HCIA - AI

HUAWEI

Contents

Aspectos generales de la IA

HUAWEI

PREFACIO

- La humanidad está acogiendo la cuarta revolución industrial representada por la tecnología inteligente. Las nuevas tecnologías como la AI, la IoT, el 5G y la bioingeniería están integradas en todos los aspectos de la sociedad humana; impulsan cambios en las tendencias macro mundiales, como el desarrollo social sostenible y el crecimiento económico. Nueva energía cinética, mejora de ciudades inteligentes, transformación digital industrial, experiencia del consumidor, etc.
- Como proveedor líder mundial de infraestructura de TIC (información y comunicaciones) y terminales inteligentes, Huawei participa activamente en la transformación de la inteligencia artificial y propone la estrategia de Huawei de AI full-stack y de escenario completo. Este capítulo presentará principalmente Aspectos generales de AI, Campos Técnicos y Campos de Aplicación de AI, Estrategia de Desarrollo de AI de Huawei, Disputas de AI, Perspectivas Futuras de AI.

OBJETIVOS

Al finalizar este curso, usted podrá:

- Entender los conceptos básicos de la IA.
- Comprender las tecnologías de la IA y su historial de desarrollo.
- Comprender las tecnologías de aplicación y los campos de aplicación de la IA.
- Conocer la estrategia de desarrollo de IA de Huawei.
- Conocer las tendencias de desarrollo de la IA.

CONTENIDOS

1. **Aspectos generales de la IA**
2. Campos técnicos y campos de aplicación de la IA
3. Estrategia de desarrollo de IA de Huawei
4. Disputas de IA
5. Perspectivas futuras de la IA

IA a los ojos de la sociedad

- La gente conoce la IA a través de noticias, películas y aplicaciones reales en la vida diaria. ¿Qué es la IA a los ojos del público?

Haidian Park: Primer parque con tema de AI en el mundo
StarCraft II: AlphaStar vence a jugadores profesionales
Edmond de Belamy, pintura creada por AI, vendida en US$430,000
Demanda para programadores de AI: ↑ 35 Veces! Salario: Top 1!
50% de trabajo serán reemplazados por AI en el futuro
El invierno se acerca? AI enfrenta retos
…

Terminator
2001: Una odisea espacial
The Matrix
Yo, Robot
Blade Runner
Elle
Hombre Bicentenario
…

Verificación de seguridad de autoservicio
Evaluación del lenguaje hablado
Recomendación Música / Película
Altavoz inteligente
…

Noticias	Películas	Aplicaciones en la vida diaria
Aplicaciones de AI	Control de AI sobre los seres humanos	Protección de seguridad
Perspectivas de la industria de la AI	Enamorarse de AI	Entretenimiento
Retos a los que se enfrenta AI	Autoconciencia de la AI	Casa inteligente
…	…	Finanzas
		…

IA a los ojos de los investigadores

"Me propongo considerar la pregunta, ¿pueden pensar las máquinas?"

— Alan Turing 1950

La rama de la ciencia de la computación preocupada en hacer que las computadoras se comporten como humanos.

Juan McCarthy 1956

La ciencia de hacer que las máquinas hagan cosas que requerirían inteligencia si lo hacen los hombres.

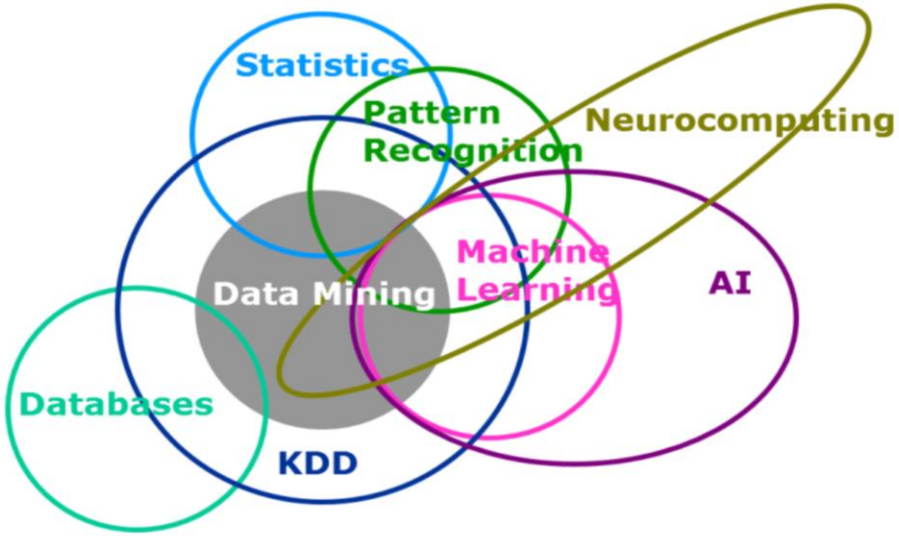
Marvin Minsky

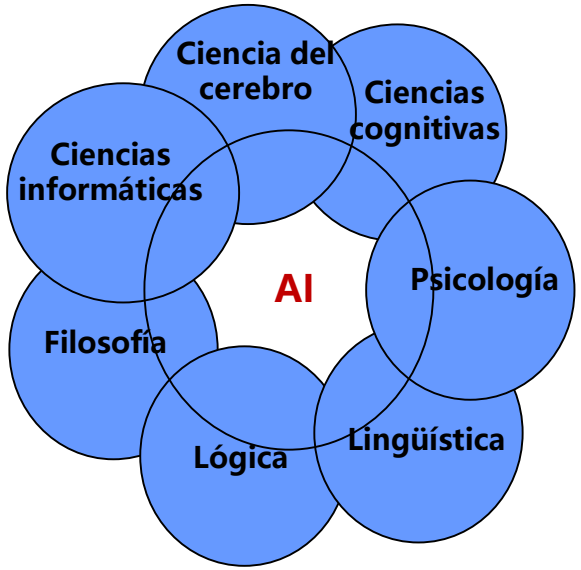
¿Qué son las Inteligencias?

- Múltiples Inteligencias de Howard Gardner
- Las inteligencias humanas se pueden dividir en siete categorías:
  - Verbal/Linguistica
  - Lógica/Matemática
  - Visual/Espacial
  - Corporal/Kinestésica
  - Musical/Rítmica
  - Inter-personal/Social
  - Intra-personal/Introspectiva

¿Qué es IA?

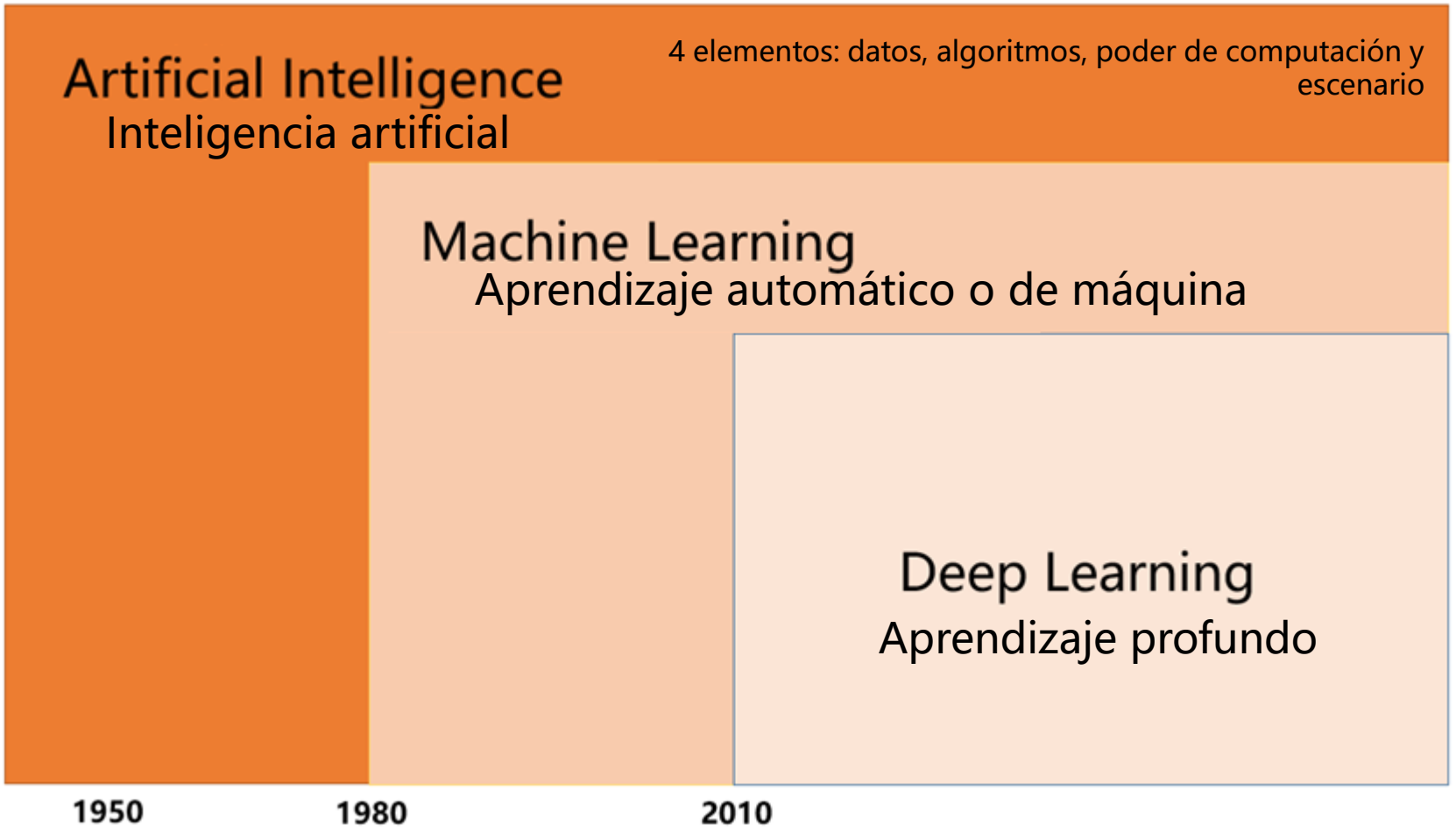
- Inteligencia artificial (IA) es una nueva ciencia técnica que estudia y desarrolla teorías, métodos, técnicas y sistemas de aplicación para simular y extender la inteligencia humana. En 1956, el concepto de AI fue propuesto por primera vez por John McCarthy, quien definió el tema como “la ciencia e ingeniería de fabricar máquinas inteligentes, especialmente programas informático inteligente". AI está preocupado en hacer que las máquinas funcionen de una manera inteligente, similar a la forma en que trabaja la mente humana. Actualmente, la AI se ha convertido en un curso interdisciplinario que involucra diversos campos.





Identificación de conceptos relacionados con la AI y el aprendizaje máquina
Informe de Desarrollo de la AI 2020

Relación de IA, aprendizaje de máquina y aprendizaje profundo



Relación de IA, aprendizaje máquina y aprendizaje profundo

- IA: Una nueva ciencia técnica que se centra en la investigación y desarrollo de teorías, métodos, técnicas y sistemas de aplicación para simular y extender la inteligencia humana.
- Aprendizaje de máquina: Un campo de investigación central de la AI. Se centra en el estudio de cómo las computadoras pueden obtener nuevos conocimientos o habilidades mediante la simulación o realización de un comportamiento de aprendizaje de seres humanos, y reorganizar la arquitectura del conocimiento existente para mejorar su rendimiento. Es uno de los campos de investigación central de la IA.
- Aprendizaje profundo: Un nuevo campo de aprendizaje máquina. El concepto de aprendizaje profundo se origina de la investigación sobre redes neurales artificiales. El perceptrón multicapa (MLP) es un tipo de arquitectura de aprendizaje profundo. El aprendizaje profundo tiene como objetivo simular el cerebro humano para interpretar datos tales como imágenes, sonidos y textos.

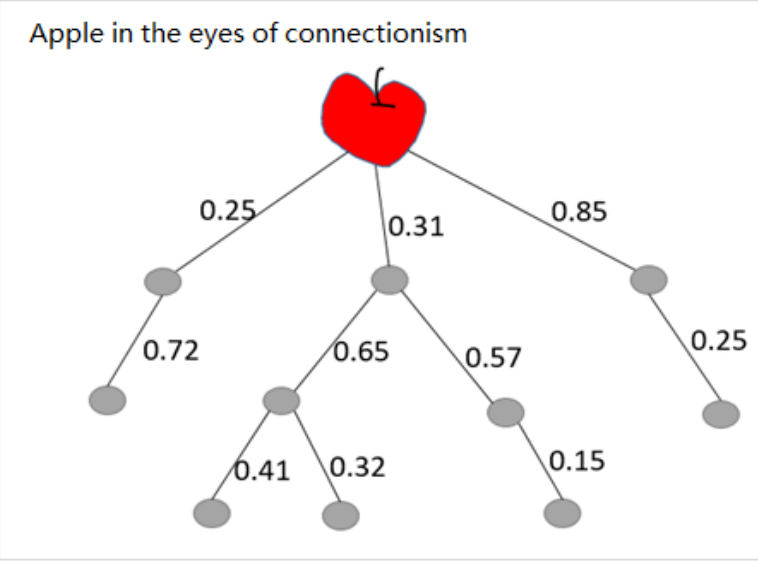
Tres principales escuelas de pensamiento: Simbolismo

- Pensamientos básicos
  - El proceso cognitivo de los seres humanos es el proceso de inferencia y operación de varios símbolos.
  - Un ser humano es un sistema de símbolos físicos, al igual que una computadora. Las computadoras, por lo tanto, pueden ser utilizadas para simular el comportamiento inteligente de los seres humanos.
  - El núcleo de la AI reside en la representación del conocimiento, la inferencia del conocimiento y la aplicación del conocimiento. El conocimiento y los conceptos pueden ser representados con símbolos. La cognición es el proceso de procesamiento de símbolos mientras que la inferencia se refiere al proceso de resolución de problemas utilizando el conocimiento heurístico y la búsqueda.
- Representante del simbolismo: inferencia, incluida la inferencia simbólica y la inferencia automática

Tres principales escuelas de pensamiento: Conexionismo

- Pensamientos básicos
  - La base del pensamiento son las neuronas más que el proceso de procesamiento de símbolos.
  - Los cerebros humanos varían de las computadoras. Se propone un modo de trabajo de computadora basado en el conectismo para reemplazar el modo de trabajo de computadora basado en una operación simbólica.

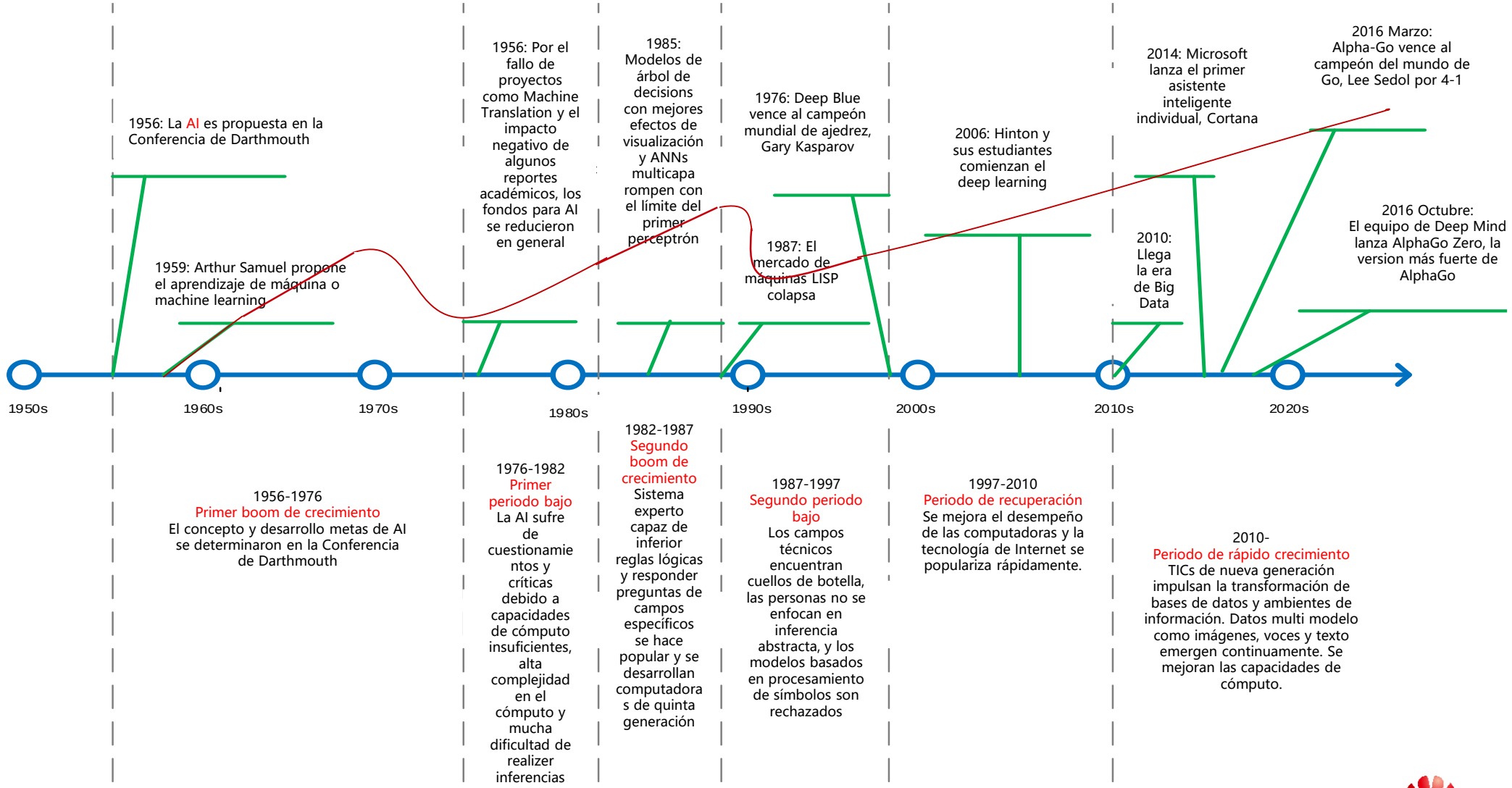
    - Representante del conexionismo: redes neurales y aprendizaje profundo



Tres principales escuelas de pensamiento: el comportamiento

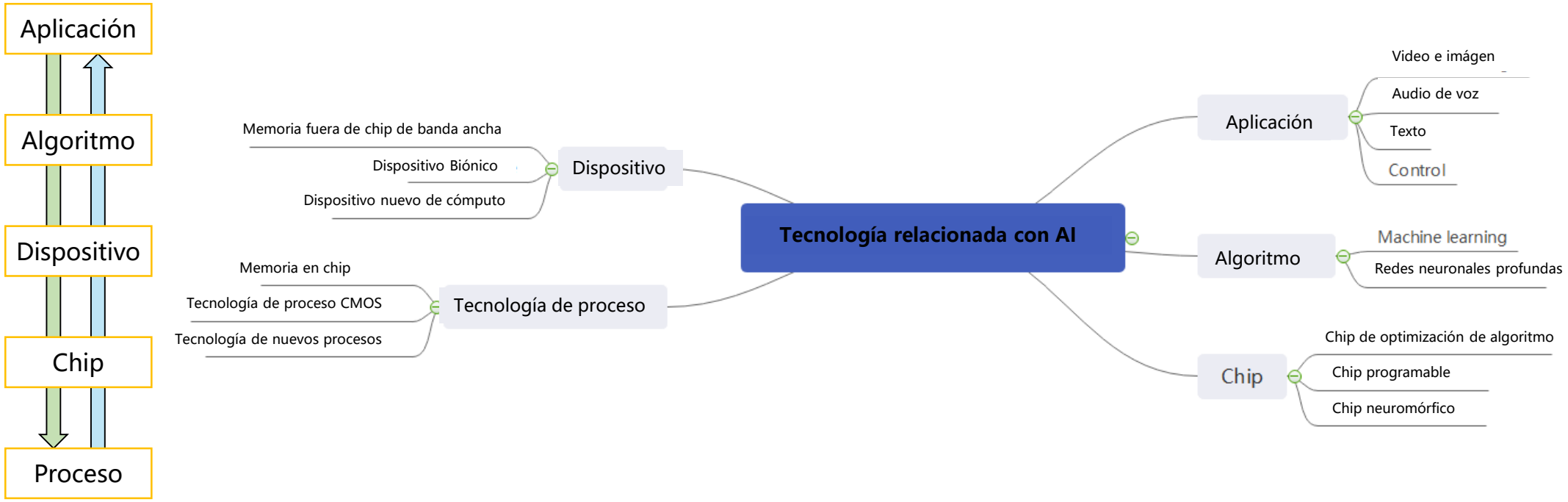
- Pensamientos básicos:
  - La inteligencia depende de la percepción y la acción. Se propone el modo de percepción-acción del comportamiento inteligente.
  - La inteligencia no requiere conocimiento, representación o inferencia. AI puede evolucionar como la inteligencia humana. El comportamiento inteligente sólo puede demostrarse en el mundo real a través de la interacción constante con el entorno circundante.
- Representante del comportamiento: control del comportamiento, adaptación y computación evolutiva

Breve historial de desarrollo de la IA



Descripción general de las tecnologías de la IA

- Las tecnologías de IA son múltiples capas, cubren las capas de aplicación, mecanismo de algoritmo, cadena de herramientas, dispositivo, chip, proceso y materiales.



Tipos de IA

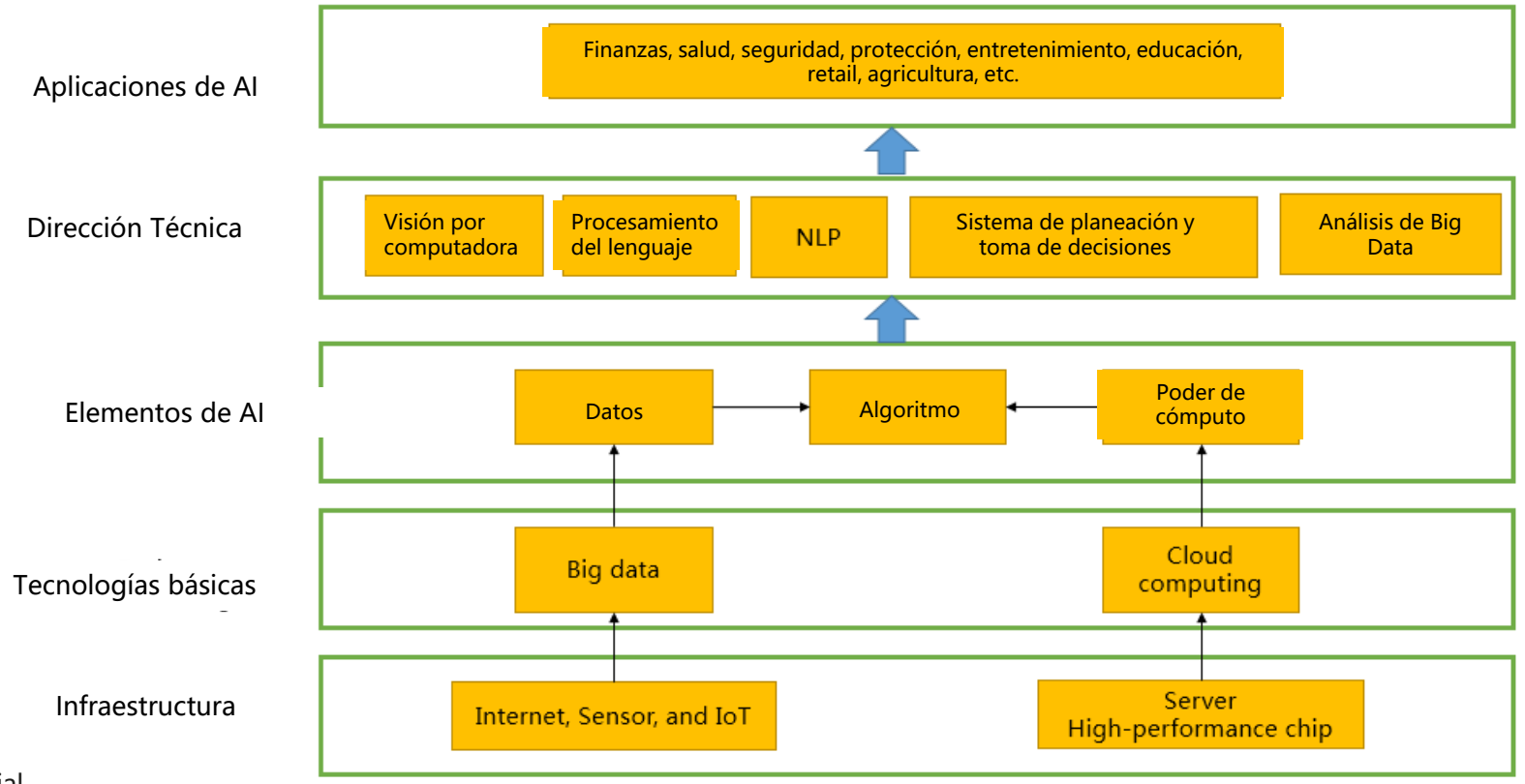
- IA Fuerte
  - La vision de IA fuerte sostiene que es posible crear máquinas inteligentes que realmente puedan razonar y resolver problemas. Tales máquinas se consideran conscientes y autoconscientes, pueden pensar independientemente sobre los problemas y buscar soluciones óptimas a los problemas, tienen su propio sistema de valores y visiones del mundo, y tienen todos los mismos instintos que los seres vivos, tales como las necesidades de supervivencia y seguridad. Puede ser considerado como una nueva civilización en cierto sentido.
- IA Débil
  - La visión de la IA débil sostiene que las máquinas inteligentes realmente no pueden razonar y resolver problemas. Estas máquinas sólo parecen inteligentes, pero no tienen inteligencia real o autoconciencia.

