

Article

Not peer-reviewed version

Graph Hyperembedding and Graph SuperHyperembedding: Extensions via HyperVector and SuperHyperVector Spaces

[Takaaki Fujita](#) *

Posted Date: 25 December 2025

doi: 10.20944/preprints202512.2284.v1

Keywords: hyperstructure; superhyperstructure; vector; HyperVector; SuperHyperVector



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

Graph Hyperembedding and Graph SuperHyperembedding: Extensions via HyperVector and SuperHyperVector Spaces

Takaaki Fujita

Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan; takaaki.fujita060@gmail.com

Abstract

Graph embedding assigns vectors to vertices (or to entire graphs) in a low-dimensional space so that adjacency information and broader structural relationships are reflected in the resulting representations. Hypervector spaces extend the classical vector-space setting by replacing ordinary scalar multiplication with a set-valued scalar action, allowing each scalar–vector pair to produce a nonempty set of vectors while maintaining familiar distributive and associative behavior in an appropriate sense. Building on this idea, (m, n) -SuperhyperVector spaces further generalize the framework by admitting an m -ary scalar superhyperoperation whose outputs belong to an n -level iterated nonempty powerset associated with the underlying abelian group. In this paper, we introduce Graph Hyperembedding and Graph SuperHyperembedding by using HyperVector and SuperHyperVector spaces, and we investigate fundamental properties of these new embedding models.

Keywords: hyperstructure; superhyperstructure; vector; HyperVector; SuperHyperVector

1. Preliminaries

This section summarizes the terminology and notation used in the sequel. Unless explicitly stated otherwise, every set and every structure discussed in this paper is assumed to be finite.

1.1. Classical Structures, Hyperstructures, and n -Superhyperstructures

We briefly recall three nested levels of abstraction: classical structures, hyperstructures, and n -superhyperstructures. Informally, hyperstructures are obtained by applying the powerset construction once to a base set [1–7], whereas n -superhyperstructures use the n^{th} iterated powerset [8,9]. Related concepts within the family of superhyperstructures include SuperHyperGraphs [10–12], chemical hyperstructure [13,14], and SuperHyperAlgebra [15].

Definition 1.1 (Base Set). A *base set* S is the ground collection on which all subsequent (higher-order) constructions are formed:

$$S = \{ x \mid x \text{ is an element of the domain} \}.$$

In particular, every object in $\mathcal{P}(S)$ and in $\mathcal{P}_n(S)$ is ultimately built from elements of S .

Definition 1.2 (Powerset). [16] For a set S , the *powerset* $\mathcal{P}(S)$ is the collection of all subsets of S , including $\emptyset$:

$$\mathcal{P}(S) = \{ A \mid A \subseteq S \}.$$

Definition 1.3 (n -th Powerset). [17] Let H be a nonempty set. The n -th *powerset* of H , denoted $\mathcal{P}_n(H)$, is defined recursively by

$$\mathcal{P}_1(H) = \mathcal{P}(H), \quad \mathcal{P}_{k+1}(H) = \mathcal{P}(\mathcal{P}_k(H)) \quad (k \geq 1).$$

Likewise, the n -th nonempty powerset $\mathcal{P}_n^*(H)$ is given by

$$\mathcal{P}_1^*(H) = \mathcal{P}^*(H), \quad \mathcal{P}_{k+1}^*(H) = \mathcal{P}^*(\mathcal{P}_k^*(H)),$$

where $\mathcal{P}^*(H) = \mathcal{P}(H) \setminus \{\emptyset\}$ denotes the family of all nonempty subsets of H .

To set a precise basis for hyperstructures and superhyperstructures, we next record the core definitions.

Definition 1.4 (Classical Structure). (cf. [17,18]) A *classical structure* consists of a nonempty set H together with one or more *classical operations* satisfying specified axioms.

A *classical operation* is a map

$$\#_0 : H^m \rightarrow H,$$

where $m \geq 1$ and H^m is the m -fold Cartesian product of H . Standard examples are addition and multiplication in groups, rings, and fields.

Definition 1.5 (Hyperoperation). (cf. [19,20]) Let S be a set. A *hyperoperation* on S is a map

$$\circ : S \times S \rightarrow \mathcal{P}(S),$$

so that the combination of two elements produces a subset of S rather than a single element.

Definition 1.6 (Hyperstructure). (cf. [4,17]) A *hyperstructure* is a structure obtained by working on the powerset of a base set. Concretely, it is a pair

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where S is a base set and $\circ$ is an operation acting on elements of $\mathcal{P}(S)$. In this way, operations may be applied to collections of elements, not only to individual elements.

Definition 1.7 (SuperHyperOperations). (cf. [17]) Let H be a nonempty set. Define the iterated powersets by

$$\mathcal{P}^0(H) = H, \quad \mathcal{P}^{k+1}(H) = \mathcal{P}(\mathcal{P}^k(H)) \quad (k \geq 0).$$

A *SuperHyperOperation* of order (m, n) is an m -ary map

$$\circ^{(m,n)} : H^m \rightarrow \mathcal{P}_*^n(H),$$

where $\mathcal{P}_*^n(H)$ denotes the n -th iterated powerset of H , with the convention that it may exclude or include $\emptyset$, depending on the setting:

- If $\emptyset$ is excluded from the codomain, then $\circ^{(m,n)}$ is called a *classical-type (m, n) -SuperHyperOperation*.
- If $\emptyset$ is allowed in the codomain, then $\circ^{(m,n)}$ is called a *Neutrosophic (m, n) -SuperHyperOperation*.

Thus, SuperHyperOperations extend hyperoperations by permitting higher-order (iterated powerset) outputs.

An (m, n) -superhyperstructure extends the hyperstructure viewpoint by introducing an m -ary operation on the hierarchical set $\mathcal{P}_n(S)$, the n^{th} iterated powerset of a finite base set.

Definition 1.8 ((m, n) -Superhyperstructure). [17,21] Let S be a nonempty set and let $\mathcal{P}_n(S)$ be its n^{th} iterated powerset (Definition 1.3). Fix an integer $m \geq 1$. An (m, n) -superhyperstructure is a pair

$$\mathcal{SH}_{m,n} = (\mathcal{P}_n(S), \circ^{(m,n)}),$$

where

$$\circ^{(m,n)} : (\mathcal{P}_n(S))^m \longrightarrow \mathcal{P}_n(S)$$

is an m -ary SuperHyperOperation of order (m, n) . That is, for any $\mathbf{X} = (X_1, \dots, X_m) \in (\mathcal{P}_n(S))^m$, the value $\circ^{(m,n)}(\mathbf{X})$ is a subset of $\mathcal{P}_n(S)$ satisfying whatever axioms (e.g. associativity or distributivity) are required in the intended application.

1.2. HyperVector Space

A *vector space* is a set V equipped with an addition $+$ and a scalar multiplication by a field K , and it satisfies the usual linearity axioms [22,23]. Classical generalizations include fuzzy vector spaces [24,25] and neutrosophic vector spaces [26,27]. Vector spaces are known to have numerous real-world applications and play a central role in fields such as machine learning.

A *hypervector space* extends the vector-space paradigm by replacing the single-valued scalar multiplication with a set-valued scalar action (an external hyperoperation)

$$\circ : K \times V \longrightarrow \mathcal{P}^*(V),$$

so that each pair $(a, x) \in K \times V$ is assigned a nonempty subset $a \circ x \subseteq V$, while distributive and associative behaviors are retained (cf. [28,29]).

