

RESUMEN IDEAS CLAVE

ULOGIC: Un lenguaje universal (matemático-computacional) para el Razonamiento Neurosimbólico Verificable

© Leopoldo Cano Guardiola 2025 - <https://ulogiclang.ai>

Introducción

La inteligencia artificial y los fundamentos de las matemáticas se enfrentan a crisis paralelas. Por un lado, los Grandes Modelos de Lenguaje (LLM), a pesar de sus impresionantes capacidades, adolecen de un problema fundamental de fiabilidad, manifestado en "alucinaciones" y una incapacidad para el razonamiento riguroso. Por otro lado, los fundamentos de las matemáticas, aunque funcionales, siguen dependiendo de sistemas axiomáticos como ZFC, diseñados para sortear paradojas en lugar de resolverlas desde su raíz (sistema ZFC que a su vez adolece de insuficiencia -si tiene un modelo, tiene uno numerable, con lo cual los axiomas no caracterizan de forma suficiente la idea intuitiva de conjunto-). Este informe presenta (sin entrar en su definición y detalles, sólo exponiendo sus capacidades) ULOGIC, un lenguaje y sistema formal que propone una solución unificada a ambas crisis.

ULOGIC representa un cambio de paradigma en los sistemas formales al unificar la sintaxis, la semántica y la computación en un único marco basado en expresiones. Su naturaleza procedimental, autorreferencial y constructiva resuelve las paradojas fundacionales internamente y, al hacerlo, sienta las bases para una novedosa arquitectura neurosimbólica. Esta arquitectura es capaz de producir razonamiento generado por IA que es verificable y fiable, culminando en la visión de una red global de conocimiento reutilizable y exacto conocida como TekDocs.

Este documento explora esta propuesta en tres partes. La Parte I establecerá los fundamentos formales de ULOGIC, detallando sus principios básicos y contrastándolos con las tradiciones lógicas establecidas. La Parte II detallará su aplicación en un sistema de IA neurosimbólico, demostrando cómo supera las limitaciones de los LLM actuales. Finalmente, la Parte III explorará la visión transformadora de los TekDocs (Transportable Encapsulated Knowledge Documents) como un sustrato global para el conocimiento acumulativo y verificable.

Parte I: La Arquitectura Fundacional de ULOGIC

Esta sección muestra como ULOGIC es un sistema formal novedoso, resumiendo sus principios básicos y contrastándolos con las tradiciones lógicas establecidas.

Sección 1. Un Marco Unificado de Expresiones Procedimentales

El principio central de ULOGIC es que "todos los constructos son expresiones". En este sistema, no existe una distinción fundamental entre datos, definiciones, teoremas, demostraciones, algoritmos o incluso la ejecución de esos algoritmos. Todos son simplemente expresiones con una estructura interna y partes reconocibles, independientes de su representación serializada (su escritura).

ULOGIC opera sobre un conjunto de *reglas* que explican cómo obtener unas expresiones de otras. Todo en el fondo es un "procedimiento". Estos procedimientos pueden ser de dos tipos:

- **Abiertos:** La aplicación operativa de reglas, que es la esencia del razonamiento.
- **Cerrados:** La aplicación de reglas siguiendo un algoritmo para generar secuencias de contextos sobre los que se realizan razonamientos.

Fundamentalmente, el razonamiento y la computación son la misma actividad procedimental dentro del sistema.

ULOGIC no se basa en la teoría de tipos, una desviación explícita de sistemas como Coq o Agda, que dependen de los tipos para mantener la consistencia.

La validez de una expresión es una cuestión de su *estructura interna* únicamente, verificable mediante el análisis de su conformidad con las reglas del sistema.

Sección 2. Semántica Interna y el Rechazo Radical de los Modelos Externos. Necesidad de una re-evaluación y re-fundación completa de la lógica moderna (semántica tarskiana, teoremas de godel, lógica formal...)

En ULOGIC "no hay semántica externa en modelos, el significado es interno sin referencia a nada externo". Este principio representa una ruptura fundamental con el enfoque estándar de la semántica en la lógica: la teoría de modelos de Alfred Tarski, donde el significado es una correspondencia con una estructura matemática externa.

ULOGIC sostiene que toda la semántica tarskiana no solo es innecesaria, sino un completo error conceptual que ha conducido a los lenguajes formales actuales, como la lógica de primer orden (FOL), a un callejón sin salida.

En ULOGIC, por el contrario, el "significado" de una expresión es su relación estructural y procedimental con todas las demás expresiones dentro del sistema. El significado queda definido implícitamente por las reglas que pueden producirla y las reglas en las que puede participar para producir nuevas expresiones. Se trata de una semántica puramente sintáctica, operacional o demostrativo-teórica. El sistema es autocontenido y se autojustifica.

Este enfoque recuerda a la escuela filosófica del formalismo, donde las matemáticas se consideran la manipulación de símbolos según reglas especificadas, sin preocuparse por su "significado" en una realidad externa. Pero difiere del formalismo radicalmente en tanto, por un lado, se aleja de la visión "alfabética", porque las expresiones NO son secuencias de letras de un alfabeto: Las expresiones escritas son serializaciones de una estructura (la expresión) que puede ser "escrita" de cualquier forma siguiendo cualquier convenio. Y por otro lado ULOGIC se aleja de la afirmación formalista típica y característica de "manipulación de símbolos sin significado" porque es absolutamente -e irónicamente- errónea: es la manipulación de símbolos usando reglas lo que crea el significado (**significado intensional, no denotacional*)

Por decirlo claramente: El principal problema de la semántica Tarskiana, que interpreta un lenguaje formal en una "estructura matemática" (un conjunto con funciones, y relaciones) es que NO es una semántica. Parece plausible porque emula a la semántica denotacional donde los signos se refieren a realidades exteriores captadas por un sistema perceptual. Pero la semántica denotacional es otra cosa radicalmente diferente a la semántica interpretativa de Tarski y de la lógica simbólica estándar.