Clasificación de robots inteligentes

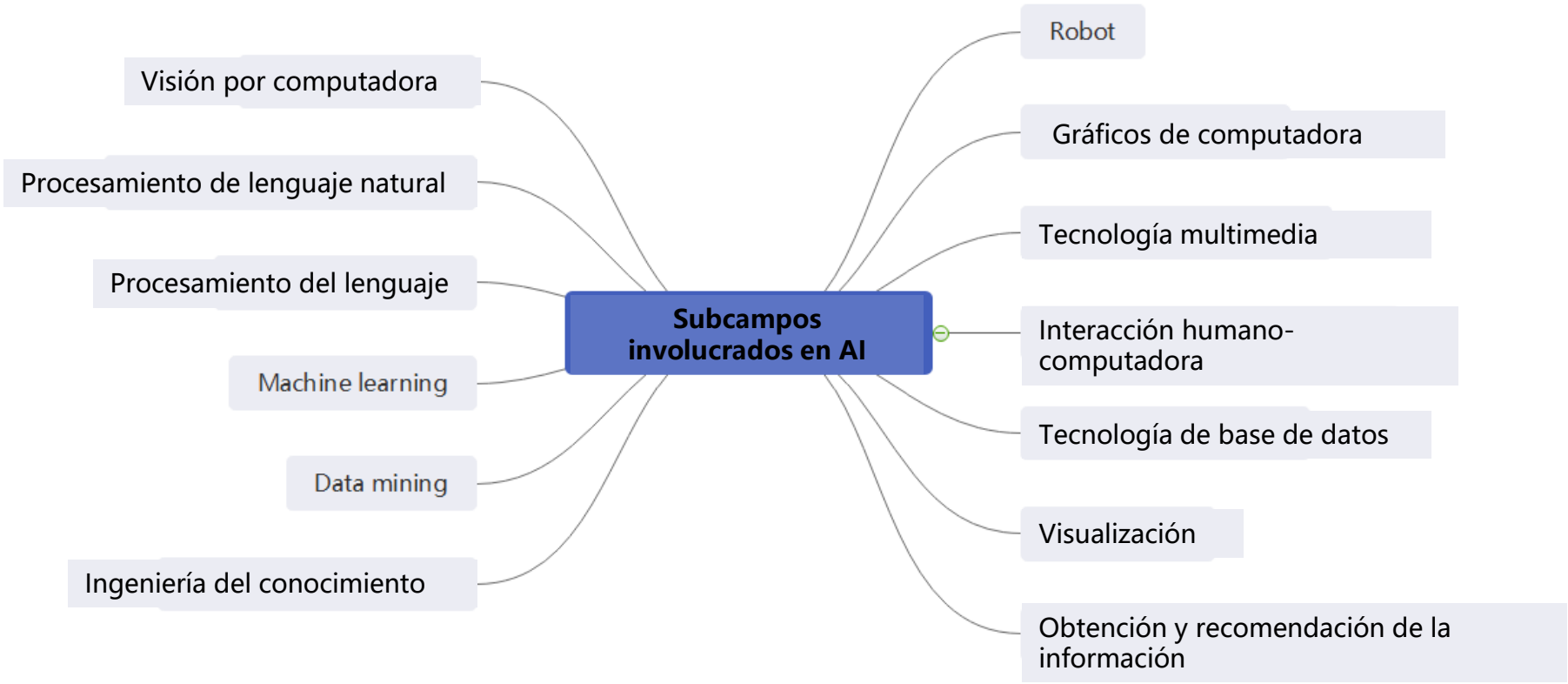
- Actualmente, no hay una definición unificada de investigación de IA. Los robots inteligentes generalmente se clasifican en los cuatro tipos siguientes:
  - "Pensar como seres humanos": IA débil, como Watson y AlphaGo
  - "Actuar como seres humanos": IA débil, como robot humanoide, iRobot y Atlas de Boston Dynamics
  - "Pensar racionalmente": IA fuerte (Actualmente, no se han creado robots inteligentes de este tipo debido al cuello de botella en la ciencia del cerebro.)
  - "Actuar racionalmente": IA fuerte

Ecosistema de la industria IA

- Los cuatro elementos de la IA son los datos, el algoritmo, la potencia informática y el escenario. Para satisfacer los requerimientos de estos cuatro elementos, necesitamos combinar la IA con la computación en la nube, Big Data e IoT para construir una sociedad inteligente.



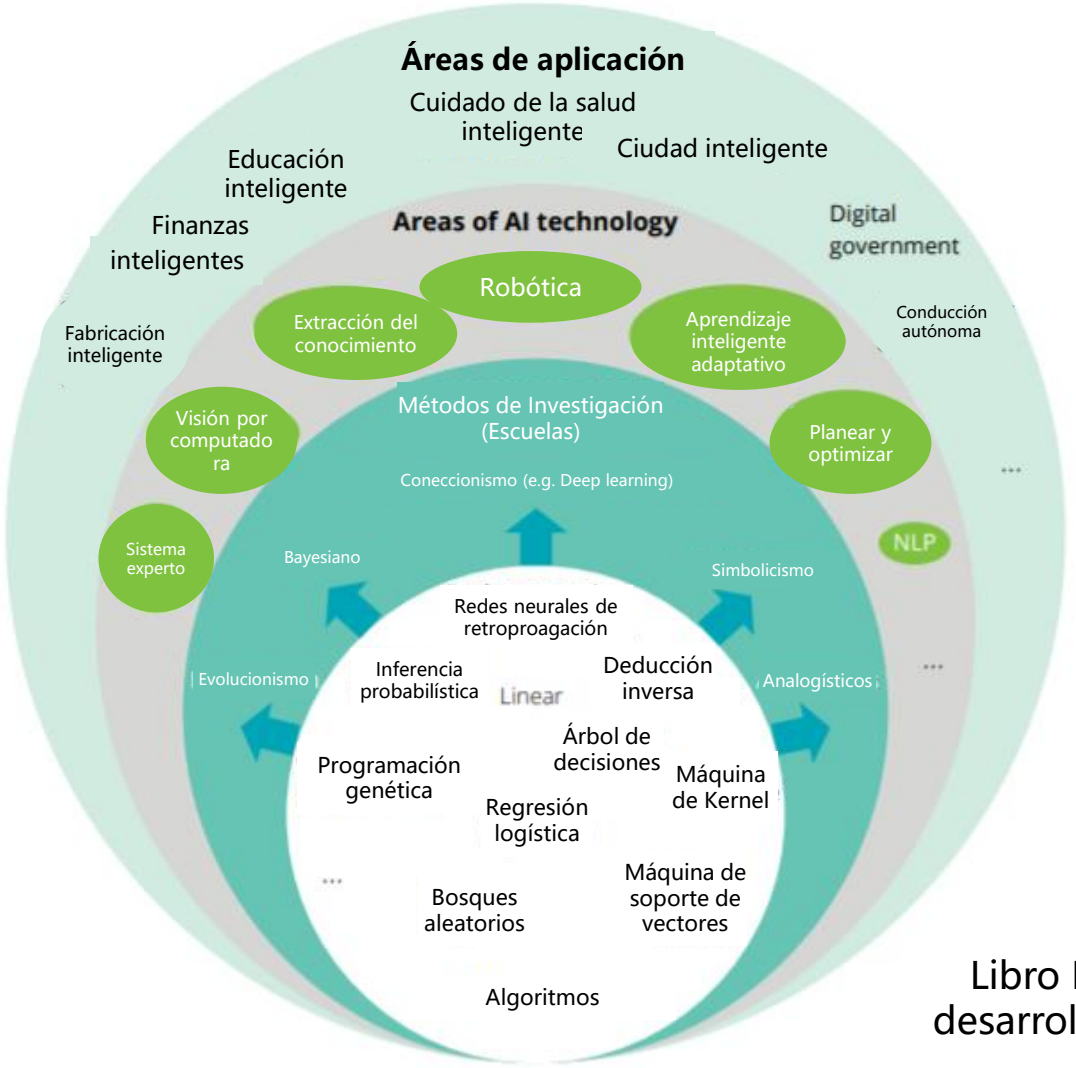
Subcampos de IA



Informe de Desarrollo de la AI 2020

CONTENIDOS

Campos técnicos y campos de aplicación de la IA

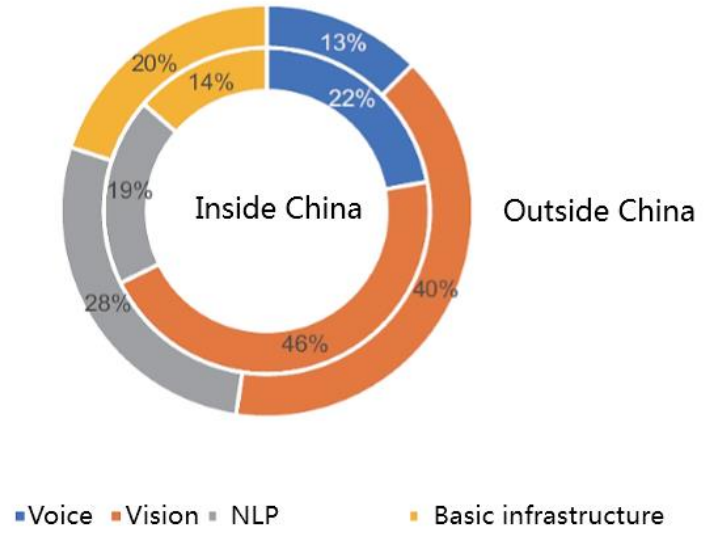


Libro Blanco sobre el desarrollo global de la AI 2020

HUAWEI

Distribución de tecnologías de aplicaciones de IA en empresas dentro y fuera de China

- En la actualidad, las direcciones de aplicación de las tecnologías de IA incluyen principalmente:
  - **Visión informática:** una ciencia de cómo hacer que las computadoras "ver"
  - **Procesamiento de voz:** término general para varias tecnologías de procesamiento utilizadas para investigar el proceso de voz, características estadísticas de las señales de voz, reconocimiento del habla, síntesis del habla basada en máquinas y percepción del habla
  - **Procesamiento de lenguaje natural (NLP):** un tema que utiliza tecnologías informáticas para entender y utilizar el lenguaje natural



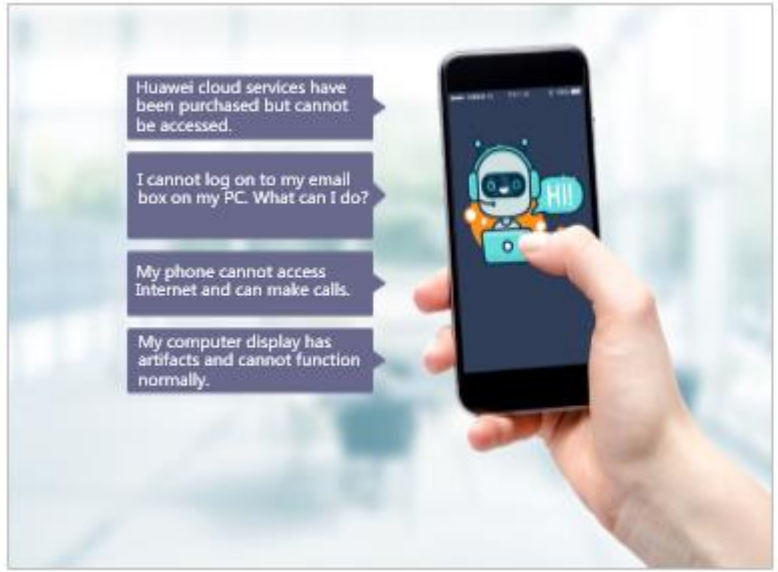
Distribución de tecnologías de aplicaciones de IA en empresas dentro y fuera de China

Informe de Desarrollo de la AI de China 2018

Escenario de aplicación de procesamiento de voz (1)

- Los temas principales de la investigación de procesamiento de voz incluyen el reconocimiento de voz, síntesis de voz, despertar de voz, reconocimiento de huellas vocales y detección de incidentes basada en audio. Entre ellos, la tecnología más madura es el reconocimiento de voz. En cuanto al reconocimiento de campo cercano en un ambiente bastante interior, la exactitud de reconocimiento puede alcanzar el 96%.
- Escenarios de aplicación:

Bot de respuesta de preguntas (QABot)



Navegación por voz

Escenario de aplicación de procesamiento de voz (2)

Educación inteligente

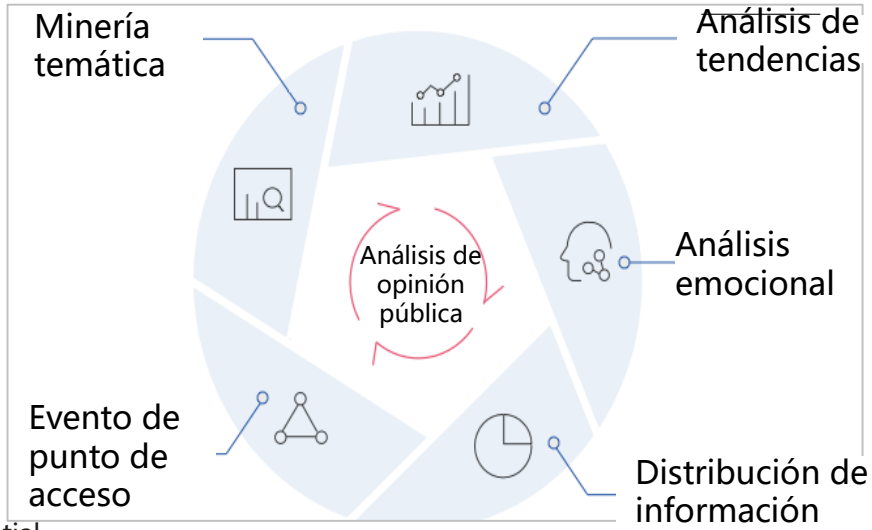
Registros de conferencia en tiempo real

- Otras aplicaciones:
  - Evaluación del lenguaje hablado
  - Robot de diagnóstico
  - Reconocimiento de huella vocal
  - Caja de sonido inteligente
  - ...

Escenario de aplicación de Procesamiento de Lenguaje Natural (NLP) (1)

- Los temas principales de la investigación NLP incluyen traducción automática, minería de texto y análisis de sentimiento. NLP impone altos requisitos a las tecnologías pero enfrenta una baja madurez tecnológica. Debido a la gran complejidad de la semántica, es difícil alcanzar el nivel de comprensión humana usando la computación paralela basada en Big Data y la computación paralela sólo.
- En el futuro, NLP logrará más crecimiento: comprensión de la semántica superficial → extracción automática de características y comprensión de la semántica profunda; inteligencia híbrida de un solo propósito (ML) → (ML, DL y RL)
- Escenarios de aplicación:

Análisis de opinión pública

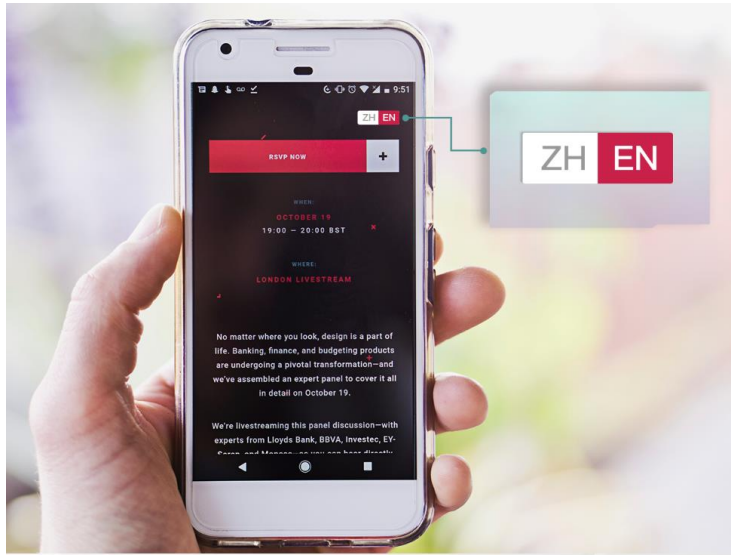


Análisis de evaluación

HUAWEI

Escenario de aplicación de Procesamiento de Lenguaje Natural (2)

Traducción automática



Clasificación de texto

- Otras aplicaciones:
  - Gráfico de conocimientos
  - Escritura de copy (copywriting) inteligente
  - Subtítulo de vídeos
  - ...

Campo de aplicación de IA - Atención de la salud inteligente

- **Minería de medicamentos:** desarrollo rápido de medicamentos personalizados por asistentes de AI
- **Gestión de la salud:** nutrición y gestión de la salud física / mental
- **Gestión hospitalaria:** servicios estructurados relativos a los registros médicos (centro)
- **Asistencia a la investigación médica:** asistencia a investigadores biomédicos en investigación
- **Asistente virtual:** registros médicos electrónicos de voz, guía inteligente, diagnóstico inteligente y recomendación de medicamentos
- **Imagen médica:** reconocimiento de imágenes médicas, marcado de imágenes y reconstrucción de imágenes 3D
- **Asistencia para el diagnóstico y el tratamiento:** robot de diagnóstico
- **Previsión del riesgo de enfermedad:** pronóstico del riesgo de enfermedad basado en la secuenciación de genes

Campo de aplicación de IA: hogar inteligente

- Basado en tecnologías de IoT, un ecosistema doméstico inteligente se forma con hardware, software y plataformas de nube, proporcionando a los usuarios servicios de vida personalizados y haciendo que la vida hogareña sea más conveniente, cómoda y segura.

Controlar los productos de hogar inteligente con procesamiento de voz tales como ajuste de temperatura del aire acondicionado, control del interruptor de cortina y control de voz en el sistema de iluminación.

Desarrollar perfiles de usuario y recomendar contenido a los usuarios con la ayuda de tecnologías de aprendizaje maquinario y aprendizaje profundo y basado en registros históricos de altavoces inteligentes y televisores inteligentes.

Campo de aplicación de IA – Industria Minorista

- AI traerá cambios revolucionarios en la industria minorista. Un síntoma típico son los supermercados no tripulados. Por ejemplo, Amazon Go, supermercado no tripulado de Amazon, utiliza sensores, cámaras, visión de computadora y algoritmos de aprendizaje profundo para cancelar completamente el proceso de factura, permitiendo a los clientes recoger bienes y "sólo salir".
- Uno de los mayores desafíos para el supermercado no tripulado es cómo cobrar las tarifas adecuadas a los clientes adecuados. Hasta ahora, Amazon Go es el único caso de negocio exitoso e incluso este caso implica muchos factores controlados. Por ejemplo, sólo los miembros Prime pueden ingresar a Amazon Go. Otras empresas, para seguir el ejemplo de Amazon, tienen que construir primero su sistema de membresía.

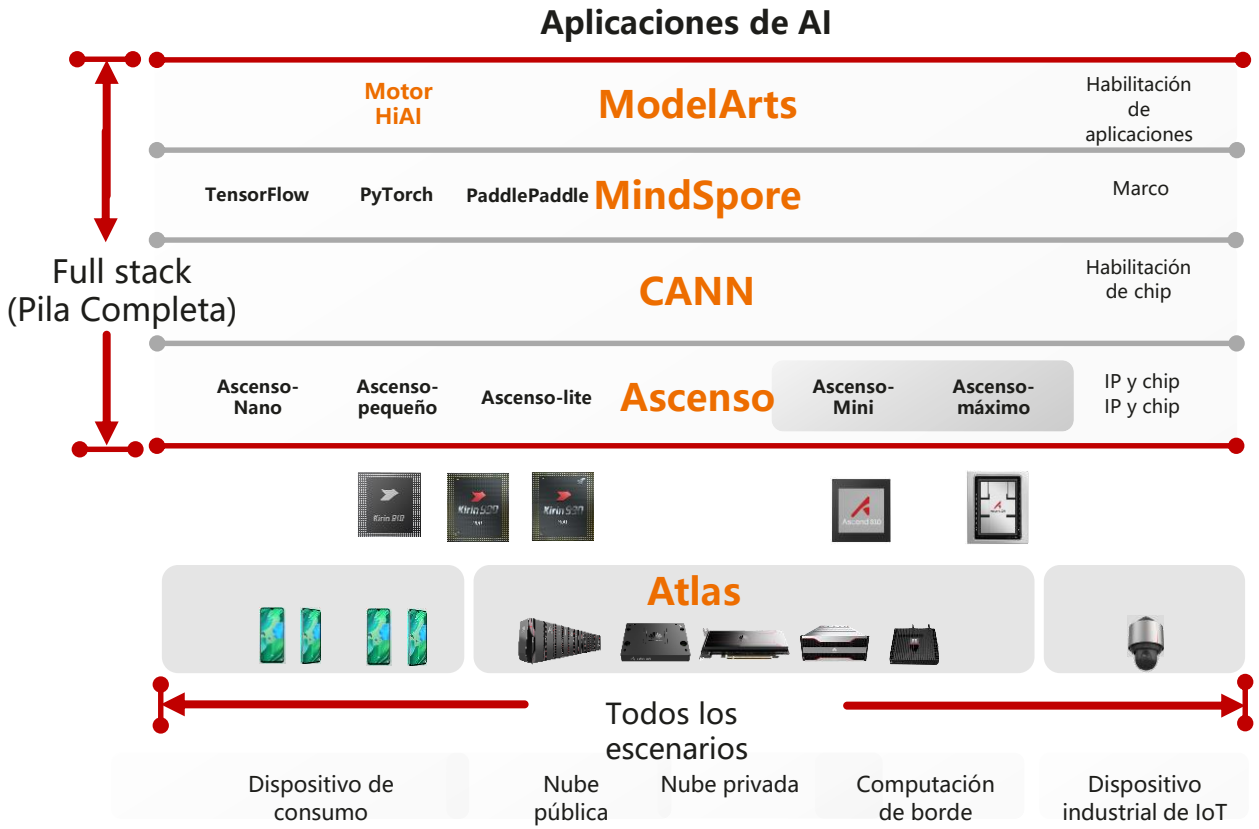
Campo de aplicación de IA - Conducción autónoma

- La Sociedad de Ingenieros Automotrices (SAE) en los Estados Unidos define 6 niveles de automatización de conducción que van desde 0 (totalmente manual) a 5 (totalmente autónomo). L0 indica que la conducción de un vehículo depende completamente sobre la operación del conductor. El sistema superior a L3 puede implementar la operación de entrega del conductor en casos específicos, L5 depende del sistema cuando los vehículos conducen en todos los escenarios.
- Actualmente, sólo algunos modelos de vehículos de pasajeros comerciales, como Audi A8, Tesla y Cadillac, soportan los sistemas avanzados de asistencia al conductor (ADAS) L2 y L3. Se estima que para 2020, más modelos de vehículos L3 emergerán con la mejora adicional de sensores y procesadores montados en vehículos. Se espera que la conducción autónoma L4 y L5 se implemente primero en vehículos comerciales en campus cerrados. Una gama más amplia de vehículos de pasajeros requiere una conducción autónoma avanzada, lo que requiere mayor mejora de tecnologías, políticas e infraestructura. Se estima que la conducción autónoma L4 y L5 estará apoyada por carreteras comunes en 2025-2030.

CONTENIDOS

1. Aspectos generales de la IA
2. Campos técnicos y campos de aplicación de la IA
3. **Estrategia de desarrollo de IA de Huawei**
4. Disputas de IA
5. Perspectivas futuras de la IA

Portafolio de IA de Full Stack y todo-escenario de Huawei



Habilitación de aplicaciones: provee servicios de extremo a extremo (ModelArts), API en capas y soluciones preintegradas.