Definition 1.9 (Hypervector Space). (cf. [28,29]) Let K be a field and let $(V, +)$ be an abelian group. Write $\mathcal{P}^*(V)$ for the family of all nonempty subsets of V . A *hypervector space* over K is a quadruple $(V, +, \circ, K)$ in which

$$\circ : K \times V \longrightarrow \mathcal{P}^*(V)$$

is an external hyperoperation satisfying, for all $a, b \in K$ and $x, y \in V$, the axioms:

- (H1) $a \circ (x + y) \subseteq (a \circ x) + (a \circ y)$, (right distributivity)
- (H2) $(a + b) \circ x \subseteq (a \circ x) + (b \circ x)$, (left distributivity)
- (H3) $a \circ (b \circ x) = (ab) \circ x$, (associativity)
- (H4) $a \circ (-x) = (-a) \circ x = -(a \circ x)$,
- (H5) $x \in 1 \circ x$.

Here, for subsets $A, B \subseteq V$,

$$A + B = \{u + v \mid u \in A, v \in B\},$$

and the expression $a \circ (b \circ x)$ is interpreted as

$$a \circ (b \circ x) = \bigcup_{y \in b \circ x} (a \circ y).$$

1.3. (m, n) -SuperhyperVector Space

An (m, n) -SuperhyperVector Space extends the hypervector-space framework by allowing an m -ary scalar SuperHyperOperation whose values lie in the n -th iterated (nonempty) powerset of the underlying abelian group [30,31].

Definition 1.10 ((m, n) -SuperhyperVector Space). [30,31] Let K be a field, let $(V, +)$ be an abelian group, and fix integers $m, n \geq 1$. Let $\mathcal{P}_n^*(V)$ denote the family of all nonempty elements of the n -th iterated powerset $\mathcal{P}_n(V)$. An (m, n) -SuperhyperVector Space over K is a quadruple

$$(V, +, \circ^{(m,n)}, K),$$

where

$$\circ^{(m,n)} : K^m \times V \longrightarrow \mathcal{P}_n^*(V)$$

is an m -ary SuperHyperOperation such that, for all $\mathbf{a} = (a_1, \dots, a_m)$, $\mathbf{b} = (b_1, \dots, b_m) \in K^m$ and all $x, y \in V$,

$$(SV1) \quad \circ^{(m,n)}(\mathbf{a}, x + y) \subseteq \circ^{(m,n)}(\mathbf{a}, x) + \circ^{(m,n)}(\mathbf{a}, y),$$

$$(SV2) \quad \circ^{(m,n)}(\mathbf{a} + \mathbf{b}, x) \subseteq \circ^{(m,n)}(\mathbf{a}, x) + \circ^{(m,n)}(\mathbf{b}, x),$$

$$(SV3) \quad \circ^{(m,n)}(\mathbf{a}, \circ^{(m,n)}(\mathbf{b}, x)) = \circ^{(m,n)}(\mathbf{a} \cdot \mathbf{b}, x),$$

$$(SV4) \quad \circ^{(m,n)}(-\mathbf{a}, x) = -\circ^{(m,n)}(\mathbf{a}, x),$$

$$(SV5) \quad x \in \circ^{(m,n)}((1, \dots, 1), x).$$

Here,

$$\mathbf{a} + \mathbf{b} = (a_1 + b_1, \dots, a_m + b_m), \quad \mathbf{a} \cdot \mathbf{b} = (a_1 b_1, \dots, a_m b_m),$$

and for subsets $A, B \subseteq V$,

$$A + B = \{u + v \mid u \in A, v \in B\}.$$

For reference, a comparison of Vector Space, HyperVector Space, and (m, n) -SuperHyperVector Space is provided in Table 1.

Table 1. Concise comparison: Vector Space vs. HyperVector Space vs. (m, n) -SuperHyperVector Space.

Category	Vector Space	HyperVector Space	(m, n) -SuperHyperVector Space
Underlying additive structure	Abelian group $(V, +)$	Abelian group $(V, +)$	Abelian group $(V, +)$
Scalars	Field K	Field K	Field K (with m -tuples of scalars)
Scalar action (type)	$K \times V \rightarrow V$	$K \times V \rightarrow \mathcal{P}^*(V)$	$K^m \times V \rightarrow \mathcal{P}_n^*(V)$
Output per scalar-vector input	Single vector	Nonempty subset of V	Nonempty element of $\mathcal{P}_n(V)$ (nested set, depth n)
Distributivity (typical form)	Equalities	Inclusions (set-valued)	Inclusions (set-valued, m -ary)
Associativity (typical form)	$a(bx) = (ab)x$	$a \circ (b \circ x) = (ab) \circ x$	$\circ^{(m,n)}(\mathbf{a}, \circ^{(m,n)}(\mathbf{b}, x)) = \circ^{(m,n)}(\mathbf{a} \cdot \mathbf{b}, x)$
Identity behavior	$1 \cdot x = x$	$x \in 1 \circ x$	$x \in \circ^{(m,n)}((1, \dots, 1), x)$
Motivation	Linear representation	Multi-valued/uncertainty-aware scalar action	Hierarchical multi-valued scalar action (iterated power-set level)

1.4. Graph Embedding

Graph embedding represents vertices (or entire graphs) by vectors in a relatively low-dimensional space, with the aim of retaining adjacency information and broader structural similarity. Such representations enable efficient downstream tasks such as prediction and retrieval [32–35].

Definition 1.11 (Graph embedding in a vector space). [34,36,37] Let $G = (V, E, w)$ be a (possibly weighted) simple graph with vertex set $V = \{v_1, \dots, v_n\}$, edge set $E \subseteq \binom{V}{2}$, and weight function $w : E \rightarrow \mathbb{R}_{>0}$. Fix an integer $d \geq 1$.

A d -dimensional graph embedding consists of a map

$$\varphi : V \rightarrow \mathbb{R}^d, \quad \text{equivalently a matrix } Z \in \mathbb{R}^{n \times d} \text{ with } Z_{i,:} = \varphi(v_i),$$

together with

$$s : V \times V \rightarrow \mathbb{R} \quad (\text{a chosen graph proximity/relation}),$$

and

$$\text{sim} : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R} \quad (\text{a chosen similarity in the embedding space}),$$

such that, for the vertex pairs of interest (u, v) , the similarity score $\text{sim}(\varphi(u), \varphi(v))$ provides an approximation to $s(u, v)$ (or at least preserves its ranking).

A standard way to express this requirement is to select φ by solving an optimization problem of the form

$$\min_{\varphi: V \rightarrow \mathbb{R}^d} \sum_{(u,v) \in \Omega} L(s(u, v), \text{sim}(\varphi(u), \varphi(v))) + \lambda \mathcal{R}(\varphi),$$

where $\Omega \subseteq V \times V$ is the set of training/evaluation pairs, L is a loss function, $\mathcal{R}$ is a regularizer, and $\lambda \geq 0$.

2. Results

This section presents the results.

2.1. Graph Hyperembedding

Graph Hyperembedding maps vertices to nonempty sets of hypervectors in a hypervector space, preserving graph proximities via aggregated similarity.

Definition 2.1 (Aggregation operator). An *aggregation operator* is a map

$$\text{Agg} : \mathcal{P}^*(\mathbb{R}) \longrightarrow \mathbb{R}$$

satisfying the singleton normalization:

$$\text{Agg}(\{r\}) = r \quad (\forall r \in \mathbb{R}).$$

Typical examples include $\text{Agg}(X) = \max X$, $\text{Agg}(X) = \min X$, or $\text{Agg}(X) = \frac{1}{|X|} \sum_{x \in X} x$ for finite X .

