Universidad
Rey Juan Carlos

Development and implementation of an ECM:
A generalist approach to user-AI interaction.

School: Escuela de Ingenieria de Fuenlabrada.
Degree: Robotics Software Engineering.

Author: Sebastián Mayorquín Posada.
Tutor: Julio Vega.

September 23, 2024

License

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Acknowledgements

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Abstract

[En proceso] Los ultimos lanzamientos de Modelos Grandes de Lenguaje (LLM por sus siglas en ingles), da paso a nuevas implementaciones de Inteligencias Artificiales Generalistas (AGI). Mas allá the optimizar modelos de machine learning, el desarrollo de AGI's requiere de aplicaciones cognitivas que habiliten a las LLM a operar de forma efectiva en aplicaciones del mundo real.

Esta memoria intruduce la Máquina de Congición-Ejecución(MCE) como un marco teórico que descompone el diseño de un AGI como un problema the aproximacion definido por 3 variables clave: El *espacio de ejecución*, textitespacio de cognición y *espacio de ejecución*. Como aproximación a este marco teórico se propone *[SystemName]* como un entorno de desarrollo para probar y diseñar AGI's implementando diferentes aproximaciones en multiples capas. Finalmente se implementan multiples técnicas incluyendo *Prompt Engineering*, codigo *Exelent*, o *Auto-Entrenamiento* con el fin de construir un AGI que soporte la interación usuario-IA en un entorno no controlado.

#TODO: Incluir en el abstract mencion a la implementacion en RaspBerry, y resultados

Acronyms

- **AI**: Artificial Intelligence
- **AP**: Agent Protocol
- A_1: Execution Layer Algorithm
- A_2: Cognition Layer Algorithm
- **LLM**: Large Language Model
- **ECM**: Execution-Cognition Machine
- **AGI**: Artificial General Intelligence
- **RAG**: Retrieval Augmented Generation
- **PMPA**: Profile-Memory-Plan-Action

Contents

1. Introducción

La palabra *"inteligencia artificial"* ha sido seleccionada por el diccionario Collins como palabra del año en el 2023. Miles de empresas ahora estan integrando tecnologías con *"IA"*. Del mismo modo, multiples medios de comunicación informan de los posibles problemas y riesgos de estas tecnologías en caso de no ser implementadas de forma apropiada. A pesar de su creciente popularidad en los ultimos años, el concepto de inteligencia artificial es dificil de definir y categorizar incluso dentro del sector cientifico.

Dado este contexto de creciente interés y debate, el presente capítulo abordará los fundamentos de la inteligencia artificial, definiendo sus características principales, su surgimiento y las interpretaciones formales mas extendidas. Este marco conceptual es clave para que el lector pueda comprender el estado del arte, que será tratado en profundidad en los capítulos siguientes.

1.1 Inteligencia Artificial

Las dos palabras acuñadas en el concepto de Inteligencia Artificial hacen referencia a la simbiosis de dos campos de estudio totalmente diferentes. Es tanto así, que en distintos idiomas esta composición lingüistica se mantiene constante (véase la terminología en ingles: *"Artificial Intelligence"*).

En este sentido, la *inteligencia* ha sido estudiada historicamente por ramas como la psicología, filosofía o educación donde se converge en multiples definiciónes que aunque distintas, resultan "intuitivamente" faciles de relacionar: la inteligencia hara referencia a la capacidad para entender, comprender o resolver problemas, al conjunto de habilidades cognitivas que incluyen la autoconciencia, creatividad o razonamiento logico. Fuera del ámbito cientifico resulta sencillo distinguir aquellas entidades que demuestran inteligencia de aquellas que no. Así pues, aunque puede conformarse un debate en torno a las capacidades intelectuales de dos especies de similar origen (como lo podría ser un delfín y un tiburon); es posible afirmar con certeza que una unidad morfológica simple como lo es una celula eucariota, es menos inteligente que un humano promedio.

