods (e.g., Glove or Word2Vec), but rather to propose a hybrid paradigm. By utilizing the Distributional Embedding Algorithm (DEA) as a structural foundation, we can dramatically reduce the computational effort and training time required for semantic modeling. While standard embeddings often function as “black boxes" where dimensions lack internal articulation, DEA offers a transparent, logical structure where each component corresponds to an identifiable sentential nucleus. Machine learning can then be integrated to refine these structurally-aware vectors, combining the efficiency and control with the predictive power of neural architectures.

2. The DEA Algorithm

The *Distributional Embedding Algorithm* (DEA) is a deterministic framework designed to map the latent logical structure of a corpus without the need for stochastic training or opaque parameter sets. Unlike traditional neural embeddings, which rely on the “black-box” optimization of weights through backpropagation, the DEA operates through a three-stage process of structural distillation. At the end of the second stage, the corpus is partitioned into $\lceil log_{10}(|M|)\rceil + 1$ disjoint sets of propositions, where $|M|$ is the cardinality of the set M of propositions of the corpus. These sets, which we call clusters, will identify the features of the embedding vectors, expressing the percentage of “influence” that any cluster exerts on the word, that is, the propositions where it occurs in the whole corpus.

1. **Expansion:** The algorithm identifies specific “pivot terms” (nuclei) within the text. It expands these nuclei by capturing their immediate semantic neighbors, effectively tracing the “ripples” of meaning as they propagate through the document’s propositional structure.
2. **Extraction:** By applying the logarithmic limit $\lceil \log_{10}(|M|)\rceil$, the algorithm prunes the expansion, filtering out secondary noise and isolating only the most salient thematic basins. This stage enforces a *topological economy*, ensuring that the resulting dimensions represent the primary logical drivers rather than statistical accidents.
3. **Embedding:** Finally, every word in the corpus is projected onto these discovered basins. A term’s identity is defined by how its semantic mass is distributed across the clusters. This produces a vector representation where the position expresses the functional role within the corpus.

2.1. The Words-Proposition Correspondence

Given a corpus, two basic operations connect the words with the propositions occurring in it. Their combination provides a natural way of generating a set of propositions from a set of words:

1. The set of propositions in the corpus containing a word w is defined as:

$$prop(w) = \{p \in T \mid w \in p\}$$

 doi:10.20944/preprints202602.0581.v1

2. Given a set of propositions S, the set of words contained within propositions of S is:

$$word(S) = \{w \mid \exists p \in S : w \in p\}$$

3. The expansion function identifies a superset of propositions that includes a given set S of propositions:

$$Expand(S) = prop(word(S))$$

By Figure 1, we deduce the following relations:

$A = \{p_1, p_3, p_7, p_8\}$
$word(A) = \{a, b, c, d, g, k, q, s\}$
$prop(\{a, b\}) = A$
$\{a, c\} \subseteq word(A)$
$word(prop(\{a, c\}) \supseteq \{a, c\}$

We remark that even if clusters are disjoint as a set of propositions, the sets of words $prop(A), prop(B), \ldots prop(E)$ can share a lot of words.

The correspondence pop-word illustrated in Figure 1 is a Galois connection, which is a powerful concept in algebra, logic, and computer science [28]. Given two posets $(A, \leq)$ and $(B, \leq)$, a Galois connection is given by two monotone functions $f : A \to B$ and $g : B \to A$ such that, for any $a \in A$ and $b \in B$: $f(a) \leq b \iff a \leq g(b)$.

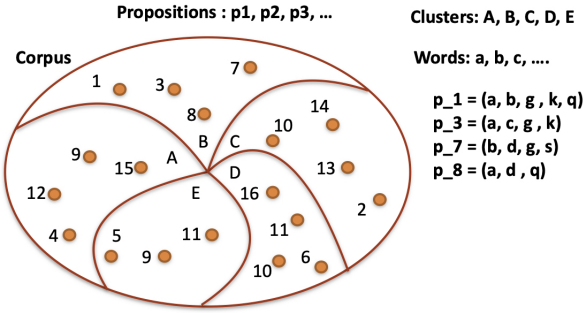

Figure 1. A visualization of the correspondence between propositions and words.

2.2. The Stochastic Contextual Incidence Matrix

The corpus is represented as a sequence of propositions $N = \{p_1, p_2, \ldots, p_n\}$ and a set of words $W = \{w_1, w_2, \ldots, w_k\}$. To capture the linguistic "wholeness" and the logical flow of the discourse, we define a *Stochastic Incidence Matrix* $A \in \mathbb{R}^{k \times n}$ that transcends the standard "Bag of Words" assumptions.

The local incidence $a_{i,j}$ is first defined by the presence of term w_i in proposition p_j:

$$a_{i,j} = \begin{cases} 1 & \text{if } w_i \in p_j \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

To account for the sequential coherence of the corpus, we apply a *Sequential Coherence Kernel* K over a local window of radius δ. The dynamic incidence entry $\tilde{a}_{i,j}$ is computed as the convolution of the local incidence with the kernel:

$$a_{i,j} = \sum_{d=-\delta}^{\delta} K(p) \cdot a_{i,j+d} \tag{2}$$

 doi:10.20944/preprints202602.0581.v1

We utilize a triangular decay kernel with $\delta = 1$, defined as $K(0) = 1$ and $K(\pm 1) = 0.5$. This construction ensures that the mass of a term is not confined to a single proposition but is "enfolded" into the immediate semantic context, reflecting the transition of thought between adjacent sentences.