Definition 2.2 (Induced similarity between nonempty subsets). Let $(H, +, \circ, K)$ be a hypervector space and let

$$\text{sim} : H \times H \rightarrow \mathbb{R}$$

be a prescribed similarity function on (hyper)vectors. Let Agg be an aggregation operator (Definition 2.1). For $A, B \in \mathcal{P}^*(H)$, define the induced similarity

$$\text{SIM}(A, B) = \text{Agg}\left(\{\text{sim}(x, y) \mid x \in A, y \in B\}\right).$$

Definition 2.3 (Graph hyperembedding in a hypervector space). Let $G = (\mathcal{V}, \mathcal{E}, w)$ be a (possibly weighted) simple graph with $\mathcal{V} = \{v_1, \dots, v_n\}$ and $\mathcal{E} \subseteq \binom{\mathcal{V}}{2}$. Fix a hypervector space $(H, +, \circ, K)$. Fix a graph proximity function $s : \mathcal{V} \times \mathcal{V} \rightarrow \mathbb{R}$, a similarity $\text{sim} : H \times H \rightarrow \mathbb{R}$, and an aggregation operator Agg .

A *graph hyperembedding* of G into $(H, +, \circ, K)$ is a set-valued map

$$\Phi : \mathcal{V} \longrightarrow \mathcal{P}^*(H),$$

together with the induced similarity SIM from Definition 2.2, such that $\text{SIM}(\Phi(u), \Phi(v))$ approximates (or preserves the ordering of) $s(u, v)$ for relevant pairs (u, v) .

As in ordinary graph embedding, one common specification is to define Φ as a minimizer of

$$\min_{\Phi : \mathcal{V} \rightarrow \mathcal{P}^*(H)} \sum_{(u,v) \in \Omega} L\left(s(u, v), \text{SIM}(\Phi(u), \Phi(v))\right) + \lambda \mathcal{R}(\Phi),$$

where $\Omega \subseteq \mathcal{V} \times \mathcal{V}$, L is a loss function, $\mathcal{R}$ is a regularizer, and $\lambda \geq 0$.

Remark 2.4 (Deterministic (singleton) case). If $\Phi(v)$ is a singleton for every $v \in \mathcal{V}$, i.e., $\Phi(v) = \{h_v\}$ with $h_v \in H$, then

$$\text{SIM}(\Phi(u), \Phi(v)) = \text{Agg}(\{\text{sim}(h_u, h_v)\}) = \text{sim}(h_u, h_v)$$

by Definition 2.1. Thus the hyperembedding reduces to an ordinary point embedding into H .

Example 2.5 (A concrete graph hyperembedding with set-valued vertex representations). Let $G = (V, E)$ be the path on three vertices

$$V = \{v_1, v_2, v_3\}, \quad E = \{\{v_1, v_2\}, \{v_2, v_3\}\}.$$

(Target hypervector space.) Let $K = \mathbb{R}$ and let $H = \mathbb{R}^2$ with the usual vector addition $+$. Define an external hyperoperation $\circ : K \times H \rightarrow \mathcal{P}^*(H)$ by

$$a \circ x = \{ax\} \quad (a \in \mathbb{R}, x \in \mathbb{R}^2).$$

Then $(H, +, \circ, K)$ is a hypervector space.

(Proximity, similarity, and aggregation.) Define the graph proximity

$$s(u, v) = \begin{cases} 1, & \{u, v\} \in E, \\ 0, & \{u, v\} \notin E, \end{cases}$$

and define a similarity on H by the negative squared Euclidean distance:

$$\text{sim}(x, y) = -\|x - y\|_2^2.$$

Choose the aggregation operator $\text{Agg} : \mathcal{P}^*(\mathbb{R}) \rightarrow \mathbb{R}$ as

$$\text{Agg}(X) = \max X.$$

Hence the induced similarity for nonempty sets $A, B \subseteq H$ is

$$\text{SIM}(A, B) = \max\{\text{sim}(x, y) \mid x \in A, y \in B\}.$$

(Set-valued embedding.) Define $\Phi : V \rightarrow \mathcal{P}^*(H)$ by

$$\Phi(v_1) = \{(0, 0), (0.1, 0)\}, \quad \Phi(v_2) = \{(1, 0), (1.1, 0)\}, \quad \Phi(v_3) = \{(2, 0), (2.1, 0)\}.$$

This Φ is a graph hyperembedding because adjacent vertices obtain higher induced similarity than the non-adjacent pair, as shown by the explicit computations below.

(Verification by direct calculation.) For $\{v_1, v_2\} \in E$, compute all four squared distances:

$$\begin{aligned} \|(0, 0) - (1, 0)\|_2^2 &= 1, & \|(0, 0) - (1.1, 0)\|_2^2 &= 1.21, \\ \|(0.1, 0) - (1, 0)\|_2^2 &= 0.81, & \|(0.1, 0) - (1.1, 0)\|_2^2 &= 1. \end{aligned}$$

Thus the corresponding similarities are

$$-1, -1.21, -0.81, -1,$$

and therefore

$$\text{SIM}(\Phi(v_1), \Phi(v_2)) = \max\{-1, -1.21, -0.81, -1\} = -0.81.$$

Similarly, for $\{v_2, v_3\} \in E$ we obtain

$$\text{SIM}(\Phi(v_2), \Phi(v_3)) = -0.81$$

(by the same distance pattern shifted by $+1$ in the first coordinate).

For the non-edge pair $\{v_1, v_3\} \notin E$, compute:

$$\begin{aligned} \|(0, 0) - (2, 0)\|_2^2 &= 4, & \|(0, 0) - (2.1, 0)\|_2^2 &= 4.41, \\ \|(0.1, 0) - (2, 0)\|_2^2 &= 3.61, & \|(0.1, 0) - (2.1, 0)\|_2^2 &= 4, \end{aligned}$$

so the similarities are

$$-4, -4.41, -3.61, -4,$$

and hence

$$\text{SIM}(\Phi(v_1), \Phi(v_3)) = \max\{-4, -4.41, -3.61, -4\} = -3.61.$$

Therefore,

$$\text{SIM}(\Phi(v_1), \Phi(v_2)) = \text{SIM}(\Phi(v_2), \Phi(v_3)) = -0.81 > -3.61 = \text{SIM}(\Phi(v_1), \Phi(v_3)),$$

which matches the intended proximity ordering $s(v_1, v_2) = s(v_2, v_3) = 1 > 0 = s(v_1, v_3)$.

Theorem 2.6 (Graph hyperembedding generalizes graph embedding). *Let $(W, +, \cdot, K)$ be a (classical) vector space over a field K . Define a hyperoperation $\circ : K \times W \rightarrow \mathcal{P}^*(W)$ by*

$$a \circ x = \{a \cdot x\} \quad (a \in K, x \in W).$$

Then $(W, +, \circ, K)$ is a hypervector space.

Moreover, let $G = (\mathcal{V}, \mathcal{E}, w)$ be a graph, let $\varphi : \mathcal{V} \rightarrow W$ be a (classical) graph embedding, and let $s : \mathcal{V} \times \mathcal{V} \rightarrow \mathbb{R}$ and $\text{sim} : W \times W \rightarrow \mathbb{R}$ be the proximity and similarity used for φ . Choose any aggregation operator Agg . Define the singleton lift

$$\Phi : \mathcal{V} \rightarrow \mathcal{P}^*(W), \quad \Phi(v) = \{\varphi(v)\}.$$

Then Φ is a graph hyperembedding of G into $(W, +, \circ, K)$ in the sense of Definition 2.3, and it reproduces the same pairwise similarities:

$$\text{SIM}(\Phi(u), \Phi(v)) = \text{sim}(\varphi(u), \varphi(v)) \quad (\forall u, v \in \mathcal{V}).$$

In particular, Graph Hyperembedding is a genuine generalization of Graph Embedding.