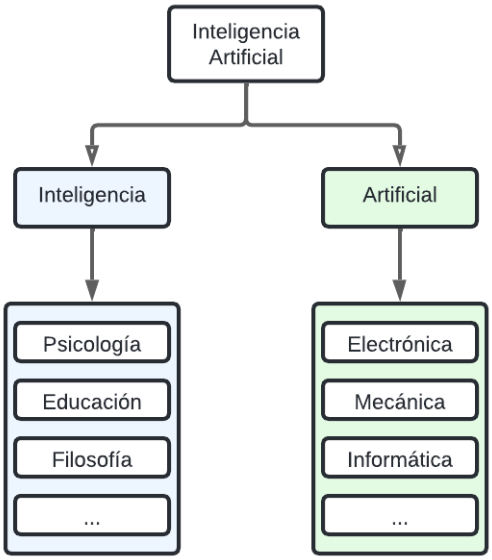

Figure 1.1: ”Inteligente” vs ”Artificial”. Elaboración Propia

De forma similar, todo aquello que es *artificial* hace referencia a aquello que no es natural, generalmente en relación con artefactos o productos creados con un proposito determinado. Mientras que un delfín se puede considerar producto de la naturaleza, un robot aspirador se considera artificial al ver que este es consecuencia de una composición de elementos electronicos diseñados y organizados por el ser humano con un fin concreto: limpiar el escenario en el que se encuentre.

Tras la Conferencia de Dartmouth organizada en 1956 por el matemático John McCarthy junto con otro investigadores como Marvin Misnky, se definieron por primera vez las bases de la inteligencia artificial:

> *”El estudio procederá sobre la base de la conjetura de que cada aspecto del aprendizaje o cualquier otra característica de la inteligencia puede, en principio, ser descrito con tal precisión que se pueda crear una máquina capaz de simularlo.” McCarthy et al. (2006)*

A lo largo de las decadas posteriores, el concepto de inteligencia artificial (IA) podrá definirse entorno a 3 interpretaciónes de distintos sectores:

- Desde la perspectiva de la IA clásica, desarrollada principalmente por los campos de las matemáticas e informática, la inteligencia artificial se concibe como un conjunto de algoritmos predefinidos, reglas de inferencia y

razonamientos lógicos que buscan tomar decisiones, realizar tareas complejas o resolver problemas de manera autónoma. El objetivo de la IA clásica es resolver tareas como encontrar el camino más corto entre dos puntos o seleccionar el movimiento con mayor probabilidad de éxito en un juego como el ajedrez.

- Desde la perspectiva del aprendizaje automático, la inteligencia artificial se define como una propiedad emergente de modelos matemáticos complejos, cuya interpretación no se basa en una secuencia de pasos predeterminada sino en las interacciones del modelo como conjunto. En este caso, de forma similar a como un cuadro artistico emerge como consecuencia del agrupamiento de trazos y colores adecuadamente posicionados, la IA emergera como resultado de las interacciones entre diferentes entidades matemáticas. [1]

- Desde la perspectiva del campo de la cognición (interpretación que utilizará de aquí en adelante), la inteligencia artificial es un sistema que replica habilidades como el razonamiento, la resolución de problemas, la toma de decisiones, el aprendizaje y la percepción de manera similar a las funciones realizadas por el cerebro humano. Dado este enfoque, el funcionamiento interno de la inteligencia no está explícitamente definido, pues se considera que la ”inteligencia” reside en el proceso de resolución de problemas y no en los datos que contiene. Es decir, la inteligencia se manifiesta en la metodologia con la que dicho sistema ”piensa”, y no únicamente en la información que posee.

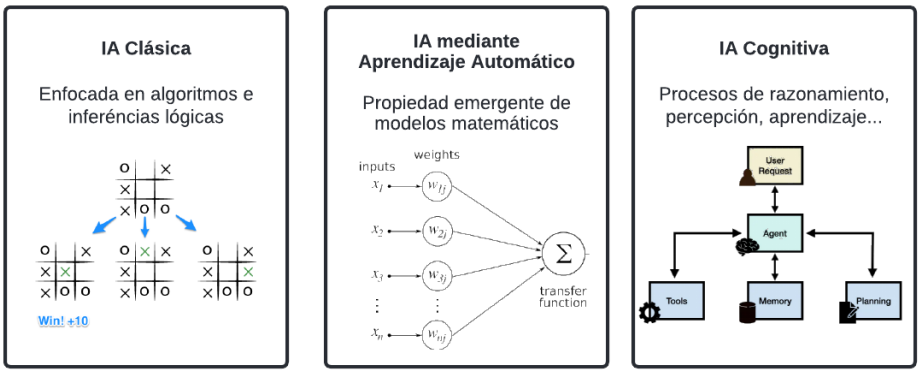

Figure 1.2: Interpretaciones de Inteligencia Artificial. Elaboración Propia

[1] Cabe señalar que, bajo este enfoque, la IA no puede ser controlada de forma determinista, ya que los modelos matemáticos no definen en su totalidad el comportamiento resultante de su aplicación.

