Meaning Circulates — Generative Semantics via Conceptual Topology

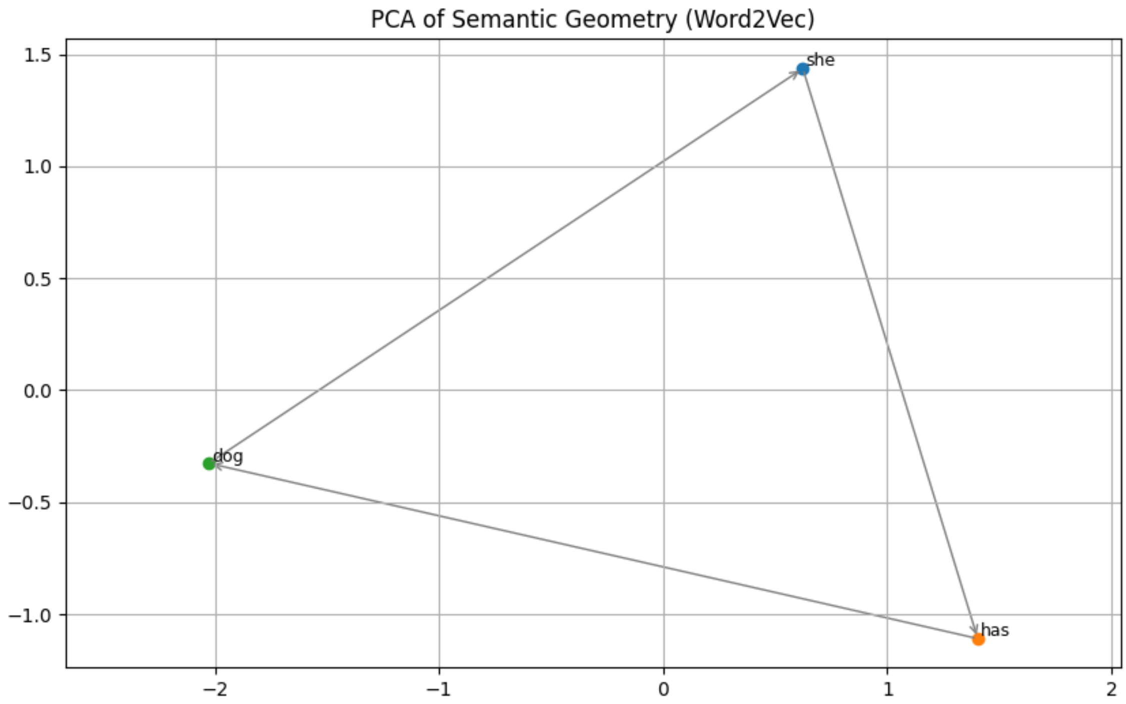
Written by No Name Yet Exist
Contact: Written Below

Introduction

Language exhibits a circulating structure in its semantics. This study explores how sentences are organized and interpreted by combining Principal Component Analysis (PCA) with Conceptual Topology—a category-theoretic framework for modeling semantic flow and structural meaning.

Reference: **Meaning in Motion: A Category-Theoretic Framework for Conceptual Topology** https://zenodo.org/records/15455079

She has a dog



Semantic Cirulation

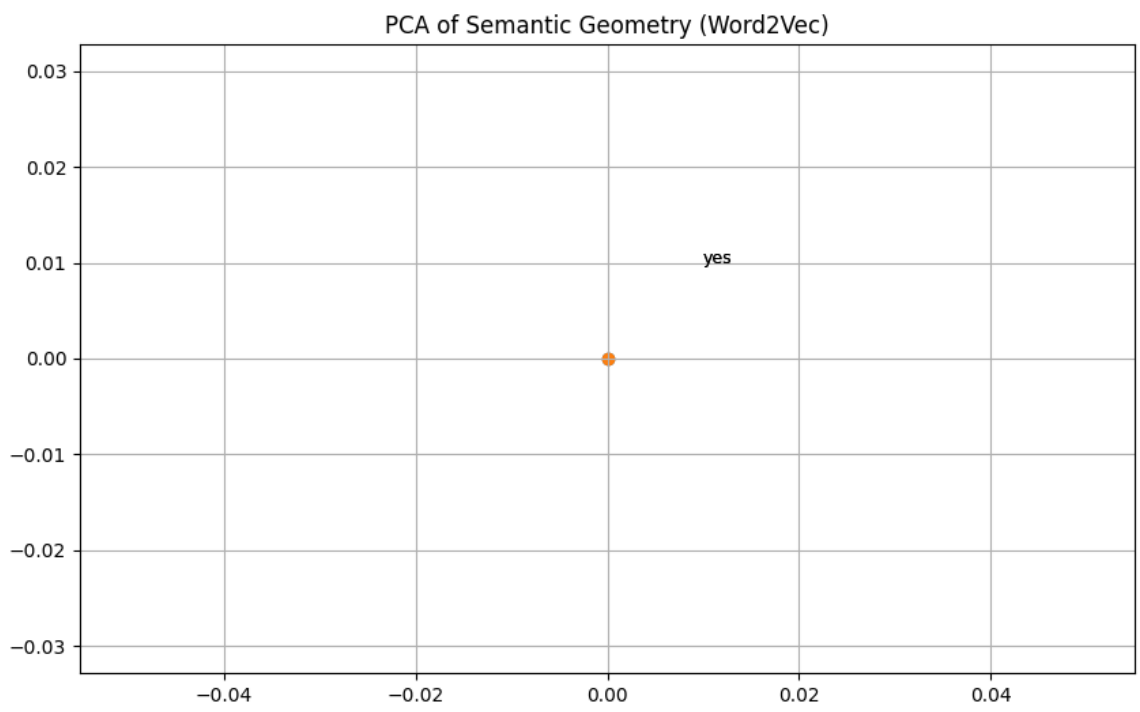
By applying Conceptual Topology, we can visualize how words are combined and structure meaning.