Proof. Step 1 (Hypervector space axioms). We verify the axioms (H1)–(H5) of the Hypervector Space definition for $(W, +, \circ, K)$ with $a \circ x = \{a \cdot x\}$. Recall that for subsets $A, B \subseteq W$,

$$A + B = \{u + v \mid u \in A, v \in B\}.$$

(H1) Right distributivity:

$$a \circ (x + y) = \{a \cdot (x + y)\} = \{a \cdot x + a \cdot y\} \subseteq \{a \cdot x\} + \{a \cdot y\} = (a \circ x) + (a \circ y).$$

(H2) Left distributivity:

$$(a + b) \circ x = \{(a + b) \cdot x\} = \{a \cdot x + b \cdot x\} \subseteq \{a \cdot x\} + \{b \cdot x\} = (a \circ x) + (b \circ x).$$

(H3) Associativity: First note $b \circ x = \{b \cdot x\}$. Hence, using the standard interpretation

$$a \circ (b \circ x) = \bigcup_{y \in b \circ x} (a \circ y),$$

we compute

$$a \circ (b \circ x) = \bigcup_{y \in \{b \cdot x\}} \{a \cdot y\} = \{a \cdot (b \cdot x)\} = \{(ab) \cdot x\} = (ab) \circ x.$$

(H4) Compatibility with additive inverses:

$$a \circ (-x) = \{a \cdot (-x)\} = \{-(a \cdot x)\} = -(\{a \cdot x\}) = -(a \circ x),$$

and similarly

$$(-a) \circ x = \{(-a) \cdot x\} = \{-(a \cdot x)\} = -(a \circ x).$$

(H5) Unit condition:

$$1 \circ x = \{1 \cdot x\} = \{x\},$$

so $x \in 1 \circ x$.

Therefore $(W, +, \circ, K)$ is a hypervector space.

Step 2 (Reduction of similarities). Let $\Phi(v) = \{\varphi(v)\}$. By Definition 2.2,

$$\text{SIM}(\Phi(u), \Phi(v)) = \text{Agg}(\{\text{sim}(x, y) \mid x \in \{\varphi(u)\}, y \in \{\varphi(v)\}\}) = \text{Agg}(\{\text{sim}(\varphi(u), \varphi(v))\}).$$

By the singleton normalization in Definition 2.1,

$$\text{Agg}(\{\text{sim}(\varphi(u), \varphi(v))\}) = \text{sim}(\varphi(u), \varphi(v)).$$

Hence $\text{SIM}(\Phi(u), \Phi(v)) = \text{sim}(\varphi(u), \varphi(v))$ for all u, v , so Φ satisfies the same proximity-preservation requirement as φ . Thus Φ is a graph hyperembedding, and Graph Hyperembedding generalizes Graph Embedding. $\square$

2.2. Graph SuperHyperembedding

Graph SuperHyperembedding maps graph vertices to n -th iterated nonempty powerset elements in an (m, n) -SuperhyperVector Space, preserving proximities.

Definition 2.7 (Lifted Minkowski sum on $\mathcal{P}_k^*(U)$). Let $(U, +)$ be an abelian group. Define binary operations $\oplus_k$ on $\mathcal{P}_k^*(U)$ recursively as follows.

For $k = 0$, set $\oplus_0 := +$ on U .

For $k \geq 0$ and $X, Y \in \mathcal{P}_{k+1}^*(U) = \mathcal{P}^*(\mathcal{P}_k^*(U))$, define

$$X \oplus_{k+1} Y = \{x \oplus_k y \mid x \in X, y \in Y\}.$$

In particular, for $k = 0$ and nonempty subsets $A, B \subseteq U$,

$$A \oplus_1 B = \{a + b \mid a \in A, b \in B\},$$

which is the usual Minkowski sum.

Definition 2.8 (Singleton tower operator). Let U be a nonempty set. Define maps $\iota_k : \mathcal{P}_0^*(U) \rightarrow \mathcal{P}_k^*(U)$ by

$$\iota_0(x) = x, \quad \iota_{k+1}(x) = \{\iota_k(x)\} \quad (k \geq 0).$$

More generally, for $X \in \mathcal{P}_j^*(U)$ and $n \geq j$, define $\iota_{j \rightarrow n}(X) \in \mathcal{P}_n^*(U)$ by

$$\iota_{j \rightarrow j}(X) = X, \quad \iota_{j \rightarrow (t+1)}(X) = \{\iota_{j \rightarrow t}(X)\} \quad (t \geq j).$$

Thus $\iota_{1 \rightarrow n}(A) = \{\{\dots \{A\} \dots\}\}$ (with $n - 1$ additional braces) for $A \in \mathcal{P}_1^*(U)$.

Lemma 2.9 (Compatibility of ι_n with lifted sum). Let $(U, +)$ be an abelian group and define $\oplus_k$ as in Definition 2.7. Then for all $n \geq 1$ and all $u, v \in U$,

$$\iota_n(u) \oplus_n \iota_n(v) = \iota_n(u + v).$$

Proof. We prove by induction on n .

Base case $n = 1$:

$$\iota_1(u) \oplus_1 \iota_1(v) = \{u\} \oplus_1 \{v\} = \{u + v\} = \iota_1(u + v).$$

Induction step: assume $\iota_n(u) \oplus_n \iota_n(v) = \iota_n(u + v)$. Then using Definition 2.7,

$$\iota_{n+1}(u) \oplus_{n+1} \iota_{n+1}(v) = \{\iota_n(u)\} \oplus_{n+1} \{\iota_n(v)\} = \{\iota_n(u) \oplus_n \iota_n(v)\}.$$

By the induction hypothesis, $\iota_n(u) \oplus_n \iota_n(v) = \iota_n(u + v)$, hence

$$\iota_{n+1}(u) \oplus_{n+1} \iota_{n+1}(v) = \{\iota_n(u + v)\} = \iota_{n+1}(u + v).$$

□

Definition 2.10 (Aggregation operator). An *aggregation operator* is a map

$$\text{Agg} : \mathcal{P}^*(\mathbb{R}) \rightarrow \mathbb{R}$$

satisfying

$$\text{Agg}(\{r\}) = r \quad (\forall r \in \mathbb{R}).$$

Definition 2.11 (Recursive induced similarity on $\mathcal{P}_n^*(U)$). Let U be a nonempty set and fix a base similarity

$$\text{sim}_0 : U \times U \rightarrow \mathbb{R}.$$

Fix an aggregation operator Agg (Definition 2.10). Define similarities $\text{SIM}_k : \mathcal{P}_k^*(U) \times \mathcal{P}_k^*(U) \rightarrow \mathbb{R}$ recursively by

$$\text{SIM}_0 = \text{sim}_0,$$

and for $k \geq 0$, for $X, Y \in \mathcal{P}_{k+1}^*(U)$,

$$\text{SIM}_{k+1}(X, Y) = \text{Agg}\left(\{\text{SIM}_k(x, y) \mid x \in X, y \in Y\}\right).$$

Lemma 2.12 (Singleton-collapse of the recursive similarity). With SIM_k defined as in Definition 2.11, for any $k \geq 0$ and any $X, Y \in \mathcal{P}_k^*(U)$,

$$\text{SIM}_{k+1}(\{X\}, \{Y\}) = \text{SIM}_k(X, Y).$$

Consequently, for $n \geq 1$ and any $A, B \in \mathcal{P}_1^*(U)$,

$$\text{SIM}_n(\iota_{1 \rightarrow n}(A), \iota_{1 \rightarrow n}(B)) = \text{SIM}_1(A, B).$$

Proof. First claim:

$$\text{SIM}_{k+1}(\{X\}, \{Y\}) = \text{Agg}\left(\{\text{SIM}_k(x, y) \mid x \in \{X\}, y \in \{Y\}\}\right) = \text{Agg}(\{\text{SIM}_k(X, Y)\}) = \text{SIM}_k(X, Y),$$

using the singleton normalization $\text{Agg}(\{r\}) = r$.