El desarrollo de la inteligencia artificial ha dado paso a la investigación de multiples areás sobre lo que se consideraba ”inteligente” y como estos sectores podian ser automatizados, esto es, convertirlos en elementos artificiales. Entre las áreas de estudio destacadas encontramos investigaciones sobre la representación del conocimiento, la inferencia, o la resolución de problemas.

De entre todas estas areas, la resolución de problemas destaca aun a día de hoy como una de las propiedades mas dificiles de alcanzar. Esto se debe a que resolver un problema, a diferencia de otras tareas, implica resolver multiples subtareas que son faciles de para un humano pero a su vez dificilmente automatizables. Como ejemplo, la tarea de distinguir los comentarios negativos de los positivos en un foro de internet, resulta trivial de forma intuitiva, sin embargo matices sutiles del lenguaje como la ironia, gracia, emoción o exageración resultan imposibles de determinar en un conjunto de pasos que ejecuta un algoritmo o maquina.

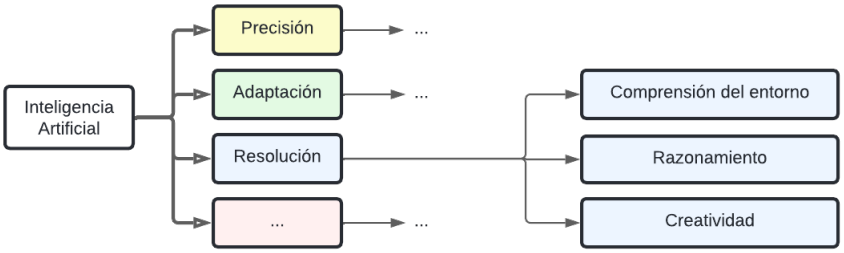

Figure 1.3: Requisitos de la IA. Elaboración Propia

1.2 Fundamentos de la IA Moderna

Entre otras arquitecturas de inteligencia artificial, los Modelos Grandes de Lenguaje (LLM por sus siglas en ingles), han destacado en los ultimos años por su capacidad de adaptarse a multiples aplicaciones. Utilizando el enfoque de inteligencia artificial basada en aprendizaje automático, estos modelos son capaces de generar imagenes, entender matices del lenguaje como la ironia, razonar matemáticamente o incluso generar modelos 3D a partir de descripciones de un objeto.

Independientemente de su estructura interna, podemos razonar que estos sistemas actuan de forma inteligente (acorde al razonamiento establecido en la sección 1.1). Por esta razón, muchas empresas han adoptado tecnologías de IA que les permiten ahorrar tiempo, esfuerzo y, en algunos casos, mejorar la calidad de sus tareas. Entre estas tareas automatizadas se encuentra la generación automática

de texto, donde la IA recibe un *prompt*, esto es, una entrada en formato de texto que contiene las instrucciones que debe seguir, y responde de forma automática. Esta respuesta puede utilizarse, por ejemplo, para actualizar contenido en redes sociales o resumir textos largos.

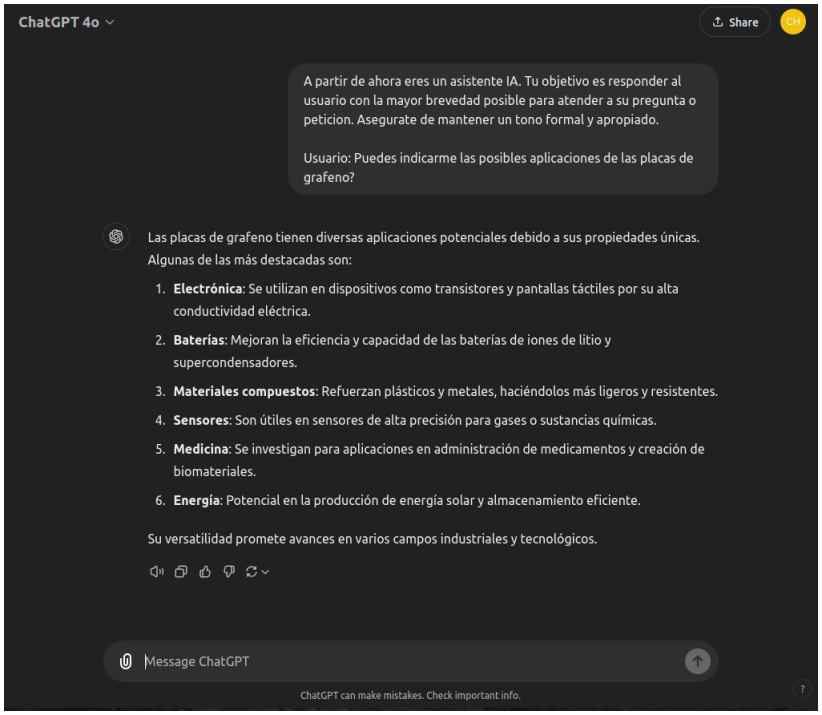

Figure 1.4: Interacción automatizada con usuario utilizando ChatGPT (modelo GPT-4o-mini). Elaboración Propia