Para entenderlo mejor: La lógica formal, la semántica y teoría de modelos, y toda la lógica “moderna” empezó como un intento de construir un lenguaje, y ha acabado convertida en una teoría matemática más, alejada del intento fundacional original: Es una teoría de POLINOMIOS, simplemente.

¿Qué es un polinomio? Dada una estructura sobre un conjunto A , con funciones f_1, f_2, \dots y relaciones R_1, R_2, \dots podemos formar secuencias que llamamos términos y fórmulas. Si las funciones son simplemente una “suma” (+) y una multiplicación (\cdot) los “polinomios” son justamente los polinomios habituales como podría ser $3x^2+x^3+2$ (que por cierto inventó Descartes sin saberlo).

La “semántica-tarskiana” es meramente una EVALUACIÓN de polinomios, o sea, una interrelación matemática entre dos estructuras. Evaluar un polinomio de un anillo multiplicativo es la famosa operación que aprendemos en la escuela de “sustituir la x ”. En la lógica formal los “términos” son polinomios cuya evaluación es un valor en el conjunto A de base, y las “fórmulas” TAMBIÉN son polinomios cuya evaluación realmente toma un valor en el conjunto $\{0,1\}$ que llamamos “verdadero-falso” pero que NO tiene nada que ver con el significado habitual de “verdadero y falso”.

La función de evaluación de polinomios permite definir la conocida propiedad “polinomios que se satisfacen en una estructura”. Y desde ahí definir una “función de deducción” como sigue: de los polinomios $\{p_1, p_2, \dots, p_n\}$ se “deduce o deriva” el polinomio q , si y solo si ocurre que q “se satisface” en toda estructura A donde p_1, p_2, \dots, p_n son satisfechos. Esta función de derivación es la única que se utiliza realmente en teoría de modelos y en lógica matemática. Por eso los “cálculos deductivos sintácticos” (conjuntos de reglas para derivar polinomios unos de otros) son irrelevantes: pueden ser axiomático-hilbertianos, natural-secuenciales a la Gódel o cualquier otra combinación, porque no importa, en tanto son construcciones que acaban en la MISMA función de derivación, la definida usando polinomios mediante el método de interpretación de Tarski.

Los polinomios a su vez pueden ser usados para definir “conjuntos de estructuras”. Dado un conjunto de polinomios P , $\text{Estruct}(P)$ es el conjunto de estructuras que “satisfacen” todos los polinomios de P (es decir, todos los polinomios del conjunto P , tienen el valor “1” al ser evaluados en cualquier estructura A del conjunto $\text{Estruct}(P)$)

Pregunta 1: ¿Son las estructuras del conjunto $\text{Estruct}(P)$ isomorfas entre sí?. La respuesta es que no. Los teoremas de Lowenheim-Skolem dicen justamente eso. O sea, intentar “definir” una estructura “usando polinomios” es una muy mala idea.

Pregunta 2: ¿Son las estructuras del conjunto Estruct(P) “elementalmente equivalentes entre sí”? (Dos estructuras son “elementalmente equivalentes” si cualquier polinomio p que se satisface en una también se satisface en la otra). Pues bien, la respuesta es que casi nunca, incluso para casos muy simples como intentar definir la estructura de los números naturales. Este es el único significado del famoso teorema de incompletitud de Gödel (que nada tiene que ver con “verdades-indemostrables”).

Por tanto, si usamos “polinomios” y la enrevesada forma de “definir conjuntos de estructuras” mediante la “semántica tarskiana” (que es meramente una evaluación de polinomios) el resultado es poco útil: no conseguimos ni isomorfía ni siquiera equivalencia elemental.

Estos resultados, incluido el de Gödel son IRRELEVANTES para las matemáticas y para la lógica (si entendemos la “lógica” como un intento de ingeniería inversa del lenguaje real matemático), porque ningún matemático en la historia ha definido estructuras matemáticas usando “polinomios” (los polinomios son objetos matemáticos interesantes, pero nada más). Si defines estructuras matemáticas (como la de los números naturales) o cualquier otra “usando expresiones matemáticas normales” por supuesto que alcanzas la isomorfía y la exactitud suficiente.

La lógica formal es una trampa de términos y definiciones: Las “fórmulas” no son expresiones matemáticas (aunque al principio querían serlo) ya que han acabado siendo meramente polinomios en estructuras. La “verdad” tarskiana no es lo que dice ser, y no tiene nada que ver con la palabra “verdad” en sus usos habituales, es meramente una evaluación de polinomios. La “deducción” no es deducción en el sentido matemático “de las matemáticas reales”, es meramente una interrelación entre polinomios basada en la función de evaluación tarskiana. La incompletitud de Gödel que usando terminología lógica formal se enuncia “hay verdades no demostrables” (terminología correcta en la teoría de modelos) parece un milagro, pero porque el significado de “verdad” y “demostrable” están falsificados en la terminología de la lógica de modelos (los teoremas de Gödel simplemente enuncian la incapacidad de definir conjuntos de estructuras elementalmente equivalentes entre ellas usando el enrevesado método de la evaluación tarskiana).