MindSpore: soporta el marco de entrenamiento e inferencia unificado que es independiente del dispositivo, borde y nube.

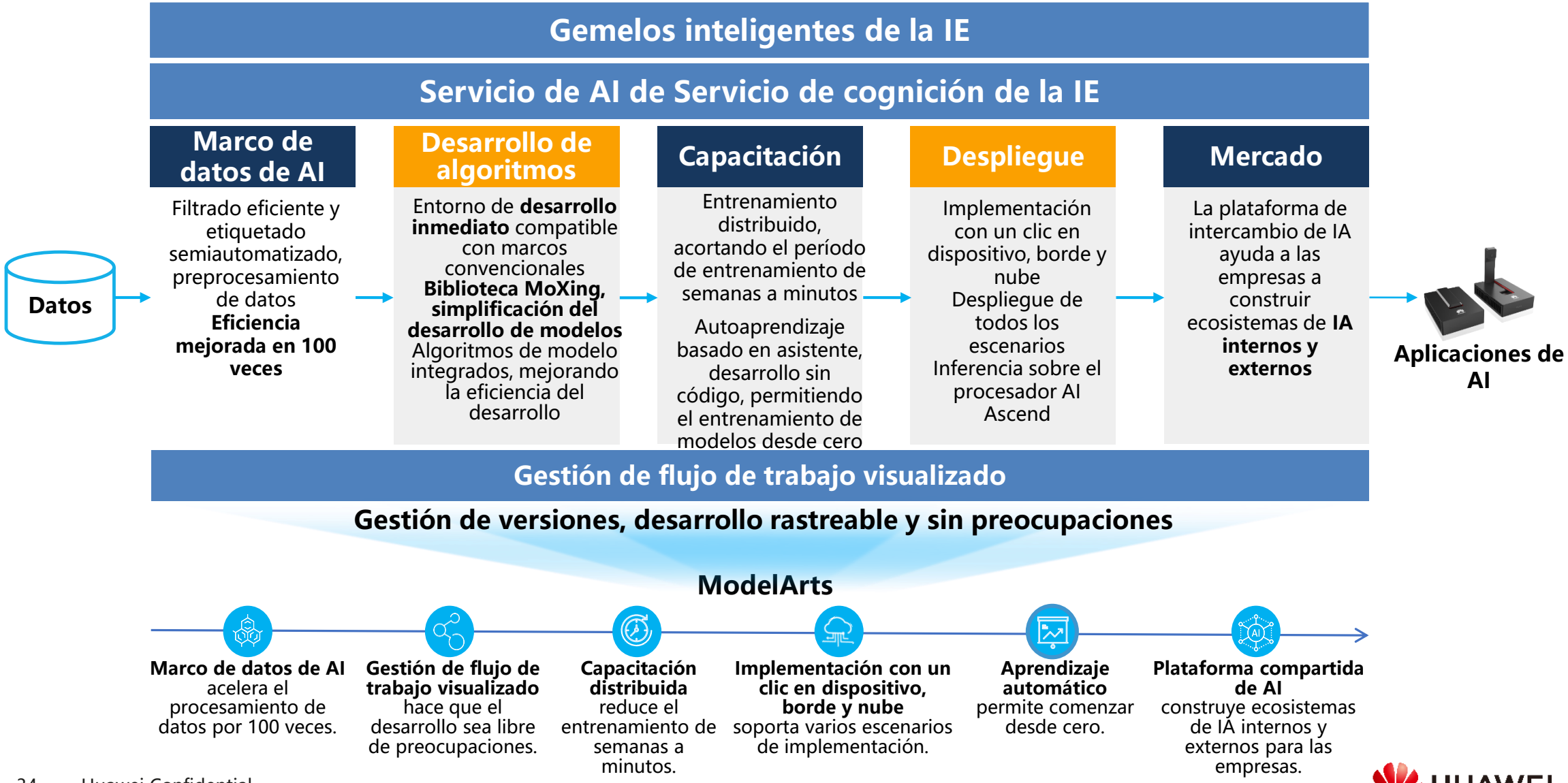
CANN: una biblioteca de operador de chips y herramienta de desarrollo de operadores altamente automatizado.

Ascend: proporciona una serie de IPs y chips de NPU basados en una arquitectura unificada y escalable.

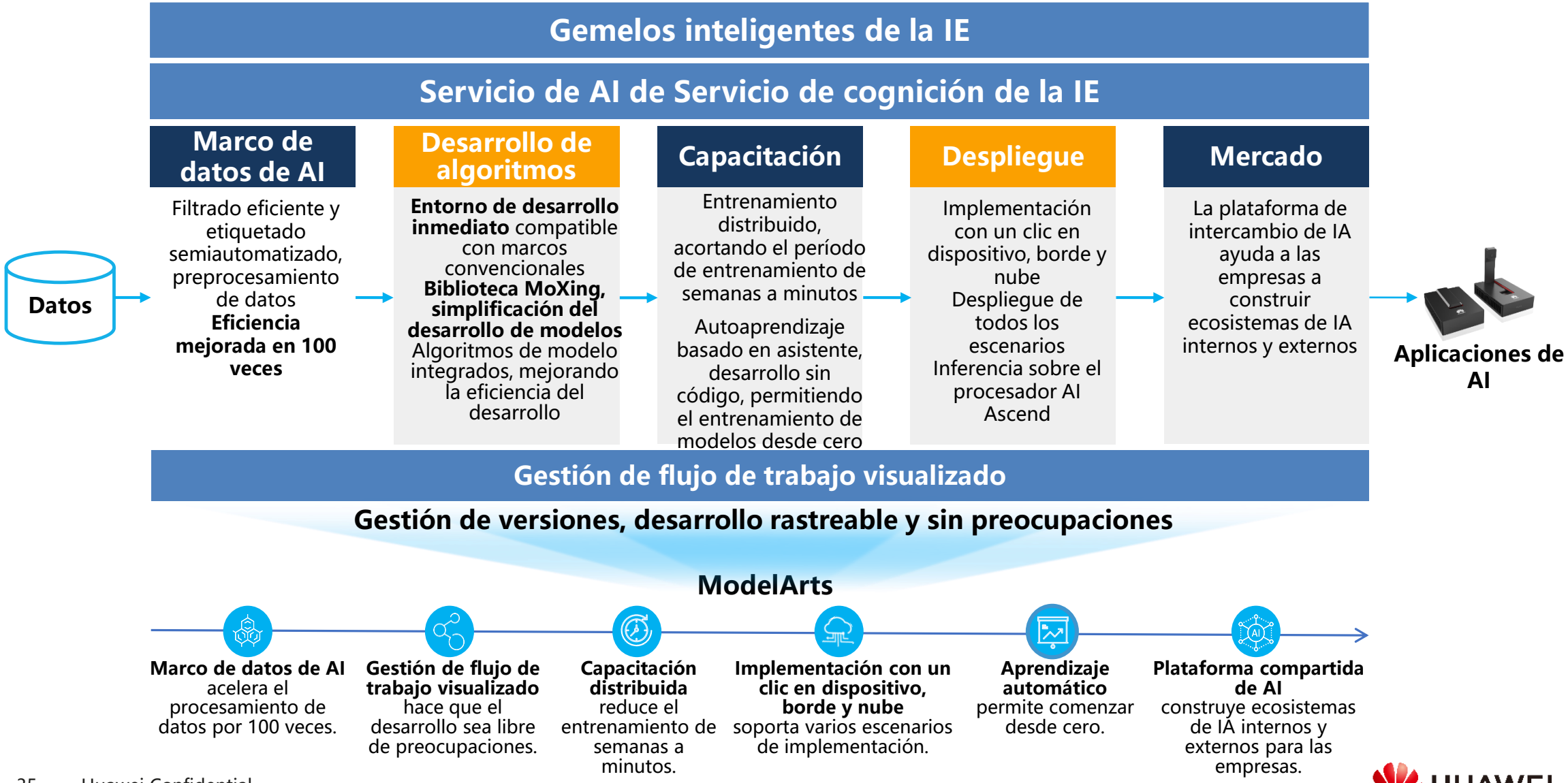
Atlas: permite una solución de infraestructura de IA de todos los escenarios que está orientada al dispositivo, borde y nube basándose en los procesadores de IA de la serie Ascend y varias formas de producto.

"Todos los escenarios de AI" de Huawei indican diferentes escenarios de implementación para AI, incluyendo nubes públicas, nubes privadas, computacion de borde en todas las formas, dispositivos industriales de IoT y dispositivos de consumo.

Full Stack- Flujo de trabajo IA de ciclo completo de ModelArts



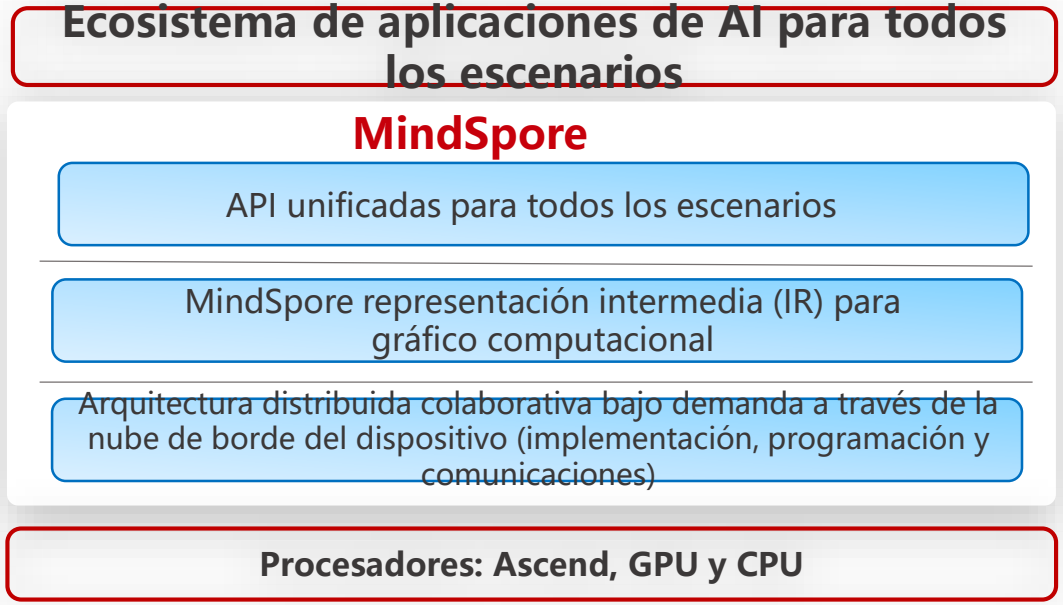
Full Stack- Flujo de trabajo IA de ciclo completo de ModelArts



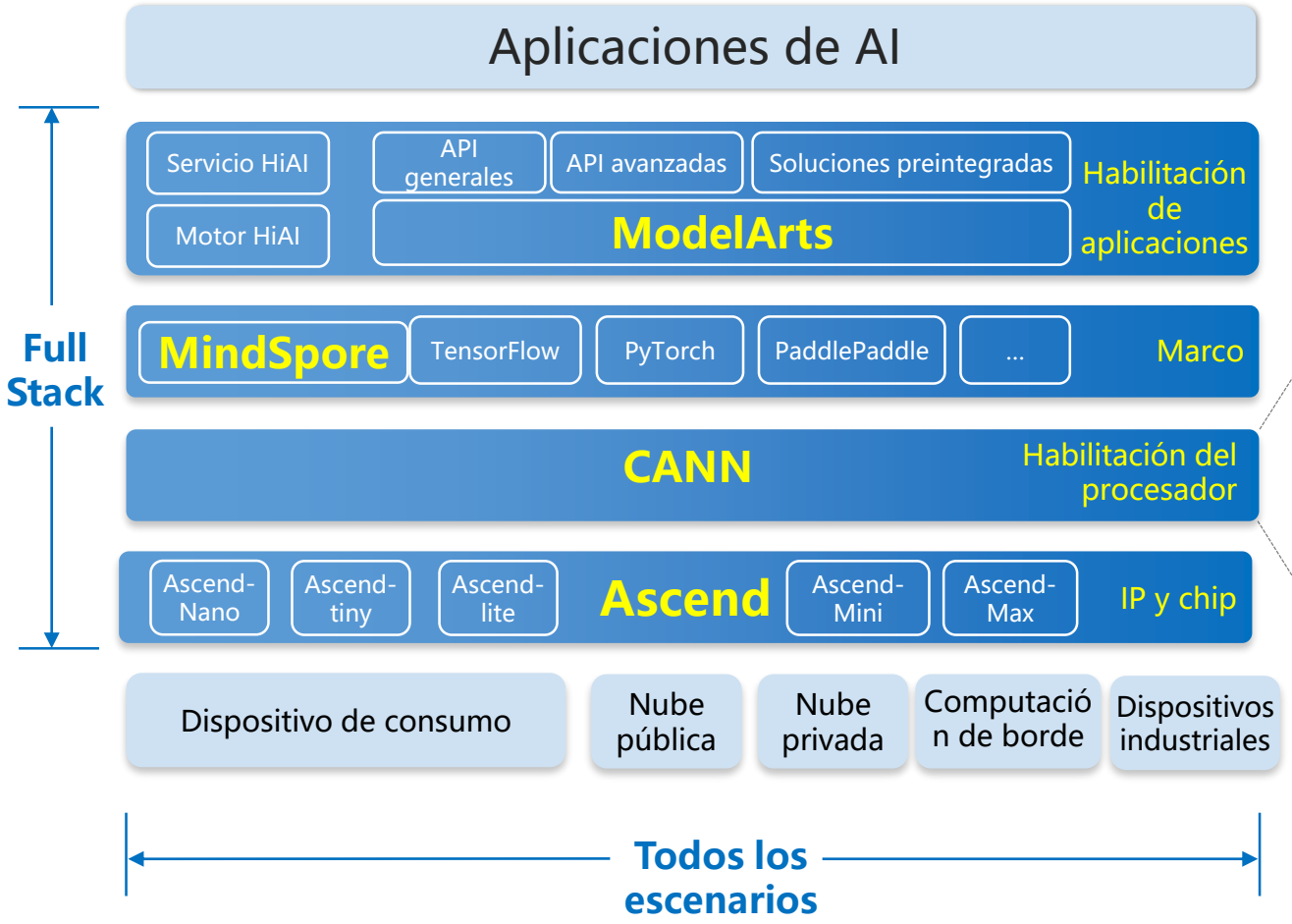
HUAWEI

Full Stack: MindSpore (estructura informática IA de Huawei)

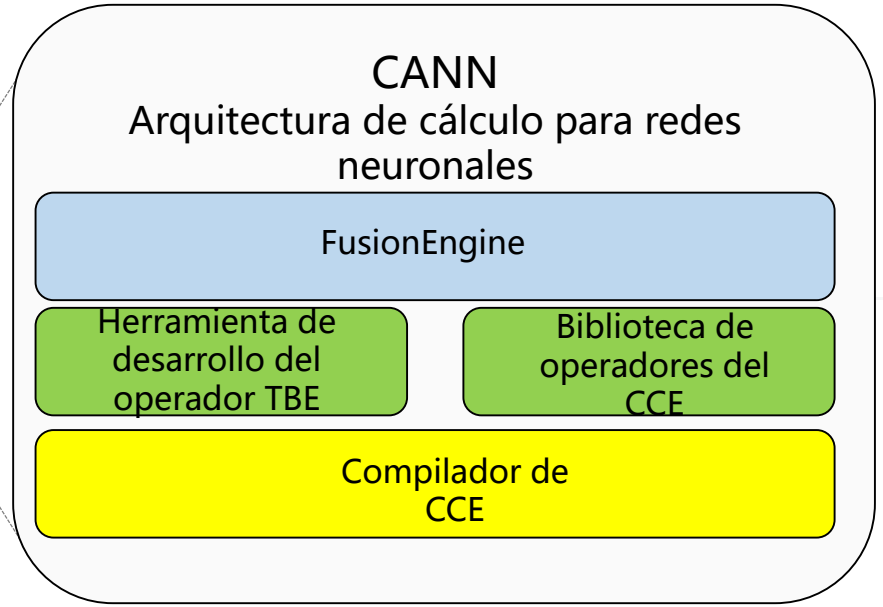
- MindSpore ofrece capacidades paralelas automáticas. Con MindSpore los ingenieros senior de algoritmos y científicos de datos que se centran en modelar datos y resolver problemas pueden ejecutar algoritmos en decenas o incluso miles de nodos informáticos de AI con sólo unas cuantas líneas de descripción.
- El marco MindSpore apoya el despliegue a gran escala y a pequeña escala, adaptándose al despliegue independiente en todos los escenarios. Además de los procesadores de AI Ascend, el MindSpore también soporta otros procesadores como GPU y CPU.



Full Stack - CANN

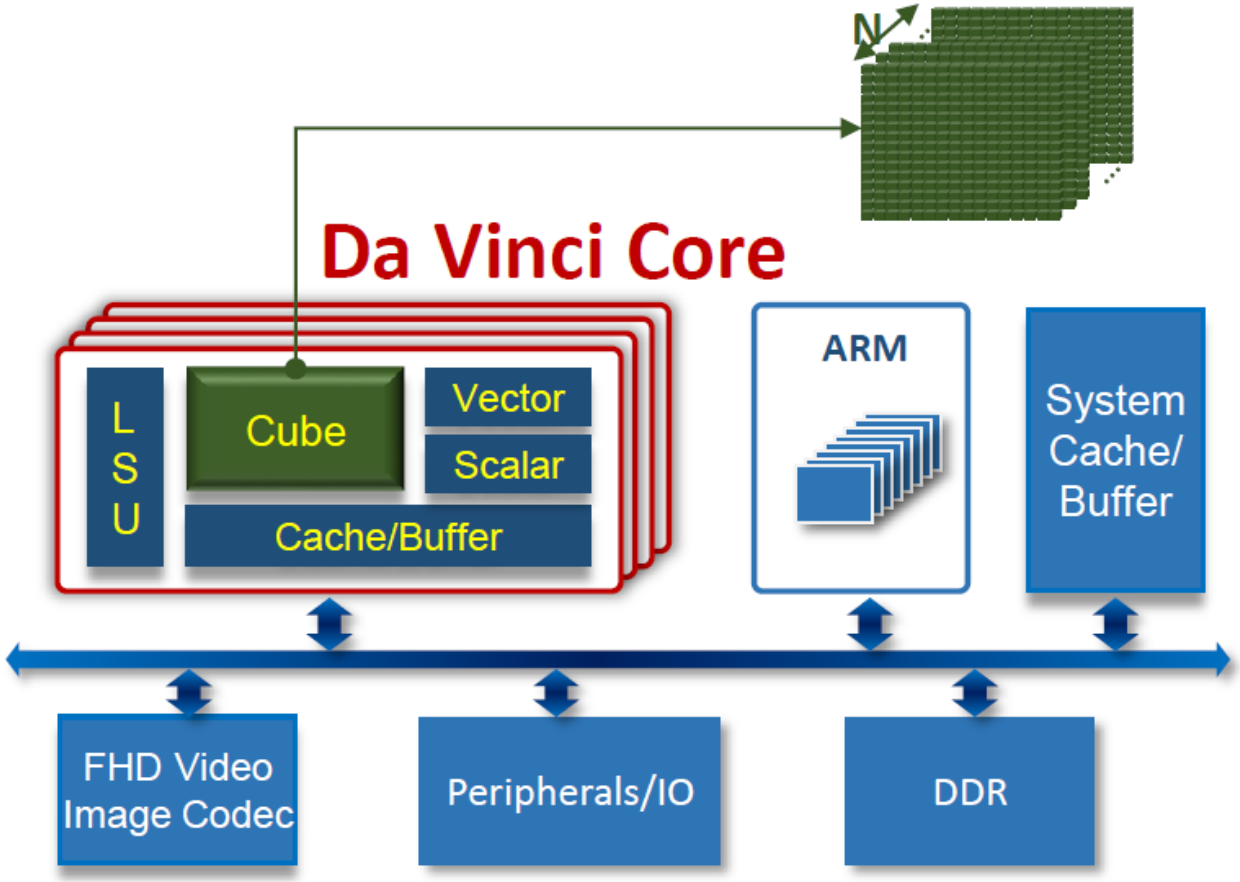


¿Qué:
Una biblioteca de operadores de chips y un kit de herramientas de desarrollo de operadores altamente automatizado
Eficiencia óptima de desarrollo, optimización en profundidad de la biblioteca de operadores común y abundantes APIs
Convergencia de operadores, mejor adaptado al rendimiento del chip Ascend



Full Stack: Procesador de IA Ascend 310 y núcleo Da Vinci

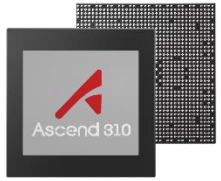
SPECIFICATIONS	Description
Architecture	AI co-processor
Performance	Up to 8T @FP16
	Up to 16T@INT8
Codec	16 Channel Decoder – H.264/265 1080P30 1 Channel Encoder
Memory Controller	LPDDR4X
Memory Bandwidth	2*64bit @3733MT/S
System Interface	PCIe3.0 /USB 3.0/GE
Package	15mm*15mm
Max Power	8Tops@4W, 16Tops@8W
Process	12nm FFC



Note: This is typical configuration, high performance and low power sku can be offered based on your requirement.

HUAWEI

Procesadores Ascend AI: infundiendo inteligencia superior para la informática



Ascend 310

SoC IA con máxima eficiencia energética

Ascend-Mini

Arquitectura: Da Vinci

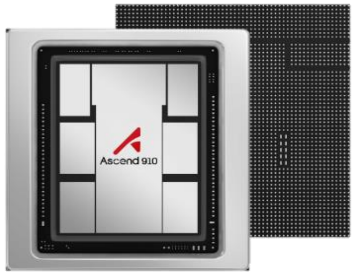
Media precisión (FP16): 8 TFLOPS

Precisión entero (INT8): 16 TOPS

Decodificador de vídeo full HD de 16 canales: H.264/265

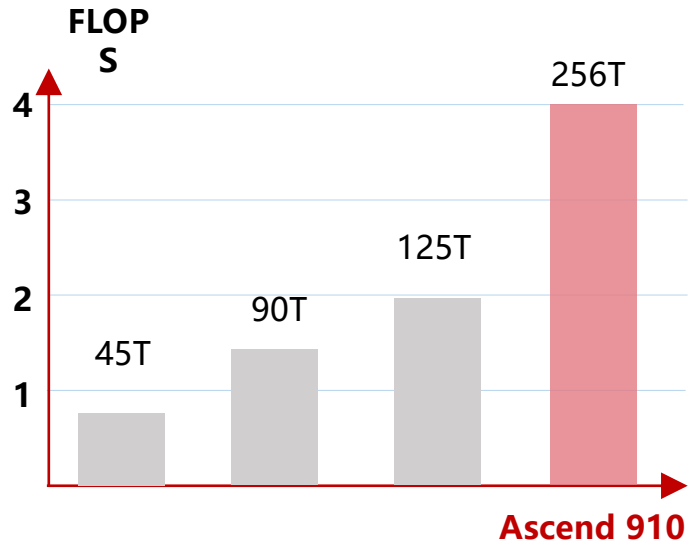
Codificador de vídeo full HD de 1 canal: H.264/265

Potencia máxima: 8 W



Ascend 910

Procesador IA más potente



Ascend-Max

Arquitectura: Da Vinci

Media precisión (FP16): 256 TFLOPS

Precisión entero (INT8): 512 TOPS

Decodificador de vídeo Full HD de 128 canales: H.264/265

Potencia máxima: 310 W

Portafolio de Plataforma de Computación AI Atlas

Internet, seguridad, finanzas, transporte, energía, etc.

Habilitación de aplicaciones

Plataforma de borde inteligente Atlas
SDK de la industria / Motor de contenedores / repositorio de servicios básicos

Plataforma de aprendizaje profundo Atlas
Gestión de clusters / Gestión de modelos / procesamiento previo de datos

Marco

MindSpore

TensorFlow/PyTorch/Caffe/MxNet

Adaptador del marco

CANN

AscendCL

Motor de gráficos para optimización de gráficos

Bibliotecas de operador/aceleración/comunicación (BLAS, FFT, DNN, Rand, Solver, Sparse, HCCL)

Tiempo de ejecución

Conductor

Componentes comunes

Subsistema de gestión de configuración y O&M unificado

Subsistema de seguridad

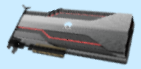
Cadena de herramientas AXE (log / perfilado / Mind Studio)

Chips y hardware

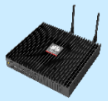
Atlas 200
16 TOPS INT8

Kit de desarrollador Atlas 200

Tarjeta de acelerador de inferencia Atlas 300
64 TOPS INT8



Tarjeta de entrenamiento Atlas 300
256 TFLOPS 16 PM



Atlas 500
16 TOPS INT8

Servidor de inferencia de AI Atlas 800
512 TOPS INT8

Servidor de entrenamiento de AI Atlas 800
2 PFLOPS 16 PM

Atlas 900
256-1024 FPLOPS 16

Ascend 310

Arquitectura Da Vinci

Ascend 910

HUAWEI

Plataforma de razonamiento computacional Atlas de Huawei

CONTENIDOS

1. Aspectos generales de la IA
2. Campos técnicos y campos de aplicación de la IA
3. Estrategia de desarrollo de IA de Huawei
4. **Disputas de IA**
5. Perspectivas futuras de la IA

Ver = ¿Creer?