Detrás de los modelos de lenguaje actuales subyacen arquitecturas avanzadas basadas en redes neuronales profundas. A diferencia de otras tecnologías, estas arquitecturas requieren un proceso intensivo de entrenamiento, optimización y refinamiento a través de múltiples capas, con el fin de garantizar tanto su robustez como su eficiencia. En este contexto, diferentes compañías proporcionan servicios basados en cómo han refinado y optimizado dichas arquitecturas.

Un ejemplo es OpenAI, que ofrece acceso a sus modelos GPT a través de su pagina web. Los modelos más recientes de esta serie destacan por su capacidad superior para la comprensión del lenguaje natural, razonamiento y control en tareas complejas. En contraste, la empresa Anthropic, ofrece proporciona acceso a sus modelos Claude Haiku, diseñados para ofrecer respuestas rápidas y con un enfoque en la optimización de costos, según lo informado en su plataforma en línea. Cada compañía, por tanto, implementa variaciones en el diseño y entrenamiento

de sus modelos, lo que conduce a distintas capacidades y enfoques en el mercado de modelos de lenguaje.

Para aprovechar en su totalidad el potencial de los modelos de lenguaje (LLMs), surge el concepto de Agente. Un agente es una tecnología diseñada para conectar a las LLMs con diversos campos de aplicación. En esencia, un agente contiene una inteligencia artificial capaz de razonar y tomar decisiones basadas en las posibles acciones a implementar. Posteriormente, utiliza herramientas preprogramadas en su arquitectura para traducir esos razonamientos en aplicaciones concretas.

Estos agentes pasan por diversas fases de razonamiento, comúnmente denominadas arquitecturas agénticas, que les permiten interactuar tanto con usuarios como con otros objetos o incluso agentes. A lo largo de este proceso, el agente evalúara la situación y ejecuta las acciones más adecuadas según las herramientas disponibles en su estructura. Un ejemplo de esta tecnología se encuentra en GEMINI ADVANCED, un agente desarrollado por Google. Este agente es capaz de procesar las solicitudes del usuario a través de comandos de voz en un dispositivo móvil, y ejecuta las acciones pertinentes para cumplir con la petición. La arquitectura de este agente le permite interpretar el lenguaje natural del usuario y, utilizando las herramientas integradas en su sistema, realizar las tareas necesarias de manera eficiente.

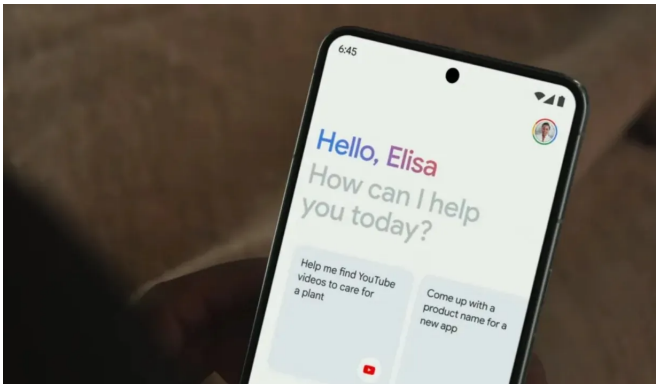

Figure 1.5: Interacción movil con Gemini Advance. Fuente: es.digitaltrends.com/.

En la presente memoria se propone implementar una Máquina de Cognición y Ejecución (MCE), esto es, una arquitectura agéntica que permita a una inteligencia artificial interactuar con su entorno de manera generalista, similar a GEMINI ADVANCED.

En el siguiente capítulo, se abordarán los principales conceptos y tecnologias relevantes para el diseño de arquitecturas agénticas. Se realizará una revisión de las

tecnologias actuales y sus capacidades cognitivas, analizando cómo se diferencian los agentes de propósito general de la inteligencia general artificial. Además, se explorarán las técnicas de comportamiento que permiten a los AGI's tomar decisiones de forma autónoma y eficiente. Finalmente, se examinará el estado actual de las arquitecturas cognitivas, enfocándose en sus limitaciones y los avances que han permitido una mayor capacidad de razonamiento y adaptabilidad en agentes de IA.