Second claim follows by iterating the first claim $n - 1$ times:

$$\text{SIM}_n(\iota_{1 \rightarrow n}(A), \iota_{1 \rightarrow n}(B)) = \text{SIM}_{n-1}(\iota_{1 \rightarrow (n-1)}(A), \iota_{1 \rightarrow (n-1)}(B)) = \cdots = \text{SIM}_1(A, B).$$

□

Definition 2.13 (Graph superhyperembedding in an (m, n) -SuperhyperVector Space). Let $G = (\mathcal{V}, \mathcal{E}, w)$ be a (possibly weighted) simple graph with $\mathcal{V} = \{v_1, \dots, v_N\}$ and $\mathcal{E} \subseteq \binom{\mathcal{V}}{2}$. Fix an (m, n) -SuperhyperVector Space

$$(U, +, \circ^{(m, n)}, K)$$

(in particular $(U, +)$ is an abelian group), and fix a graph proximity function

$$s : \mathcal{V} \times \mathcal{V} \rightarrow \mathbb{R}.$$

Fix also a base similarity $\text{sim}_0 : U \times U \rightarrow \mathbb{R}$ and an aggregation operator Agg . Let SIM_n be the induced similarity on $\mathcal{P}_n^*(U)$ from Definition 2.11.

A graph superhyperembedding of G into $(U, +, \circ^{(m,n)}, K)$ is a map

$$\Psi : \mathcal{V} \longrightarrow \mathcal{P}_n^*(U)$$

such that $\text{SIM}_n(\Psi(u), \Psi(v))$ approximates (or preserves the ordering of) $s(u, v)$ for relevant pairs (u, v) .

One common specification is to define Ψ as a minimizer of

$$\min_{\Psi : \mathcal{V} \rightarrow \mathcal{P}_n^*(U)} \sum_{(u,v) \in \Omega} L(s(u, v), \text{SIM}_n(\Psi(u), \Psi(v))) + \lambda \mathcal{R}(\Psi),$$

where $\Omega \subseteq \mathcal{V} \times \mathcal{V}$, L is a loss function, $\mathcal{R}$ a regularizer, and $\lambda \geq 0$.

Example 2.14 (A level- $n = 2$ Graph SuperHyperembedding on a short path). Let $G = (\mathcal{V}, \mathcal{E})$ be the path on three vertices

$$\mathcal{V} = \{v_1, v_2, v_3\}, \quad \mathcal{E} = \{\{v_1, v_2\}, \{v_2, v_3\}\}.$$

Define the proximity

$$s(u, v) = \begin{cases} 1, & \{u, v\} \in \mathcal{E}, \\ 0, & \{u, v\} \notin \mathcal{E}. \end{cases}$$

(Target (m, n) -SuperhyperVector Space.) Let $K = \mathbb{R}$ and let $U = \mathbb{R}^2$ with the usual addition $+$. Fix $(m, n) = (2, 2)$ and define

$$\circ^{(2,2)} : K^2 \times U \rightarrow \mathcal{P}_2^*(U), \quad \circ^{(2,2)}((a_1, a_2), x) = \{\{(a_1 a_2)x\}\} = \iota_2((a_1 a_2)x).$$

This is the canonical singleton-tower lifting of ordinary scalar multiplication; hence the axioms (SV1)–(SV5) follow from the usual vector-space identities exactly as in the singleton reduction arguments used for hypervector spaces.

(Base similarity and aggregation.) Let

$$\text{sim}_0(x, y) = -\|x - y\|_2^2 \quad (x, y \in U),$$

and choose the aggregation operator $\text{Agg}(X) = \max X$. Let SIM_2 be the induced similarity on $\mathcal{P}_2^*(U)$ constructed recursively (Definition 2.11).

(Definition of the superhyperembedding.) Define $\Psi : \mathcal{V} \rightarrow \mathcal{P}_2^*(U)$ by the following nonempty families of nonempty subsets:

$$\Psi(v_1) = \{A_1, A_2\}, \quad \Psi(v_2) = \{B_1, B_2\}, \quad \Psi(v_3) = \{C_1, C_2\},$$

where

$$A_1 = \{(0, 0), (0.2, 0)\}, \quad A_2 = \{(0, 0.1), (0.2, 0.1)\},$$

$$B_1 = \{(1, 0), (1.2, 0)\}, \quad B_2 = \{(1, 0.1), (1.2, 0.1)\},$$

$$C_1 = \{(3, 0), (3.2, 0)\}, \quad C_2 = \{(3, 0.1), (3.2, 0.1)\}.$$

Thus each $\Psi(v_i)$ is an element of $\mathcal{P}_2^*(U) = \mathcal{P}^*(\mathcal{P}^*(U))$.

(Verification: edge pairs have larger induced similarity than the non-edge pair.) First compute $\text{SIM}_1(A_1, B_1)$. We list the four squared distances:

$$\begin{aligned}\|(0,0) - (1,0)\|_2^2 &= 1, & \|(0,0) - (1.2,0)\|_2^2 &= 1.44, \\ \|(0.2,0) - (1,0)\|_2^2 &= 0.64, & \|(0.2,0) - (1.2,0)\|_2^2 &= 1.\end{aligned}$$

Hence the four similarities sim_0 are

$$-1, -1.44, -0.64, -1,$$

so (since $\text{Agg} = \max$)

$$\text{SIM}_1(A_1, B_1) = \max\{-1, -1.44, -0.64, -1\} = -0.64.$$

By the same computation (shifted in the second coordinate),

$$\text{SIM}_1(A_2, B_2) = -0.64.$$

Therefore, using the level-2 recursion $\text{SIM}_2(\{A_1, A_2\}, \{B_1, B_2\}) = \max_{i,j} \text{SIM}_1(A_i, B_j)$, we obtain

$$\text{SIM}_2(\Psi(v_1), \Psi(v_2)) = -0.64.$$

Next, for the edge $\{v_2, v_3\} \in \mathcal{E}$, compute $\text{SIM}_1(B_1, C_1)$. The closest pair is $(1.2, 0) \in B_1$ and $(3, 0) \in C_1$, giving

$$\|(1.2, 0) - (3, 0)\|_2^2 = (1.8)^2 = 3.24 \Rightarrow \text{SIM}_1(B_1, C_1) \geq -3.24.$$

A direct enumeration of all four pairs in $B_1 \times C_1$ shows the maximum similarity is indeed -3.24 , so $\text{SIM}_1(B_1, C_1) = -3.24$, and similarly $\text{SIM}_1(B_2, C_2) = -3.24$. Hence

$$\text{SIM}_2(\Psi(v_2), \Psi(v_3)) = -3.24.$$

Finally, for the non-edge $\{v_1, v_3\} \notin \mathcal{E}$, the closest pair is $(0.2, 0) \in A_1$ and $(3, 0) \in C_1$, giving

$$\|(0.2, 0) - (3, 0)\|_2^2 = (2.8)^2 = 7.84 \Rightarrow \text{SIM}_2(\Psi(v_1), \Psi(v_3)) = -7.84.$$

Consequently,

$$\min\{\text{SIM}_2(\Psi(v_1), \Psi(v_2)), \text{SIM}_2(\Psi(v_2), \Psi(v_3))\} = \min\{-0.64, -3.24\} = -3.24 > -7.84 = \text{SIM}_2(\Psi(v_1), \Psi(v_3)),$$

which matches the intended proximity ordering $1 > 0$ for edges versus the non-edge. Thus Ψ is a concrete Graph SuperHyperembedding of G at level $n = 2$.