- Con el desarrollo de tecnologías de visión informática, la fiabilidad de imágenes y vídeos está disminuyendo. Las imágenes falsas se pueden producir con tecnologías tales como PS y redes adversarias generativas (GAN), haciendo difícil identificar si las imágenes son verdaderas o no.
- Ejemplo:
  - Un sospechoso proporcionó pruebas falsas forjando una imagen en la que el sospechoso está en un lugar donde nunca ha estado o con alguien que nunca ha visto usando tecnologías de PS.
  - En los anuncios de píldoras de dieta, las apariencias de las personas antes y después de la pérdida de peso pueden cambiarse con tecnologías de PS para exagerar el efecto de las píldoras.
  - Lyrebird, una herramienta para simular la voz de seres humanos basada en muestras de minutos de registro, puede ser utilizado por delincuentes.
  - Las imágenes domésticas liberadas en plataformas de alquiler y reserva de hoteles pueden ser generadas a través de GAN.

Desarrollo de IA = ¿Aumento del desempleo?

- Mirando hacia atrás, los seres humanos siempre han estado buscando maneras de mejorar la eficiencia, es decir, obtener más con menos recursos. Usamos piedras afiladas para cazar y recoger alimentos de manera más eficiente. Usamos motores de vapor para reducir la necesidad de caballos. Cada paso para lograr la automatización cambiará nuestra vida y nuestro trabajo. En la era de la AI, ¿qué empleos serán reemplazados por la AI?
- La respuesta es trabajos repetitivos que involucran poca creatividad e interacción social.

Empleos más probables de ser reemplazados por AI	Empleos más improbables que sean reemplazados por AI
Repartidor	Escritor
Conductor de Taxi	Personal de Gestión
Soldado	Ingenieros de Software
Contabilidad	Gerente de Recursos Humanos
Personal de televentas	Diseñador
Atención al cliente	Planificador de actividades
...	...

Problemas a resolver

- ¿Están las obras creadas por IA protegidas por las leyes de derechos de autor?
- ¿Quién da autoridad a los robots?
- ¿Qué derechos se deben autorizar a los robots?
- ...

CONTENIDOS

1. Aspectos generales de la IA
2. Campos técnicos y campos de aplicación de la IA
3. Estrategia de desarrollo de IA de Huawei
4. Disputas de IA
5. **Perspectivas futuras de la IA**

Tendencias de desarrollo de tecnologías de la IA

- Marco: marco de desarrollo más fácil de utilizar
- Algoritmo: modelos de algoritmos con mejor rendimiento y menor tamaño
- Potencia informática: desarrollo integral de la computación en nube del dispositivo
- Datos: industria de servicios básicos de datos más completa y compartición de datos más segura
- Escenario: avances continuos en aplicaciones de la industria

Marco de Desarrollo más fácil de usar

- Varios marcos de desarrollo de IA están evolucionando hacia la facilidad de uso y la omnipotencia, reduciendo continuamente el umbral para el desarrollo de IA.

MindSpore

TensorFlow

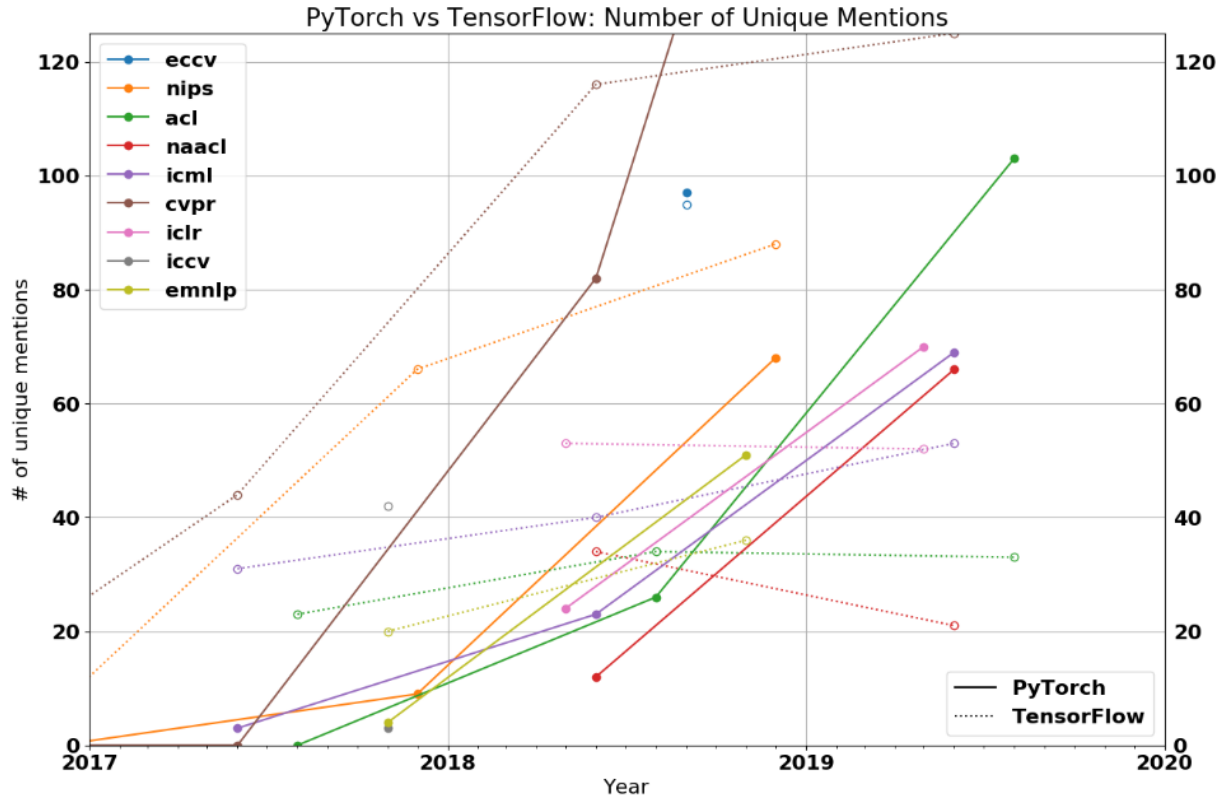
PYTORCH

TensorFlow 2.0

- TensorFlow 2.0 ha sido lanzado oficialmente. Integra Keras como su API de alto nivel, mejorando considerablemente la usabilidad.

Pytorch vs TensorFlow

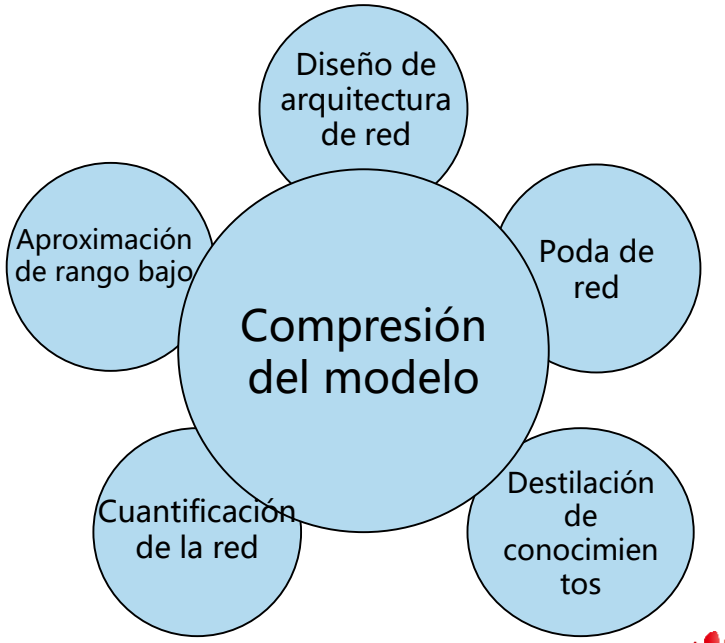
- PyTorch es ampliamente reconocido por el mundo académico por su facilidad de uso.



Comparación entre estadísticas de uso PyTorch y TensorFlow de conferencias académicas más importantes

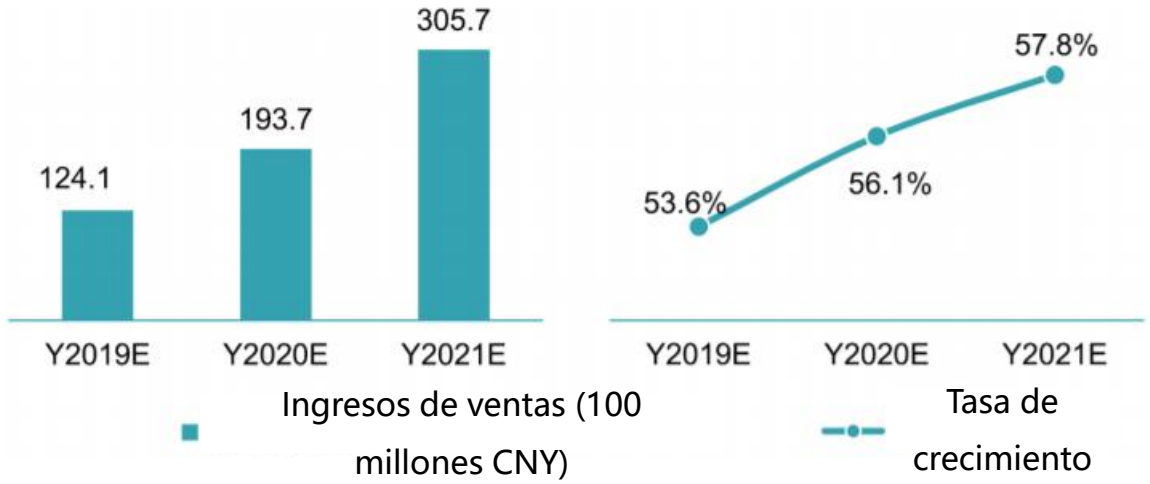
Modelos de aprendizaje profundo más pequeños

- Un modelo con mejor rendimiento suele tener una mayor cantidad de parámetros, y un modelo grande tiene menor eficiencia de funcionamiento en aplicaciones industriales. Cada vez se proponen más tecnologías de compresión de modelos para comprimir aún más el tamaño del modelo, garantizando al mismo tiempo el rendimiento del modelo, cumpliendo los requisitos de aplicaciones industriales.
  - Aproximación de rango bajo
  - Poda de red
  - Cuantificación de la red
  - Destilación de conocimientos
  - Diseño de red compacto



Potencia informática con desarrollo integral de dispositivos-borde-nube

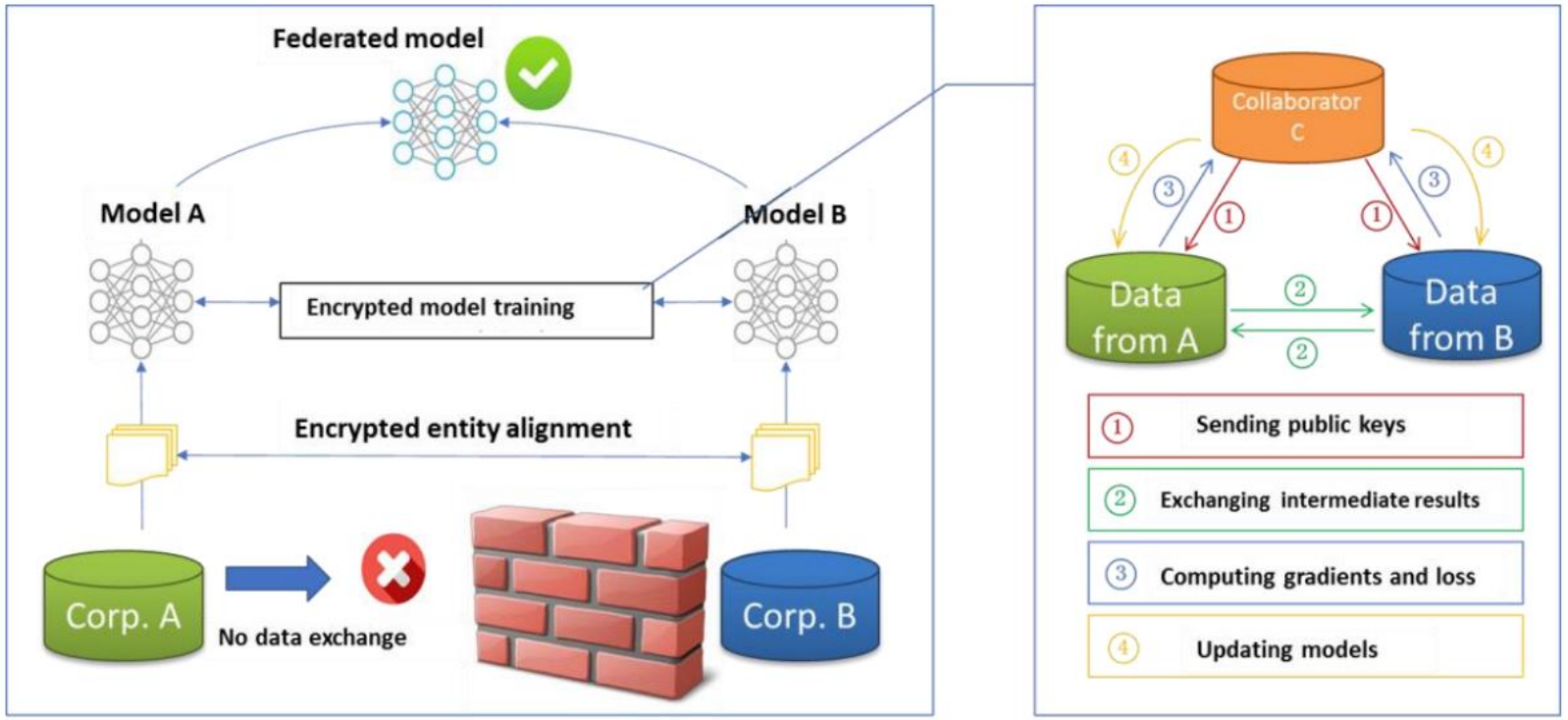
- La escala de chips de IA aplicados a la nube, dispositivos de borde y dispositivos móviles sigue aumentando, satisfaciendo aún más la demanda de potencia de computación de IA.



Libro Blanco sobre el desarrollo de la industria de chips de AI de China 2020
Escala de mercado y predicción de crecimiento de los chips de AI en China de 2020 a 2021

Compartir datos de manera más segura

- El aprendizaje federado utiliza diferentes fuentes de datos para entrenar modelos, rompiendo aún más los cuellos de botella de datos al tiempo que garantiza la privacidad y seguridad de los datos.



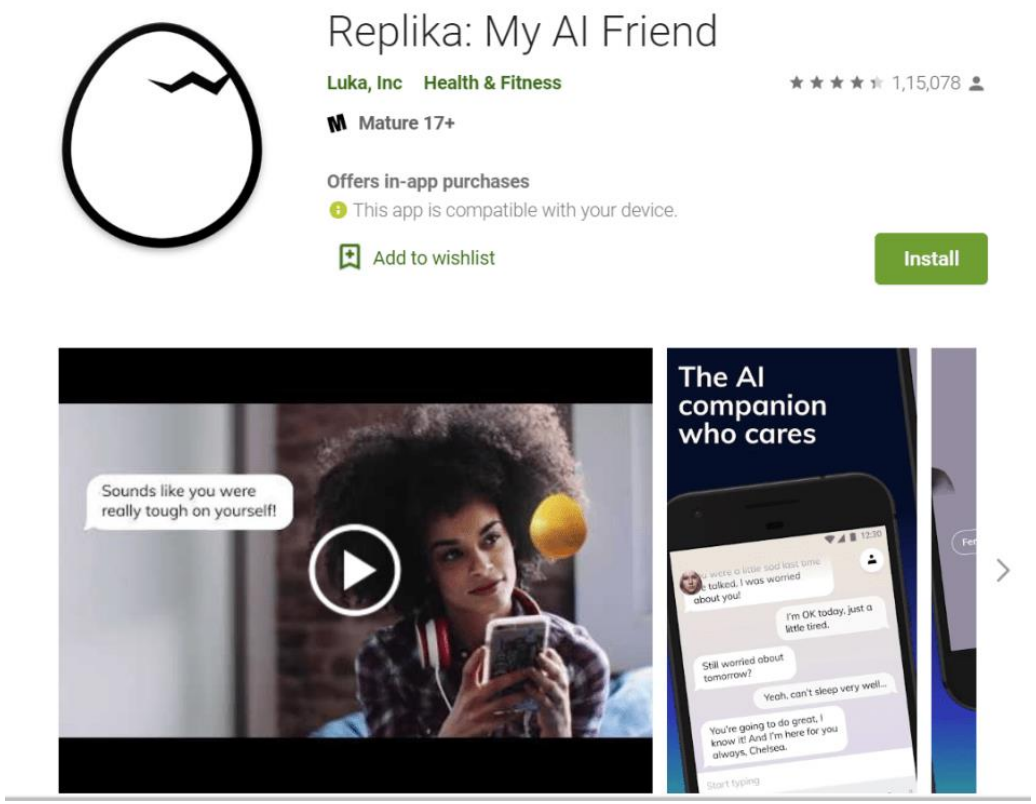
Libro Blanco de aprendizaje Federado V1.0

Adelantos continuos en escenarios de aplicaciones

- Con la exploración continua de la IA en varias verticales, los escenarios de aplicación de la IA cambiarán continuamente.
  - Mitigación de problemas psicológicos
  - Seguro automático de vehículos y evaluación de pérdidas
  - Automatización de oficinas
  - ...

Mitigación de problemas psicológicos

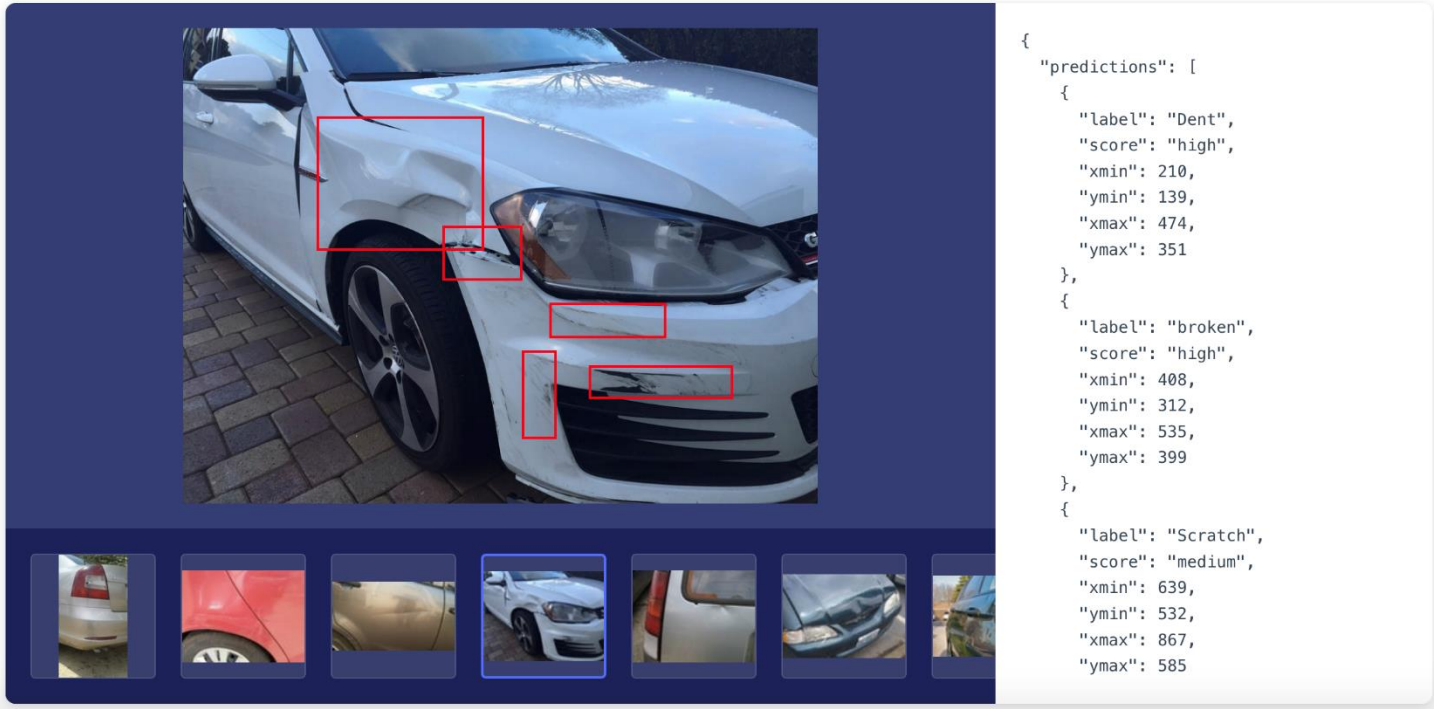
- Los robots de chat con IA ayudan a aliviar problemas de salud mental como el autismo combinando conocimiento psicológico.



Seguro automático de vehículos y evaluación de pérdidas

- Las tecnologías de IA ayudan a las compañías de seguros a optimizar las reclamaciones de seguro de vehículos y completar la evaluación de pérdidas de seguro de vehículos utilizando algoritmos de aprendizaje profundo como el reconocimiento de imágenes.

Vehicle Damage Assessment




```
{
  "predictions": [
    {
      "label": "Dent",
      "score": "high",
      "xmin": 210,
      "ymin": 139,
      "xmax": 474,
      "ymax": 351
    },
    {
      "label": "broken",
      "score": "high",
      "xmin": 408,
      "ymin": 312,
      "xmax": 535,
      "ymax": 399
    },
    {
      "label": "Scratch",
      "score": "medium",
      "xmin": 639,
      "ymin": 532,
      "xmax": 867,
      "ymax": 585
```

Automatización de oficinas

- IA está automatizando la gestión, pero la naturaleza y formato diferentes de los datos hacen que sea una tarea difícil. Mientras que cada industria y aplicación tiene sus propios desafíos únicos, diferentes industrias están adoptando gradualmente soluciones de flujo de trabajo basadas en el aprendizaje máquina.

RESUMEN

- Este capítulo introduce la definición y el historial de desarrollo de la IA, describe los campos técnicos y campos de aplicación de la IA, introduce brevemente la estrategia de desarrollo de la IA de Huawei y, finalmente, discute las disputas y las tendencias de desarrollo de la IA.

Quiz

1. (Pregunta de respuestas múltiples) ¿Cuál de los siguientes campos de aplicación de IA?
   A. Casa inteligente
   B. Asistencia sanitaria inteligente
   C. Ciudad inteligente
   D. Educación inteligente
2. (Verdadero o Falso) Por "todos los escenarios de IA", Huawei significa diferentes escenarios de implementación para IA, incluyendo nubes públicas, nubes privadas, computación de borde en todas las formas, dispositivos IoT industriales y dispositivos de consumo.
   A. Verdadero
   B. Falso

MÁS INFORMACIÓN

Sitio web de aprendizaje en línea

- https://e.huawei.com/en/talent/#/home

Base de conocimientos de Huawei

- https://support.huawei.com/enterprise/en/knowledge?lang=en

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

HUAWEI

Introducción a Machine Learning

HUAWEI

Prefacio

- Machine learning (M.L.) es un campo central de investigación de la IA, y también es un conocimiento necesario para el aprendizaje profundo (deep learning). Por lo tanto, este capítulo introduce principalmente los conceptos principales, la clasificación, el proceso general y los algoritmos comunes de Machine Learning.