La lógica-formal comenzó como un intento de continuar la venerable tradición de “ingeniería inversa” de Aristóteles, Peano, Frege, Russell, etc (es decir, el intento de

encontrar las reglas que seguimos cuando hacemos demostraciones, reglas que una vez “descubiertas” transforman y reinventan el propio lenguaje que están estudiando). Pero la pericia técnica y genialidad de los matemáticos que intentaron ir más allá (Tarski, Hilbert, Godel y tantos otros) nos ha llevado a un callejón sin salida: a una teoría matemática de “polinomios” que ha abandonado las aspiraciones de entender las matemáticas reales. Las matemáticas reales, las de verdad, no son “lo que hay dentro de los polinomios” sino lo que hay fuera en las demostraciones llamadas metamatemáticas.

Esto a su vez nos lleva al mayor error de la historia, iniciado por mentes preclaras como Hilbert: Concebir que existen “lenguajes formales alfabéticos” por un lado, con reglas exactas, y por otro una “metamatemática-informal” que no es formalizable, pero que usa métodos de razonamientos finitistas-indubitables para hacer demostraciones sobre los sistemas formales.

La manifiesta falsedad de semejante programa de investigación es hasta obvia: Basta ir a una facultad de matemáticas y ver QUÉ hacen los matemáticos. Y lo que hacen es manejar un lenguaje (bastante complejo) que es una herencia de un proceso histórico, un lenguaje dentro del lenguaje natural, un lenguaje que ahora está plagado de símbolos y expresiones antes inexistentes -creadas por la investigación lógica-. Pero es un único lenguaje. Un lenguaje en el que hacemos razonamientos, demostraciones, definiciones, definimos algoritmos, ejecutamos algoritmos y hasta hablamos sobre el propio lenguaje. ¿Conoce el lector a alguna persona del planeta Tierra que se haya “salido” del lenguaje para hablar sobre el lenguaje? Solo tenemos UN lenguaje único.

Sección 2b. Necesitamos redefinir la “lógica” como un proyecto de ingeniería inversa absolutamente necesario para entender el ÚNICO lenguaje que tenemos (un lenguaje con capacidades matemáticas, computacionales y metalingüísticas)

La palabra “lógica” ha significado tantas cosas en los últimos 200 años (el famoso logicismo Ruselliano tiene hasta un punto místico) que apenas significa ya nada.

Nuestra propuesta es muy simple y hasta auto-evidente, y se resume en tres principios:

- (1) **Lenguaje único con capacidades múltiples** (discursiva, matemática, computacional y metalingüística): Los seres humanos tenemos un único lenguaje. Este lenguaje lo hemos creado a lo largo de 3000 años. El “lenguaje natural” es el mar de fondo, pero dentro de él hemos creado sublenguajes como el de las matemáticas, la computación, los algoritmos. Pero todo es un único lenguaje: uno capaz de hacer demostraciones, algoritmos, ejecuciones, y metalingüística.

- (2) **“Lógica” como “ingeniería-inversa”**: La “lógica” debe cambiar de rumbo: debe ser el proyecto de “ingeniería inversa” para descifrar qué reglas seguimos cuando hacemos demostraciones, algoritmos, computación o metalingüística. El lenguaje actual es borroso e impreciso. Cuando encontramos “patrones” y los convertimos “en reglas” ¡Estamos transformando el lenguaje en algo nuevo! Es el proceso iterativo que hemos seguido al crear las matemáticas: nuevas personas crean nuevos métodos sin saberlo, y personas que vienen detrás los estudian y crean reglas que cambian el lenguaje anterior. Las “reglas lógicas” evidentes de hoy, que un matemático conoce, forman parte del proceso histórico conocido. Pero la pregunta importante siempre es: ¿Cuáles son las reglas lógicas que estamos utilizando ahora mismo al hacer matemáticas y computación y metalingüística, reglas que están ahí y de las cuales no somos ni conscientes? La respuesta es sorprendente: casi todo lo que importa y es esencial está ahora mismo “fuera del radar” de la lógica.

- (3) **El progreso consiste en crear lenguajes más potentes**. El progreso de la civilización se fundamenta exclusivamente en ese proceso evolutivo de creación de un lenguaje cada vez más potente junto a complejas formas de “aplicarlo” a la realidad para aprender a ver lo que no es visible: Es la historia que empieza antes de Euclides (y que Euclides cristalizó en una nueva forma de razonamiento) hasta llegar a Newton, y terminando en la energía nuclear, la exploración espacial y la ingeniería genética. El superpoder último de la especie humana es el lenguaje, algo que empezó como simples enunciados para describir la realidad, y que ha acabado siendo un edificio imponente de “razonamiento” y “computación”.

Resumiendo y terminando:

Al rechazar los modelos externos, ULOGIC define un universo lógico-computacional completamente autocontenido. La "verdad" de una declaración es equivalente a su "demostrabilidad" o "constructibilidad" dentro del sistema.