Example 2.15 (A level- $n = 3$ Graph SuperHyperembedding on a 4-cycle). Let $G = (\mathcal{V}, \mathcal{E})$ be the cycle C_4 with

$$\mathcal{V} = \{a, b, c, d\}, \quad \mathcal{E} = \{\{a, b\}, \{b, c\}, \{c, d\}, \{d, a\}\}.$$

Define the proximity s by $s(u, v) = 1$ for $\{u, v\} \in \mathcal{E}$ and $s(u, v) = 0$ otherwise.

(Target (m, n) -SuperhyperVector Space.) Let $K = \mathbb{R}$, $U = \mathbb{R}^2$ with usual addition, and fix $(m, n) = (3, 3)$. Define the canonical singleton-tower scalar SuperHyperOperation

$$\circ^{(3,3)} : K^3 \times U \rightarrow \mathcal{P}_3^*(U), \quad \circ^{(3,3)}((\alpha, \beta, \gamma), x) = \iota_3((\alpha\beta\gamma)x).$$

(Base similarity and aggregation.) Let

$$\text{sim}_0(x, y) = -\|x - y\|_2^2, \quad \text{Agg}(X) = \max X,$$

and let SIM_3 be the induced similarity on $\mathcal{P}_3^*(U)$ obtained from Definition 2.11.

(Mode construction in $\mathcal{P}_3^*(U)$.) We first define nonempty subsets of U (level 1 objects):

$$A_1 = \{(0, 0), (0, 0.1)\}, \quad A_2 = \{(0.1, 0), (0.1, 0.1)\},$$

$$B_1 = \{(1, 0), (1, 0.1)\}, \quad B_2 = \{(1.1, 0), (1.1, 0.1)\},$$

$$C_1 = \{(1, 1), (1, 1.1)\}, \quad C_2 = \{(1.1, 1), (1.1, 1.1)\},$$

$$D_1 = \{(0, 1), (0, 1.1)\}, \quad D_2 = \{(0.1, 1), (0.1, 1.1)\}.$$

Next define level 2 objects (nonempty sets of the above subsets):

$$X_a^{(1)} = \{A_1, A_2\}, \quad X_a^{(2)} = \{A_1\},$$

$$X_b^{(1)} = \{B_1, B_2\}, \quad X_b^{(2)} = \{B_1\},$$

$$X_c^{(1)} = \{C_1, C_2\}, \quad X_c^{(2)} = \{C_1\},$$

$$X_d^{(1)} = \{D_1, D_2\}, \quad X_d^{(2)} = \{D_1\}.$$

Finally, define the level 3 embedding $\Psi : \mathcal{V} \rightarrow \mathcal{P}_3^*(U)$ by

$$\Psi(a) = \{X_a^{(1)}, X_a^{(2)}\}, \quad \Psi(b) = \{X_b^{(1)}, X_b^{(2)}\}, \quad \Psi(c) = \{X_c^{(1)}, X_c^{(2)}\}, \quad \Psi(d) = \{X_d^{(1)}, X_d^{(2)}\}.$$

Each $\Psi(\cdot)$ is a nonempty set of level-2 objects, hence an element of $\mathcal{P}_3^*(U)$.

(Verification: an edge similarity dominates a diagonal similarity.) Consider first the edge $\{a, b\} \in \mathcal{E}$. Compute the level-1 induced similarity $\text{SIM}_1(A_2, B_1)$. The closest pair is $(0.1, 0) \in A_2$ and $(1, 0) \in B_1$:

$$\|(0.1, 0) - (1, 0)\|_2^2 = (0.9)^2 = 0.81 \quad \Rightarrow \quad \text{SIM}_1(A_2, B_1) = -0.81.$$

One checks similarly that all other pairs among A_i and B_j yield similarities ≤ -0.81 , so

$$\text{SIM}_2(X_a^{(1)}, X_b^{(1)}) = \max_{i \in \{1, 2\}, j \in \{1, 2\}} \text{SIM}_1(A_i, B_j) = -0.81.$$

Because SIM_3 is the max-aggregation over mode pairs,

$$\text{SIM}_3(\Psi(a), \Psi(b)) = \max_{p \in \{1, 2\}, q \in \{1, 2\}} \text{SIM}_2(X_a^{(p)}, X_b^{(q)}) \geq \text{SIM}_2(X_a^{(1)}, X_b^{(1)}) = -0.81.$$

In fact, the other mode combinations give values ≤ -0.81 , so

$$\text{SIM}_3(\Psi(a), \Psi(b)) = -0.81.$$

Now consider the diagonal non-edge $\{a, c\} \notin \mathcal{E}$. Compute $\text{SIM}_1(A_2, C_1)$ using the closest pair $(0.1, 0.1) \in A_2$ and $(1, 1) \in C_1$:

$$\|(0.1, 0.1) - (1, 1)\|_2^2 = (0.9)^2 + (0.9)^2 = 0.81 + 0.81 = 1.62 \quad \Rightarrow \quad \text{SIM}_1(A_2, C_1) = -1.62.$$

Enumerating the remaining pairs among A_i and C_j gives similarities ≤ -1.62 , hence

$$\text{SIM}_2(X_a^{(1)}, X_c^{(1)}) = -1.62.$$

The other mode combinations yield values ≤ -1.62 (for instance, $\text{SIM}_2(X_a^{(2)}, X_c^{(2)}) = \text{SIM}_1(A_1, C_1) = -1.81$), so

$$\text{SIM}_3(\Psi(a), \Psi(c)) = \max_{p,q \in \{1,2\}} \text{SIM}_2(X_a^{(p)}, X_c^{(q)}) = -1.62.$$

Therefore,

$$\text{SIM}_3(\Psi(a), \Psi(b)) = -0.81 > -1.62 = \text{SIM}_3(\Psi(a), \Psi(c)),$$

which matches $s(a, b) = 1 > 0 = s(a, c)$. By the symmetric construction of the four corners, the same edge-versus-diagonal ordering holds for $\{b, c\}, \{c, d\}, \{d, a\}$ versus the non-edges $\{a, c\}, \{b, d\}$. Hence Ψ is a concrete Graph SuperHyperembedding of C_4 at level $n = 3$.