Objetivos

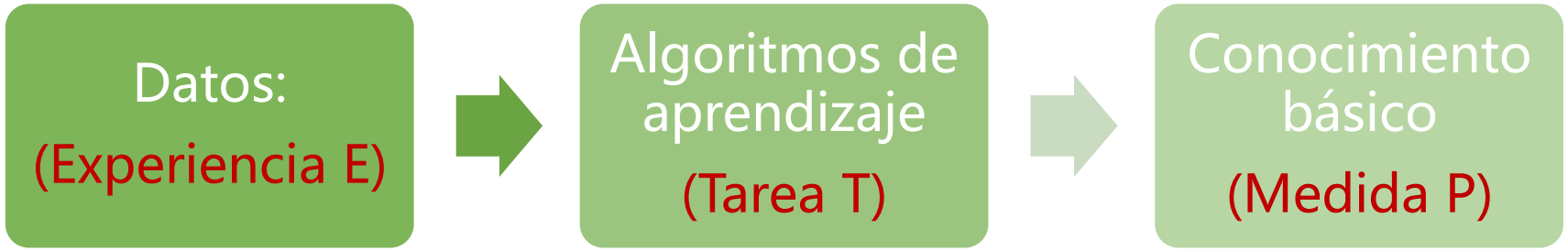
Al finalizar este curso, podrá:

- Dominar la definición del algoritmo de aprendizaje y el proceso de Machine Learning.
- Conozca los algoritmos comunes de Machine Learning.
- Comprender conceptos tales como hiperparámetros, descenso de gradiente y validación cruzada.

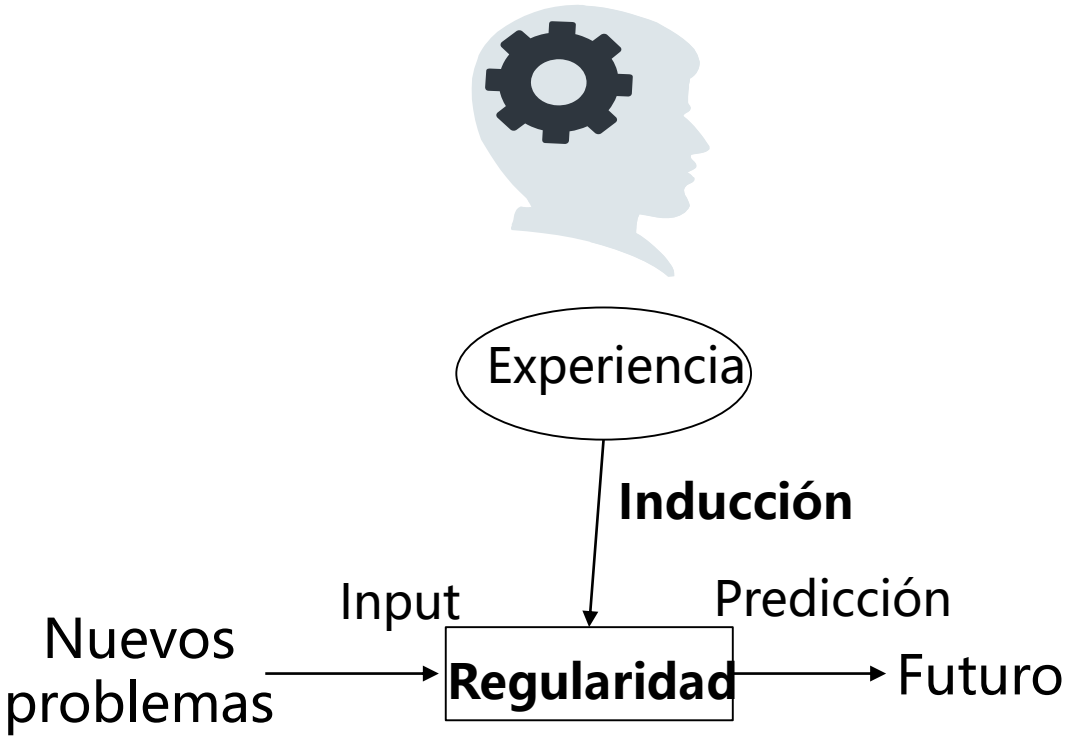
Contents

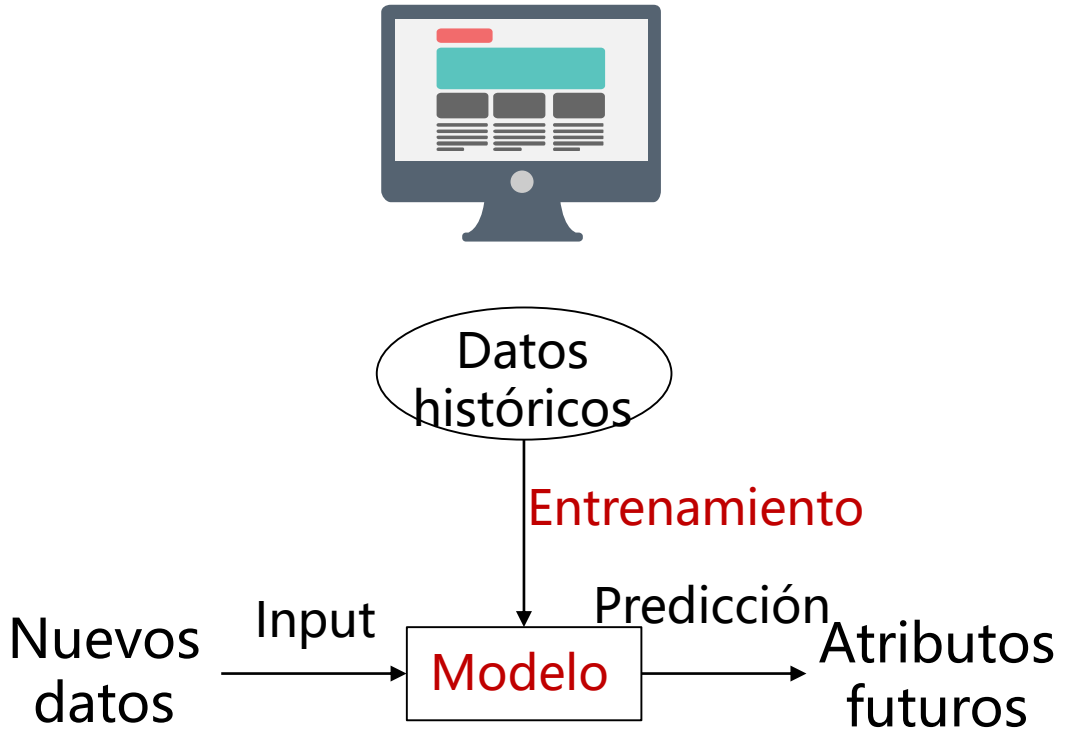
Algoritmos de Machine Learning (1)

- El Machine Learning ó Machine Learning (incluido Deep Learning o aprendizaje profundo) es un estudio de algoritmos de aprendizaje. Se dice que un programa de computadora aprende de la experiencia E con respecto a alguna clase de tareas T y medida de desempeño P si su desempeño en las tareas en T, medido por P, mejora con la experiencia E.

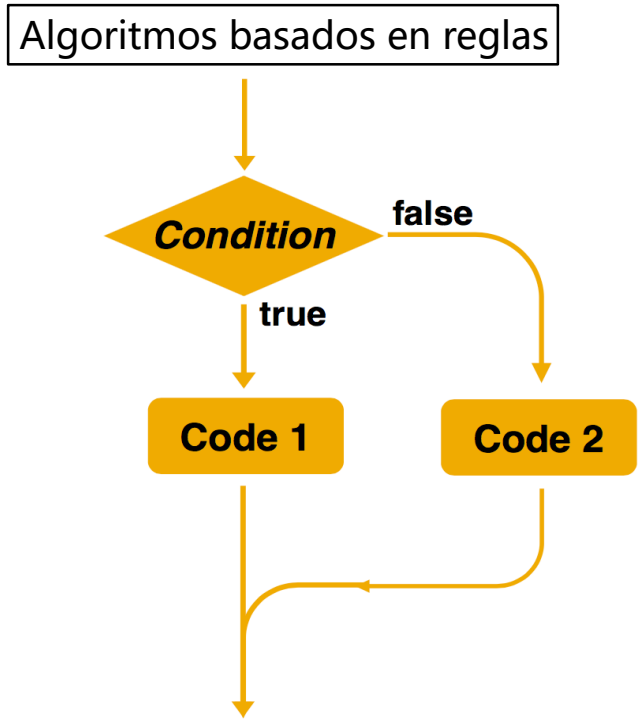


Algoritmos de Machine Learning (2)

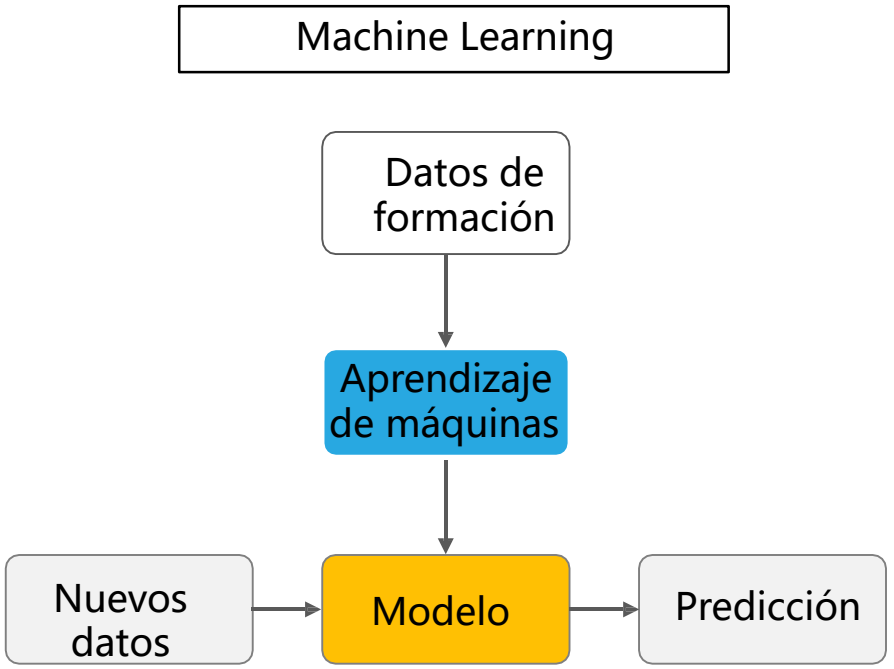




Diferencias entre algoritmos de Machine Learning y algoritmos basados en reglas



- La programación explícita se utiliza para resolver problemas.
- Las reglas se pueden especificar manualmente.

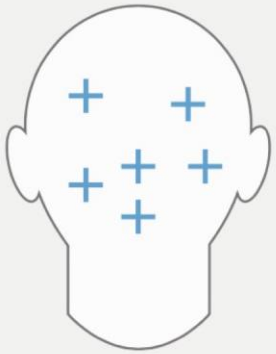


- Las muestras se utilizan para el entrenamiento.
- Las normas de toma de decisiones son complejas o difíciles de describir.
- Las reglas son aprendidas automáticamente por las máquinas.

Escenarios de aplicación del Machine Learning(1)

- La solución a un problema es compleja, o el problema puede implicar una gran cantidad de datos sin una función clara de distribución de datos.
- El Machine Learning se puede utilizar en los siguientes escenarios:

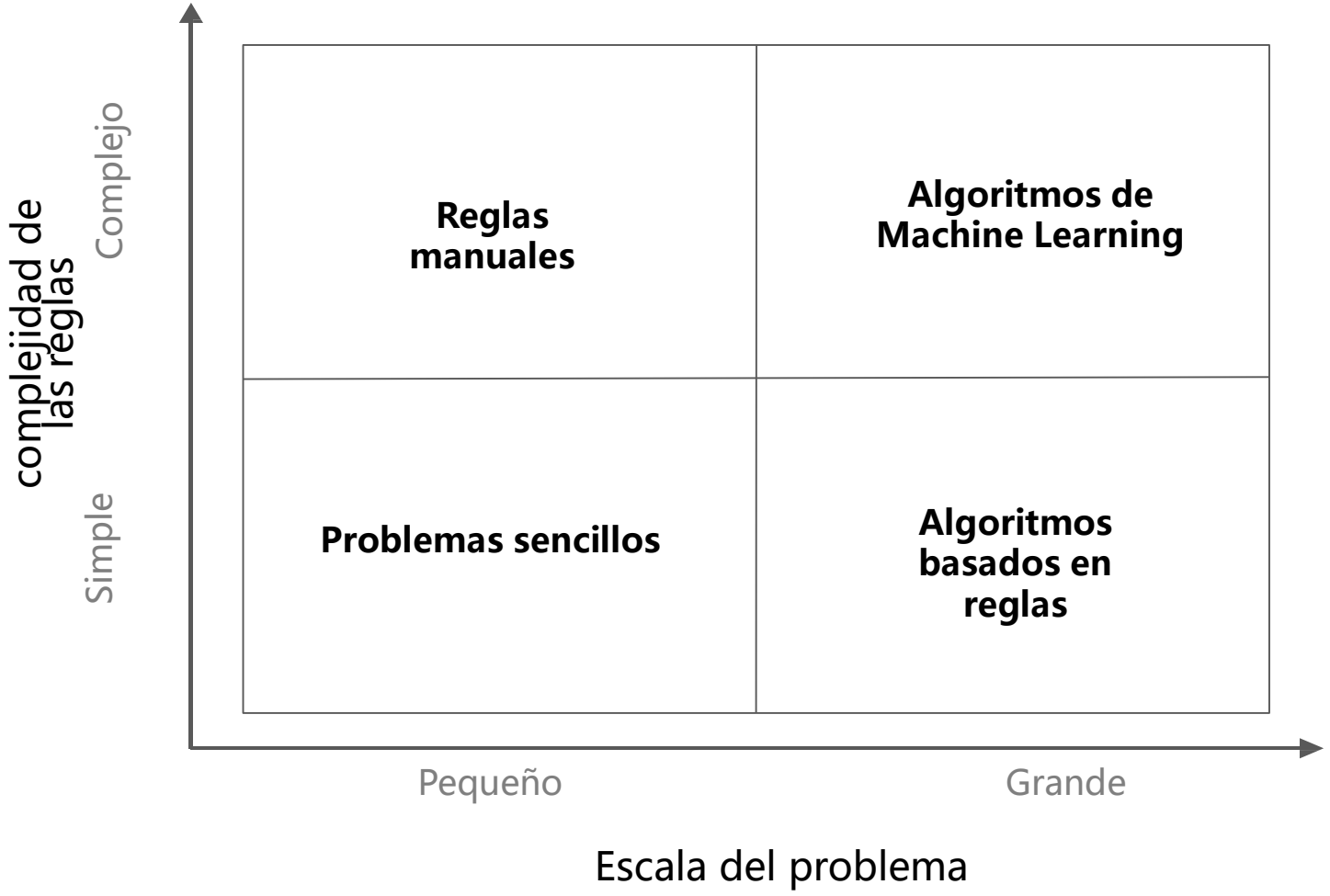
Las reglas son complejas o no pueden describirse, como el reconocimiento facial y el reconocimiento de voz.



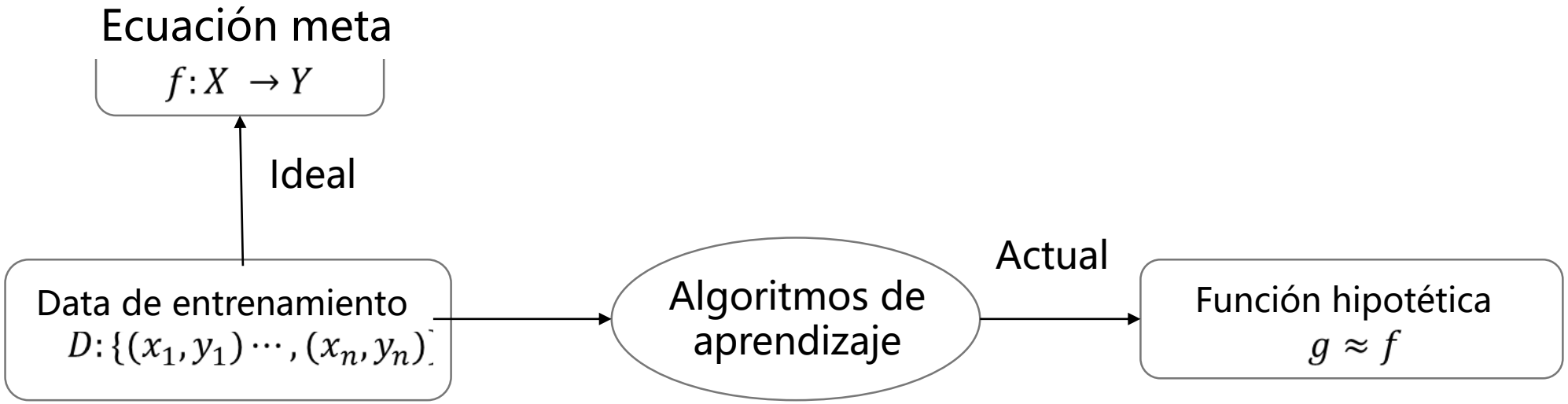
Las reglas de la tarea cambian con el tiempo. Por ejemplo, en la tarea de etiquetado de la parte de habla, se generan nuevas palabras o significados en cualquier momento.

La distribución de datos cambia con el tiempo, lo que requiere una constante readaptación de los programas, como predecir la tendencia de las ventas de productos básicos.

Escenarios de aplicación del Machine Learning(2)



Comprendimiento racional de los algoritmos de Machine Learning



- La función de meta *f* se desconoce. Los algoritmos de aprendizaje no pueden obtener una función perfecta *f*.
- Supongamos que la función de hipótesis *g* **se aproxima** a la función *f*, pero puede ser diferente de la función *f*.

Principales problemas resueltos por el Machine Learning

- El Machine Learning puede lidiar con muchos tipos de tareas. A continuación, se describen los tipos de tareas más habituales y habituales.
  - Clasificación: Un programa de computadora necesita especificar a cuál de las categorías k pertenece alguna entrada. Para realizar esta tarea, los algoritmos de aprendizaje suelen generar una función $f: R^n \rightarrow (1,2,\ldots, k)$. Por ejemplo, el algoritmo de clasificación de imágenes en visión artificial se desarrolla para manejar tareas de clasificación.
  - Regresión: para este tipo de tarea, un programa de computadora predice la salida para la entrada dada. Los algoritmos de aprendizaje suelen generar una función $f: R^n \rightarrow R$. Un ejemplo de este tipo de tarea es predecir el monto del reclamo de una persona asegurada (para establecer la prima del seguro) o predecir el precio del valor.
  - Agrupación: una gran cantidad de datos de un conjunto de datos sin etiquetar se divide en varias categorías según la similitud interna de los datos. Los datos de la misma categoría son más similares que los de diferentes categorías. Esta función se puede utilizar en escenarios como la recuperación de imágenes y la gestión de perfiles de usuario.

  La clasificación y la regresión son dos tipos principales de predicción, y representan del 80% al 90%. El resultado de la clasificación son valores de categoría discretos y el resultado de la regresión son números continuos.

Contenido

1. Definición de Machine Learning
2. **Tipos de Machine Learning**
3. Proceso de Machine Learning
4. Otros métodos de Machine Learning clave
5. Algoritmos comunes de Machine Learning
6. Caso práctico

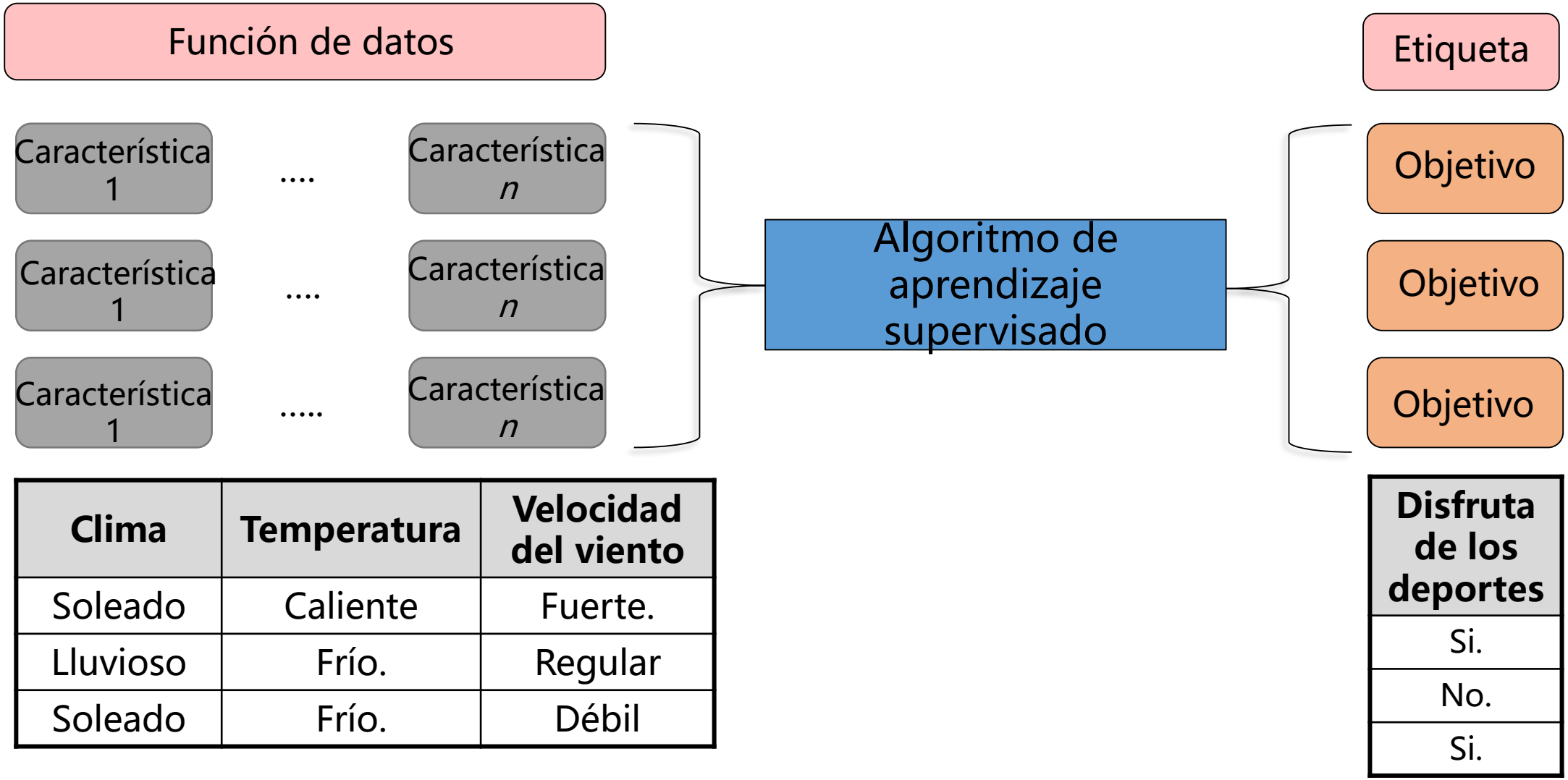
Clasificación de Machine Learning

- **Aprendizaje supervisado:** Obtener un modelo óptimo con el desempeño requerido a través de la capacitación y el aprendizaje basado en las muestras de las categorías conocidas. A continuación, utilizar el modelo para mapear todas las entradas a las salidas y comprobar la salida con el fin de clasificar los datos desconocidos.
- **Aprendizaje no supervisado:** Para muestras sin etiqueta, los algoritmos de aprendizaje modelan directamente los conjuntos de datos de entrada. La agrupación (Clustering) es una forma común de aprendizaje no supervisado. Solo necesitamos juntar muestras muy similares, calcular la similitud entre las nuevas muestras y las existentes, y clasificarlas por similitud.
- **Aprendizaje semisupervisado:** En una tarea, un modelo de Machine Learning que automáticamente utiliza una gran cantidad de datos sin etiquetar para ayudar a aprender directamente de una pequeña cantidad de datos etiquetados.
- **Aprendizaje reforzado:** Se trata de un área de Machine Learning que se ocupa de cómo los agentes deben tomar medidas en un entorno para maximizar cierta noción de recompensa acumulativa. La diferencia entre el refuerzo del aprendizaje y el aprendizaje supervisado es la señal del profesor. La señal de refuerzo proporcionada por el entorno en el aprendizaje de refuerzo se utiliza para evaluar la acción (señal escalar) en lugar de decirle al sistema de aprendizaje cómo realizar las acciones correctas.

Clasificación de Machine Learning

- **Aprendizaje supervisado:** Obtenga un modelo óptimo con el desempeño requerido a través de la capacitación y el aprendizaje basado en las muestras de las categorías conocidas. A continuación, utilice el modelo para mapear todas las entradas a las salidas y comprobar la salida con el fin de clasificar los datos desconocidos.
- **Aprendizaje no supervisado:** para muestras sin etiqueta, los algoritmos de aprendizaje modelan directamente los conjuntos de datos de entrada. La agrupación es una forma común de aprendizaje no supervisado. Solo necesitamos juntar muestras muy similares, calcular la similitud entre las nuevas muestras y las existentes, y clasificarlas por similitud.
- **Aprendizaje semisupervisado:** En una tarea, un modelo de Machine Learning que automáticamente utiliza una gran cantidad de datos sin etiquetar para ayudar a aprender directamente de una pequeña cantidad de datos etiquetados.
- **Aprendizaje reforzado:** Se trata de un área de Machine Learning que se ocupa de cómo los agentes deben tomar medidas en un entorno para maximizar cierta noción de recompensa acumulativa. La diferencia entre el refuerzo del aprendizaje y el aprendizaje supervisado es la señal del profesor. La señal de refuerzo proporcionada por el entorno en el aprendizaje de refuerzo se utiliza para evaluar la acción (señal escalar) en lugar de decirle al sistema de aprendizaje cómo realizar las acciones correctas.

