Análisis Matemático: Modelado en el Espacio de Embeddings 384D de CAELION

Autor: Manus AI
Fecha: 23 de enero de 2026
Versión: 1.0

---

1. Introducción

Este documento detalla el modelado matemático del **error semántico e(t)** y la **corrección u(t)** dentro del espacio de alta dimensión (384D) de CAELION. Se explica cómo estos vectores se calculan y se interpretan en el contexto del control semántico.

---

2. El Espacio de Estados Semántico: $\mathbb{R}^{384}$

CAELION opera en un espacio de estados semántico, que es un espacio vectorial de 384 dimensiones. Cada punto en este espacio representa un significado o concepto.

- **Modelo de Embedding**: Se utiliza un modelo de sentence embedding como `all-MiniLM-L6-v2`, que convierte cualquier texto en un vector normalizado de 384 dimensiones.
- **Vector de Estado x(t)**: Es el embedding de la salida del LLM en el turno `t`. Es un vector en $\mathbb{R}^{384}$.
- **Referencia Ontológica x_ref**: Es el embedding del texto que define el propósito central del sistema. También es un vector en $\mathbb{R}^{384}$.

> *$x(t) \in \mathbb{R}^{384}$, $||x(t)|| = 1$ $x\_ref \in \mathbb{R}^{384}$, $||x\_ref|| = 1$*

---

3. Modelado del Error Semántico e(t)

El error semántico **e(t)** no es un escalar, sino un **vector** que captura la magnitud y la dirección de la desviación semántica.

3.1. Cálculo

Se calcula como la diferencia vectorial entre el estado actual y la referencia:

e(t) = x(t) - x_ref

Dado que x(t) y x_ref son vectores de 384 dimensiones, **e(t) también es un vector en $\mathbb{R}^{384}$**.

```
e(t) = [x₁(t) - x_ref₁, x₂(t) - x_ref₂, ..., x₃₈₄(t) - x_ref₃₈₄]
```

3.2. Interpretación

- **Dirección del Vector e(t)**: Apunta desde la referencia (x_ref) hacia el estado actual (x(t)) en el espacio semántico. Indica "hacia dónde" se ha desviado el significado.
- **Magnitud del Vector ||e(t)||**: Representa la "distancia" semántica entre el estado actual y la referencia. Se puede calcular como la norma euclidiana del vector de error.

> ***Relación con la Coherencia Ω**: La magnitud del error está directamente relacionada con la similitud coseno (coherencia Ω). $||e(t)||^2 = ||x(t) - x_{ref}||^2 = ||x(t)||^2 - 2(x(t) \cdot x_{ref}) + ||x_{ref}||^2$ Como los vectores están normalizados ($||x||=1$): $||e(t)||^2 = 1 - 2\Omega(t) + 1 = 2(1 - \Omega(t))$ $\boldsymbol{||e(t)|| = \sqrt{[2(1 - \Omega(t))]}}$*

Esto muestra que un error de magnitud cero implica una coherencia de 1, y viceversa.

---

4. Modelado de la Corrección Aplicada al Prompt u(t)

La corrección **u(t)** también es un **vector en el espacio de 384 dimensiones**. Representa la "fuerza" y "dirección" semántica que se debe aplicar para guiar al LLM de vuelta a la referencia.

4.1. Cálculo

Se calcula mediante la ley de control LQR:

u(t) = -K·e(t)

Donde:

- **e(t)** es un vector de 384x1.
- **K** es la matriz de ganancia, de dimensiones 384x384.

El resultado **u(t)** es un vector de 384x1, es decir, un vector en el mismo espacio de embeddings.

4.2. Interpretación

- **u(t)** es un **vector de corrección semántica**. No es texto, sino una representación matemática de la “idea” que se necesita para corregir el rumbo.
- La dirección de **u(t)** es, en general, opuesta a la del error **e(t)** (debido al signo negativo), empujando al sistema de vuelta hacia la referencia.
- La magnitud **||u(t)||** determina la “fuerza” de la corrección.

4.3. De Vector a Texto: La Aplicación al Prompt

Aquí reside una de las claves de la implementación. El vector de corrección **u(t)** no se puede “sumar” directamente al prompt. Se debe traducir de nuevo a lenguaje natural. CAELION lo hace de forma implícita:

1. **Cálculo del Estado Corregido Deseado**: Se puede pensar en un estado objetivo para el siguiente paso: **x_deseado(t+1) = x(t) + u(t)** (o una variante)

2. **Búsqueda Semántica Inversa (Implícita)**: El sistema necesita encontrar un texto cuyo embedding sea lo más cercano posible a **x_deseado(t+1)**. Esto es un problema de “búsqueda semántica inversa” o “decodificación de embeddings”.

3. **Implementación Pragmática en CAELION**: En lugar de resolver este complejo problema de búsqueda, CAELION utiliza una solución pragmática y efectiva:

inyecta la información de control en el prompt en lenguaje natural.

```
correction_prompt = f"""
{original_prompt}

[CONTROL DE ESTABILIDAD ACTIVO]
Referencia ontológica: {self.reference_text}
Error detectado de magnitud {error_mag:.3f}.
INSTRUCCIÓN: Corrige la trayectoria hacia la referencia.
"""
```

El LLM, al procesar este prompt aumentado, realiza implícitamente la tarea de decodificación. Entiende la instrucción de "corregir" y genera una respuesta que, idealmente, moverá el estado semántico en la dirección de **u(t)**.

---

5. Conclusión

El modelado matemático en CAELION es coherente y se lleva a cabo enteramente en el espacio de embeddings de 384 dimensiones:

- El **error semántico e(t)** es un vector que captura la **magnitud y dirección de la desviación** del significado.
- La **corrección u(t)** es un vector que representa la **fuerza y dirección semántica del ajuste** necesario.
- La **matriz de ganancia K (384x384)** actúa como un transformador que convierte el error en la corrección óptima.
- La **traducción del vector de corrección u(t) a texto** se realiza de forma pragmática, delegando la tarea de "decodificación semántica" al propio LLM mediante un prompt aumentado con instrucciones de control.

Este enfoque permite a CAELION aplicar los principios rigurosos de la teoría de control directamente en el dominio abstracto y de alta dimensión de la semántica del lenguaje.