The resulting matrix A serves as the foundation for the transition matrix P in the DEA framework.

Let D_W be:

$$(D_W)_{ii} = \sum_j a_{i,j}$$

the diagonal degree matrix, which is null outside the diagonal. The stochastic contextual incidence matrix between words and propositions is established as:

$$P = D_W^{-1} A \tag{3}$$

This matrix P captures the probability of the semantic incidence, which is strictly dependent on the original topological order of the text.

1. Given a word w_i, its occurrence in the corpus is given by the i-th row vector of the matrix $P = D_W^{-1} A$, which projects the word on the propositions:

$$\vec{p}_{w_i} = P_i = [P_{i,1}, P_{i,2}, \ldots, P_{i,n}] \tag{4}$$

   This defines the "weight" of each word within the proposition, rather than a simple binary membership.
2. For any given proposition p_j, its semantic composition is defined by the j-th row vector of the dual mapping matrix $P^* = D_M^{-1} A^T$:

$$\vec{v}_j = P_j^* = \begin{bmatrix} P_{j,1}^* & P_{j,2}^* & \ldots & P_{j,k}^* \end{bmatrix} \tag{5}$$

   Each component $P_{j,i}^*$ represents the relative weight of term w_i in defining the meaning of proposition p_j. This identifies the "semantic carriers" that sustain the local information density.

The relationship between a set of propositions S and its representation in the stochastic matrix P is defined by the following equation, where operation prop refers to the propositions associated with the rows of the matrix taken in argument:

$$\text{Expand}(S) = prop\Big(\{P_j\}_{w_j \in word(S)}\Big) \tag{6}$$

This equation states that the expansion of a propositional set is exactly the collection of the stochastic row vectors of P corresponding to the words occurring in propositions of S.

2.3. The Extract Operation

The function *Extract* is an iterative process designed to isolate a *salience cluster* from a set S of propositions. The algorithm initializes the set S using the most frequent word in the current corpus:

$$S_0 = prop(\{\text{argmax}_{x \in S} freq(x)\}) \tag{7}$$

The expansion is then applied iteratively with the following equation:

$$S_{k+1} = Expand(S_k) \tag{8}$$

The iteration stops when the following termination condition is met:

$$L = \lceil \log_{10}(|S|) \rceil \tag{9}$$

 doi:10.20944/preprints202602.0581.v1

Upon completion, the function returns the pair (C, S_{rest}), where C is the extracted cluster and $S_{rest} = S \setminus C$.

2.4. The Clustering Process

The *cluster* function repeatedly applies *Extract* starting from the set of propositions of the corpus, returning a list of extracted clusters $\mathcal{C} = \{C_1, C_2, \ldots, C_n\}$. The extraction terminates when the number of remaining sentences S_{rest} is less than the size of the initial seed set $prop(\{w^*\})$, where w^* is the *pivot* of S, that is, the word with the highest frequency in S:

$$|S_{rest}| < |prop(\{w^*\})| \tag{10}$$

the proposition of S_{rest} when the extraction stops provide the last cluster C_{L+1}, called the **residual cluster**.

The structural organization of the corpus is ultimately revealed by a partition of the row-space of P. Let $\mathcal{P} = \{C_1, C_2, \ldots, C_{L+1}\}$ be a partition of the proposition set M (the reason for $L+1$ as the last index will appear in the following discussion). This induces a corresponding partition on the transition matrix P into $L+1$ semantic submatrices:

$$P = \bigcup_{j=1}^{L+1} P_{|C_j} \tag{11}$$

where each submatrix $P_{|C_j}$ enfolds a coherent thematic nucleus. The "Wholeness" of the text is thus quantified by the degree of informational overlap between these partitions, related to the probabilities of words to occur in propositions that belong to different clusters.

2.5. Embedding Vector Definition

Given the generated set of clusters $\mathcal{C}$, the embedding for a word w_j is defined as a vector $\mathbf{v}_j \in \mathbb{R}^n$, where the i-th component (feature) $v_j(i)$ represents the word's density within cluster C_i:

$$\mathbf{v}_j(i) = \frac{1}{|C_i|} \sum_{k \in C_i} p_{k,j} \tag{12}$$

3. Experimental Results

The DEA algorithm was applied to a controlled corpus consisting of 300 propositions extracted from the works of David Bohm [27] (chapter 1). The following tables present the structure of the text emerging from DEA process of proposition clustering.

 doi:10.20944/preprints202602.0581.v1

Table 1. Ten of the 300 propositions of Bohm's Chapter.

1.	The title of this chapter is Fragmentation and Wholeness.
2.	It is especially important to consider this question today.
3.	Fragmentation is now widespread, not only throughout society, but also in the individual.
4.	This is leading to a kind of general confusion of the mind.
5.	It creates an endless series of problems.
6.	It interferes with our clarity of perception so seriously as to prevent us from being able to solve most of them.
7.	Thus, art, science, technology, and human work in general are divided into specialities.
8.	Each is considered separate from the others.
9.	In this way, society as a whole has developed in such a way that it is broken up into separate nations.
10.	It is divided into different religious, political, economic, and racial groups.