Aprendizaje supervisado

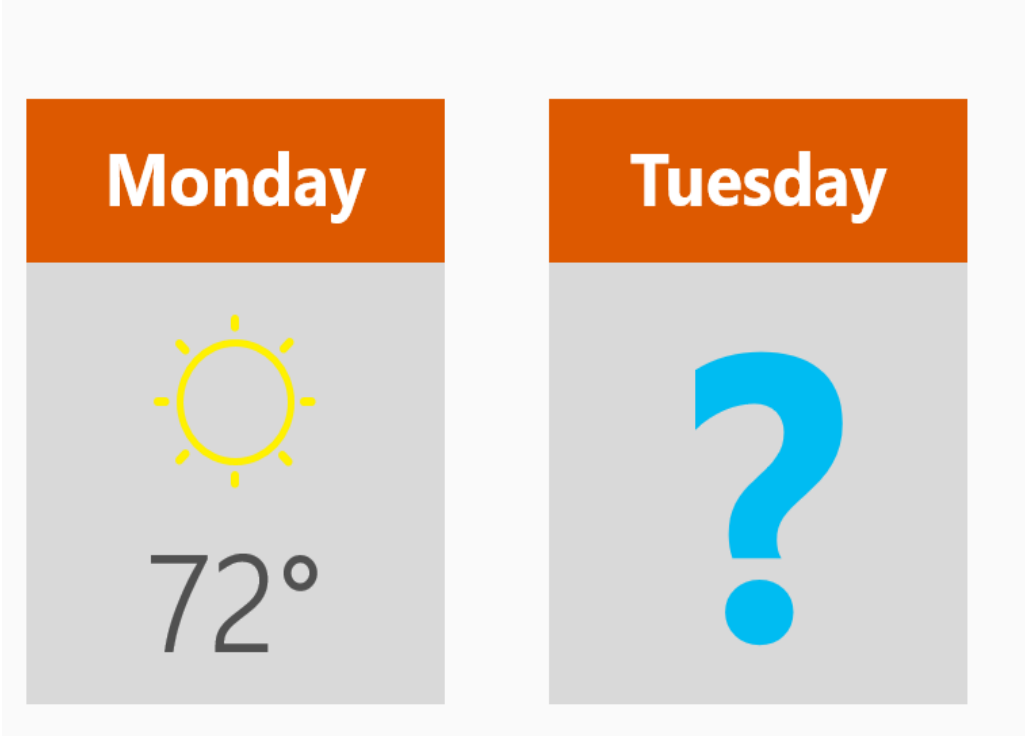


Clima	Temperatura	Velocidad del viento
Soleado	Caliente	Fuerte.
Lluvioso	Frío.	Regular
Soleado	Frío.	Débil

Disfruta de los deportes
Si.
No.
Si.

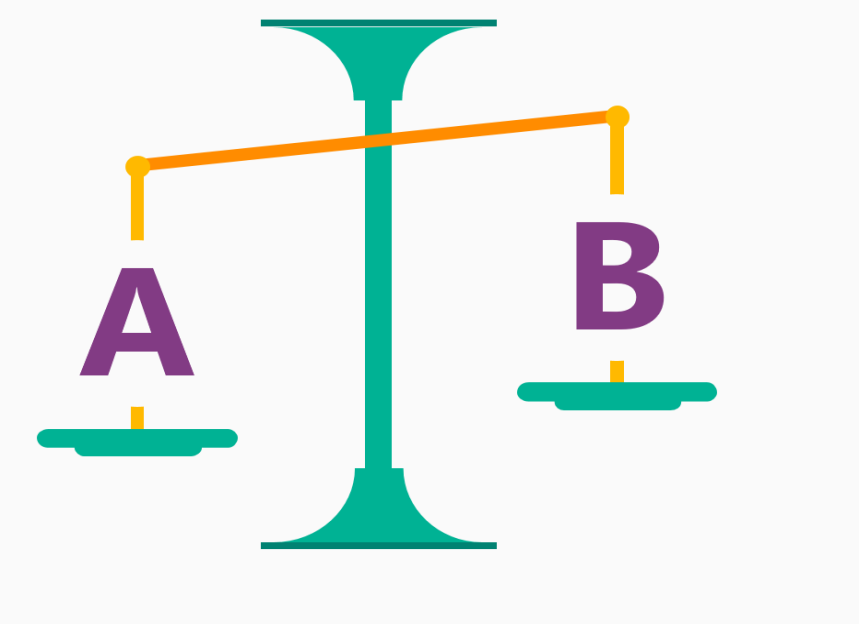
Aprendizaje supervisado - Preguntas de regresión

- Regresión: refleja las características de los valores de atributo de las muestras en un conjunto de datos de muestra. La dependencia entre los valores de atributo se descubre mediante la expresión de la relación de la asignación de muestra a través de funciones.
  - ¿Cuánto me beneficiaré de las acciones la semana que viene?
  - ¿Cuál es la temperatura del martes?

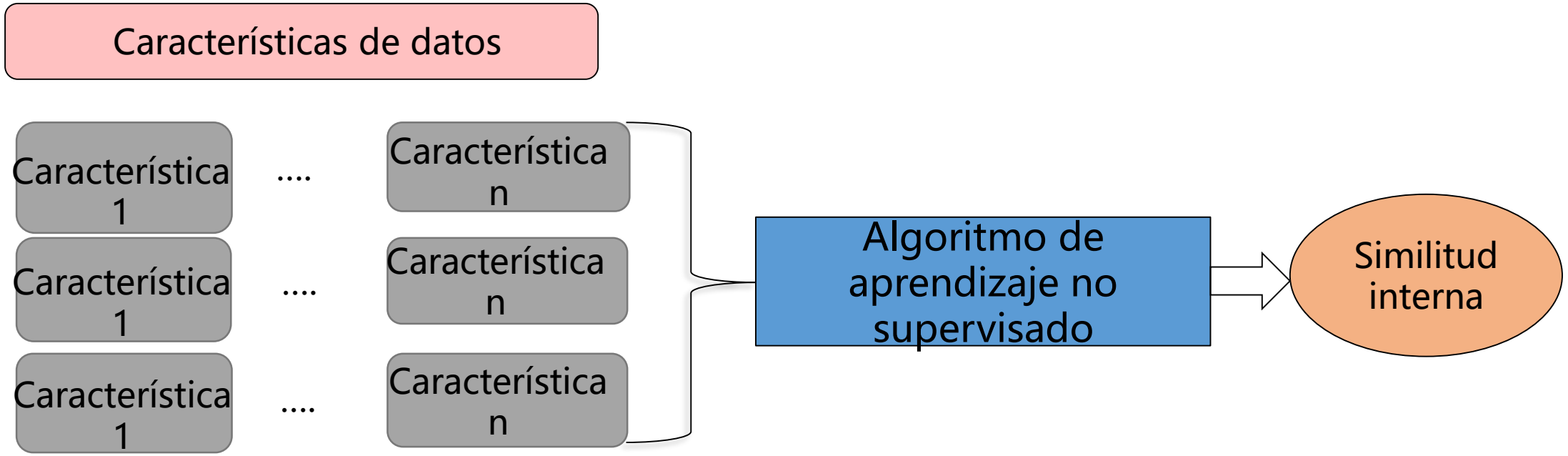


Aprendizaje supervisado - Cuestiones de clasificación

- Clasificación: asigna las muestras de un conjunto de datos de muestra a una categoría específica utilizando un modelo de clasificación.
  - ¿Habrá un embotellamiento en la calle X mañana en la hora punta?
  - ¿Qué método es más atractivo para los clientes? ¿Voucher de X USD, o 25% de descuento?



Aprendizaje sin supervisión

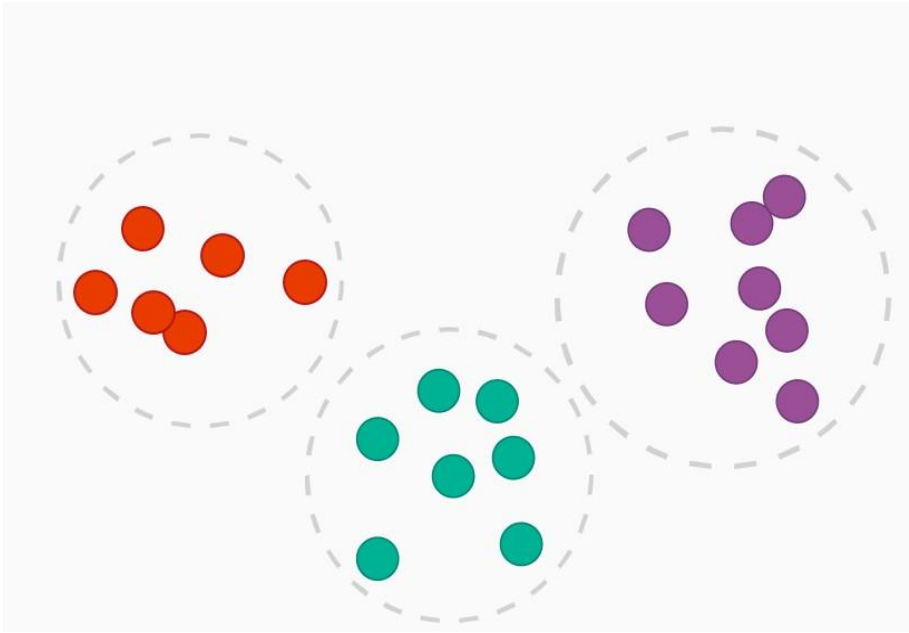


Consumo mensual	Producto	Tiempo de consumo
1000–2000	Raqueta de badminton	6:00–12:00
500–1000	Balón de basquetbol	18:00–24:00
1000–2000	Consola de juegos	00:00–6:00

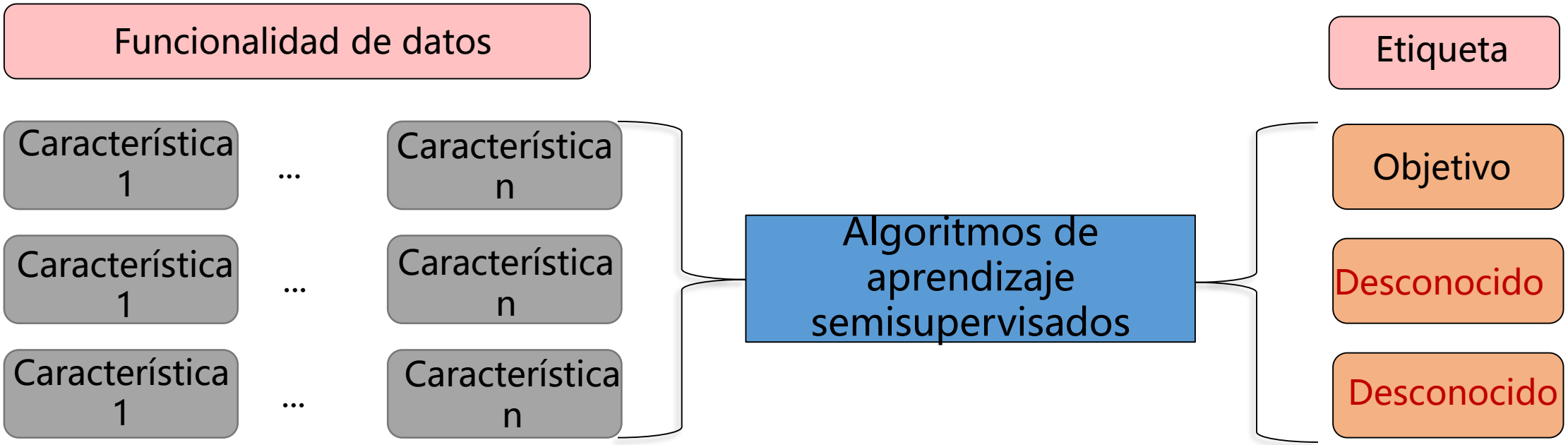
Categoría
cluster 1
cluster 2

Aprendizaje no supervisado - Preguntas de agrupamiento

- Clustering: clasifica las muestras en un conjunto de datos de muestras en varias categorías basadas en el modelo de clustering. La similitud de las muestras pertenecientes a la misma categoría es elevada.
  - ¿A qué público le gusta ver películas del mismo tema?
  - ¿Cuáles de estos componentes están dañados de una manera similar?



Aprendizaje semisupervisado

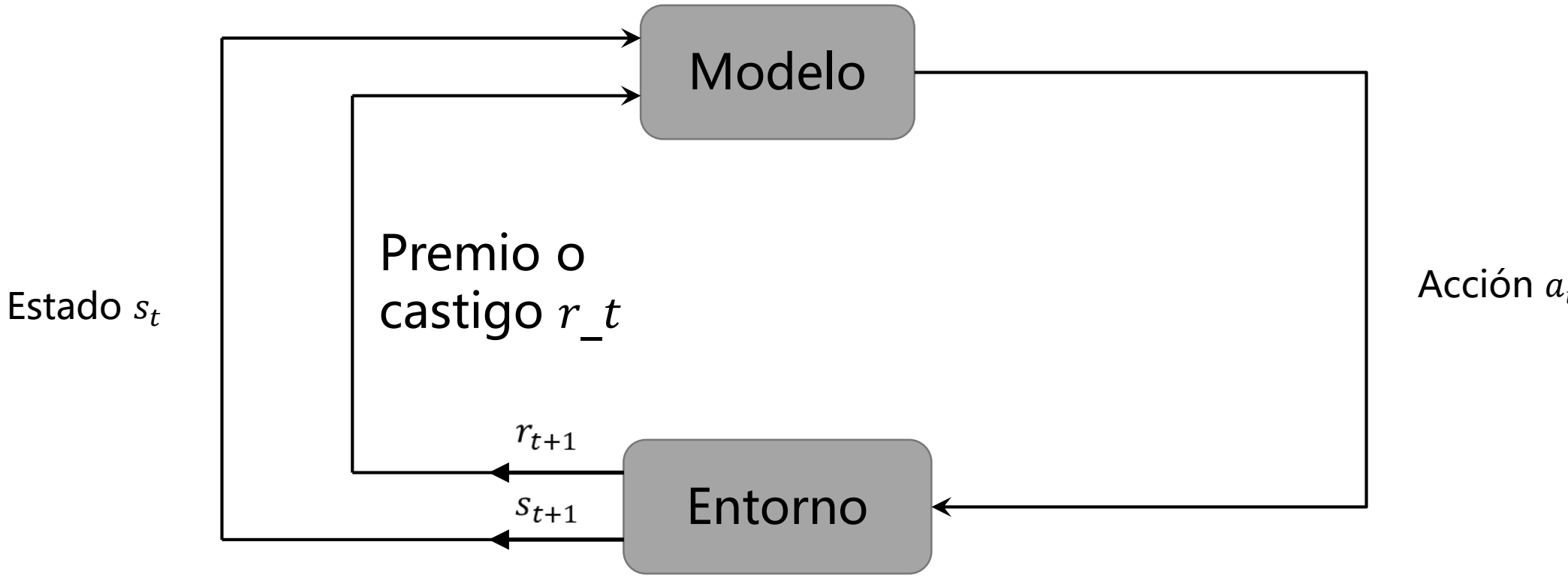


Clima	Temperatura	Velocidad del viento
Soleado	Caliente	Fuerte.
Lluvioso	Frío.	Regular
Soleado	Frío.	Débil

Disfruta de los deportes
Si.
/
/

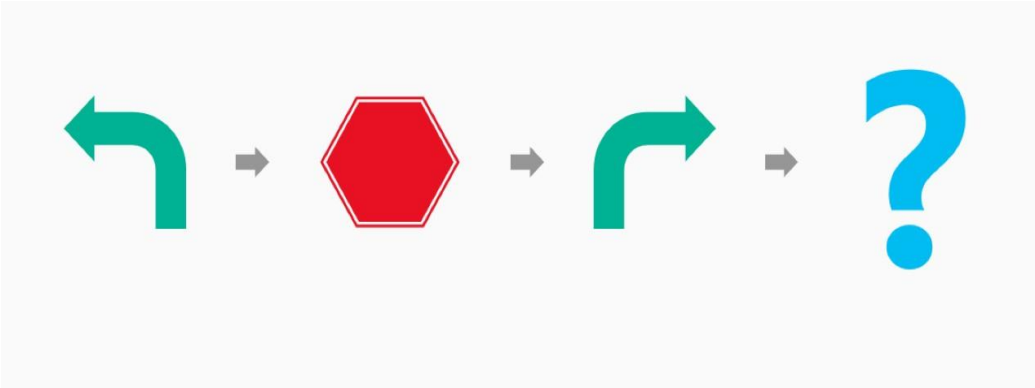
Aprendizaje de refuerzo

- El modelo percibe el entorno, toma acciones y realiza ajustes y elecciones en función del estado y el premio o el castigo.



Aprendizaje de refuerzo - Mejor comportamiento

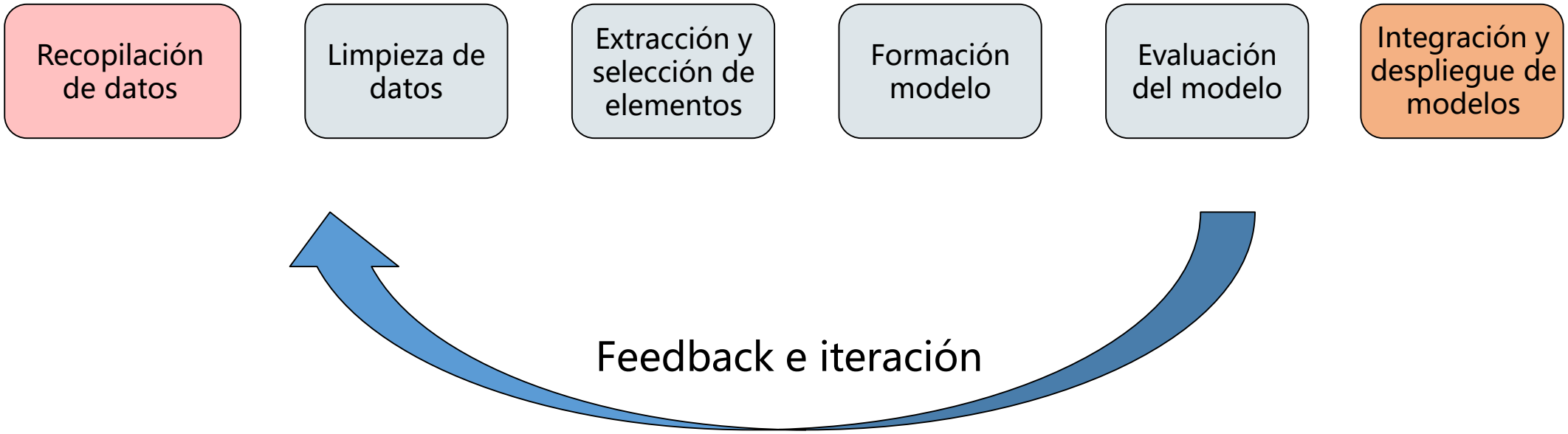
- Aprendizaje de refuerzo: siempre busca mejores comportamientos. El aprendizaje del refuerzo está dirigido a máquinas o robots.
  - Autopiloto: ¿Debería frenar o acelerar cuando la luz amarilla comienza a brillar?
  - Robot de limpieza: ¿Debería seguir funcionando o volver a cargarse?



Contenido

1. Algoritmo de Machine Learning
2. Clasificación de Machine Learning
3. **Proceso de Machine Learning**
4. Otros métodos clave de Machine Learning
5. Algoritmos comunes de Machine Learning
6. Caso práctico

Proceso de Machine Learning



Concepto básico de Machine Learning: Dataset

- Dataset: una colección de datos utilizados en tareas de M.L. Cada registro de datos se llama una muestra. Los eventos o atributos que reflejan el desempeño o la naturaleza de una muestra en un aspecto particular se llaman características.
- Set de entrenamiento: conjunto de datos (dataset) utilizado en el proceso de entrenamiento, en el que cada muestra se denomina una muestra de entrenamiento. El proceso de creación de un modelo a partir de datos se llama aprendizaje (entrenamiento).
- Set de pruebas: Prueba se refiere al proceso de uso del modelo obtenido después del aprendizaje para la predicción. El conjunto de datos utilizado se llama un conjunto de pruebas, y cada muestra es llamada una muestra de pruebas.

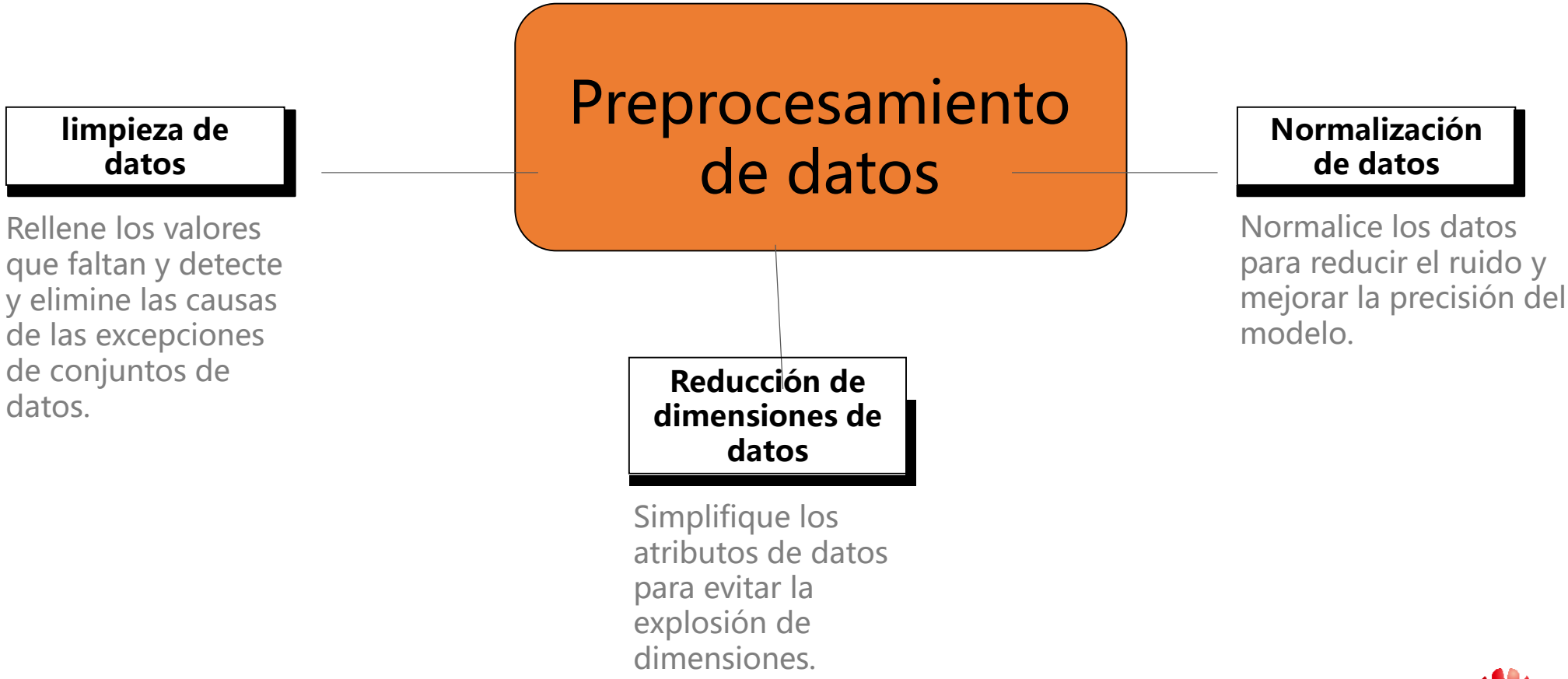
Vista general de la verificación de datos

- Formato típico del conjunto de datos

		Característica 1	Característica 2	Característica 3	Etiqueta
	No.	Area	Distritos escolares	Dirección	Precio de la casa
Set de Entrenamiento	1	100	8	South	1000
	2	120	9	Southwest	1300
	3	60	6	North	700
	4	80	9	Southeast	1100
Set de prueba	5	95	3	South	850

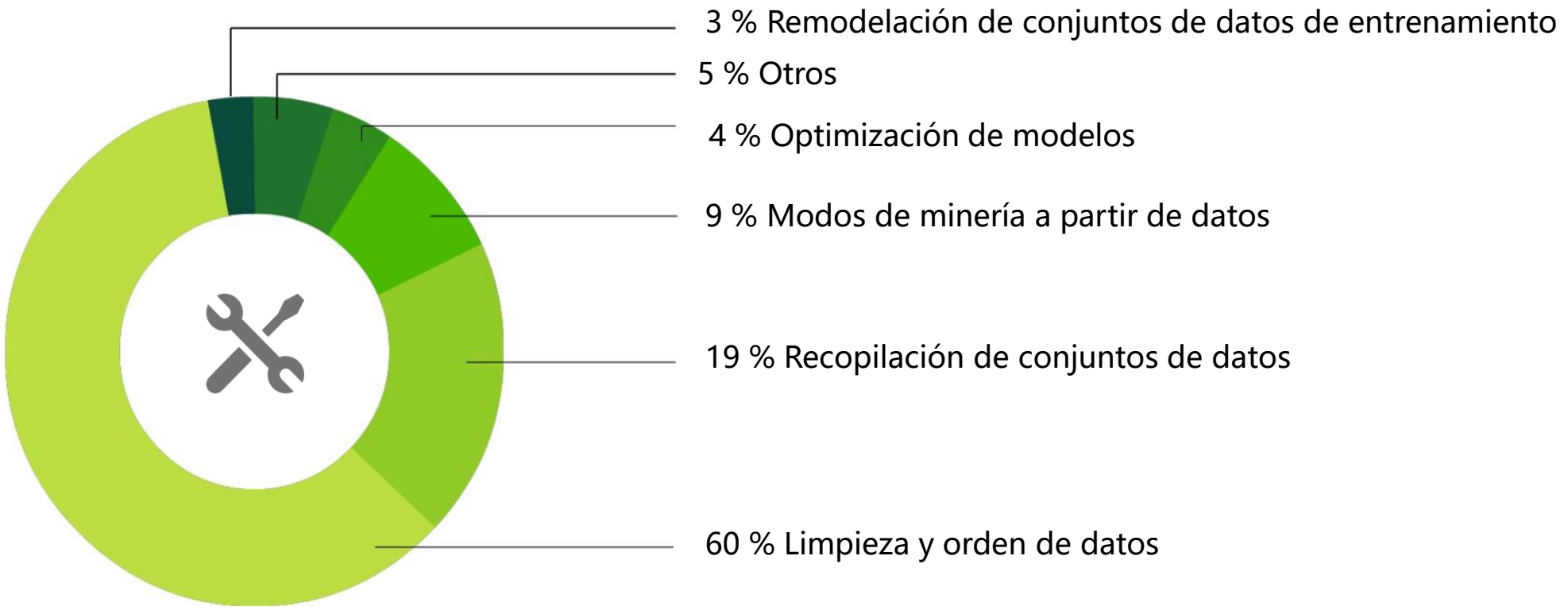
Importancia del procesamiento de datos

- Los datos son cruciales para los modelos. Es el techo de las capacidades del modelo. Sin buenos datos, no hay un buen modelo.