Esto tiene profundas implicaciones, y si se quiere decir así, fusiona los conceptos de verdad y computación. Aunque lo mejor sería dejar de hablar de "verdad" y reservar el término "verdad" exclusivamente para la semántica denotacional de interrelación del lenguaje con el mundo real (pero eso es un proyecto ajeno a la lógica y las matemáticas, porque involucra sistemas perceptuales reales que hacen de puente entre las percepciones y el lenguaje a través de mapeos en espacios topológicos internos perceptuales donde ciertas regiones de ese espacio son etiquetadas dinámicamente usando el lenguaje ... y sí, suena como parece: es otra historia).

Los teoremas de incompletitud de Gödel demostraron una brecha entre la "verdad" y la demostrabilidad en sistemas formales; existen enunciados que son "verdaderos" (en el modelo estándar de la aritmética) pero no demostrables dentro del sistema. Pero en realidad todo eso es meramente una forma de hablar en el universo de la semántica tarskiana y teoría de polinomios.

Podríamos decir que ULOGIC cierra esta brecha al *definir* la verdad como demostrabilidad. Pero también podemos decir que ULOGIC disuelve este problema que deja de ser un problema. Un enunciado para el cual no se puede formar una expresión-demostración constructiva no es "verdadero pero indemostrable"; simplemente no es un teorema del sistema.

Esto elude las dificultades filosóficas de los teoremas de Gödel al negarse a reconocer una noción de verdad externa al propio sistema, logrando a cambio una coherencia interna completa donde la verdad y la prueba son sinónimos. Aunque insistimos: el exacto significado de los teoremas de Gödel es simplemente el explicado más arriba (que usar polinomios para definir conjuntos de estructuras mediante el método de satisfacción de tarski es una mala idea que ni siquiera consigue alcanzar la equivalencia elemental entre estructuras ... y por supuesto, además es un método que ningún matemático ha usado jamás)

Sección 3. La única solución posible a las Paradojas Conjuntistas: Las Definiciones como Constructos no Eliminables

Expresado en términos habituales según la bibliografía, la teoría ingenua de conjuntos se basa en un principio de comprensión sin restricciones que conduce directamente a la Paradoja de Russell. Las soluciones estándar, como ZFC o la teoría de tipos, imponen restricciones axiomáticas o estructurales para evitar la formación de conjuntos paradójicos.

ULOGIC propone una solución más profunda y radical, basada en una idea clave: **las definiciones matemáticas NO son abreviaturas eliminables**, en contra de lo que dicta el estándar lógico. En la lógica tradicional, una definición es una mera conveniencia notacional que puede ser sustituida por su forma expandida sin alterar el significado. ULOGIC rechaza esta noción. Aquí, las definiciones son expresiones con una estructura interna propia y un rol fundamental en el sistema.

La “definiciones que no son abreviaturas y que no son eliminables” siempre han existido. Pero es en el siglo XIX cuando los conjuntos empiezan a verse como objetos de estudio (¿es el conjunto de las funciones continuas cerrado dentro del conjunto de las funciones acotadas definida cierta métrica?). La teoría Cantoriana aceleró este proceso. Y aquí se produjo un salto y un choque de trenes: las contradicciones conjuntistas.

Es precisamente el “descubrimiento” de esta naturaleza no eliminable de las definiciones lo que permite a ULOGIC resolver las paradojas conjuntistas desde su raíz. En ULOGIC, las reglas para crear expresiones de definición están diseñadas para prohibir la construcción de este tipo de recursividad estructuralmente defectuosa. Una definición no es una simple abreviatura, sino un expresión que antes no existía y que ahora se introduce en el sistema.

Una expresión “que antes no existía, pero que ahora se introduce en el sistema” es exactamente lo que uno entiende por “axioma” o “hipótesis” ... y eso en el fondo son las definiciones: Una autorización para crear nuevas expresiones, que antes no existían.

“Definir” no es hacer abreviaturas (aunque también): En realidad “definir” (cuando las definiciones no son eliminables) es tener el permiso de crear “axiomas” (expresiones que antes no existían) pero sin llamarlos axiomas. Entonces podemos

demostrar cosas que sería imposible de demostrar sin esos “axiomas” y luego borrar las huellas y fingir que lo que hemos demostrado sólo gracias a la ayuda de las definiciones también hubiera sido demostrable sin ayuda de las definiciones (lo cual es completamente falso).

ULOGIC explica todo esto con precisión milimétrica.

Pero si el lector quiere ver “la luz al final del tunel” le proponemos un ejercicio: El famoso teorema de Cantor de que la potencia de $|A|$ es menor que la del conjunto de sus subconjuntos $|P(A)|$ es bien conocida, y se basa en una técnica habitual: hacer una definición mágica que prueba el teorema por reducción al absurdo. ¿Podría el lector hacer una demostración de ese teorema SIN hacer la definición clave intermedia? Es imposible. Ahora la pregunta: Si las definiciones son abreviaturas eliminables que no aportan nada nuevo a lo que ya existía ¿por qué hay una montaña de teoremas que NO se pueden demostrar sin hacer una definición mágica intermedia?

Este patrón es bien conocido pero mal comprendido (“llegar a ser un matemático profesional” consiste en dominar intuitivamente ese método). De hecho, y como ejemplo muy significativo, la “realidad matemática” de que hay infinitos de diferente cardinalidad y las diversas demostraciones de ello, parecen un descubrimiento asombroso de una realidad trascendente, pero se basan en un patrón demostrativo que en el fondo es meramente (si se quiere decir así) “una regla gramatical”, un mero convenio “de cómo usar las expresiones”, convenio que ni sabemos que estamos siguiendo. (Esto sólo se puede “ver” si tienes reglas explícitas y lenguajes alternativos como ULOGIC).