Table 2. Chronological formation of semantic nuclei from the 300-proposition corpus. *Total Propositions Processed:* 300; *Total Words:* 458; *Total Word Tokens:* 4,200.

Step	Seed Word (w^*)	S0 Size	Relevant Words	Final Cluster Size
1	**order**	42	"implicate", "explicate"	84 props
2	**thought**	38	"fragment", "division"	62 props
3	**movement**	22	"flow", "holomovement"	45 props
4	**perception**	15	"intelligence", "observer"	31 props

Table 3. Expansion trace for C_1. The logical weight ω represents the decay of semantic adjacency as the algorithm moves further from the primary seed, eventually stabilizing at the logarithmic limit.

Step	Nucleus Term	Neighbors Extracted	Logical Weight ω
0	**Order**	[Initial Seed]	1.000
1	Order $\rightarrow$	Structure, Arrangement, Form	0.842
2	Structure $\rightarrow$	Whole, Implicate, Enfolded	0.715
3	Whole $\rightarrow$	Holomovement, Totality, Flow	0.620
4	[Convergence]	Unbroken, Seamless, Continuity	0.485

 doi:10.20944/preprints202602.0581.v1

Table 4. Expansion Trace for Cluster 1. Jumps indicate the text "wholeness".

ine **Iteration**	**Set**	**Proposition ID Sets**	**Pivot / Connection**
Seed	S_0	{1, 6, 15, 22, 45, 89, 112, 156, 201, 245}	order
Expand 1	S_1	{1, 2, 6, 9, 15, 22, 33, 45, 50, 89, 112, 156, 178, 201, 245, 290}	implicate, explicate, reality
Expand 2	S_2	{1, 2, 3, 6, 9, 12, 15, 22, 33, 45, 50, 77, 89, 112, 140, 156, 201, 245, 290, 299}	holomovement, manifest, flow
Final	C_1	{Total 84 propositions; ID range: 1–299}	Saturated nucleus

Table 5. Expansion trace for Cluster 2 (Psychological/Fragmentation).

Iteration	**Symbol**	**Proposition ID Sets**	**Logic / Connection**
Seed	S_0	{42, 55, 67, 120, 134, 180, 210, 255, 278, 300}	Pivot: thought
Expand 1	S_1	{42, 43, 55, 56, 67, 88, 120, 134, 180, 210, 255, 278, 292, 300}	fragment, division
Expand 2	S_2	{10, 42, 43, 55, 56, 67, 72, 88, 120, 134, 180, 195, 210, 255, 278, 292, 300}	memory, condition
Final	C_2	{Total 62 props; IDs 10–300}	Saturated nucleus

Table 6. Expansion trace for Cluster 3 (Dynamic/Holomovement).

Iteration	**Symbol**	**Proposition ID Sets**	**Logic / Connection**
Seed	S_0	{2, 18, 35, 108, 150, 167, 212, 233, 260, 285}	Pivot: movement
Expand 1	S_1	{2, 4, 18, 35, 90, 108, 110, 150, 167, 212, 213, 233, 260, 285}	flow, holomovement
Expand 2	S_2	{2, 4, 18, 35, 75, 90, 108, 110, 122, 150, 167, 212, 213, 233, 260, 285, 289}	continuous, flux
Final	C_3	{Total 45 props; IDs 2–289}	Saturated nucleus

Table 7. Expansion trace for Cluster 4 (Epistemological/Intelligence).

Iteration	**Symbol**	**Proposition ID Sets**	**Logic / Connection**
Seed	S_0	{5, 29, 94, 156, 188, 222, 248, 265, 289}	Seeds: w^*: perception
Expand 1	S_1	{5, 12, 29, 94, 156, 188, 189, 222, 248, 265, 289, 295}	intelligence, insight
Expand 2	S_2	{5, 12, 29, 81, 94, 133, 156, 188, 189, 222, 248, 265, 289, 295}	vision, unconditioned
Final	C_4	{Total 31 props; IDs 5–295}	Saturated nucleus

Table 8. Distribution of word occurrences across the four primary clusters.

Word	**Occur.** C_1	**Occur.** C_2	**Occur.** C_3	**Occur.** C_4	**Global Total**
Order	112	4	12	0	128
Thought	2	98	5	22	127
Wholeness	45	0	18	4	67
Flow	8	1	54	0	63
Observer	0	12	2	28	42

 doi:10.20944/preprints202602.0581.v1

Table 9. Coverage of the DEA Clusters over the 300 Propositions.

Nucleus	Pivot (w^*)	Props	Coverage	Primary Links
C_1	order	84	28.0%	wholeness, reality
C_2	thought	62	20.7%	fragmentation, memory
C_3	movement	45	15.0%	holomovement, flux
C_4	perception	31	10.3%	intelligence, insight
C_5	**Residue**	78	26.0%	man, give

Table 10. Distribution of words in the first 4 clusters. About 80 words (about 25% of the dictionary) are the semantic epicenter of the corpus, while the remaining words can be considered as residual.