2. State of Art

2.1 IA Suave y Fuerte

Para definir la Inteligencia Artificial Searle (1980) distingue dos tipos de IA. La IA *suave*[1] hace referencia a un conjunto de algoritmos o herramientas que aproximan una solucion a un problema determinado utilizando estrategias basadas en analisis y reacción a traves de múltiples estados. En contraste, la IA *fuerte* esta caracterizada por sus capacidades para entender de forma profunda, razonar y reaccionar a estados impredecibles, adaptandose y aprendiendo a lo largo del proceso de resolución del problema.

Desde el diseño de las *redes neuronales* en 1949, nuevas tecnicas para el desarrollo de IA fuerte han estado en auge. Aunque la IA fuerte muestra propiedades aparentemente utópicas, muchos de los subproblemas que encierra su diseño han fomentado la evolución de multiples sectores. Como ejemplo de esto, el campo del aprendizaje automatic se ha expandido creando nuevos campos de estudio como el aprendizaje profundo. Será este nuevo campo es el responsable de la creación de los Modelos Grandes de Lenguaje (LLMs), donde modelos como GPT-4 (Achiam et al. (2023)), LLAMA2 o (Touvron et al. (2023)), o GEMINI (Saeidnia (2023)) utilizan el entrenamiento de millones de iteraciones sobre el conocimiento humano para recrear propiedades emergentes como lo son el entendimiento del sentido comun, habilidades de razonamiento y resolución de problemas, generación de respuestas con contexto, etc. Estos modelos aun estan en etapa de desarrollo, donde el objetivo apunta a propiedades como la multimodalidad[2], reconocimiento de patrones y/o explicabilidad, acercando a estos modelos a la integración de agentes cognitivos completos.

#TODO: En el parrafo, menciono 3 ejemplos de LLMs del estado del arte: GPT4, Llama2 y Gemini ¿Deberia explicar cada una por separado?

[1] Nótese que aunque similares, es importante no confundir IA suave con IA clásica.

[2] (Capacidad para trabajar con multiples tipos de entrada, como audios o imagenes.)

2.2 Agentes y Cognición

Estandarizando la salida de los modelos, es posible conseguir que los LLMs sean capaces de usar múltiples *herramientas*. En el ámbito de la IA, las herramientas son un conjunto de funciones o utilidades que permiten que el LLM interactúe con su entorno. Gracias a esto, mientras que un LLM solo puede generar una salida a partir de una consulta dada, los agentes traducen esas salidas en acciones que modifican el mundo y, opcionalmente, la propia configuración del agente.

> *"Los agentes son tan buenos como las herramientas que tienen."*
> (LangChain, 2024).

El uso de agentes se ha extendido en los últimos años, con una implementación notable siendo la *Generación Aumentada por Recuperación (RAG)*. Esta arquitectura se sustenta sobre un agente conectado a una herramienta que recolecta información de una base de datos, sustentando al LLM con conocimiento específico de un campo o empresa y mejorando su precisión en entornos controlados.

La aparición de agentes IA abre habilita la oportunidad de realizar más investigaciones en el campo de la cognición. La *cognición* es una rama de la informática que investiga el desarrollo de sistemas capaces de replicar propiedades de la inteligencia humana como el razonamiento, la percepción, la memoria, la planificación y la toma de decisiones. El cognitivismo clásico (la primera aproximación hacia los algoritmos cognitivos) se ha estudiado desde 1956; sin embargo, la relación interdependiente entre múltiples módulos necesarios para una arquitectura cognitiva ha planteado una limitación bloqueante en investigaciones previas.

> *"Tenemos círculos dentro de círculos. La dificultad central para la resolución de problemas eran los problemas de formulación limitada (PFL). Para enfrentarnos a ellos, recurrimos a la reestructuración. Para abordar esto, recurrimos a la analogía y en su núcleo encontramos la resolución de un PFL como un componente crucial. El análisis y la formalización se han visto seriamente frustrados en todo este esfuerzo." Vervaeke (1999)*

A partir de los intentos de definir alguna forma de inteligencia cognitiva, han surgido múltiples areas de estudio que evitan depender de un sistema completamente inteligente. Por ejemplo, la resolución de problemas se ha abordado utilizando IA clásica (Russell and Norvig, 2016), STRIPS (Fikes and Nilsson, 1971),

o PDDL (Aeronautiques et al., 1998). Aunque estos enfoques han sido exitosos en múltiples aplicaciones, la llegada de agentes de IA abre la puerta a retomar el estudio original de la cognición.