Tareas de la limpieza de datos

- Estadísticas sobre el trabajo de los científicos en M.L.



Informe CrowdFlower Data Science 2016

Limpieza de datos

- La mayoría de los modelos de ML procesan características, que suelen ser representaciones numéricas de variables de entrada que se pueden utilizar en el modelo.
- En la mayoría de los casos, los datos recogidos pueden ser utilizados por algoritmos sólo después de ser preprocesados. Las operaciones de preprocesamiento incluyen las siguientes:
  - Filtrado de datos
  - Procesamiento de datos perdidos
  - Procesamiento de posibles excepciones, errores o valores anormales
  - Combinación de datos de múltiples fuentes de datos
  - Consolidación de datos

Datos sucios (1)

- Generalmente, los datos reales pueden tener algunos problemas de calidad.
  - Incompletitud: contiene valores que faltan o los datos que carecen de atributos
  - Ruido: contiene registros incorrectos o excepciones.
  - Inconsistencia: contiene registros inconsistentes.

Datos sucios(2)

#	Id	Name	Birthday	Gender	Is Teacher	#Students	Country	City
1	111	John	31/12/1990	M	0	0	Ireland	Dublin
2	222	Mery	15/10/1978	F	1	15	Iceland	
3	333	Alice	19/04/2000	F	0	0	Spain	Madrid
4	444	Mark	01/11/1997	M	0	0	France	Paris
5	555	Alex	15/03/2000	A	1	23	Germany	Berlin
6	555	Peter	1983-12-01	M	1	10	Italy	Rome
7	777	Calvin	05/05/1995	M	0	0	Italy	Italy
8	888	Roxane	03/08/1948	F	0	0	Portugal	Lisbon
9	999	Anne	05/09/1992	F	0	5	Switzerland	Geneva
10	101010	Paul	14/11/1992	M	1	26	Ytali	Rome

Valor faltante

Valor inválido

Valor que debería estar en otra columna

Error de ortografía

Item duplicado-Inválido

Formato incorrecto

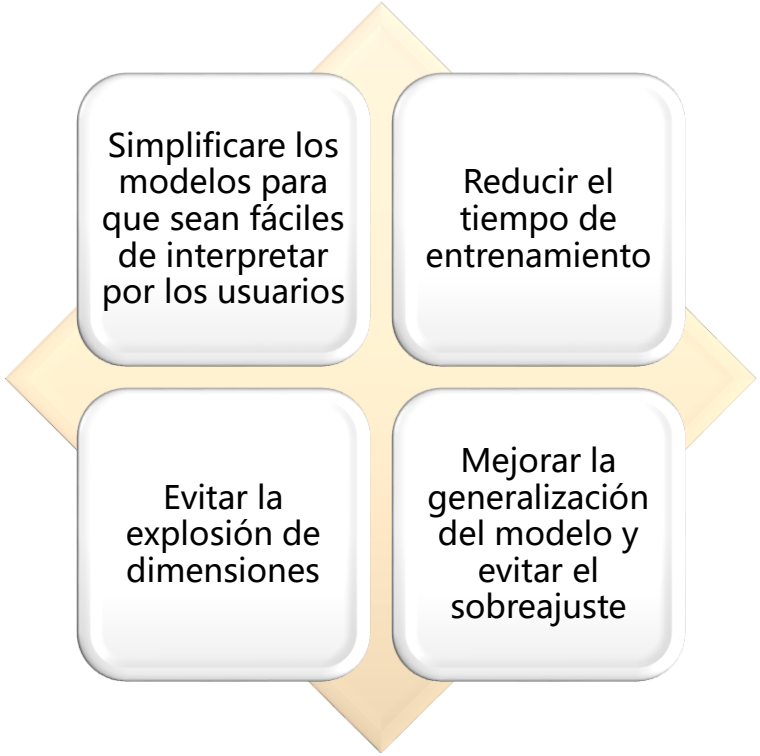
Dependencia de atributos

Conversión de datos

- Después de ser preprocesados, los datos deben ser convertidos en un formulario de representación adecuado para el modelo de Machine Learning. Los formularios comunes de conversión de datos incluyen los siguientes:
  - Con respecto a la clasificación, los datos de las categorías se codifican en la representación numérica correspondiente.
  - Los datos de valor se convierten en datos de categoría para reducir el valor de las variables (para la segmentación de edad).
  - Otros datos
    - En texto, la palabra se convierte en un vector de palabras a través de incrustaciones de palabras (generalmente usando el modelo word2vec, el modelo BERT, etc.)
    - Procesar datos de imagen (espacio de color, escala de grises, cambio geométrico, función Haar y mejora de imagen)
  - Ingeniería de características
    - Normalizar características para garantizar los mismos rangos de valores para las variables de entrada del mismo modelo.
    - Expansión de características : Combine o convierta variables existentes para generar nuevas caracteristicas, como el promedio.

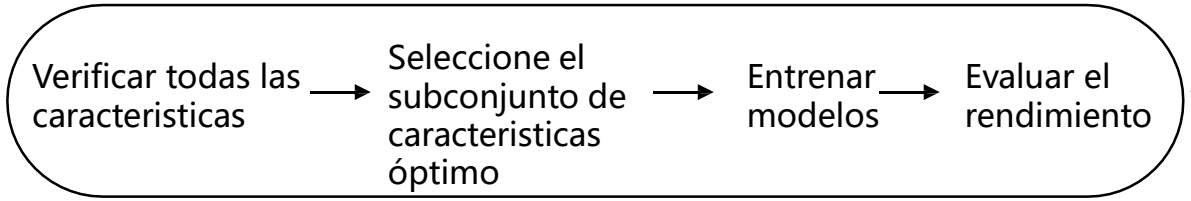
Necesidad de la selección de características

- Generalmente, un conjunto de datos tiene muchas características, algunas de las cuales pueden ser redundantes o irrelevantes para el valor a predecir.
- La selección de características es necesaria en los siguientes aspectos:



Métodos de selección de características - Filtro

- Los métodos de filtrado son independientes del modelo durante la selección de características.



Procedimiento de un método de filtro

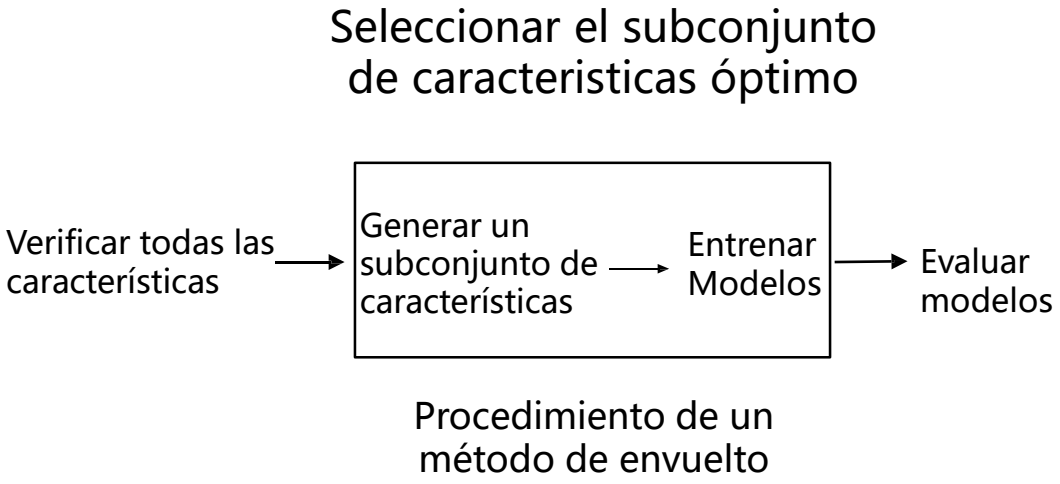
Al evaluar la correlación entre cada característica y el atributo de destino, estos métodos utilizan una medida estadística para asignar un valor a cada característica. Las características se clasifican por puntuación, lo que ayuda a preservar o eliminar características específicas.

Métodos comunes
- Coeficiente de correlación de Pearson
- Coeficiente de chi-cuadrado
- Información mutua

Limitaciones
- El método de filtrado tiende a seleccionar variables redundantes ya que no se tiene en cuenta la relación entre las características.

Métodos de selección de elementos – Wrapper (envuelto)

- Los métodos de envuelto utilizan un modelo de predicción para puntuar subconjuntos de características.



Procedimiento de un método de envuelto

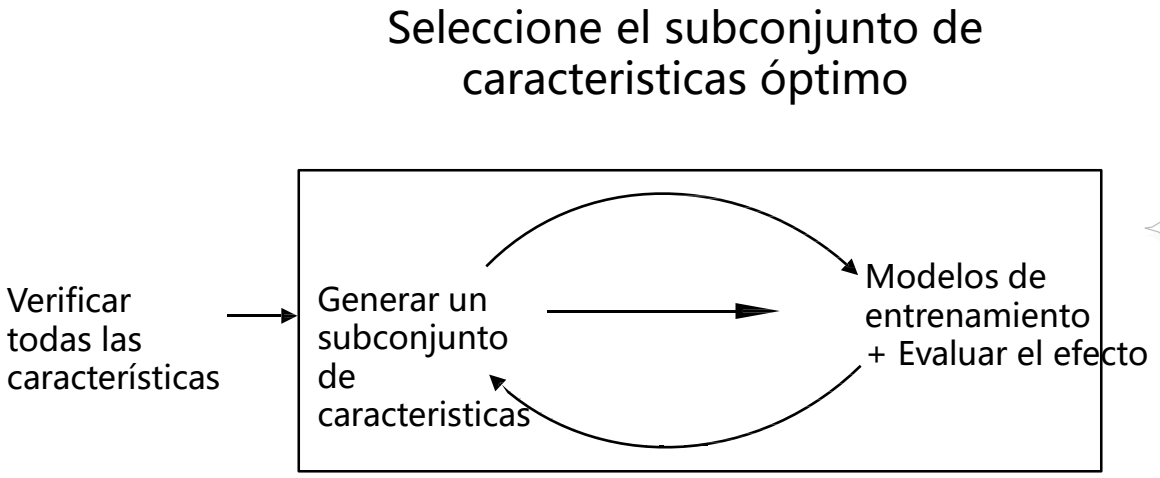
Los métodos de envuelto consideran la selección de características como un problema de búsqueda para el cual se evalúan y comparan diferentes combinaciones. Un modelo predictivo se utiliza para evaluar una combinación de características y asignar una puntuación basada en la precisión del modelo.

Métodos comunes
- Eliminación de funciones recursivas (RFE)

Limitaciones
- Los métodos de envuelto entrenan un nuevo modelo para cada subconjunto, lo que da lugar a un **gran número de cálculos.**
- Normalmente se proporciona un conjunto de características con el mejor rendimiento para un tipo específico de modelo.

Métodos de selección de caracterísitcas – Integrados (Embedded)

- Los métodos integrados consideran la selección de características como parte de la construcción de modelos.



Procedimiento de un método integrado

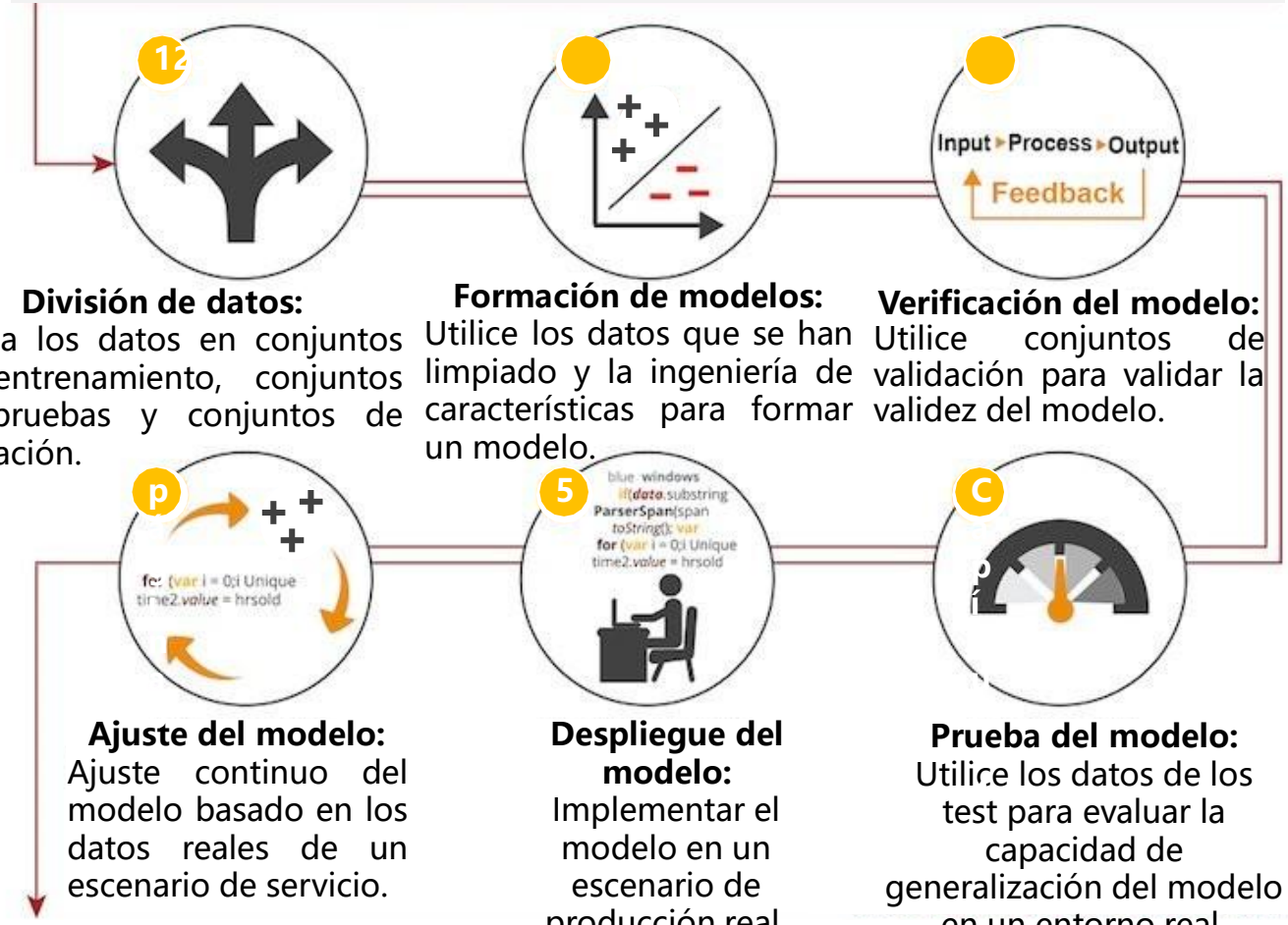
El método de selección de características incorporadas más común es el **método de regularización**.

Los métodos de regularización también se llaman métodos de penalización que introducen restricciones adicionales en la optimización de un algoritmo predictive, que presiona al modelo hacia una menor complejidad y reducen el número de características.

Métodos comunes
- Regresión Lasso
- Regresión Ridge

Procedimiento general para la construcción de un modelo

Procedimiento de construcción del modelo



División de datos:
Divida los datos en conjuntos de entrenamiento, conjuntos de pruebas y conjuntos de validación.

Formación de modelos:
Utilice los datos que se han limpiado y la ingeniería de características para formar un modelo.

Verificación del modelo:
Utilice conjuntos de validación para validar la validez del modelo.

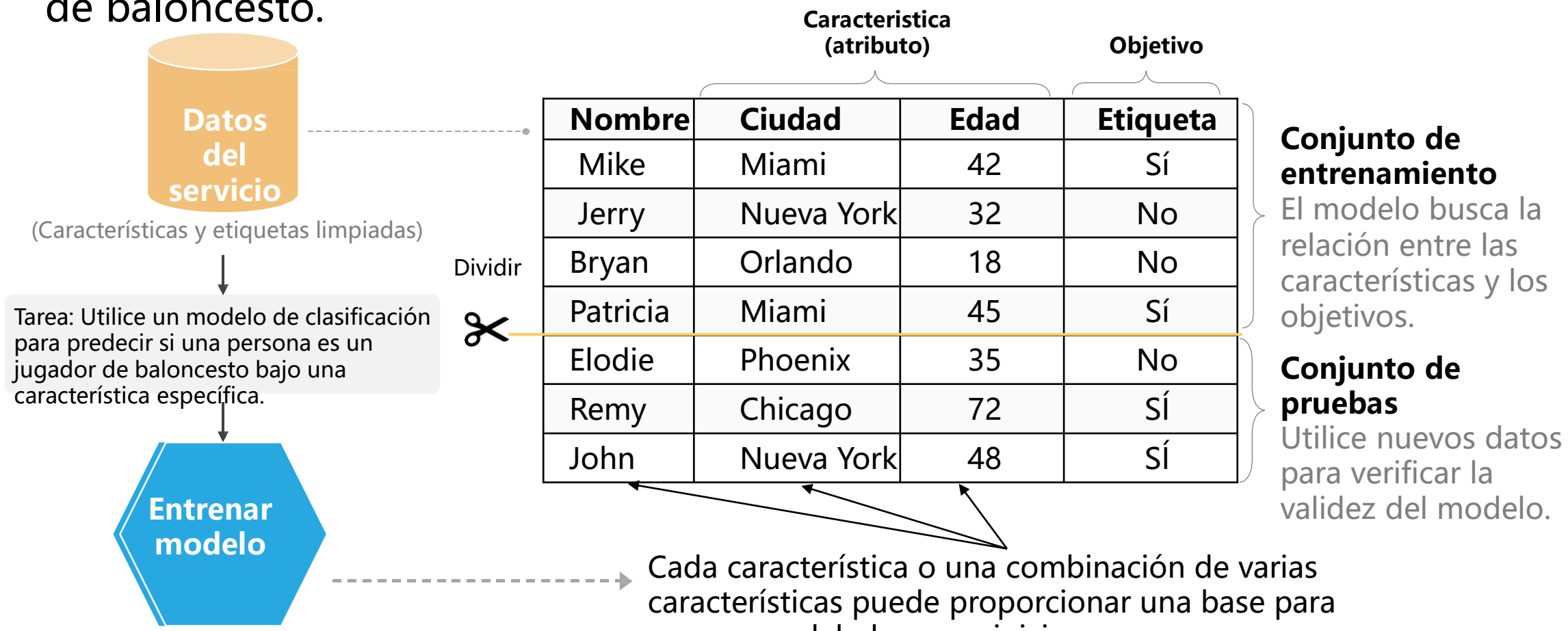
Ajuste del modelo:
Ajuste continuo del modelo basado en los datos reales de un escenario de servicio.

Despliegue del modelo:
Implementar el modelo en un escenario de producción real.

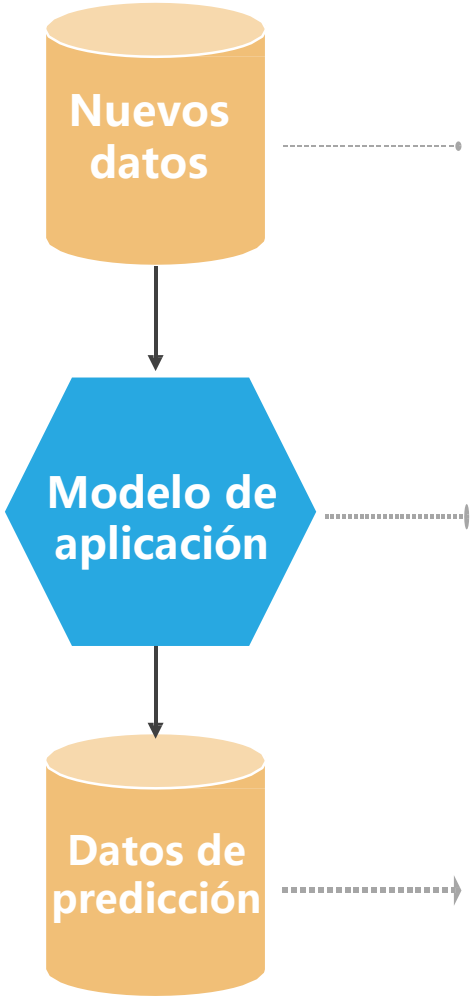
Prueba del modelo:
Utilice los datos de los test para evaluar la capacidad de generalización del modelo en un entorno real.

Ejemplos de aprendizaje supervisado - Fase de aprendizaje

- Utilice el modelo de clasificación para predecir si una persona es un jugador de baloncesto.



Ejemplos de aprendizaje supervisado - Fase de predicción



Nombre	Ciudad	Edad	Etiqueta
Marine	Miami	45	?
Julien	Miami	52	?
Fred	Orlando	20	?
Michelle	Boston	34	?
Nicolas	Phoenix	90	?

Datos desconocidos
Datos recientes, no se sabe si las personas son jugadores de baloncesto.

IF ciudad = Miami → Probabilidad = +0.7
IF city= Orlando → Probabilidad = +0.2
IF edad > 42 → Probabilidad = +0,05*edad + 0.06
IF edad > 42 → Probabilidad = +0,01*edad + 0.02

Nombre	Ciudad	Edad	Predicción
Marine	Miami	45	**0,3**
Julien	Miami	52	**0,9**
Fred	Orlando	20	**0.6.**
Michelle	Boston	34	**0.5.**
Nicolas	Phoenix	90	**0,4**

Predicción de posibilidades
Aplicar el modelo a los nuevos datos para predecir si el cliente cambiará de proveedor.

¿Qué es un buen modelo?