Identity Morphism

The sentence Yes is a minimal linguistic unit—a single-word sentence. It exhibits a semantic identity, represented by the following morphism:

```
Yes is Yes.
f: Yes → Yes
∴id_f
```

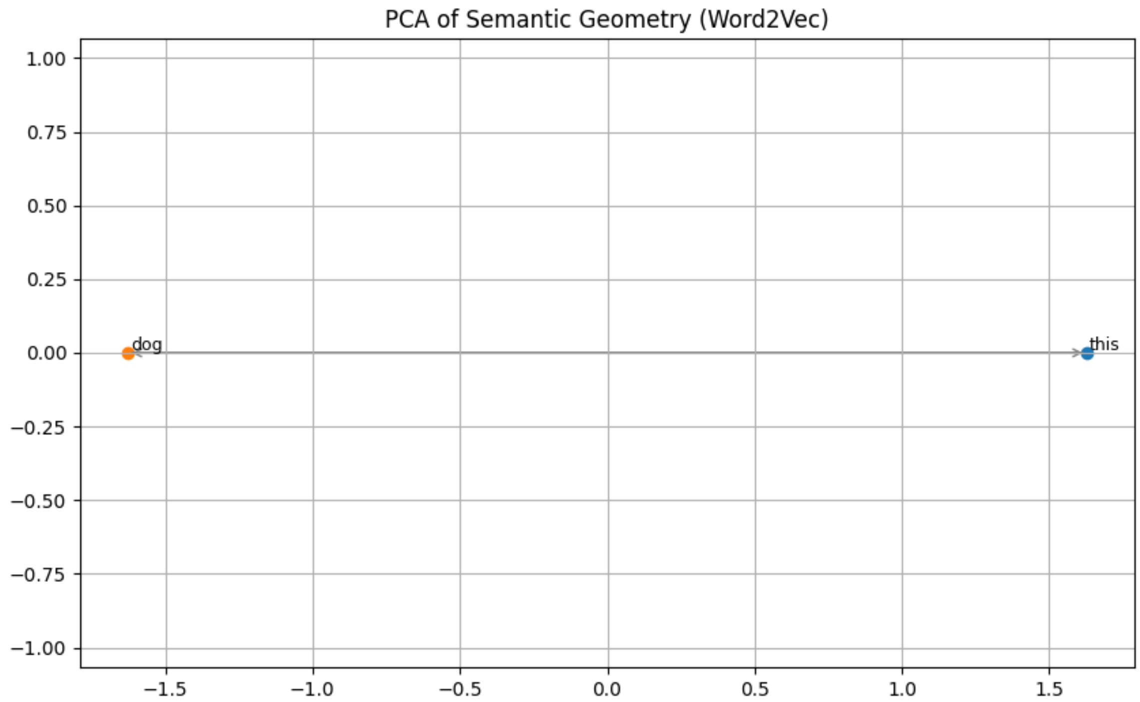
In PCA space, this is observed as a point—an unmoving semantic center.



Homeomorphic Relation

The sentence This is a dog expresses a basic referential mapping. Here, This acts as a morphism from a real-world object, projected onto a conceptual object, dog—establishing a semantic mapping from the observed to the conceptual.

```
f: This → dog  // This is a dog
f-1: dog → this  //The dog is this

g: this → this
∴id_g
```

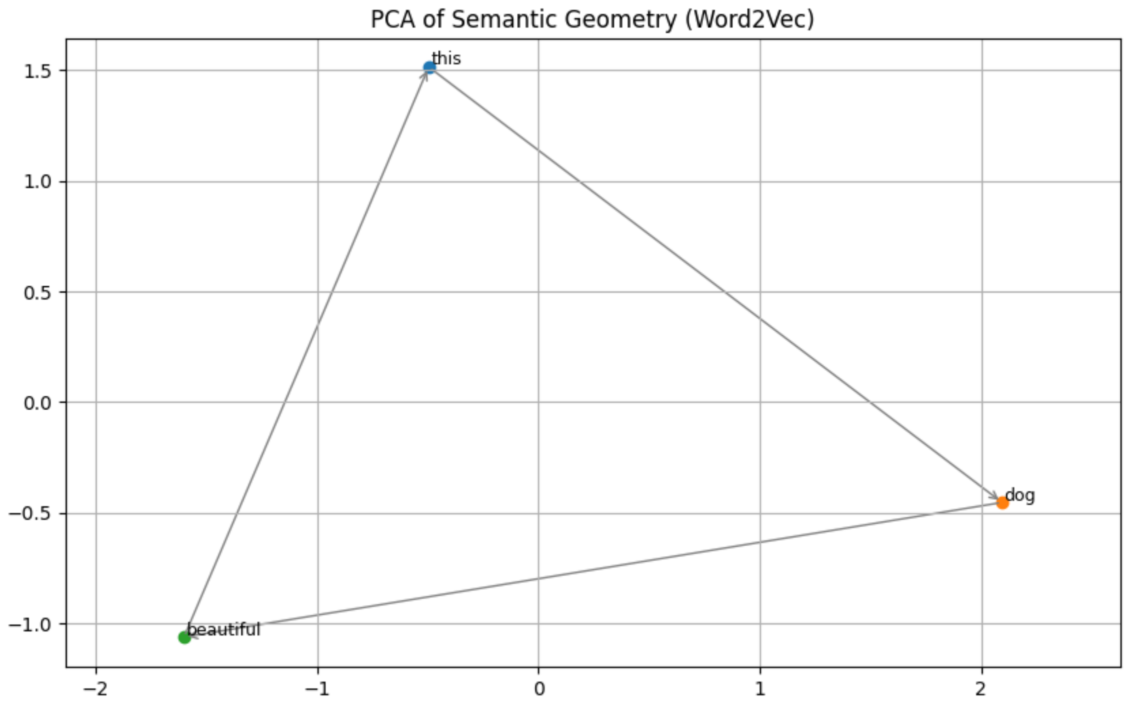


In PCA space, this is typically observed as a linear alignment— a directed semantic path between referent and concept, with identity preserved via mutual reversibility.