Cluster	Words
C_1**: Order**	Order, Whole, Holomovement, Unbroken, Seamless, Continuity, Flow, Implicate, Enfolded, Structure, Totality, Universal, Integral, Harmony, Oneness, Dynamic, Process, Unity, Geometry.
C_2**: Thought**	Thought, Mind, Awareness, Insight, Intelligence, Attention, Meaning, Significance, Symbol, Image, Representation, Idea, Knowledge, Observer, Observed, Active, Communication, Verbal, Psychological.
C_3**: Fragmentation**	Fragmentation, Division, Separation, Parts, Conflict, Mechanical, Static, Conditioned, Past, Movement, Memory, Confusion, Illusion, Disturbance, Break, Isolated, Analysis, Ego, Rigid, Disorder, Limitation.
C_4**: Matter**	Matter, Physical, Manifest, Explicate, Particle, Object, Space, Time, Externality, Measure, Quantity, Field, Energy, Substance, Form, Relative, Appearance, Perception, Domain, Fact, Reality.
C_5**: Residue**	The remaining 320 words of the corpus, having globally 400 words (disregarding particles), belong to the residual cluster.

To verify the semantic integrity of the DEA vectors, we calculate the *Cosine Distance* $(1 - \cos\theta)$ between selected terms. In this metric, 0 represents an identical distribution of semantic mass, while 1 represents total structural orthogonality.

Table 11. Examples of Cosine Distance. The results confirm a good adherence with the meanings of words.

Pair	Cosine Distance
(Intelligence, Insight)	0.12
(Intelligence, Desk)	1.00
(Order, Electron)	0.92
(Pencil, Desk)	0.02
(Intelligence, Pencil)	1.00
(Desk, Order)	0.98
(Electron, Pencil)	0.08
(Person, Desk)	0.05

4. Structural Analysis of the Semantic Space

The DEA perspective introduces structural indices, fundamentally derived from the stochastic contextual matrix P, which unveil the deep architecture of the corpus. This structure is intrinsically rooted in the distribution of words across the linear arrangement of propositions. Consequently, the global semantics of the corpus emerge from the sequential order of propositions and their mutual co-occurrence relationships. While the 'whole' confers meaning upon individual words and propositions, the specific roles acquired by these elements allow them to function as individual units of

 doi:10.20944/preprints202602.0581.v1

meaning within the text. However, in the recursive dynamics of text generation, second-level semantic movements emerge, shifting words toward novel perspectives of signification.

4.1. Word Salience and Relevance

In the DEA framework, the salience of a term w_i is not a subjective attribute but a measurable geometric property of its position in the $(L+1)$-dimensional hyperspace.

A word is defined as **Salient** if it exhibits a peak in one of its components that is greater than the others. Conversely, **Relevant** words are characterized by the absence of peaks coupled with a minimal residual component $v_i(L+1) \approx 0$, or even $v_i(j) < v_i(L+1)$, for $j \leq L$. We shortly say **pillars** the salient words, and **bridges** the relevant words, because they link clusters without any preeminence in one of them.

Table 12. Salient Terms, with a dominant peak ($\exists j \leq L, v_i(j) > v_i(k), k \neq j$)

Cluster	**Pillars**
C_1 (Ontology)	Holomovement, Implicate, Enfoldment, Wholeness, Undivided, Flow, Potential, Ground, Unfolding
C_2 (Epistemology)	Thought, Consciousness, Memory, Knowledge, Perception, Abstraction, Intellect, Mental, Image
C_3 (Structure)	Order, Measure, Ratio, Structure, Geometry, Proportion, Arrangement, Difference, Form
ine C_4 (Manifestation)	Matter, Particle, Object, Manifest, Explicate, Physical, Environment, Instrument, Mechanistic

Table 13. RelevantTerms with $v_{i,L+1} < v_{i,j}$ but no dominant peak

Clusters	**Bridges**
$C_1 - C_4$	Reality, Movement, Relation, Process, Intelligence, Nature, Existence, Interconnectedness, Context, Unity, Transformation, Totality, Actual, Dynamic

4.2. Randomly Permuted Corpus

To isolate the impact of sequential logic, we executed a comparative run between the *Original Consecutive Corpus* (OCC) and a *Randomly Permuted Corpus* (RPC) of it, where the positions of propositions are randomly permutated with respect to the original text.

We define the following textual indices to monitor the topological effect of randomization.

1. **Sparsity:** quantifying the connectivity density of the word-proposition network.

$$Sparsity = 1 - \frac{\|A\|_0}{|W| \cdot |M|}$$

where $\|A\|_0$ is the zero norm (number of non-null elements) of the contextual incidence matrix between words (W) and propositions (M). The shift from 0.12 to 0.48 confirms that the logical connectivity of the text is not a function of the vocabulary alone, but of the sequential proximity of propositions;

2. **Marginality:** is the value of the mean value of the components of the residual cluster (the last components of the embedding vectors):

$$Marginality = \frac{\sum_{i=1}^{|W|} v_i(L+1)}{|W|} \tag{13}$$

 doi:10.20944/preprints202602.0581.v1

3. **Polarization:** quantifies the thematic specificity of a word w_i, as the distance of the embedding vector v_i from the average $\bar{v}_i = \frac{1}{L+1}\sum_{j=1}^{L+1} v_i(j)$ of its components:

$$Polarization(w_i) = \sqrt{\sum_{j=1}^{L+1}(v_i(j) - \bar{v}_i)^2} \tag{14}$$

A high (average) polarization indicates a word with strong topical focus, while near-zero polarization characterizes words that are uniformly distributed (enfolded) across the entire discourse. A high *Polarization* (0.82) indicates a well-defined semantic specificity, while a low polarization (0.31) signals semantic genericity.