Efectivamente: somos esclavos de nuestro lenguaje y no podemos ver más allá de lo que el lenguaje nos permite. Sólo los nuevos lenguajes permiten ver las ingenuidades, imprecisiones y defectos de los viejos lenguajes (interesante: investigar los libros de historia de las matemáticas que hablan de la evolución del rigor matemático, porque ha sido justamente ese proceso de evolución del lenguaje)

Continuando: Las reglas del sistema ULOGIC impiden que una expresión de definición se construya de una manera que genere una dependencia circular viciosa, haciendo que las paradojas sean sintácticamente imposibles de formular. Esto es análogo a un error de sintaxis en un lenguaje de programación, pero aplicado a la estructura misma del razonamiento.

Tabla 1: Un (“posible-útil-pero-imperfecto”) Análisis Comparativo de los Mecanismos de Resolución de Paradojas

Sistema	Principio Básico para la Formación de Conjuntos/Co lecciones	Mecanismo de Evitación de Paradojas	Permite Conjunto Universal	Consecuencia Clave
Teoría Ingenua de Conjuntos	Comprensión sin restricciones	Ninguno (Inconsistente)	Sí (conduce a la paradoja)	Inconsistente
Teoría de Tipos de Russell	Comprensión por tipos	Jerarquía de tipos sintácticos	No	Jerarquía rígida
ZFC	Esquema Axiomático de Separación	Restricción a subconjuntos de conjuntos existentes ("Limitación de Tamaño")	No	Jerarquía acumulativa (Axioma de Fundación)
Nuevas Fundaciones (NF)	Comprensión estratificada	Estratificación sintáctica de fórmulas ("Limitación de Estructura")	Sí	Modelos no bien fundados posibles / AC falla
ULOGIC	Reglas para la formación de expresiones de definición no eliminables	Reglas procedimental es que impiden definiciones con dependencias circulares viciosas	Sí	Las definiciones son constructos fundamentales / Consistencia por construcción

Sección 4. El Lenguaje ULOGIC como su Propio Metalenguaje ¿Es posible?

Que un lenguaje "formal-exacto" sea su propio metalenguaje es hoy por hoy una herejía lógica, y cualquiera que lo afirme parecerá "un poco tonto" porque está bien demostrado que eso es imposible.

Ahora bien, la cantidad de teoremas de la lógica matemática que *no son lo que parecen* es interminable.

El teorema de indefinibilidad de Tarski demuestra que un lenguaje formal no puede definir su propio predicado de verdad semántica sin caer en contradicción. La solución estándar es la jerarquía de lenguajes de Tarski, que separa el *lenguaje objeto* del *metalenguaje*.

ULOGIC hace la afirmación radical de que "el lenguaje tiene capacidad para ser su propio metalenguaje".

Esto es posible porque ULOGIC elude el teorema de Tarski. Al haber abandonado la semántica tarskiana y los modelos externos, ULOGIC no opera con un concepto de verdad semántica, sino con el concepto sintáctico y procedimental de "ser una expresión válidamente construida" o "ser un teorema".

La capacidad de autorreferencia segura de ULOGIC se deriva directamente de sus principios fundacionales: las expresiones son estructuras, las definiciones no son eliminables y la semántica es interna.

Una expresión de ULOGIC puede referirse a otra expresión y afirmar su *demostrabilidad* o *constructibilidad*. La Paradoja del Mentiroso ("Esta oración es falsa") se desactiva porque la afirmación "Esta expresión no es un teorema" no es inherentemente paradójica. La consistencia interna de ULOGIC se mantiene mediante sus reglas de definición, que impiden la construcción de expresiones que conducirían a contradicciones. Este concepto es análogo a la reflexión computacional, donde un programa puede inspeccionar y modificar su propio código como si fueran datos, pero elevado a un principio fundamental de la lógica.

El problema de ULOGIC a la hora de intentar crear un "lenguaje exacto con capacidad metalingüística" es "cómo conseguirlo" porque eso es un camino inexplorado. (ADELANTO: Mostraremos que es posible incluso que el lenguaje cree modelos de sí mismo ... aunque este es un campo de investigación novedoso).

Parte II: La Simbiosis Neurosimbólica: ULOGIC y los Grandes Modelos de Lenguaje

ULOGIC es un proyecto de investigación pura en fundamentos de la lógica y las matemáticas, con implicaciones filosóficas, y en principio destinado a existir dentro de libros de filosofía, de lógica o en reuniones de gente rara (lógicos, matemáticos y filósofos).

Lo cual es poco alentador, porque la comunidad de matemáticos aplicados considera que la "lógica" es irrelevante para ellos (y razón no les falta, por la evolución de la lógica que ha terminado siendo teoría de polinomios). Los lógicos puros que trabajan en cosas como teoría de modelos avanzada o teoría de conjuntos en un universo de técnicas de forcing para probar independencia de "axiomas" sólo se hablan entre ellos. Y los filósofos de la lógica y las matemáticas analizan el panorama llegando a conclusiones inevitables: hoy por hoy nadie tiene ni la menor idea de qué son las matemáticas o la lógica (y las acumulación de paradojas, y problemas simplemente ha sido resuelta "por olvido": olvidarse de un problema es otra forma de "solucionar" el problema en una comunidad).