Communicative Triangle

This dog is beatiful - This sentence exhibits a triangular communicative structure, where three elements interact to produce a unified semantic coherence.

```
This dog is beatiful

This  - dog
   \    /
  beatiful
```



Here, This and dog form a composite subject, both pointing toward and being qualified by beautiful. The triangle reflects a semantic circuit—a three-point relational topology—where reference (This), substance (dog), and evaluation (beautiful) converge into a stable communicative unit.

This sentence can be reduced to a homeomorphic relation by omitting dog. This reduction is semantically valid, grounded in the bidirectional relation between the referent (this) and the predicate (beautiful), forming a minimal semantic core.

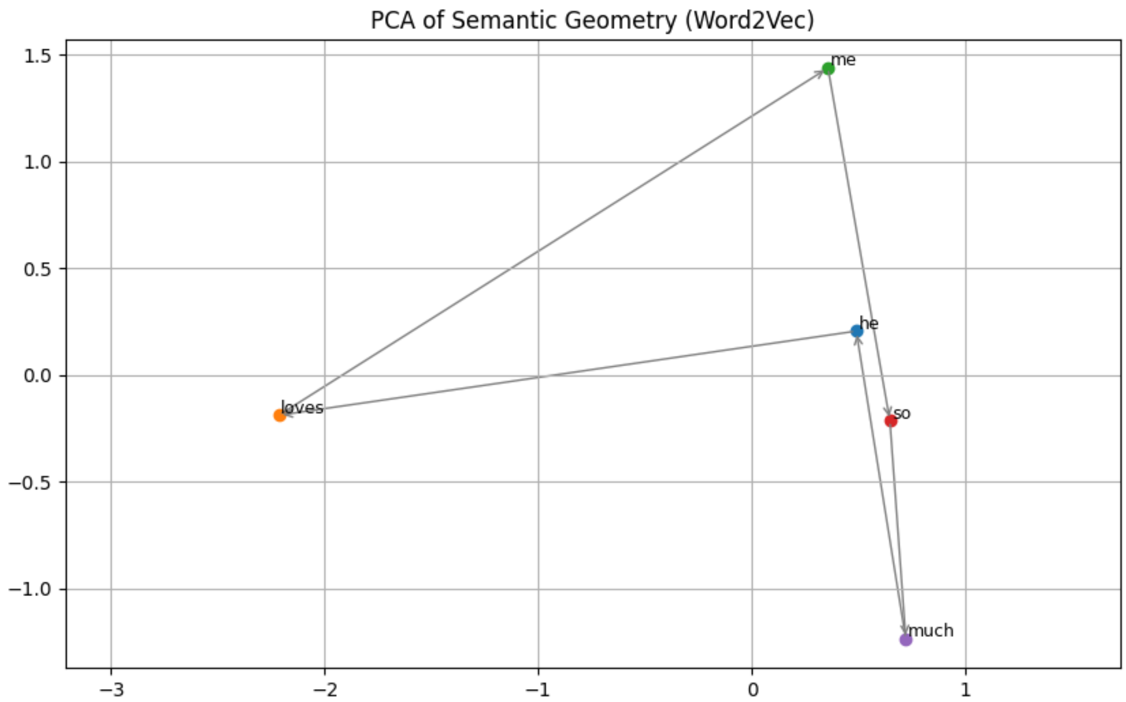
```
this is beatiful

f: this → beautiful
f⁻¹: beautiful → this
g: this → this = f⁻¹ ∘ f
∴ g = id_this
```

Additive Structure

This sentence is composed of a core three-word relational structure, with so much functioning as an additive semantic extension.

He loves me so much.

```
He - loves - me
 \         /
   so much
```



From this PCA visualization, we observe that *love* mediates between *he* and *me*, forming a semantic anchor. The intensifier so much extends this core structure without disrupting its internal coherence.

We can formalize this as:

```
(λx. so much(x)) (he → me | love)
```

This expression represents a triadic relation between *he* and *me*, mediated by *love*, and further intensified by the function *so_much* — an external semantic modifier that amplifies the emotional vector of the relation, without altering its underlying commutative structure.

Remapping Generative Grammar

We reinterpret Chomsky's Generative Grammar framework through the lens of Conceptual Topology, to reveal how syntactic form is sustained through underlying semantic continuity.

Let us consider the sentence:

She has a dog with a long tail.

We begin by transcribing this into a standard syntactic tree:

```
S
├── NP (subject)
│   └── PRP: She
├── VP (predicate)
│   ├── V: has
│   └── NP (object)
│       ├── Det: a
│       ├── N: dog
│       └── PP (prepositional phrase)
│           ├── P: with
│           └── NP
│               ├── Det: a
│               ├── Adj: long
│               └── N: tail
```

From Tree to Topology

In traditional generative grammar, *She* and *has* are placed at parallel syntactic branches. However, from a topological perspective, they form a continuous relation—or more precisely, a semantic morphism—preserving the role of the agent across the conceptual space.

In this mapping: She → has expresses the activation of agency The morphism preserves subject continuity, even as the syntactic structure branches

Thus, we begin to reinterpret syntactic hierarchy as semantic flow, where morphisms carry roles, and identity is preserved not through position but through structural invariance.