4. **Cluster Dispersion**. The mean squared distance of words from their respective cluster centroids is a measure of the semantic cohesion of the cluster:

$$Dispersion(C_k) = \frac{1}{|C_k|}\sum_{v_i \in C_k} \|v_i - \mu_k\|^2 \tag{15}$$

and μ_k is the centroid of the cluster C_k, definited as $\frac{1}{|C_k|}\sum v_i$, and $\|v_i - \mu_k\|^2$ is the euclidean squared distance. The significant shift in mean cluster dispersion from 0.012 to 0.045 serves as a diagnostic tool for structural decay. In the *Original Consecutive Corpus*, terms gravitate toward their respective nuclei with high precision, forming dense and coherent logical basins. In the *Randomly Permuted Corpus*, the disruption of sequential paths forces terms into erratic trajectories. Even when a term maintains a primary association with a cluster, its distance from the centroid increases, signaling a loss of semantic solidarity.

5. **Corpus Cohesion** measures the logical cohesion of the corpus $\mathcal{C}$:

$$Cohesion = \frac{1}{|W_{rel}|}\sum_{w_i \in W_{rel}} \frac{\frac{1}{L}\sigma^2_{word}(i)}{\sigma^2_{max}} \cdot (1 - v_i(L+1)) \tag{16}$$

where:

$$\sigma^2_{word}(w_i) = \sum_{j=1}^{L+1}(v_i(j) - \bar{v}_i)^2 \tag{17}$$

and where $\bar{v}_i$ is the average value of the components of the non-residual clusters; W_{rel} are the relevant words of the corpus; $(1 - v_i(L+1))$ is the relevance of v, that is, the total of non-residual percentage of v; and σ^2_{max} is the theoretical maximum of the variance (when only one component is 1 and the others are 0).

A value of $Cohesion > 1$ signifies that the document possesses a non-stochastic structural architecture. The observed value of $5,66$ of Bohm's discourse confirms that it is characterized by a high degree of *logical enfoldment*, where the meaning of individual terms is strictly dependent on their sequential context within the 300-proposition flow.

The decrease of σ^2_{word} toward zero, under random permutations of the text, provides empirical evidence that salience is a product of the sequential order and the logical enfoldment of the propositions.

The comparisons between the considered corpus and its randomized version are summarized in the following metrics.

 doi:10.20944/preprints202602.0581.v1

Table 14. Comparison of DEA performance metrics between the Original Consecutive Corpus (OCC) and the Randomly Permuted Corpus (RPC). The dramatic increase in the residue mass and the collapse of polarization confirm the path-dependency of the semantic structure.

Metric	Original (OCC)	Permuted (RPC)
Sparsity	0.12	0.48
Marginality	23.8%	59.2%
Polarization	0.82	0.31
Mean Cluster Dispersion	0.012	0.045
Cohesion	5.66	1.00 (Baseline)

The Random Permutation Test reveals that the *implicate order* is not merely a statistical property of the vocabulary, but a dynamic property of the propositional sequence.

The collapse of the clusters manifold into the residual cluster (increasing from 25% to 58%) under permutation proves that semantic salience in Bohm's work is *path-dependent*. We conclude that the DEA effectively captures the "logical friction" of the discourse: in the original sequence, ideas flow with minimal resistance between clusters; in the shuffled sequence, the topological connectivity breaks, leading to a state of semantic entropy.

Other structural indices are based on the entropy of probabilistic distributions [25,26]. The **Corpus Entropy** of the text is given by:

$$H_{tot} = -\frac{1}{|M|} \sum_{i=1}^{|M|} \sum_{j=1}^{|W|} p_{i,j} \log_2 p_{i,j} \tag{18}$$

where $p_{i,j}$ is the probability (an element of stochastic incidence matrix P) expressing the probability that word w_i belongs to the proposition p_j.

For the analyzed corpus ($|M| = 300$, $|W| = 458$), the exact calculation yields:

$$H_{tot} = 3.425 \text{ bits}$$

The **Cluster Entropy** of the cluster C_j for $j = 1, \ldots, L+1$ is the sum of terms $v_j(i) \log_2(v_j(i))$ extended to all the embedding vectors.

$$H_j = -\sum_{i=1}^{|W|} v_j(i) \log_2(v_j(i)) \tag{19}$$

The **Entropic Vector** of the corpus is the vector of the entropies of clusters:

$$H_1, H_2, \ldots, H_{L+1}$$

The entropic jump is the difference between the entropy of the last non-residual cluster and the residual cluster:

$$\Delta_H = H_{L+1} - H_L \tag{20}$$

4.3. The Logarithmic Bound Hypothesis

While contemporary Large Language Models (LLMs) rely on hyper-dimensional manifolds—often exceeding $4,000$ dimensions to process corpora of 10^{13} propositions—the DEA framework suggests a logarithmic collapse of semantic complexity.

 doi:10.20944/preprints202602.0581.v1

According to the $L = \lceil \log_{10}(|M|) \rceil$ principle, the digitized large corpus of human knowledge available to the training of modern chatbots could theoretically be mapped onto manifolds of merely 14 or 15 dimensions. Even if we were to adopt a hyper-refined resolution of $L = 30$ components to capture the most subtle conceptual nuances, the resulting dimensionality would remain orders of magnitude lower than those of the embedding vectors of the current neural architectures.