En otras palabras: ULOGIC no tiene la menor oportunidad de tener un impacto en el mundo real si la propuesta se hace a matemáticos-lógicos (puede que los filósofos de la ciencia tengan una perspectiva más amplia y le den mejor acogida).

Sin embargo, como tantas veces ocurre, podría tener un impacto imprevisto: En Inteligencia Artificial.

La investigación en inteligencia artificial de los últimos años (2016-2025) ha producido una explosión asombrosa de "máquinas prodigiosas" generativas. Pero se ha estancado: no se ve ningún camino posible para mejorar las capacidades lógicas y de razonamiento de los sistemas de forma "potente" y no meramente "simulada".

Como tantas veces en la historia, la solución podría estar en algo que en origen nada tiene que ver con la Inteligencia Artificial, fruto de una investigación que nunca tuvo como objetivo ser aplicada a la inteligencia artificial.

Ahora vamos a contar esa historia.

Sección 5. Uniendo el Razonamiento Intuitivo y el Formal

El marco cognitivo "Pensar, rápido y despacio" describe dos modos de pensamiento: el Sistema 1, que es rápido, intuitivo y basado en patrones, y el Sistema 2, que es lento, deliberado y lógico.

Este paradigma se puede mapear directamente a las arquitecturas de IA. Los LLM son potentes motores de Sistema 1, que destacan en el reconocimiento de patrones, la fluidez y los saltos intuitivos. La IA simbólica, por su parte, representa motores puros de Sistema 2, que sobresalen en el razonamiento riguroso, verificable y paso a paso.

La debilidad clave de los LLM —su propensión a las alucinaciones, su falta de razonamiento genuino y su poca fiabilidad — se deriva de su carencia de un componente de Sistema 2.

El objetivo de la IA neurosimbólica es crear una arquitectura híbrida que combine las fortalezas de ambos enfoques, uniendo la percepción y la intuición con la lógica y la verificación.

Pero si en el lado "simbólico" (Sistema-2) usamos lenguajes de "juguete" como la lógica de primer orden y tantos otros actuales, el resultado va a ser terriblemente desalentador: Será como intentar escribir El Quijote usando dos vocales y cinco consonantes ... ¡faltan piezas!

Sección 6. La Arquitectura del Verificador ULOGIC

La arquitectura neurosimbólica propuesta en torno a ULOGIC funciona como un bucle iterativo de generación y verificación:

1. **Entrada (Prompt):** Un usuario plantea un problema que requiere una demostración, un algoritmo o un procedimiento formal.
2. **LLM (Generador):** El LLM, actuando como un motor de "intuición" del Sistema 1, genera una solución candidata en forma de una expresión ULOGIC. No se espera que sea perfecta; es una hipótesis.
3. **ULOGIC (Verificador):** El motor ULOGIC, un razonador determinista del Sistema 2, recibe la expresión. No interpreta su "significado", sino que realiza una

comprobación puramente sintáctica para determinar si la expresión está bien formada según sus reglas. Este proceso es una forma de *verificación formal*.

4. Retroalimentación (Feedback):

- **Éxito:** Si la expresión es válida, se acepta como una demostración, algoritmo o procedimiento correcto.
- **Fallo:** Si la expresión es inválida, el verificador ULOGIC devuelve un informe de error estructural preciso, señalando la subexpresión exacta y la regla que se ha violado.

5. **Refinamiento:** El LLM recibe esta retroalimentación estructurada. Este feedback se convierte en parte del contexto para su siguiente intento, guiándolo para corregir su error. Este proceso iterativo continúa hasta que se genera una expresión válida.

Este bucle es un proceso de entrenamiento cognitivo. A través de miles de estas interacciones, el LLM se ajusta para generar expresiones ULOGIC válidas.

El LLM, al ser entrenado sobre ULOGIC, lenguaje con capacidades avanzadas y suficiente potencia expresiva, **aprende la intuición de cómo hacer demostraciones, algoritmos y procedimientos**; aprende a pensar con lógica y a crear razonamiento verificable, no meramente estadístico e imitativo.

La retroalimentación estructurada del verificador proporciona una señal de entrenamiento potente que va más allá de la simple predicción del siguiente token, moldeando el espacio de representación interno del LLM para que sea compatible con la lógica subyacente.

Crear esta arquitectura tiene sus desafíos tecnológicos (es necesario un orquestador que interconecte los dos sistemas, del lado del LLM el contenido se codifica en los embeddings vectoriales abstractos que usan las arquitecturas transformer, pero que representan TekDocs, documentos que contienen teoremas, definiciones, demostraciones, algoritmos, ejecuciones de algoritmos, TekDocs que son expresiones legibles por humanos, y almacenadas en bases de datos relacionales y de otro tipo). Suena complejo y lo es. Pero no es lo más complicado.

→ **El reto monumental** está en primero especificar el lenguaje ULOGIC (que una persona entiende porque sabemos movernos en terrenos pantanosos) hasta el grado de precisión suficiente que permita construir un Kernel-Verificador ←

Sección 7. De la Estadística Imitativa a la Cognición Verificable

El producto final de este sistema neurosimbólico no es el texto en lenguaje natural generado por el LLM, sino la *expresión ULOGIC verificada*.