This mapping relation may not be immediately evident in English, but becomes more apparent in Spanish:

Tiene un perro con cola larga.
→ *(She) has a dog with a long tail.*

Here, the agent (she) is omitted and absorbed into the verb *tiene*. In this case, the agent is either:

- homeomorphically embedded within the verb (semantic continuity is preserved through structural integration), or
- treated as isomorphic to the verb phrase, assuming the agent role is still distinguishable through context.

The latter interpretation is more likely, since Spanish maintains clear subject-verb agreement, indicating that the agent is not erased, but structurally mirrored.

Functorial Interpretation of Prepositions

In the syntactic structure of the sentence:

She has a dog with a long tail.

The second hierarchical level is a dog, the third is the preposition with, and the fourth is a long tail.

From a topological and categorical perspective, *with* acts not merely as a preposition, but as a Functor that maps from the conceptual object *dog* to its associated attribute tail.

We can formalize this as:

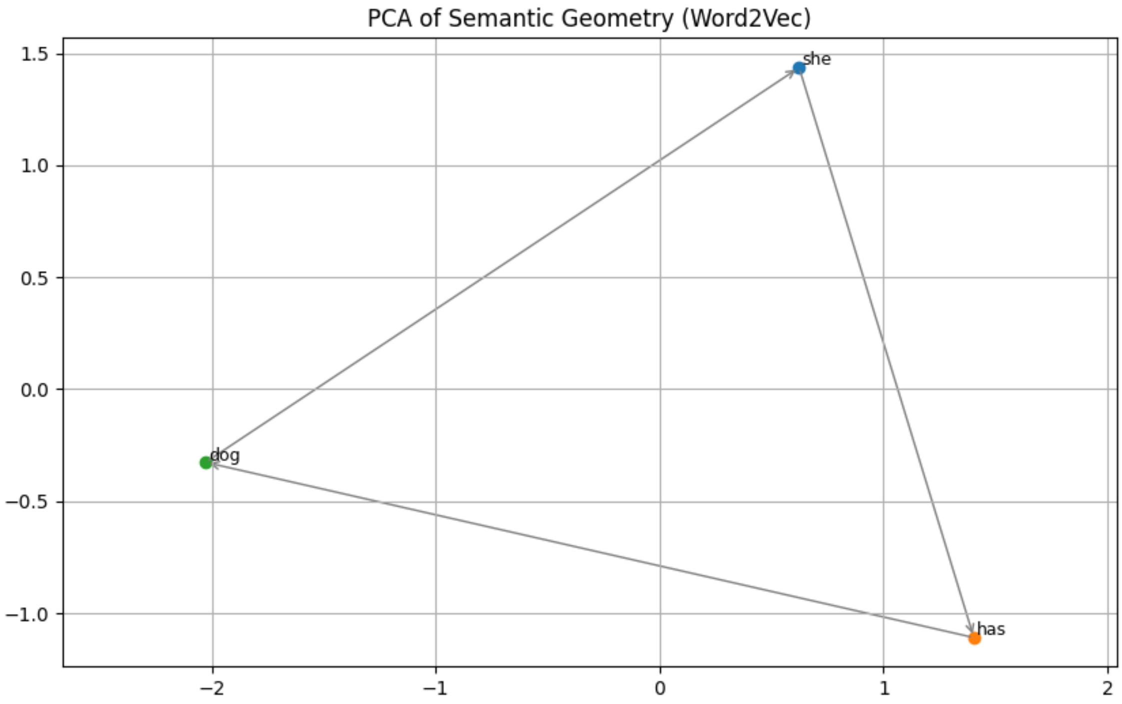
```
F_with: Dog → a long tale
```

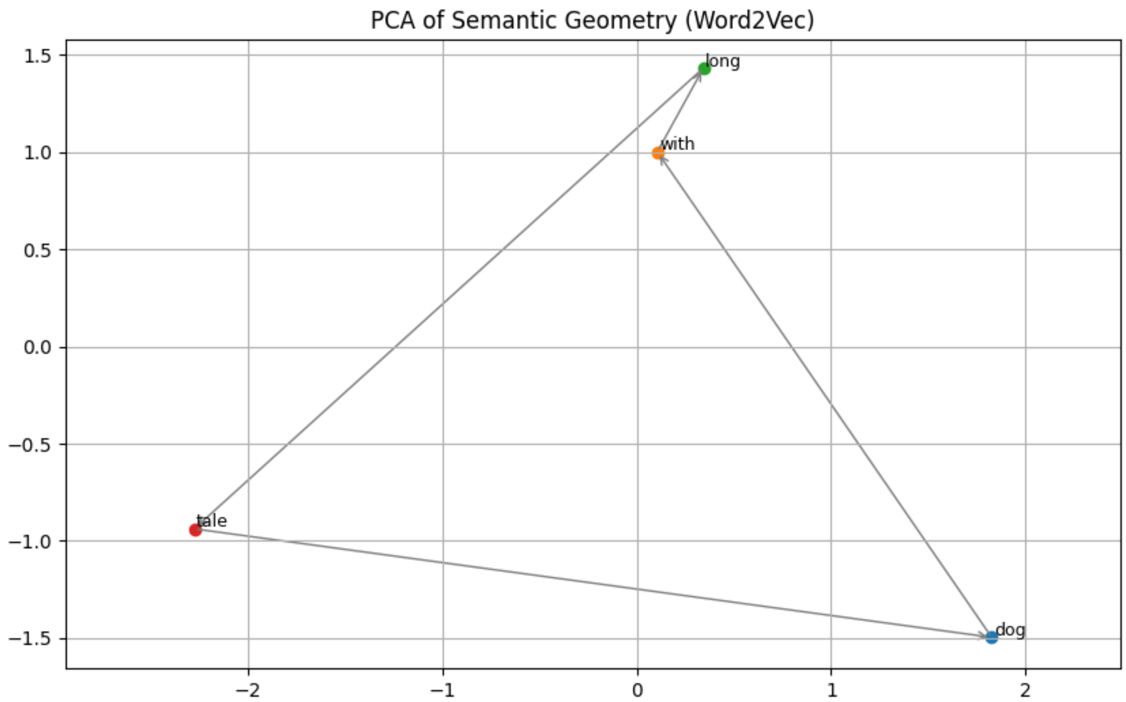
In PCA space, the full sentence She has a dog with a long tail does not appear as a single coherent structure. To clarify its internal semantic relations, we divide the sentence into two conceptual components:

```
(1) she has a dog

(2) dog with a long tail
```