The selection of the cluster cardinality to the logarithmic bound $L = \lceil \log(|M|) \rceil$ (in our case, $|M| = 300, L = 4$) is supported by the correspondence between the geometric projection and the information-theoretic density of the corpus. We defined the total corpus entropy H_{tot} as the mean entropy per proposition within the stochastic matrix P:

For the specific corpus under analysis ($|W| = 458$), the empirical calculation yields:

$$H_{tot} = 3.425 \text{ bits}$$

Therefore, the value $L = 4$ represents the intrinsic informational "grain" of the text. The relationship between H_{tot} and the number of non-residual clusters L defines the stability of the semantic mapping.

The sharp increase in entropy observed in the fifth component, the **Entropic Jump** in passing to the residual cluster C_5, serves as proof of the adequacy of the value L. Once the 3.425 bits of structural information are mapped into the first four clusters, the fifth cluster is forced to capture only stochastic noise.

Table 15. Entropy across Clusters ($H_{tot} = 3.425$ bits)

Cluster (C_k)	**Order Type**	**Cluster Entropy (H_k)**		**Ratio (H_k / H_{tot})**
C_1	Primary	1.30		0.37
C_2	Secondary	1.45		0.42
C_3	Tertiary	1.90		0.55
C_4	Quaternary	**2.15**		**0.62**
C_5	Residual	**4.20**	Entropic Jump	**1.22**

While the first four clusters maintain an entropy level below this threshold, the fifth cluster exhibits an 'entropic jump' exceeding 4 bits. This overshoot indicates that in C_5 the system has surpassed the total information density of the corpus, transitioning from semantic extraction to meaning fragmentation.

This suggests that meaning does not scale linearly with data, but rather stabilizes into a limited set of invariant basins of attraction. By shifting the computational focus from probabilistic next-token prediction to the extraction of these structural nuclei.

We conclude that the 10^{13} propositions of the modern digital era could be be distilled into a crystalline structure of L dimensions, which represents the true dimension of the corpus's structure.

These findings resonate with Johnson-Lindenstrauss' Lemma [29–31], according to which a set of n points can be projected in a space of $k \approx \log(n)$ dimensions by preserving the Euclidean distances,where ϵ is the error threshold and C is a constant:

$$k \geq \frac{C \cdot \log(n)}{\epsilon^2}$$

5. Conclusions

The fundamental point of departure for this framework is the conceptualization of a text not as a static database, but as a sequential list of propositions. Within this structure, a *word* is perceived as an entity engaged in a "walk" across the propositional landscape. Along its trajectory through linear space, the encounters it makes with other "traveling entities" define its specific identity.

 doi:10.20944/preprints202602.0581.v1

Crucially, this is an **intrinsically collective process**: the path taken by each word is shaped by, and simultaneously shapes, the paths of all others. Semantics, therefore, emerges as the result of two interacting linear orders: the sequential arrangement of words within a single statement, and the sequential flow of propositions within the global text.

However, these are not merely static sequences; they are two linear dimensions upon which a population of **non-linear, interwoven paths** unfolds. In this view, the meaning of a term is not found in the nodes themselves, but in the topology of the intersections formed during this collective journey. The identity of a concept is the history of its encounters within the *holomovement* of the text (a key term in Bohm's analysis).

One of the main motivations of this paper is that: *features of embedding vectors are inside the corpus*. This can be considered obvious. However, what is not obvious is the existence of a purely distributional method for finding the right features that represent the meaning of words in the corpus. The theoretical analysis and the experimental results of the paper seem to confirm this intuition. Maybe, the DEA algorithm, in the present formulation, is not the best answer to the quest underlying the intuition inspiring the research of the paper, but the search for analogous and more powerful methods is not only a technical matter for improving the efficiency of transformers; it is an important step in the advancement of our comprehension of human language and human thought.

The Galois correspondence between the functions *prop* and *word* of a given corpus shows the theoretical strength of embendings and their intrinsic relationship with the corpus from which they are extracted. Even if there are different ways to discover them, they are shaped in the set of relationships they are involved in, which essentially depend on the semantic relations that a very large corpus establishes among propositions and words of a language.

If the logarithmic limit hypothesis is confirmed by the analysis and application to corpora of significant magnitude (10^{14}), and proves successful by scaling current embeddings by several orders of magnitude, this will undoubtedly be the case since the DEA perspective structurally leverages the full cohesion and informational density of the entire text. This globality, or 'wholeness' in the Bohmian sense, is the secret behind such efficiency. It is, therefore, truly remarkable—one might even say 'magical'—that the foundational analysis of DEA has been based upon this very masterpiece by David Bohm.