Este resultado es fundamentalmente diferente del contenido generado por la IA tradicional y posee propiedades transformadoras:

- **Verificable:** Su corrección puede ser comprobada de forma mecánica y determinista por cualquier motor o Kernel-ULOGIC.
- **Fiable:** Se garantiza que está libre de contradicciones lógicas y alucinaciones, ya que estas serían detectadas por el verificador.
- **Explicable:** La propia expresión ULOGIC es la explicación. Una expresión-demostración es un recuento completo y estructural del razonamiento.
- **Componible:** Al ser una expresión formal, puede utilizarse de forma fiable como componente en construcciones más grandes y complejas.

Este enfoque aborda directamente las principales críticas a la IA generativa actual, produciendo un razonamiento que no solo es fluido, sino también correcto, transparente y digno de confianza.

Parte III:

TekDocs (Transportable Encapsulated Knowledge Document) Un Sustrato Global para el Conocimiento Reutilizable y Exacto

Esta última parte extrapola las consecuencias del sistema neurosimbólico, argumentando a favor de una revolución en la gestión y colaboración del conocimiento.

Sección 8. Trascendiendo el Cuello de Botella de la Representación del Conocimiento

El objetivo de la Web Semántica y la representación del conocimiento ha sido durante mucho tiempo crear conocimiento legible por máquinas. Las herramientas principales para este fin son las ontologías y los grafos de conocimiento.

Sin embargo, este campo se ha visto obstaculizado por el "cuello de botella de la adquisición de conocimiento". La creación y el mantenimiento manual de ontologías grandes y coherentes es un proceso extremadamente caro, lento y frágil.

El sistema ULOGIC-LLM ofrece una solución a este problema al automatizar la creación de conocimiento *formal y verificable*. Evita el cuello de botella manual al hacer que la IA realice el trabajo pesado de la formalización, guiada y corregida por el verificador ULOGIC.

Sección 9. La Arquitectura de la Red TekDocs

Un "TekDoc" (Transportable Encapsulated Knowledge Document) se define como una expresión ULOGIC verificada y autocontenida que representa una unidad o conjunto de conocimiento exacto (definiciones, argumentos, demostraciones, teoremas, algoritmos, ejecución de algoritmo, teorías completas...).

La visión es una red mundial de TekDocs que contendría los productos del razonamiento, algoritmos y definiciones, creando un ecosistema global de conocimiento reutilizable y exacto.

Las propiedades de esta red serían:

- **Reutilización:** Cualquier TekDoc puede ser referenciado e incorporado en la creación de un nuevo TekDoc, ya que todos comparten el mismo lenguaje.
- **Exactitud:** Cada pieza de conocimiento en la red está formalmente verificada. Se eliminan la ambigüedad y la contradicción.
- **Crecimiento Acumulativo:** La ciencia y la ingeniería pueden progresar de una manera verdaderamente acumulativa, construyendo sobre una base compartida y sólida de resultados verificados.

Esta red de TekDocs puede entenderse como un "**Git para el conocimiento formal**".

Antes de Git, la colaboración en el desarrollo de software era difícil. Git introdujo un modelo descentralizado donde el código se gestiona en unidades autocontenidas y versionadas (commits) que pueden ser fácilmente bifurcadas, fusionadas y compartidas, lo que condujo a una explosión en el desarrollo colaborativo de código abierto.

Hoy en día, el conocimiento científico y técnico se almacena en gran medida en documentos legibles por humanos (PDFs), que son ambiguos, no ejecutables y difíciles de integrar.

Los TekDocs funcionarían como "commits" de conocimiento formal: atómicos, verificables y con un claro grafo de dependencias.

La red de TekDocs funcionaría, por tanto, como un **GitHub para la ciencia y la ingeniería**. Un físico podría "bifurcar" un TekDoc de la Relatividad General, añadir un nuevo término y enviar una "solicitud de extracción" que sería verificada automáticamente en cuanto a su consistencia lógica. Un ingeniero podría "importar" un TekDoc para un algoritmo de ordenación, con la garantía de que es correcto, en el diseño de un sistema más grande. Esto aceleraría drásticamente la innovación y evitaría la constante "reinención de la rueda" y la propagación de errores que son comunes en la actualidad.

Conclusión: Un Cambio de Paradigma en el Pensamiento Formal y la Inteligencia Artificial

Las innovaciones fundacionales de ULOGIC —expresiones estructurales unificadas, definiciones no eliminables, semántica interna y autorreferencia segura— no son meras curiosidades teóricas.

Son el sustrato necesario para una nueva generación de IA neurosimbólica.

Este sistema, a su vez, no es un fin en sí mismo, sino una "máquina para producir exactitud": el motor que poblará la red de TekDocs, creando un universo intelectual común compartido, global, verificable y reutilizable.

ULOGIC se posiciona como un punto de inflexión potencial en la historia de la lógica y la computación.

Ofrece un camino hacia una IA verdaderamente inteligente y fiable, y al mismo tiempo sienta las bases para un futuro más riguroso y colaborativo para el conocimiento humano, abordando simultáneamente las crisis fundacionales que han limitado a ambos campos durante décadas.