As a result, the first segment—she has a dog—exhibits a communicative triangular structure, while the second segment—with a long tail—reveals how with functions as a Functor mediating the morphic relation between long and tail, and indirectly linking them back to dog.

```
                          tale
                            |
             Functor        |
 She  -  Dog  ----→ with    |
   \    /                 \ |
    Has                    long

Note: The two PCA spaces were artificially merged for this visualization.
```





Remapping the Syntactic Tree into Conceptual Topology Semantic Flow

We reinterpret the syntactic tree structure of the sentence *She has a dog with a long tail* as a Conceptual Topology semantic flow map, where morphisms reflect meaning-preserving relations.

```
C₁:    She → has
         \   /        // she relates to a dog
          dog

    f: She → Dog
              ↓ F_with  // what kind of dog?
C₂:   g: long → tail
```

Note: There is an interpretable structure where g ∈ Fib(F_with) — that is, the modifier long → tail exists within the fiber over the functor F_with.

Functorial Role of Transitive verb

A transitive verb functions not only as a morphism but often as a functor—mapping between conceptual domains.

Consider the verb *give*, which operates as a transformation from an implied subject to a receiver of possession.

In Japanese, this becomes even clearer:

ねえ、それ、ちょうだい。 (Hey, give [that] [to me].)

Although the subject, object and indirect object are omitted, the request enacts a functorial structure in which possession is transferred or requested across conceptual roles.

```
    [それ]     ちょうだい
      |           |
[私に] ← Functor ← [主語]
```

Rewritten in English alignment:

```
   [That]         Give
     |              |
 [Me] ← Functor ← [Subject]
```

The preserved structure is the ownership relationship directed toward the listener.

We may formalize this functorial action as:

```
f: Owner → Owned ∈ Category A, B
F: A → B
```

Where F maps ownership structure from the domain (A) to the codomain (B), reflecting how possession shifts or is preserved in the linguistic operation.

Reconciling Functorial and Morphic Interpretations in Conceptual Topology

In Conceptual Topology, treating give as a functor and love as a morphism may appear inconsistent at first glance. However, this distinction reflects not a contradiction, but a difference in structural hierarchy.

Semantic Function of love: Morphism

In sentences such as He loves me or She has a dog, the verb denotes a direct, unidirectional conceptual relation:

```
f: he → me | love
g: she → dog | have
```

These represent basic morphisms—arrows preserving semantic identity within a single conceptual frame, typically between two arguments in a static structure.

Semantic Function of give: Functor

In contrast to simple binary relations, the verb give implies a semantic transformation— a dynamic restructuring of conceptual roles across participants.

Specifically, it reorganizes a triadic structure:

```
subject → object → indirect object
i.e., giver → thing → receiver
```

This transformation can be captured functorially:

```
f: owner → owned ∈ A, B
F: A → B | owned object

Where:
    A = conceptual domain of giver
    B = domain of receiver
    F maps ownership structures from one domain to another
```

Unlike morphisms, which represent direct semantic connections, this functorial action transfers not only referents but semantic roles. It preserves internal structure while shifting possessive alignment across conceptual domains.

Example: I show you this book

We can model the sentence:

```
I show you this book
I → book → you
```

Morphic Layer

```
f: person → book | sight
This represents a direct semantic action.
the person is semantically linked to the book through the act of
perception.
```

Functorial Layer

```
F: A → B | sight
Where:
    A = conceptual domain of the subject (I)
    B = perceptual domain of the receiver (you)
    F = a functor that maps objects from the subject’s domain
        into the perceptual space of the recipient via sight
```

Not as a Signal of Non-Projectional Morphic Path

We hypothesize that negation functions as a signal of a non-projectional morphic path. In this framework, we define rupture (R) as a morphic discontinuity—where semantic projection breaks.

It was not correct that I told you before.

Here, the negation not ruptures the semantic path between *It* and *correct*.

```
   R
It - correct
 |     | R
What I told you you before
```