Finally, there is another important aspect showing the novelty of the DEA embedding vectors. Namely, the feature components are arranged in order of significance. The first components represent the most salient clusters, while the following ones represent the most peculiar characteristics. This order is strongly related to the meaning composition in the construction of phases. Namely, embedding vectors to not simply represent the meaning of words, but meanings in a wide sense: of phrases, propositions, discourses, and concepts. For example, in a modification such as "good policeman" the features that are peculiar in "good" direct a new arrangement of the internal semantic mass of "policeman" by enforcing some aspect of the concept. The normalized Hadamard product of the embedding vectors expresses the semantics of modification. Analogously, the "abstraction" of a concept corresponds to an operation in which the weights of the most peculiar features are reduced, thereby enforcing the principal ones. In other words, the internal organization of features in embedding vectors can be directly related to logical and semantic relations among meanings [32], introducing a new perspective that can adequately represent complex semantic relationships. This transparency allows for the mapping of logical operators—such as modification, complementation, or predicative abstraction—directly onto tensor transformations. In this view, a logical operation is expressed as a formal redistribution of semantic mass across different groups of features, providing a geometric calculus for propositional semantics. Therefore, head-driven attention mechanisms could be more efficiently guided toward the best semantic composition of complex meanings. [33–35].

 doi:10.20944/preprints202602.0581.v1

6. Appendix: Python Code of DEA

```
%Python

import numpy as np
from scipy.sparse import csr_matrix

def compute_dea_lplus1_embeddings(propositions, vocab, clusters_list):
    """
    CONSTRUCTION OF THE DYNAMIC INCIDENCE MATRIX AND L+1 EMBEDDINGS

    Parameters:
    - propositions: List of lists containing word indices for each sentence.
    - vocab: Dictionary mapping words to indices.
    - clusters_list: List of L lists, each containing indices of propositions assigned to that cluster.
    """

    M = len(propositions)
    W = len(vocab)
    L = len(clusters_list)

    # --- STEP 1: COMPUTE LOCAL INCIDENCE (STATIC) ---
    # CREATING THE BASE MATRIX OF WORD-PROPOSITION RELATIONSHIPS
    rows = []
    cols = []
    data = []
    for m_idx, prop in enumerate(propositions):
        for w_idx in prop:
            rows.append(w_idx)
            cols.append(m_idx)
            data.append(1.0)

    A_local = csr_matrix((data, (rows, cols)), shape=(W, M)).toarray()

    # --- STEP 2: APPLY SEQUENTIAL COHERENCE KERNEL (DYNAMIC) ---
    # INCORPORATING THE FLOW OF THOUGHT BY CONVOLVING ADJACENT PROPOSITIONS
    # THIS SECTION CAPTURES THE TOPOLOGICAL ORDER THAT PERMUTATIONS DESTROY
    A_dynamic = np.zeros_like(A_local)
    for m in range(M):
        A_dynamic[:, m] += A_local[:, m]  # Central weight K(0) = 1
        if m > 0:
            A_dynamic[:, m] += 0.5 * A_local[:, m-1]  # Backward link K(-1) = 0.5
        if m < M - 1:
            A_dynamic[:, m] += 0.5 * A_local[:, m+1]  # Forward link K(+1) = 0.5

    # --- STEP 3: DEFINE THE RESIDUAL CLUSTER L+1 ---
    # IDENTIFYING ALL PROPOSITIONS NOT BELONGING TO THE PRIMARY L NUCLEI
    assigned_m = set().union(*clusters_list)
```

 doi:10.20944/preprints202602.0581.v1


```
    residual_m = [m for m in range(M) if m not in assigned_m]

    # COMBINING INTO A FULL SET OF L+1 CLUSTERS
    all_clusters = clusters_list + [residual_m]

    # --- STEP 4: NORMALIZE TRANSITION MATRIX P ---
    # CALCULATING PROBABILITIES ALONG THE SEMANTIC WALK
    row_sums = A_dynamic.sum(axis=1)[:, np.newaxis]
    row_sums[row_sums == 0] = 1.0  # Avoid division by zero for rare terms
    P = A_dynamic / row_sums

    # --- STEP 5: CALCULATE L+1 DIMENSIONAL EMBEDDINGS ---
    # PROJECTING WORD MASS INTO THE L STRUCTURAL CLUSTERS AND THE RESIDUAL SPACE
    embeddings = np.zeros((W, L + 1))
    for j, cluster_indices in enumerate(all_clusters):
        if cluster_indices:
            # The embedding value is the sum of transition probabilities to that cluster
            embeddings[:, j] = P[:, cluster_indices].sum(axis=1)

    return embeddings

# --- STEP 6: HIERARCHICAL CLASSIFICATION LOGIC ---
# CATEGORIZING WORDS BASED ON PEAK DOMINANCE VS. RESIDUAL MAGNITUDE
def classify_entity(v_i, L):
    """
    v_i: vector with L+1 components
    L: number of structural clusters
    """
    residue = v_i[L]
    structural_components = v_i[:L]
    max_idx = np.argmax(structural_components)
    max_val = structural_components[max_idx]

    # CRITERION 1: SALIENT (PILLAR)
    # The term has a clear peak in one of the L clusters
    if all(max_val > structural_components[k] for k in range(L) if k != max_idx):
        if max_val > residue:
            return f"Salient (Cluster {max_idx + 1})"

    # CRITERION 2: BRIDGE (RELEVANT)
    # Low residue but no single dominant peak; mass is distributed
    if residue < (1.0 / L) or all(residue < val for val in structural_components):
        return "Bridge (Relevant)"

    # CRITERION 3: RESIDUAL
    # The term's mass is mainly lost in the unstructured L+1 space
    return "Residual"
```