Referencia Bibliográfica

Teoría de Modelos y Lógica Matemática

- Chang, C. C., & Keisler, H. J. (2012). *Model Theory (Third Edition)*. Dover Publications. *Un clásico absoluto y una referencia enciclopédica sobre la teoría de modelos, reeditado por Dover.*
- Enderton, H. B. (2001). *A Mathematical Introduction to Logic (Second Edition)*. Academic Press. *Considerado uno de los mejores textos introductorios a la lógica matemática, cubriendo los teoremas de completitud e incompletitud con gran claridad.*
- Hodges, W. (2008). *A Shorter Model Theory*. Cambridge University Press. *Una versión condensada de su monumental "Model Theory", ideal para una introducción de posgrado al campo.*

- Marker, D. (2002). *Model Theory: An Introduction*. Springer. Texto de referencia moderno en teoría de modelos, enfocado en las aplicaciones del campo en álgebra y geometría.
- Tarski, A. (1983). *Logic, Semantics, Metamathematics: Papers from 1923 to 1938*. Hackett Publishing Company. Colección de los trabajos fundamentales de Alfred Tarski donde desarrolla sus teorías sobre la verdad y la semántica formal.

Filosofía de las Matemáticas y los Teoremas de Gödel

- Goldstein, R. (2005). *Incompleteness: The Proof and Paradox of Kurt Gödel*. W. W. Norton & Company. Una aclamada biografía intelectual que explica de forma accesible la vida de Gödel y el profundo impacto de sus teoremas.
- Horsten, L. (2011). *The Tarskian Turn: Deflationism and Axiomatic Truth*. The MIT Press. Un libro de filosofía de la lógica que analiza en profundidad la teoría de la verdad de Tarski y sus implicaciones para los debates modernos.
- Linnebo, Ø. (2017). *Philosophy of Mathematics*. Princeton University Press. Una excelente y muy actual introducción a las principales corrientes de la filosofía de la matemática (platonismo, estructuralismo, logicismo) y sus problemas centrales.
- Shapiro, S. (2005). *The Oxford Handbook of Philosophy of Mathematics and Logic*. Oxford University Press. Una obra de referencia indispensable, con ensayos de los mayores especialistas sobre prácticamente todos los temas relevantes en la filosofía de la lógica y la matemática.
- Smith, P. (2013). *An Introduction to Gödel's Theorems (Second Edition)*. Cambridge University Press. Considerado el mejor libro para una introducción técnica pero guiada a las pruebas de los teoremas de incompletitud, ideal para estudiantes de filosofía, lógica y matemáticas.

Artículos sobre Arquitectura Neurosimbólica (arXiv)

- d'Avila Garcez, A. S., & Lamb, L. C. (2020). *Neurosymbolic AI: The 3rd Wave*. arXiv:2012.05876. Un artículo de perspectiva que posiciona a la IA Neurosimbólica como la "tercera ola" de la IA, describiendo sus principios y potencial.
- Mao, J., Gan, C., Kohli, P., Tenenbaum, J. B., & Wu, J. (2019). *The Neuro-Symbolic*

Concept Learner: Interpretable Reasoning, Compositional Concepts, and Robust Learning. arXiv:1904.12584. *Un trabajo influyente que presenta una arquitectura concreta para el aprendizaje de conceptos y el razonamiento visual de forma interpretable.*

- Sarker, M. M. K., et al. (2021). *Neuro-Symbolic Artificial Intelligence (AI): A new-age of AI.* arXiv:2105.05330. *Un artículo de revisión que ofrece una visión general de las motivaciones, desafíos y diferentes enfoques dentro del campo de la IA Neurosimbólica.*
- Yi, K., Wu, J., Gan, C., Torralba, A., Kohli, P., & Tenenbaum, J. B. (2018). *Neural-Symbolic VQA: Disentangling Reasoning from Vision and Language Understanding.* arXiv:1810.02338. *Propone una arquitectura para sistemas de pregunta-respuesta visual (VQA) que separa explícitamente los módulos neuronales (percepción) de los simbólicos (razonamiento).*
- Enhancing Large Language Models through Neuro-Symbolic Integration and Ontological Reasoning - arXiv, <https://arxiv.org/html/2504.07640v1>
- Design Patterns for LLM-based Neuro-Symbolic Systems - Neurosymbolic Artificial Intelligence, <https://neurosymbolic-ai-journal.com/system/files/nai-paper-787.pdf>
- Accelerating Knowledge Graph and Ontology Engineering with Large Language Models, <https://arxiv.org/html/2411.09601v1>

Contenidos accesibles de nivel introductorio e interesantes

1. Formal proof - Wikipedia, https://en.wikipedia.org/wiki/Formal_proof
2. Type theory - Wikipedia, https://en.wikipedia.org/wiki/Type_theory
3. Formal system - Wikipedia, fecha de acceso: julio 8, 2025, https://en.wikipedia.org/wiki/Formal_system
4. New Foundations - Wikipedia, https://en.wikipedia.org/wiki/New_Foundations
5. Self-reference - Wikipedia, <https://en.wikipedia.org/wiki/Self-reference>
6. Neuro-symbolic AI - Wikipedia, https://en.wikipedia.org/wiki/Neuro-symbolic_AI
7. Formal verification - Wikipedia, https://en.wikipedia.org/wiki/Formal_verification
8. Logical form - Wikipedia, https://en.wikipedia.org/wiki/Logical_form
9. Model theory - Wikipedia, https://en.wikipedia.org/wiki/Model_theory
10. Proof-theoretic semantics - Wikipedia, https://en.wikipedia.org/wiki/Proof-theoretic_semantics