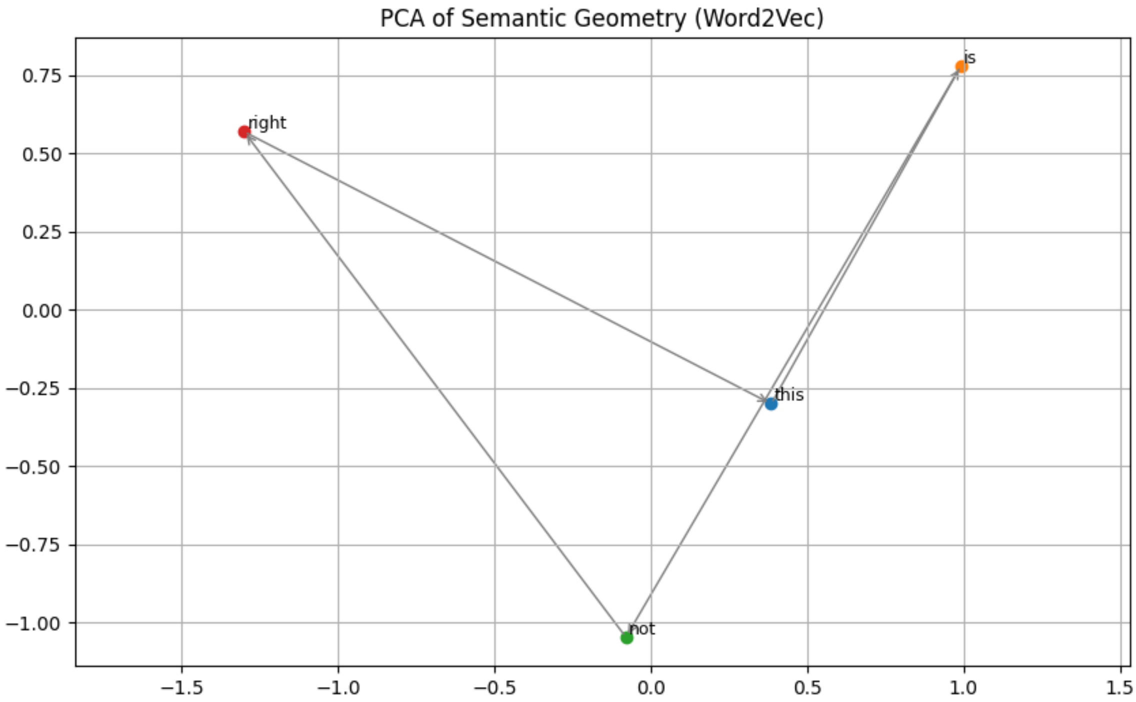
We can express this with a simplified schematic:

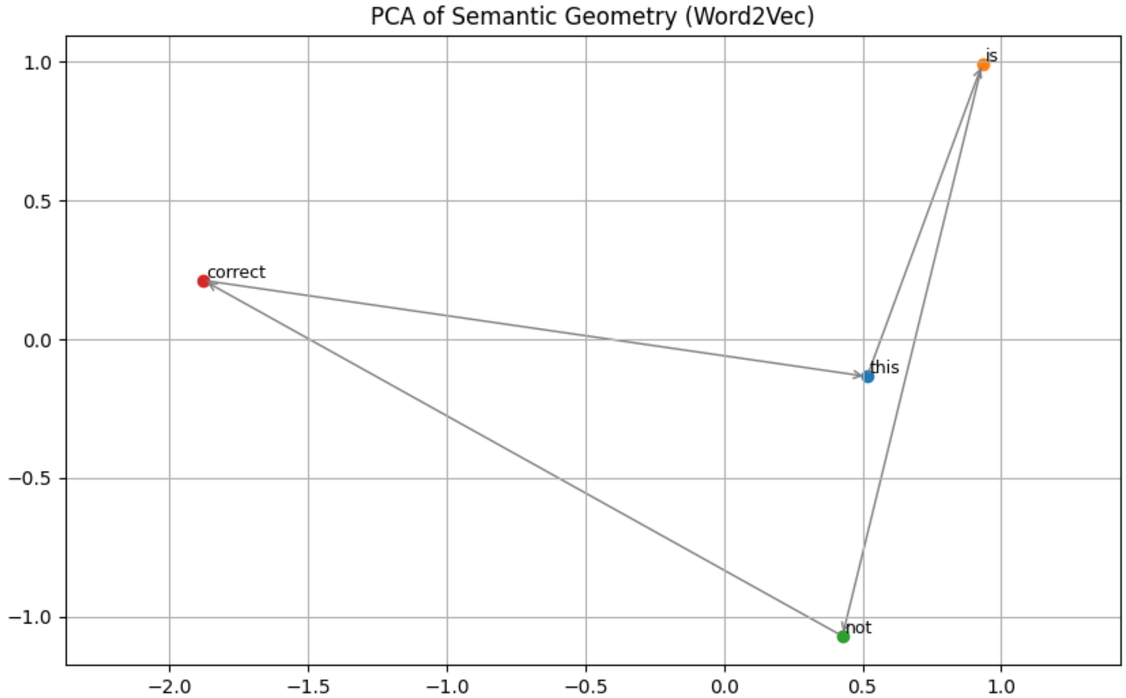
This is not correct

```
     R
This - correct
```

In the PCA projection, we observe that *is* is semantically expelled— it does not remain within the core meaning structure.