Gracias a los LLMs, es posible *romper el bucle* de la restricción de interdependencia al abordar el razonamiento como un problema de optimización computable. Será aqui donde los agentes de IA permitirán que los LLMs se integren en un sistema cognitivo, aportando una nueva aproximación a la cognición clasica. En línea con estas arquitecturas, se define un *agente cognitivo* como un agente del cual emergen capacidades cognitivas, y que es capaz de interactuar, aprender y modificar su comportamiento o entorno.

2.3 GPS vs AGI

El *General Problem Solver (GPS)* ha sido estudiado en el campo de la cognición desde 1959 con Newell et al. (1959), quien introdujo el GPS como un algoritmo hipotético o conjunto de técnicas que descomponen un problema en la ejecución de una secuencia de operadores que, combinados de la manera adecuada, pueden explorar multiples estados y subobjetivos del problema hasta alcanzar una solución válida. Aunque se lograron avances en el diseño de un GPS, las investigaciones sobre este tema se discontinuaron debido a las limitaciones de la IA clásica y la cognición descritas en la Sección 2.2.

Con propiedades similares al GPS, la *Inteligencia Artificial General (AGI)* es un nuevo enfoque basado en las técnicas de IA moderna y que preretende entender, aprender y aplicar conocimientos en cualquier tarea cognitiva, emulando las capacidades de un ser humano. Nótese que dado que la AGI se centra en replicar la inteligencia humana en un sentido más amplio que una exploración de estados, la capacidad de resolución de problemas emerge como parte de la generalización del conocimiento humano, donde se requiere un nuevo marco que interactúe con esta AGI para llevar esas capacidades a la realidad.

Es importante tener en cuenta que aunque la AGI no requiere necesariamente interacción con el entorno, este término es comúnmente utilizado por marcos que despliegan la AGI como su funcionalidad principal (otros nombres comunes incluyen IA autónoma, agentes AGI, etc.). En adelante, nos referiremos a estas arquitecturas como *despliegues AGI*.

"Demostrar que un sistema puede realizar un conjunto requerido de tar-

> *eas en un nivel de rendimiento dado debería ser suficiente para declarar al sistema como AGI; el despliegue de dicho sistema en el mundo abierto no debería ser inherente a la definición de AGI." Morris et al. (2023)*

Ha habido múltiples implementaciones de despliegues de AGI. Algunas se centran en especializar el conocimiento general en una aplicación específica, donde está encapsulado dentro de un agente cognitivo que guía a la IA informándola sobre el estado actual del entorno o las herramientas disponibles.

Un ejemplo de un despliegue especializado es VOYAGER. Este agente utiliza una arquitectura basada en tres módulos: el *currículo automático*, que describe el estado actual del agente y guarda información relevante; el *mecanismos de prompting iterativos*, que mantiene un bucle de retroalimentación entre las acciones codificadas por la IA y la retroalimentación obtenida; y la *biblioteca de habilidades*, que permite a la IA aprender y almacenar acciones previas y subobjetivos completados para fomentar la reutilización de herramientas. Utilizando esta arquitectura, VOYAGER demuestra la capacidad de jugar de manera autónoma al videojuego *Minecraft*, logrando múltiples objetivos solicitados.

> *"VOYAGER exhibe un rendimiento superior en el descubrimiento de nuevos ítems, desbloqueo del árbol tecnológico de Minecraft, recorrido de diversos terrenos y aplicación de su biblioteca de habilidades aprendidas a tareas no vistas en un mundo recién creado. VOYAGER sirve como punto de partida para desarrollar agentes generalistas potentes sin necesidad de ajustar los parámetros del modelo." Wang et al. (2023)*

Otro proyecto relevante es AUTOGPT. Este proyecto facilita el despliegue de agentes autónomos para tareas menores. Maneja la gestión de tareas, la selección de herramientas, múltiples técnicas de prompting y más. Para el despliegue de agentes, AUTOGPT proporciona lo que se conoce como la FORGE, que conecta automáticamente todos los mecanismos y servidores necesarios no solo para empezar a ejecutar el agente, sino también para proporcionar al usuario diversas herramientas para interactuar y depurar en tiempo real.

> *"AutoGPT utiliza el concepto de apilamiento para llamarse recursivamente a sí mismo [...], usando este método y con la ayuda de GPT 3.5 y GPT 4, crea proyectos completos iterando sobre sus propios prompts." Fezari and Ali-Al-Dahoud (2023)*

Al igual que AutoGPT, los despliegues de AGI se están extendiendo a proyectos para la asistencia de software en empresas (MetaGPT, Wu (2023)), despliegues autónomos (SuperAGI, TransformerOptimus (2023)), asistencia en el diseño-creatividad (AgentGPT, Reworkd (2023)), y más.