References

1. Rosenblatt, F., The perceptron: A probabilistic model for information storage and organization in the brain. *Psychol. Rev.*, *65*, 386–408 (1958)
2. Hinton G. E., Implementing semantic networks in parallel hardware. In *Parallel Models of Associative Memory*; Hinton, G. E., Anderson, J.A., eds.; Lawrence Erlbaum Associates., pp. 191–217 (1981), Available online: https:taylorfrancis.com/chapters/edit/10.4324/9781315807997-13/implementing-semantic-networks-parallel-hardware-geoffrey-hinton (accessed on 1 December 2024).
3. Hopfield, J. J., Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. USA* , *79*, 2554–2558 (1982)
4. Rumelhart, D., Hinton, G., Williams, R. *Learning representations by back-propagating errors. Nature* 323, 533–536 (1986)
5. Rumelhart, D. E., McClelland, J. L., (eds.) *Parallel distributed processing: explorations in the microstructure of cognition, vol. 1: foundations*, MIT Press, Cambridge, MA, United States (1986)
6. Mitchell, T., *Machine Learning*, McGraw Hill (1997)

 doi:10.20944/preprints202602.0581.v1

7. Nielsen, M., *Neural Networks and Deep Learning*. Online (2013)
8. Bahdanau, D., Cho, K., Bengio, Y., Neural Machine Translation by Jointly Learning to Align and Translate, *arXiv:1409.0473* (2014)
9. Goodfellow, I., Bengio, Y. Courville A., *Deep Learning*. MIT Press (2016)
10. Kaplan, J. et al., Scaling Laws for Neural Language Models arXiv:2001.08361 (2020)
11. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Language Models are Few-Shot Learners. *NEURIPS*, *33*, 1877–1901, (2020)
12. OpenAI, GPT4-Technical Report, ArXiv: submit/4812508 [cs.CL] 27 Mar. (2023)
13. Mikolov, T., Chen, K., Corrado, G., Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. arXiv preprint arXiv:1301.3781.
14. Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. Advances in Neural Information Processing Systems (NeurIPS), 26, 3111–3119.
15. Pennington, J., Socher, R., Manning, C. D. GloVe: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 1532–1543 (2014)
16. Levy, O., Goldberg, Y., Neural Word Embedding as Implicit Matrix Factorization. Advances in Neural Information Processing Systems (NeurIPS) (2014)
17. Shi, T., Liu, Z., Linking GloVe with word2vec. arXiv preprint arXiv:1411.5595 (2014)
18. Raunak, V., Gupta, V., Metze, F., Effective Dimensionality Reduction for Word Embeddings. Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP) (2019)
19. Mu, J., Viswanath, P., All-but-the-Top: Simple and Effective Postprocessing for Word Representations, *International Conference on Learning Representations*, https://openreview.net/forum?id=HkuGJ3kCb (2018)
20. Al-Anzi, A., AbuZeina, D. (2026). A Comparative Study of Different Dimensionality Reduction Techniques for Compressing Word Embeddings. ACM Transactions on Asian and Low-Resource Language Information Processing.
21. Harris, Z. S. (1954). Distributional structure. *Word*, 10(2-3), 146–162 (1954)
22. Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6), 391–407 (1990)
23. Baroni, M., & Lenci, A., Distributional Memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4), 673–721 (2010)
24. Turney, P. D., & Pantel, P. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37, 141–188 (2010)
25. Shannon, C. E. *A Mathematical Theory of Communication*, Bell System Technical Journal, **1948**
26. Cover, T. M., Thomas, J. A., *Elements of Information Theory*, John Wilwey & Sons, New York **1991**
27. Bohm, D., *Wholeness and the Implicate Order*. Routledge & Kegan Paul (1980)
28. Denecke, K., Erné, M., Wismath, S. L. *Galois connections and applications*, Springer (2004)
29. Johnson, W. B., Lindenstrauss, J., Extensions of Lipschitz mappings into a Hilbert space, In Conference in modern analysis and probability, 26, 89-206, *Contemporary Mathematics, American Mathematical Society* (1984)
30. Indyk, P., & Motwani, R., (Approximate nearest neighbors in high dimensions. *Proceedings of the 30th annual ACM symposium on Theory of computing*, (1998)
31. Dasgupta, S., Gupta, A., An elementary proof of a theorem of Johnson and Lindenstrauss. *Random Structures & Algorithms*, 22(1), 60-65, (2003)
32. Manca, V., Functional Language Logic, *Electronics MDPI*, 14, 3, 460 (2025)
33. Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, 46(1-2), 159–216.
34. Manca, V., Knowledge and Information in Epistemic Dynamics, *Preprint.org MDPI* (2026)
35. Manca, V., Tensor Logic of Embedding Vectors in Neural Networks, *Preprint.org MDPI* (2026)