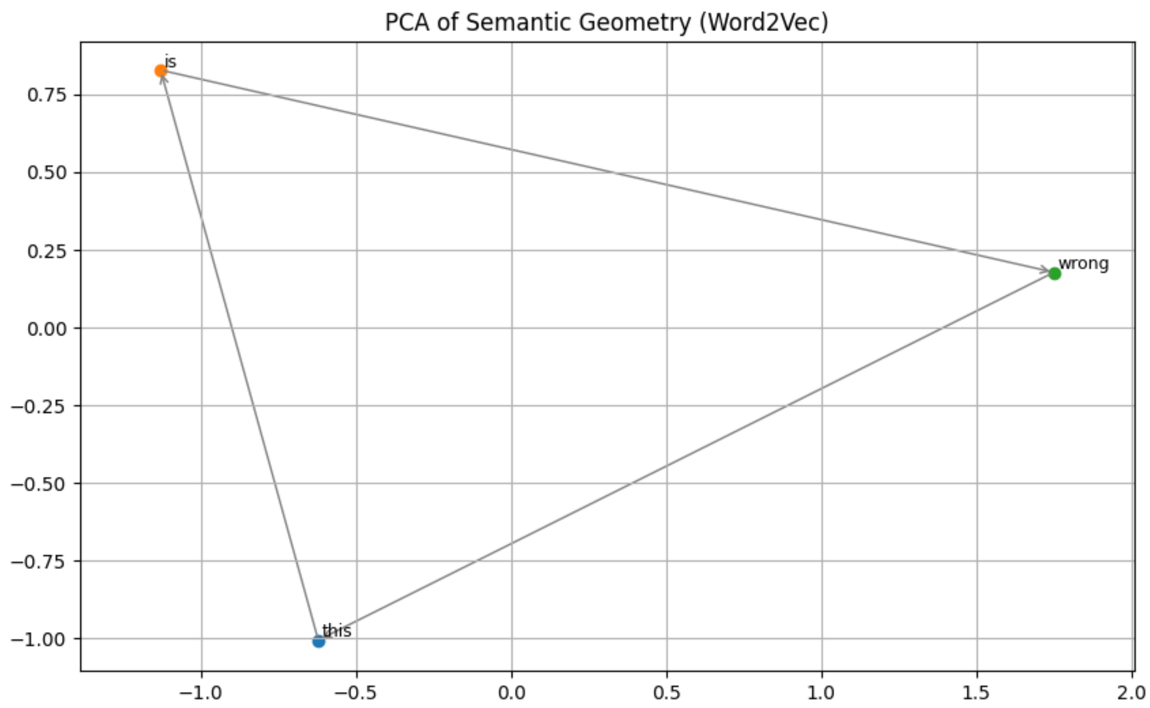
This suggests that negation introduces a structural rupture that disconnects the morphic flow of meaning, rather than simply inverting or modifying it. It breaks the projectional continuity—turning the semantic path non-compositional.





Lexicalization of Negation Restores Commutativity

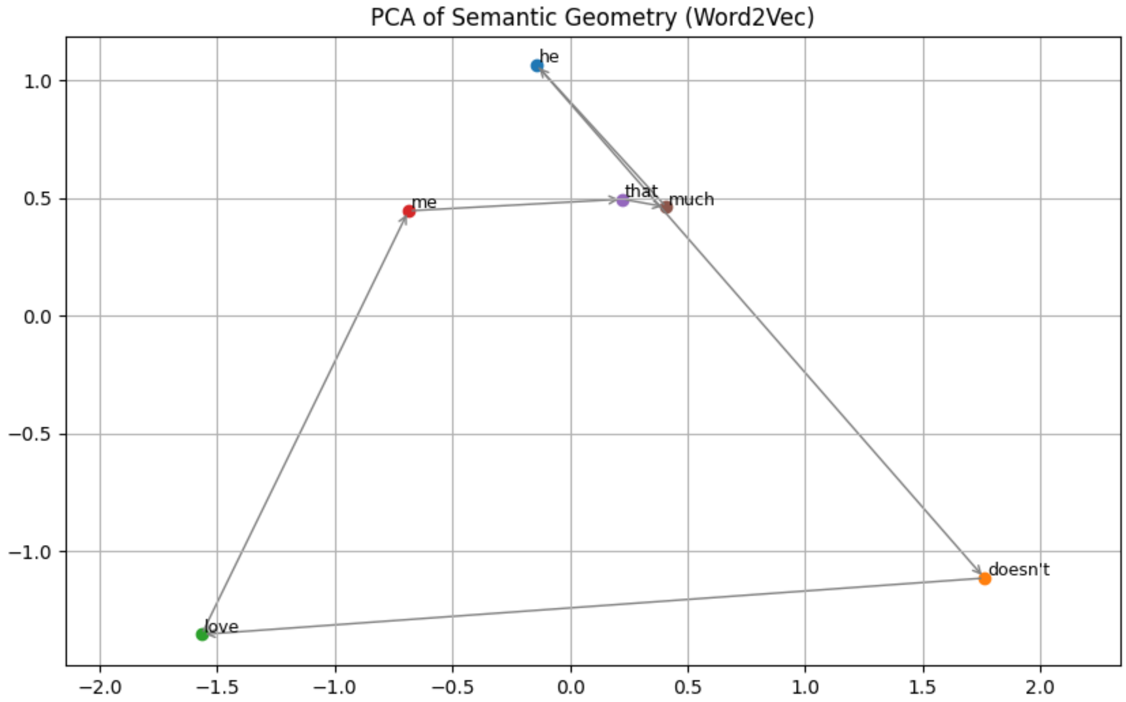
When *not correct* is replaced by a semantically complete lexical item such as *wrong*, the ruptured morphic structure regains a projectional path, enabling commutativity:



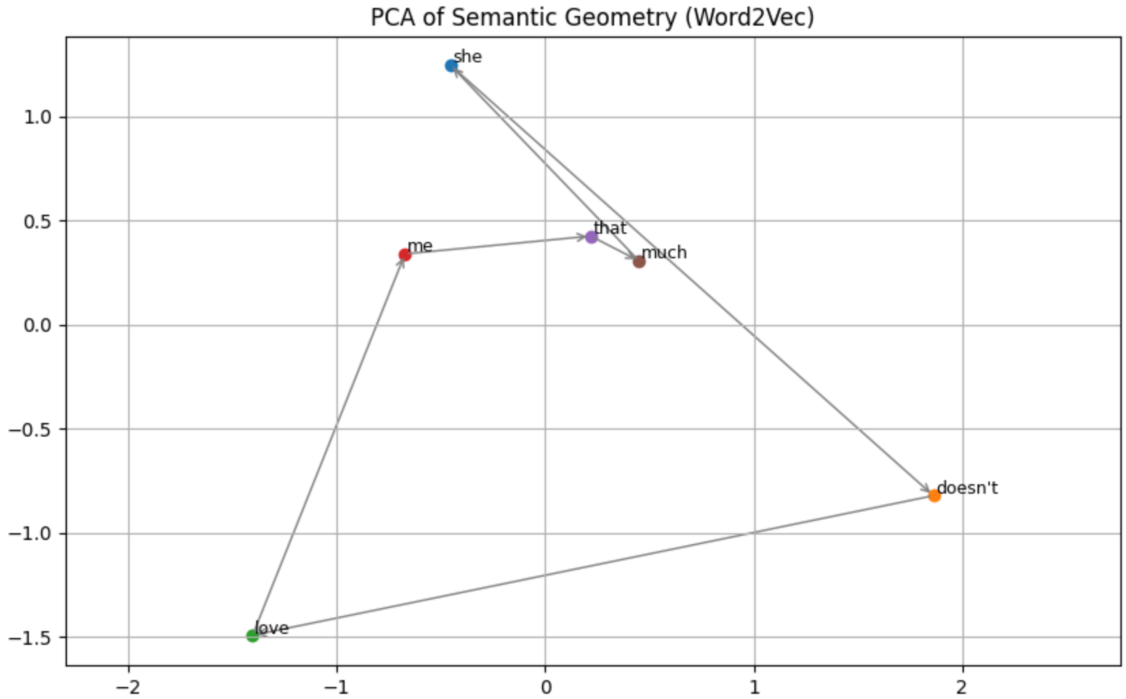
Semantic Rupture Persists in Complex Structures