2.4 Técnicas de Comportamiento para AGI's

En secciones anteriores se ha discutido cómo las LLMs pueden ser utilizadas para construir y desplegar AGIs. Sin embargo, existen múltiples maneras de modificar o *tunear* el comportamiento de una IA para que pueda implementarse con éxito en agentes. A continuación, se destacan las técnicas clave en el estado del arte para construir agentes a partir de LLMs:

- **Tuneado (Fine-Tuning):** Todos los LLMs consisten en una o varias capas en una red neuronal basada en aprendizaje profundo. Estas capas contienen un conjunto de parámetros que definen el conocimiento o el comportamiento de la IA. Al entrenar un LLM ya estable con un conjunto de datos específicos de un campo, podemos mejorar la precisión de la IA con el conocimiento proporcionado y definir el formato y/o las directrices que debe seguir. Aunque este método obtiene mejores resultados que las siguientes técnicas, puede perder propiedades emergentes del LLM original y requiere un análisis previo del conjunto de datos proporcionado (sesgos, expectativas vs. resultados, limpieza de datos, etc.). En este campo, técnicas como el entrenamiento personalizado o el *congelamiento* de capas pueden ser usadas para mejorar los resultados del ajuste fino.
- **Ingeniería de Prompts (Prompting):** Sin modificar los parámetros del LLM, todavía podemos cambiar el comportamiento esperado introduciendo un *prompt* diseñado para un objetivo específico que la IA recibirá como entrada y utilizará como directrices. Aunque este método no garantiza una mejor precisión en comparación con el uso de fine-tunning, no altera los parámetros del modelo y permite la definición de estructuras de razonamiento más complejas. Es importante señalar que, aunque este método puede generar riesgos de seguridad debido a un mal comportamiento de la IA, técnicas como la Cadena de Pensamiento (Chain of Thought, CoT), la Generación Aumentada por Recuperación (RAG) o los *Few-Shot prompting* conducen a propiedades de razonamiento de alto nivel que no se han logrado con otros métodos. Sahoo et al. (2024) profundiza más en este campo.

- **Composición:** Al utilizar tanto la ingeniería de prompts como los métodos de fine-tunning, es posible optimizar múltiples agentes dividiendo el comportamiento esperado en varios subobjetivos. A estos agentes se les suele llamar *Expertos*, y reducen las alucinaciones de los LLMs distribuyendo la atención necesaria para completar una consulta dada en múltiples ejecuciones en lugar de una sola instancia. Algunos marcos de AGI, como AutoGen Wu et al. (2023), están completamente basados en este método.
- **Excitación de Características Interpretables:** Al usar LLMs, las características abstractas parecen tener una relación con patrones visibles dentro de los parámetros de la red neuronal. Estos parámetros pueden ser excitados (por ejemplo, aumentando la influencia de esos parámetros en la salida) para regular comportamientos relacionados con esa característica. La investigación realizada por Viteri et al. (2024) en este tema abre la puerta a usar este método en futuros diseños de agentes.

2.5 Arquitecturas Cognitivas

Independientemente de la metodología utilizada para construir un agente, la composicion de mecanismos, herramientas y agentes constituye una *arquitectura cognitiva*.

Usando IA clásica y aprendizaje por refuerzo, Laird (2019) introdujeron SOAR como una arquitectura cognitiva unificada que integra varias funciones cognitivas, como el aprendizaje, la memoria y la resolución de problemas, en un único marco. SOAR emplea un ciclo de decisión que implica proponer, evaluar y seleccionar operadores en función del estado actual.

En la investigación contemporánea, las arquitecturas cognitivas han evolucionado al incorporar más herramientas e integrar las capacidades de los LLMs como mecanismos principales. Wang et al. (2024) proporcionan una encuesta exhaustiva de las principales arquitecturas para despliegues de AGI en los últimos años, describiendo el modelo PMPA como un marco unificado que abarca todas las arquitecturas estudiadas.

El modelo PMPA aborda cuatro temas principales que deben ser implementados por la arquitectura cognitiva:

- *Perfil*: Define las principales directrices y reglas para los agentes implemen-

tados, orientados a los objetivos principales, la base de conocimientos y el comportamiento.