Theorem 2.16 (Graph SuperHyperembedding generalizes Graph Hyperembedding and Graph Embedding). *Let $G = (\mathcal{V}, \mathcal{E}, w)$ be a graph, $s : \mathcal{V} \times \mathcal{V} \rightarrow \mathbb{R}$ a proximity, $\text{sim}_0 : U \times U \rightarrow \mathbb{R}$ a base similarity, and Agg an aggregation operator.*

(I) (Generalizes Graph Hyperembedding) *Assume $n \geq 1$. Let $\Phi : \mathcal{V} \rightarrow \mathcal{P}_1^*(U) = \mathcal{P}^*(U)$ be a graph hyperembedding in the sense of Definition 2.3 (with induced SIM_1). Define*

$$\Psi(v) = \iota_{1 \rightarrow n}(\Phi(v)) \in \mathcal{P}_n^*(U) \quad (v \in \mathcal{V}).$$

Then Ψ is a graph superhyperembedding (Definition 2.13), and for all $u, v \in \mathcal{V}$,

$$\text{SIM}_n(\Psi(u), \Psi(v)) = \text{SIM}_1(\Phi(u), \Phi(v)).$$

(II) (Generalizes Graph Embedding) *Let $(W, +, \cdot, K)$ be a classical vector space. Fix $\text{sim}_0 : W \times W \rightarrow \mathbb{R}$ and Agg . Let $\varphi : \mathcal{V} \rightarrow W$ be a classical graph embedding (Definition: Graph embedding in a vector space). Define a lifted map*

$$\Phi(v) = \{\varphi(v)\} \in \mathcal{P}_1^*(W), \quad \Psi(v) = \iota_{1 \rightarrow n}(\Phi(v)) \in \mathcal{P}_n^*(W).$$

Then Ψ is a graph superhyperembedding into $\mathcal{P}_n^(W)$, and for all $u, v \in \mathcal{V}$,*

$$\text{SIM}_n(\Psi(u), \Psi(v)) = \text{sim}_0(\varphi(u), \varphi(v)).$$

In particular, Graph SuperHyperembedding strictly contains the usual Graph Embedding as the singleton case.

Proof. (I) Let $n \geq 1$ and define $\Psi(v) = \iota_{1 \rightarrow n}(\Phi(v))$. For any $u, v \in \mathcal{V}$, by Lemma 2.12,

$$\text{SIM}_n(\Psi(u), \Psi(v)) = \text{SIM}_n(\iota_{1 \rightarrow n}(\Phi(u)), \iota_{1 \rightarrow n}(\Phi(v))) = \text{SIM}_1(\Phi(u), \Phi(v)).$$

Therefore the proximity-preservation property for Ψ at level n is identical to that for Φ at level 1. Hence Ψ is a graph superhyperembedding.

(II) Let $\varphi : \mathcal{V} \rightarrow W$ be a classical graph embedding and define $\Phi(v) = \{\varphi(v)\}$ and $\Psi(v) = \iota_{1 \rightarrow n}(\Phi(v))$. First compute SIM_1 on singletons:

$$\text{SIM}_1(\Phi(u), \Phi(v)) = \text{Agg}(\{\text{SIM}_0(x, y) \mid x \in \{\varphi(u)\}, y \in \{\varphi(v)\}\}) = \text{Agg}(\{\text{sim}_0(\varphi(u), \varphi(v))\}) = \text{sim}_0(\varphi(u), \varphi(v)).$$

Next apply Lemma 2.12 to lift from level 1 to level n :

$$\text{SIM}_n(\Psi(u), \Psi(v)) = \text{SIM}_n(\iota_{1 \rightarrow n}(\Phi(u)), \iota_{1 \rightarrow n}(\Phi(v))) = \text{SIM}_1(\Phi(u), \Phi(v)) = \text{sim}_0(\varphi(u), \varphi(v)).$$

Thus the induced similarities in the superhyperembedding match the classical similarities of φ , so Ψ preserves the same proximity information and is a graph superhyperembedding. $\square$

3. Conclusion

In this paper, we introduced two extensions of classical graph embedding by employing Hyper-Vector and SuperHyperVector structures: namely, Graph Hyperembedding and Graph SuperHyperembedding. For reference, we summarize Graph embedding, Graph Hyperembedding, and Graph SuperHyperembedding in a comparative Table 2.

Table 2. Concise comparison: Graph embedding vs. Graph Hyperembedding vs. Graph SuperHyperembedding.

Category	Graph embedding	Graph Hyperembedding	Graph SuperHyperembedding
Target space	$\mathbb{R}^d$	$(U, +, \circ, K)$	$(U, +, \circ^{(m,n)}, K)$
Embedding map	$\varphi : V \rightarrow \mathbb{R}^d$	$\Phi : V \rightarrow \mathcal{P}^*(U)$	$\Psi : V \rightarrow \mathcal{P}_n^*(U)$
Representation	One vector	Nonempty set in U	Nested nonempty set in $\mathcal{P}_n^*(U)$
Similarity	$\text{sim}(\cdot, \cdot)$ on $\mathbb{R}^d$	SIM_1 induced from sim_0 on U	SIM_n induced recursively from sim_0 on U
Objective	$\text{sim}(\varphi(u), \varphi(v)) \approx s(u, v)$	$\text{SIM}_1(\Phi(u), \Phi(v)) \approx s(u, v)$	$\text{SIM}_n(\Psi(u), \Psi(v)) \approx s(u, v)$
Reduction	—	$\Phi(v) = \{x_v\}$ gives classical embedding into U	$\Psi(v) = \iota_n(x_v)$ gives $\text{SIM}_n = \text{sim}_0$
Motivation	Compact encoding	Ambiguity-aware encoding	Hierarchical ambiguity/multi-scale encoding

In future work, we aim to investigate extended frameworks based on Fuzzy Sets[38], Neutrosophic Sets[39,40], Uncertain Sets[41,42], Plithogenic Sets[43,44], and related generalizations.

Funding: No external or organizational funding supported this work.

Data Availability Statement: This paper is purely theoretical and does not involve any empirical data. We welcome future empirical studies that build upon and test the concepts presented here.

Acknowledgments: We wish to thank everyone whose guidance, ideas, and assistance contributed to this research. Our appreciation goes to the readers for their interest and to the scholars whose publications provided the groundwork for this study. We are also indebted to the individuals and institutions that supplied the resources and infrastructure necessary for completing and disseminating this paper. Lastly, we extend our gratitude to all who offered their support in various capacities.

Ethical Approval: As this work is entirely conceptual and involves no human or animal subjects, ethical approval was not required.

Conflicts of Interest: The authors declare no conflicts of interest in connection with this study or its publication.

Research Integrity: The authors affirm that, to the best of their knowledge, this manuscript represents their original research. It has not been previously published in any journal, nor is it currently being considered for publication elsewhere.

Disclaimer on Computational Toolst: No computer-based tools—such as symbolic computation systems, automated theorem provers, or proof assistants (e.g., Mathematica, SageMath, Coq)—were employed in the development, analysis, or verification of the results contained in this paper. All derivations and proofs were conducted manually through analytical methods by the authors.

Use of Artificial Intelligence: I use generative AI and AI-assisted tools for tasks such as English grammar checking, and I do not employ them in any way that violates ethical standards.

Disclaimer on Scope and Accuracy: The theoretical models and concepts proposed in this manuscript have not yet undergone empirical testing or practical deployment. Future work may investigate their utility in applied or experimental contexts. While the authors have taken care to maintain accuracy and provide appropriate citations, inadvertent errors or omissions may remain. Readers are encouraged to consult original references for

confirmation and further study. The authors assert that all mathematical results and justifications included in this work have been carefully reviewed and are believed to be correct. Should any inaccuracies or ambiguities be discovered, the authors welcome constructive feedback and will provide clarification upon request. The conclusions presented are valid only within the specific theoretical framework and assumptions described in the text. Generalizing these results to other mathematical contexts may require further investigation. All opinions and interpretations expressed herein are solely those of the authors and do not necessarily reflect the views of their respective institutions.