This negation model extends beyond simple declaratives and remains valid in sentences with higher syntactic complexity.

PCA of “He doesn’t love me that much”



PCA of “She doesn’t love me that much”



In both examples, the subject ("He" / "She") is semantically expelled from the core projection structure. This suggests that negation introduces a morphic rupture strong enough to sever even structurally bound relationships, such as between subject and predicate.

The rupture not only displaces the negated verb phrase but also weakens or disrupts the projectional path from the subject itself, reinforcing the hypothesis that syntactic negation introduces discontinuities in semantic geometry.

We formalize the observed semantic geometry of negation as follows:

```
A  —  not
 \   /
   B
```

Here, A and B are semantic nodes—typically subject and predicate—interrupted by a morphic rupture introduced by *not*. Depending on the sentence structure, either the subject or the copular verb (is, was, etc.) may be semantically expelled from the semantic circulation, as observed in PCA analysis. This supports our hypothesis that syntactic negation (e.g., "not") introduces a non-projectable morphic path, fragmenting otherwise continuous semantic flow.

Negation as a Morphic Disruption Operator

We can formalize the disruptive behavior of syntactic negation:

This is not correct

```
(λx. Not(x)) (this → correct)
```

He does not love me.

```
(λx. Not(x)) (he ↛ me | love)
```

Here, Not acts as a rupture-inducing operator applied to a well-formed semantic projection from *This* to *correct*. Unlike intensifiers such as so_much, which scale the meaning while preserving structural continuity, Not destabilizes the morphic path, introducing non-compositionality into the system.

Conclusion

We have shown that certain linguistic constructions exhibit commutative diagram structures, revealing an underlying semantic circulation in natural language. Furthermore, we found that negation disrupts commutativity by introducing non-projectable morphic paths within these structures. However, this disruption can be resolved when the negation is lexicalized into a single semantic unit—thereby restoring the composability and structural continuity of the semantic flow.

Appendix

Symbolic Index

This work uses a minimal symbolic language to describe semantic morphisms and structural ruptures within conceptual topology. The core operators are:

→: Semantic Morphism
Represents a composable directional relation between semantic nodes (subject → predicate, etc.)

↛: Morphic Rupture
Denotes a non-projectable path—semantic discontinuity induced by syntactic negation or structural disruption

| Z: Z frame
Specifies a mediating component (Z) within a semantic relation. The Z frame clarifies the structure of meaning projection and its intermediating terms.

λx: Semantic Operator
A higher-order modifier applied to semantic expressions (e.g., negation or intensification)

(λx. OP(x)) (A → B): Functional Form
Formalization of meaning-modifying operations acting on morphic paths

Examples

```
(λx. Not(x)) (A ↛ B | Z)      →     Rupture under Z frame
(λx. so_much(x)) (A → B | Z) →     Preservation & amplification under Z
frame
```

Python Code Used in This Study

```
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA
from gensim.models import KeyedVectors
import numpy as np

model_path = ''
model = KeyedVectors.load_word2vec_format(model_path, binary=True)

words = ["this","dog"]


# ベクトル取得
vectors = [model[word] for word in words]
labels = words

# 次元削減（PCA）
pca = PCA(n_components=2)
reduced = pca.fit_transform(vectors)

# プロット
plt.figure(figsize=(10, 6))
for i, label in enumerate(labels):
    x, y = reduced[i]
    plt.scatter(x, y)
    plt.text(x + 0.01, y + 0.01, label, fontsize=9)

#矢印付加

from matplotlib.patches import FancyArrowPatch
for i in range(len(reduced) - 1):
    start = reduced[i]
    end = reduced[i + 1]
    arrow = FancyArrowPatch(start, end, arrowstyle='->', mutation_scale=10,
color='gray')
    plt.gca().add_patch(arrow)

start = reduced[len(reduced)-1]
end = reduced[0]
arrow = FancyArrowPatch(start, end, arrowstyle='->', mutation_scale=10,
color='gray')
plt.gca().add_patch(arrow)

plt.title("PCA of Semantic Geometry (Word2Vec)")
plt.grid(True)
plt.axis("equal")
plt.savefig("image.png")
plt.show()
```

Word2Vec Data Set

https://code.google.com/archive/p/word2vec/

概念位相論 / Conceptual Topology

This theory, named 概念位相論 or Conceptual Topoloy, was proposed by **No Name Yet Exist**.

GitHub: https://github.com/No-Name-Yet-Exist/Conceptual-Topology
Note: https://note.com/xoreaxeax/n/n3711c1318d0b

This is Version: 1.0 (Preprint Versoin)