- *Memoria*: Define la información y los datos obtenidos del entorno del agente y establece una estructura y mecanismo para recuperar, codificar y clasificar el conocimiento relevante.
- *Planificación*: Define el mecanismo que permite al agente descomponer el objetivo principal en múltiples subobjetivos, emulando las capacidades de planificación humana.
- *Acción*: Define el mecanismo que conecta o traduce las órdenes solicitadas por el agente en un conjunto de herramientas que interactuarán con el entorno o comportamiento del agente.

Es importante señalar que, a diferencia de SOAR, este marco no prescribe métodos específicos para conectar cada módulo. En su lugar, describe las propiedades primarias que una arquitectura cognitiva debe poseer para ser viable en un despliegue de AGI. Además del modelo PMPA, existen otros marcos de AGI, como el Protocolo de Agentes-AI-Engineer-Foundation (2023)-, que proponen estándares alternativos de API, comportamiento y módulos. Sin embargo, se necesitan más avances en las arquitecturas cognitivas para establecer qué estándar será finalmente adoptado.

2.6 Estado del Arte en arquitecturas cognitivas

#TODO: Añadir información sobre arquitecturas más relevantes del estado del arte: ReAct, VerifyAgain y/o expectativas sobre Q* de OpenAI

3. Objectives

3.1 Problem Description

3.2 Requirements

3.3 Methodology

3.4 Workplan

4. Development Platform

4.1 Python

4.2 OpenAI (library)

4.3 Langchain

4.4 LangGraph

4.5 FastAPI and Requests

4.6 AgentProtocol

4.7 ROS2

4.8 PyAutoGUI and Pynput

4.9 OpenCV

5. Software Development

6. Hardware Development

Bibliography

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. (2023). Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Aeronautiques, C., Howe, A., Knoblock, C., McDermott, I. D., Ram, A., Veloso, M., Weld, D., Sri, D. W., Barrett, A., Christianson, D., et al. (1998). Pddl— the planning domain definition language. *Technical Report, Tech. Rep.*

AI-Engineer-Foundation (2023). Agentprotocol.

Fezari, M. and Ali-Al-Dahoud, A. A.-D. (2023). From gpt to autogpt: a brief attention in nlp processing using dl. *Preprint*.

Fikes, R. E. and Nilsson, N. J. (1971). Strips: A new approach to the application of theorem proving to problem solving. *Artificial intelligence*, 2(3-4):189–208.

Laird, J. E. (2019). *The Soar cognitive architecture*. MIT press.

LangChain (2024). Langchain documentation: Agents module. Accessed: 2024-07-05.

McCarthy, J., Minsky, M. L., Rochester, N., and Shannon, C. E. (2006). A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955. *AI magazine*, 27(4):12–12.

Morris, M. R., Sohl-Dickstein, J., Fiedel, N., Warkentin, T., Dafoe, A., Faust, A., Farabet, C., and Legg, S. (2023). Levels of agi: Operationalizing progress on the path to agi. *arXiv preprint arXiv:2311.02462*.

Newell, A., Shaw, J. C., and Simon, H. A. (1959). Report on a general problem solving program. In *IFIP congress*, volume 256, page 64. Pittsburgh, PA.

Reworkd (2023). Agentgpt.

Russell, S. J. and Norvig, P. (2016). *Artificial intelligence: a modern approach*. Pearson.

Saeidnia, H. R. (2023). Welcome to the gemini era: Google deepmind and the information industry. *Library Hi Tech News*, (ahead-of-print).

Sahoo, P., Singh, A. K., Saha, S., Jain, V., Mondal, S., and Chadha, A. (2024). A systematic survey of prompt engineering in large language models: Techniques and applications. *arXiv preprint arXiv:2402.07927*.

Searle, J. R. (1980). Minds, brains, and programs. *Behavioral and brain sciences*, 3(3):417–424.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. (2023). Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

TransformerOptimus (2023). Superagi.

Vervaeke, J. A. (1999). *The naturalistic imperative in cognitive science*. PhD thesis.

Viteri, S., Nanda, N., and Smith, J. (2024). Scaling monosemanticity in transformer circuits. Accessed: 2024-07-08.

Wang, G., Xie, Y., Jiang, Y., Mandlekar, A., Xiao, C., Zhu, Y., Fan, L., and Anandkumar, A. (2023). Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*.

Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., Chen, Z., Tang, J., Chen, X., Lin, Y., et al. (2024). A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345.

Wu, A. (2023). Metagpt.

Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., Jiang, L., Zhang, X., Zhang, S., Liu, J., Awadallah, A. H., White, R. W., Burger, D., and Wang, C. (2023). Autogen: Enabling next-gen llm applications via multi-agent conversation framework.