References

1. Adebisi, S.A.; Ajuebishi, A.P. The Order Involving the Neutrosophic Hyperstructures, the Construction and Setting up of a Typical Neutrosophic Group. *HyperSoft Set Methods in Engineering* **2025**, *3*, 26–31.
2. Corsini, P.; Leoreanu, V. *Applications of hyperstructure theory*; Vol. 5, Springer Science & Business Media, 2013.
3. Davvaz, B.; Nezhad, A.D.; Nejad, S.M. Algebraic hyperstructure of observable elementary particles including the Higgs boson. *Proceedings of the National Academy of Sciences, India Section A: Physical Sciences* **2020**, *90*, 169–176.
4. Al Tahan, M.; Davvaz, B. Weak chemical hyperstructures associated to electrochemical cells. *Iranian Journal of Mathematical Chemistry* **2018**, *9*, 65–75.
5. Vougiouklis, T. The fundamental relation in hyperrings. The general hyperfield. In Proceedings of the Proc. Fourth Int. Congress on Algebraic Hyperstructures and Applications (AHA 1990), World Scientific. World Scientific, 1991, pp. 203–211.
6. Vougiouklis, T. On some representations of hypergroups. *Annales scientifiques de l'Université de Clermont. Mathématiques* **1990**, *95*, 21–29.
7. Vougiouklis, T. Hypermathematics, Hv-structures, hypernumbers, hypermatrices and Lie-Santilli admissibility. *American Journal of Modern Physics* **2015**, *4*, 38–46.
8. Fujita, T. Toward a Unified Framework for Knot Theory, Hyperknot Theory, and Superhyperknot Theory via Superhyperstructures. *Neutrosophic Knowledge* **2025**, *6*, 55–71.
9. Al-Odhari, A. A Brief Comparative Study on HyperStructure, Super HyperStructure, and n-Super SuperHyperStructure. *Neutrosophic Knowledge* **2025**, *6*, 38–49.
10. Smarandache, F. *Extension of HyperGraph to n-SuperHyperGraph and to Plithogenic n-SuperHyperGraph, and Extension of HyperAlgebra to n-ary (Classical-/Neutro-/Anti-) HyperAlgebra*; Infinite Study, 2020.
11. Ghods, M.; Rostami, Z.; Smarandache, F. Introduction to Neutrosophic Restricted SuperHyperGraphs and Neutrosophic Restricted SuperHyperTrees and several of their properties. *Neutrosophic Sets and Systems* **2022**, *50*, 480–487.
12. Hamidi, M.; Smarandache, F.; Davneshvar, E. Spectrum of superhypergraphs via flows. *Journal of Mathematics* **2022**, *2022*, 9158912.
13. Al-Tahan, M.; Davvaz, B. Chemical hyperstructures for elements with four oxidation states. *Iranian Journal of Mathematical Chemistry* **2022**, *13*, 85–97.
14. Davvaz, B.; Dehghan Nezhad, A.; Benvidi, A. Chemical hyperalgebra: Dismutation reactions. *Match-Communications in Mathematical and Computer Chemistry* **2012**, *67*, 55.
15. Smarandache, F. History of SuperHyperAlgebra and Neutrosophic SuperHyperAlgebra (revisited again). *Neutrosophic Algebraic Structures and Their Applications* **2022**, p. 10.
16. Jech, T. *Set theory: The third millennium edition, revised and expanded*; Springer, 2003.
17. Smarandache, F. Foundation of SuperHyperStructure & Neutrosophic SuperHyperStructure. *Neutrosophic Sets and Systems* **2024**, *63*, 21.
18. Smarandache, F. Introduction to SuperHyperAlgebra and Neutrosophic SuperHyperAlgebra. *Journal of Algebraic Hyperstructures and Logical Algebras* **2022**.
19. Vougioukli, S. Helix hyperoperation in teaching research. *Science & Philosophy* **2020**, *8*, 157–163.
20. Vougioukli, S. Hyperoperations defined on sets of S-helix matrices. *Journal of Algebraic Hyperstructures and Logical Algebras* **2020**, *1*.
21. Smarandache, F. Foundation of revolutionary topologies: An overview, examples, trend analysis, research issues, challenges, and future directions. *Neutrosophic Systems with Applications* **2024**, *13*.
22. Hatip, A.; Olgun, N.; et al. On the Concepts of Two-Fold Fuzzy Vector Spaces and Algebraic Modules. *Journal of Neutrosophic and Fuzzy Systems* **2023**, *7*, 46–52.
23. Şahin, M.; Kargin, A.; Yıldız, İ. Neutrosophic triplet field and neutrosophic triplet vector space based on set valued neutrosophic quadruple number. *Quadruple Neutrosophic Theory And Applications* **2020**, *1*, 52.

24. Sameena, K.; et al. FUZZY MATROIDS FROM FUZZY VECTOR SPACES. *South East Asian Journal of Mathematics and Mathematical Sciences* **2021**, 17, 381–390.
25. Lubczonok, P. Fuzzy vector spaces. *Fuzzy sets and systems* **1990**, 38, 329–343.
26. Agboola, A.; Akinleye, S. Neutrosophic vector spaces. *Neutrosophic Sets and Systems* **2014**, 4, 9–18.
27. Ibrahim, M.A.; Badmus, B.; Akinleye, S.; et al. On refined neutrosophic vector spaces I. *International Journal of Neutrosophic Science* **2020**, 7, 97–109.
28. Motameni, M.; Ameri, R.; Sadeghi, R. Hypermatrix Based on Krasner Hypervector Spaces. 2013.
29. Dehghan, O. An Introduction to NeutroHyperVector Spaces. *Neutrosophic Sets and Systems* **2023**, 58, 21.
30. Fujita, T. HyperVector and SuperHyperVector Spaces with Applications in Machine Learning: Feature, Support, and Relevance Vectors, 2025.
31. Fujita, T. New Concepts Using HyperStructure and n -SuperHyperStructure: Sequence, Equation, Statistics, Number, Vector, and More, 2024.
32. Wang, Q.; Mao, Z.; Wang, B.; Guo, L. Knowledge graph embedding: A survey of approaches and applications. *IEEE transactions on knowledge and data engineering* **2017**, 29, 2724–2743.
33. Ou, M.; Cui, P.; Pei, J.; Zhang, Z.; Zhu, W. Asymmetric Transitivity Preserving Graph Embedding. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* **2016**.
34. Cai, H.; Zheng, V.W.; Chang, K.C.C. A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE transactions on knowledge and data engineering* **2018**, 30, 1616–1637.
35. Wang, Z.; Zhang, J.; Feng, J.; Chen, Z. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the Proceedings of the AAAI conference on artificial intelligence*, 2014, Vol. 28.
36. Goyal, P.; Ferrara, E. Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems* **2018**, 151, 78–94.
37. Xu, M. Understanding graph embedding methods and their applications. *SIAM Review* **2021**, 63, 825–853.
38. Zadeh, L.A. Fuzzy sets. *Information and control* **1965**, 8, 338–353.
39. Broumi, S.; Talea, M.; Bakali, A.; Smarandache, F. Single valued neutrosophic graphs. *Journal of New theory* **2016**, 10, 86–101.
40. Wang, H.; Smarandache, F.; Zhang, Y.; Sunderraman, R. *Single valued neutrosophic sets*; Infinite study, 2010.
41. Fujita, T.; Smarandache, F. A Unified Framework for U -Structures and Functorial Structure: Managing Super, Hyper, SuperHyper, Tree, and Forest Uncertain Over/Under/Off Models. *Neutrosophic Sets and Systems* **2025**, 91, 337–380.
42. Fujita, T.; Smarandache, F. *A Dynamic Survey of Fuzzy, Intuitionistic Fuzzy, Neutrosophic, , and Extensional Sets*; Neutrosophic Science International Association (NSIA), 2025.
43. Smarandache, F.; Martin, N. *Plithogenic n -super hypergraph in novel multi-attribute decision making*; Infinite Study, 2020.
44. Smarandache, F. *Plithogenic set, an extension of crisp, fuzzy, intuitionistic fuzzy, and neutrosophic sets-revisited*; Infinite study, 2018.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.