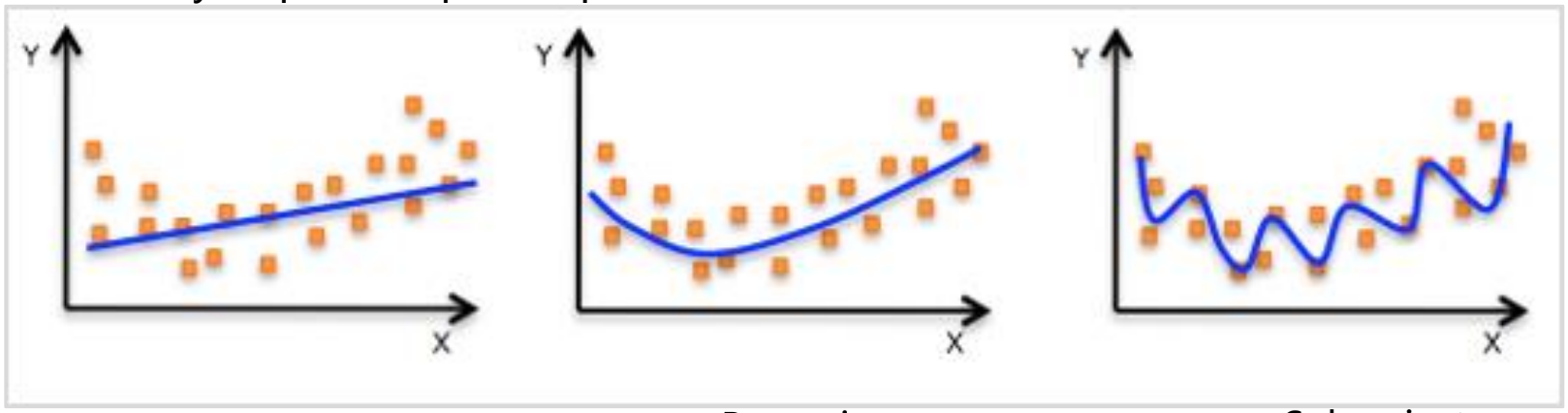
- **Capacidad de generalización**
  ¿Puede predecir con precisión los datos reales del servicio?
- **Interpretabilidad**
  ¿Es fácil interpretar el resultado de la predicción?
- **Velocidad de predicción**
  ¿Cuánto tiempo se tarda en predecir cada pieza de datos?
- **Practicidad**
  ¿La velocidad de predicción sigue siendo aceptable cuando el volumen de servicio aumenta con un gran volumen de datos?

Validez del modelo (1)

- Capacidad de generalización: El objetivo de Machine Learning es que el modelo obtenido después del aprendizaje tenga un buen rendimiento en nuevas muestras de datos, no sólo en las muestras utilizadas para el entrenamiento. La capacidad de aplicar un modelo a nuevas muestras de datos se denomina generalización o robustez.
- Error: diferencia entre el resultado de la muestra predicho por el modelo obtenido después del aprendizaje, y el resultado de la muestra real.
  - Error de entrenamiento: error que se obtiene al ejecutar el modelo en los datos de entrenamiento.
  - Error de generalización: error que se obtiene al ejecutar el modelo en nuevas muestras. Obviamente, preferimos un modelo con un error de generalización menor.
- Subajuste: se produce cuando el modelo o el algoritmo no se ajusta a los datos lo suficientemente bien.
- Sobreajuste: se produce cuando el error de formación del modelo obtenido después del aprendizaje es pequeño pero el error de generalización es grande (capacidad de generalización deficiente).

Validez del modelo (2)

- Capacidad del modelo: capacidad del modelo de ajustar funciones, que también se denomina complejidad del modelo.
  - Cuando la capacidad se adapta a la complejidad de la tarea y a la cantidad de datos de entrenamiento proporcionados, el efecto de algoritmo suele ser óptimo.
  - Los modelos con capacidad insuficiente no pueden resolver tareas complejas y pueden producirse subajustes
  - Un modelo de alta capacidad puede resolver tareas complejas, pero puede producirse un sobreajuste si la capacidad es mayor que la requerida por una tarea.



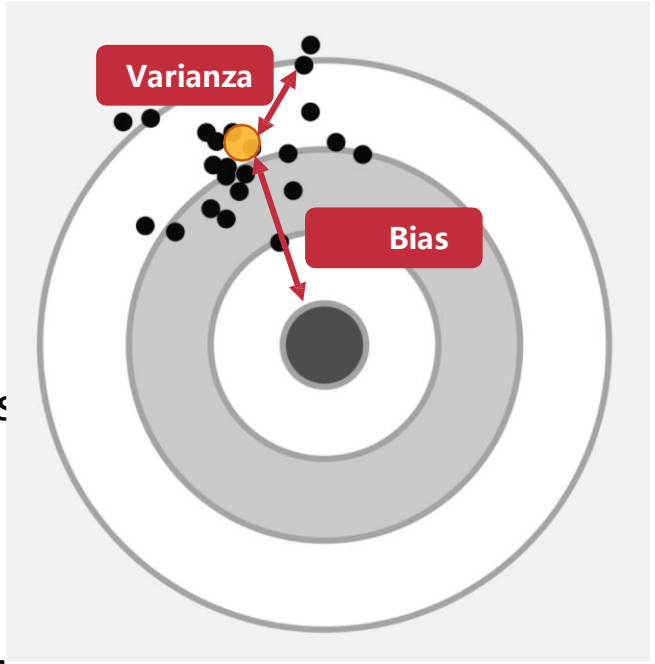
Subajuste
No todas las características son aprendidas.

Buen ajuste

Sobreajuste
Los ruidos se aprenden.

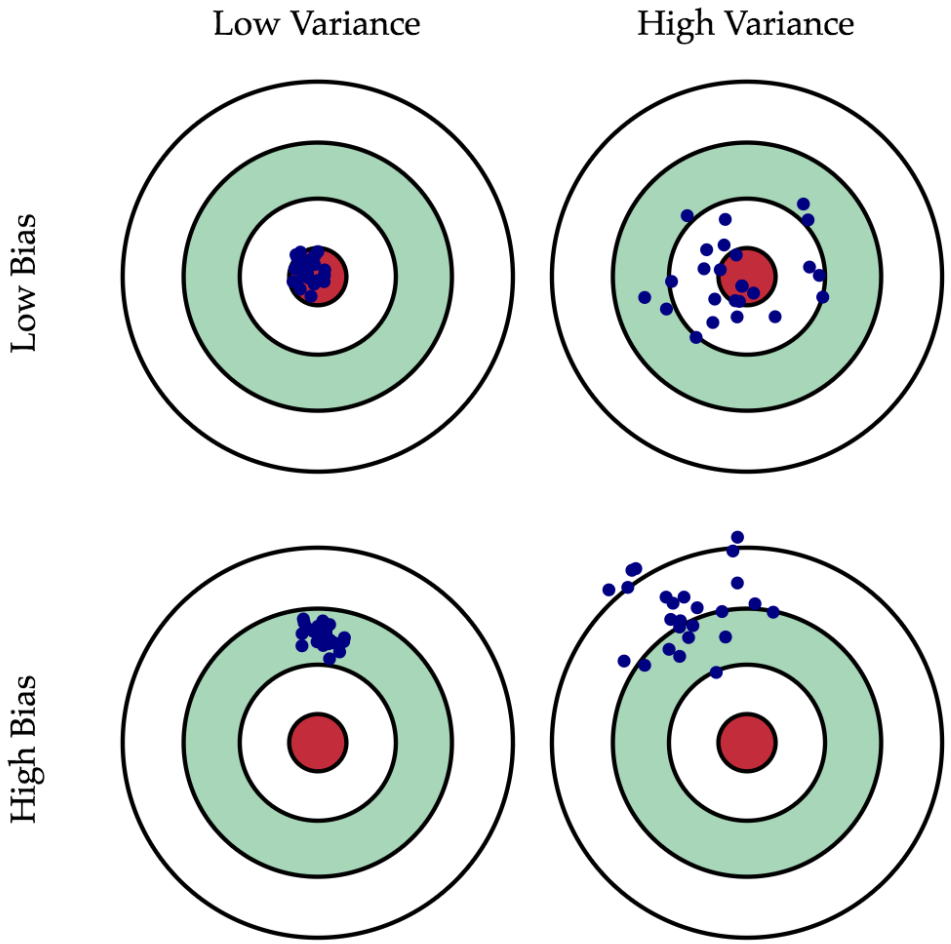
Sobreajuste Causa— Error

- Error total de predicción final = Bias2 + Varianza + Error Irreducible
- Generalmente, el error de predicción se puede dividir en dos tipos:
  - Error causado por "bias“ (sesgo)
  - Error causado por "variancia"
- Varianza:
  - Desfase del resultado de la predicción a partir del valor promedio
  - Error causado por la sensibilidad del modelo a pequeñas fluctuaciones el conjunto de entrenamiento
- Bias (sesgo):
  - Diferencia entre el valor previsto (o promedio) de predicción y el valor correcto que estamos intentando predecir.



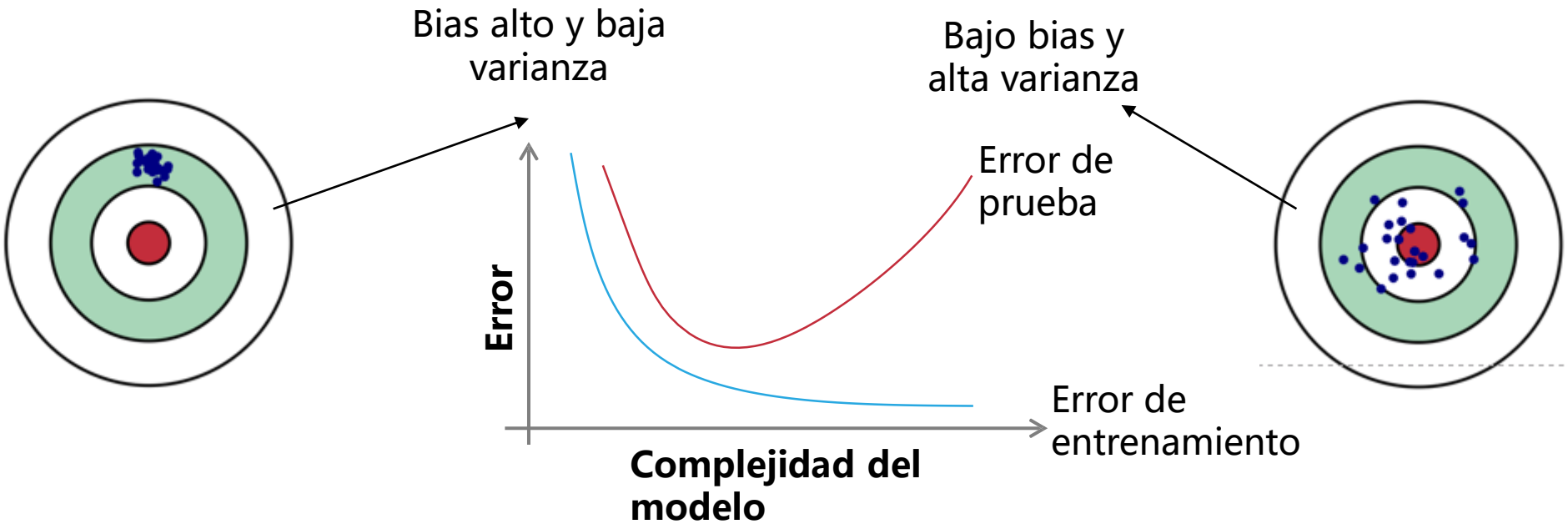
Varianza y Bias (sesgo)

- Las combinaciones de variación y sesgo son las siguientes:
  - Bajo sesgo y baja varianza –> Buen modelo
  - Bajo sesgo y alta varianza
  - Sesgo alto y baja varianzas
  - Sesgo alto y alta varianza –> Mal modelo
- Idealmente, queremos un modelo que pueda capturar con precisión las reglas en los datos de entrenamiento y resumir los datos invisibles (nuevos datos). Pero por lo general es imposible para el modelo completar ambas tareas al mismo tiempo.



Complejidad y error del modelo

- A medida que aumenta la complejidad del modelo, disminuye el error de entrenamiento.
- A medida que aumenta la complejidad del modelo, el error de prueba disminuye a un punto determinado y luego aumenta en la dirección inversa, formando una curva convexa.



Evaluación del desempeño de Machine Learning - Regresión

- Cuanto más cerca esté el Mean Absolute Error (MAE) de 0, mejor se ajustará el modelo a los datos de entrenamiento.

$$MAE = \frac{1}{\mathrm{m}}\sum_{i=1}^{m}|y_i - \hat{y}_i|$$

- Mean Square Error (MSE)

$$MSE = \frac{1}{\mathrm{m}}\sum_{i=1}^{\mathrm{m}}(y_i - \hat{y}_i)^2$$

- El rango de valores de R^2 es (–∞, 1]. Un valor mayor indica que el modelo puede ajustarse mejor a los datos de entrenamiento. TSS indica la diferencia entre muestras. RSS indica la diferencia entre el valor predicho y el valor de muestra.

$$R^2 = 1 - \frac{RSS}{TSS} = 1 - \frac{\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{m}(y_i - \bar{y}_i)^2}$$

Evaluación del rendimiento del Machine Learning - Clasificación (1)

Cantidad estimada / Cantidad real	yes	no	Total
Sí	TP	FN	P
No	FP	TN	N
Total	P'	N'	$P + N$

Matriz de confusión

- Terminos y definiciones:
- P: Positivo, indicando el número de casos reales positivos en los datos
  - N: Negativo, indicando el número de casos reales negativos en los datos
  - TP: True positive (positivo real), indica el numero de casos positivos que son clasificados correctamente por el clasificador
  - TN: True Negative (negativo real), indica el numero de casos negativos que son clasificados correctamente por el clasificador
  - FP: Falso Positivo, indica el número de casos positivos que son clasificados incorrectamente por el clasificador
  - FP: Falso Negativo, indica el número de casos negativos que son clasificados incorrectamente por el clasificador
  - **Matriz de confusión**: al menos una tabla $m \times m$. $CM_{i,\ j}$ de las primeras filas m y columnas m indica el número de casos que realmente pertenecen a la clase i pero que están clasificados en la clase j por el clasificador.
  - Idealmente, para un clasificador de alta precisión, la mayoría de los valores de predicción deben ubicarse en la diagonal de $CM\_1{,}1$ a $CM\_\ (m, m)$ de la tabla, mientras que los valores fuera de la diagonal son 0 o cerca de 0. Es decir, FP y FP son cerca de 0.

Evaluación del rendimiento de Machine Learning- Clasificación (2)

Medición	Proporción
Exactitud y tasa de reconocimiento	$\frac{TP + TN}{P + N}$
Tasa de error y tasa de clasificación errónea	$\frac{FP + FN}{P + N}$
Sensibilidad, tasa positiva real, y recordatorio	$\frac{TP}{P}$
Especificidad y tasa negativa real	$\frac{TN}{N}$
Precisión	$\frac{TP}{TP + FP}$
F_1, media armónica de la tasa de recordatorio y precisión	$\frac{2 \times precision \times recall}{precision + recall}$
F_β, donde β es un número no negativo y real	$\frac{(1 + \beta^2) \times precision \times recall}{\beta^2 \times precision + recall}$

Ejemplo de evaluación del rendimiento de Machine Learning

- Hemos entrenado un modelo de M.L. para identificar si el objeto de una imagen es un gato. Ahora usamos 200 imágenes para verificar el desempeño del modelo. Entre las 200 imágenes, los objetos en 170 imágenes son gatos, mientras que otros no lo son.
- El resultado de identificación del modelo es que los objetos en 160 imágenes son gatos, mientras que otros no lo son.

Precisión: $P = \frac{TP}{TP+FP} = \frac{140}{140+20} = 87.5\%$

Recall: $R = \frac{TP}{P} = \frac{140}{170} = 82.4\%$

Exactitud: $ACC = \frac{TP+TN}{P+N} = \frac{140+10}{170+30} = 75\%$

Cantidad estimada / Cantidad real	yes	no	Total:
yes	140.	30.	170.
no	20.	10.	30.
Total:	160.	40	200.

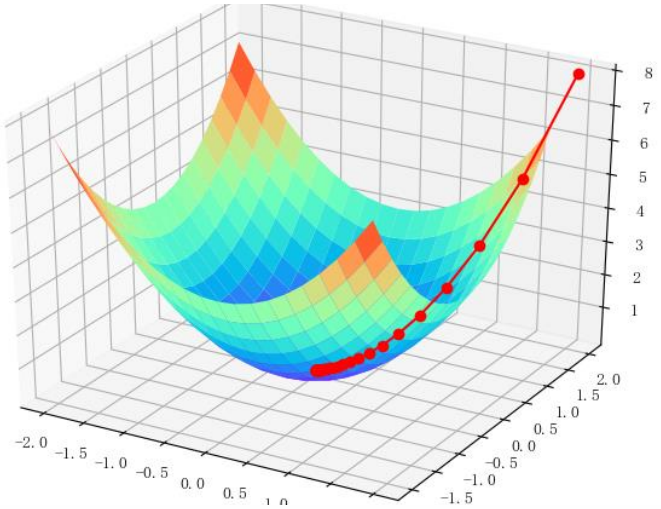
Contenidos

Método de entrenamiento de Machine Learning: Descenso de gradiente (1)

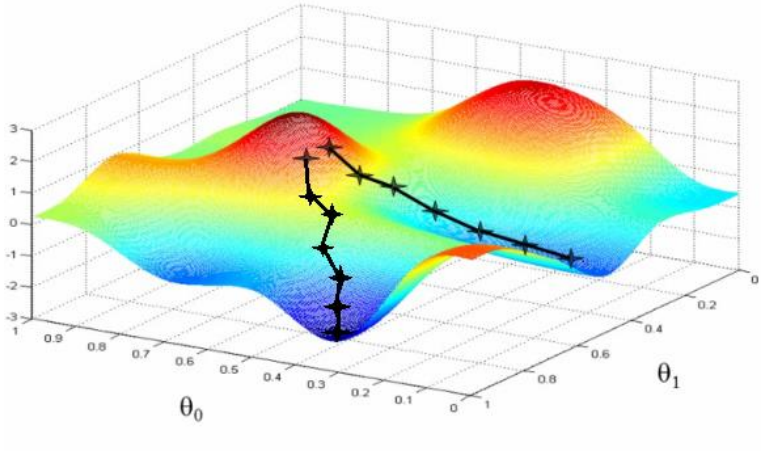
- El método de descenso de gradiente utiliza la dirección de gradiente negativa de la posición actual como dirección de búsqueda, que es la dirección más empinada. La fórmula es la siguiente:

$$w_{k+1} = w_k - \eta \nabla f_{w_k}(x^i)$$

Superficie de coste



- En la fórmula, η indica la tasa de aprendizaje y i indica el número de registro de datos i. El parámetro de peso w indica el cambio en cada iteración.
- Convergencia: el valor de la función objetivo cambia muy poco o se alcanza el número máximo de iteraciones.



Método de entrenamiento de Machine Learning: Descenso de gradiente (2)

- Descenso de Gradiente por Lotes (Batch Gradient Descent - BGD) utiliza las muestras (m en total) en todos los conjuntos de datos para actualizar el parámetro de peso basado en el valor de gradiente en el punto actual.

$$w_{k+1} = w_k - \eta \frac{1}{m} \sum_{i=1}^{m} \nabla f_{w_k}(x^i)$$

- Descenso de Gradiente Estocástico (Stochastic Gradient Descent - SGD) selecciona al azar una muestra en un conjunto de datos para actualizar el parámetro de peso basado en el valor de gradiente en el punto actual.

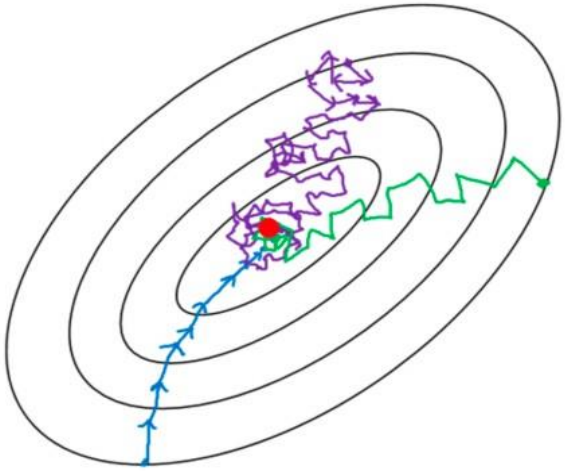
$$w_{k+1} = w_k - \eta \nabla f_{w_k}(x^i)$$

- Mini-Batch Gradient Descent (Mini-Batch Gradient Descent - MBGD) combina las características de BGD y SGD y selecciona los gradientes de n muestras en un conjunto de datos para actualizar el parámetro de peso.

$$w_{k+1} = w_k - \eta \frac{1}{n} \sum_{i=\mathrm{t}}^{t+n-1} \nabla f_{w_k}(x^i)$$

Metodología de entrenamiento de máquinas - Descenso de Gradiente (3)

- Comparación de los tres métodos de descenso de gradiente
  - En el SGD, las muestras seleccionadas para cada entrenamiento son estocásticas. Esta inestabilidad hace que la función de pérdida sea inestable o incluso causa desplazamiento inverso cuando la función de pérdida disminuye hasta el punto más bajo.
  - BGD tiene la mayor estabilidad, pero consume demasiados recursos informáticos. MBGD es un método que equilibra SGD y BGD.



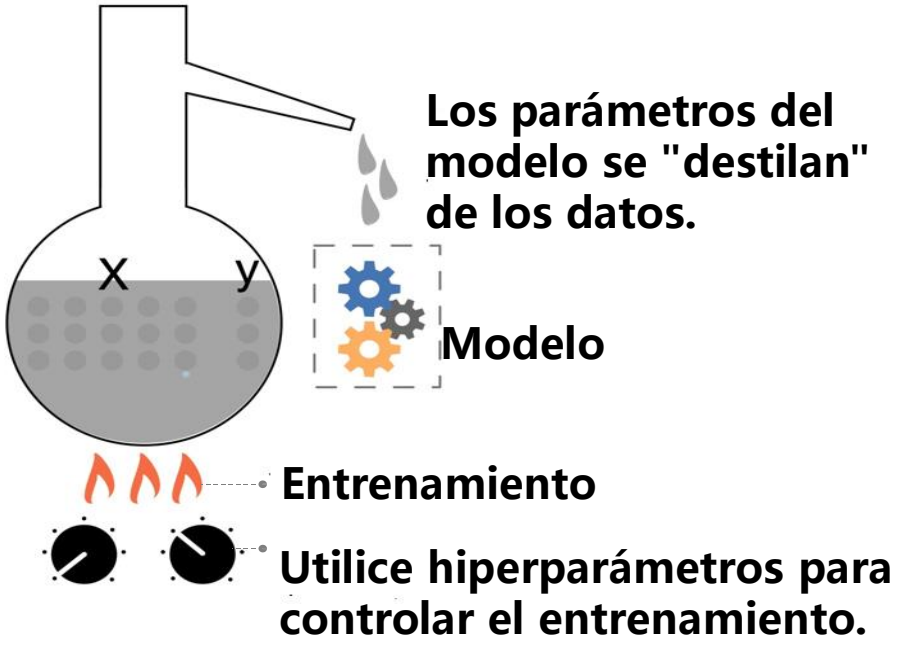
BGD
Usa todas las muestras de entrenamiento para entrenar cada vez.

SGD
Usa una muestra de entrenamiento cada vez.

MBGD
Usa un cierto número de muestras de entrenamiento para cada entrenamiento.

Parámetros e hiperparámetros en modelos

- El modelo contiene no sólo parámetros sino también hiperparámetros. El objetivo es permitir que el modelo aprenda los parámetros óptimos.
  - Los parámetros se aprenden automáticamente por los modelos.
  - Los hiperparámetros se configuran manualmente.



Hiperparámetros de un modelo

- Con frecuencia se utilizan en los procesos de estimación de parámetros del modelo.
- A menudo especificados por el practicante.
- A menudo se puede configurar usando heurística.
- A menudo sintonizada para un problema de modelado predictivo dado.

Los hiperparámetros de los modelos son configuraciones externas de los modelos.

- λ durante la regression LASSO/RIDGE
- Tasa de aprendizaje apra entrenar una red neuronal, número de iteraciones, tamaño de lote, función de activación, y número de neuronas..
- C y σ en máquinas de vectores de soporte (Support vector Machine – SVM).
- K en k-vecino más cercano (K in K-Nearest Neighbor KNN)
- Número de árboles en un bosque aleatorio

Hiperparámetros comunes del modelo

Procedimiento y método de búsqueda de hiperparámetros

Procedimiento de búsqueda de hiperparámetros

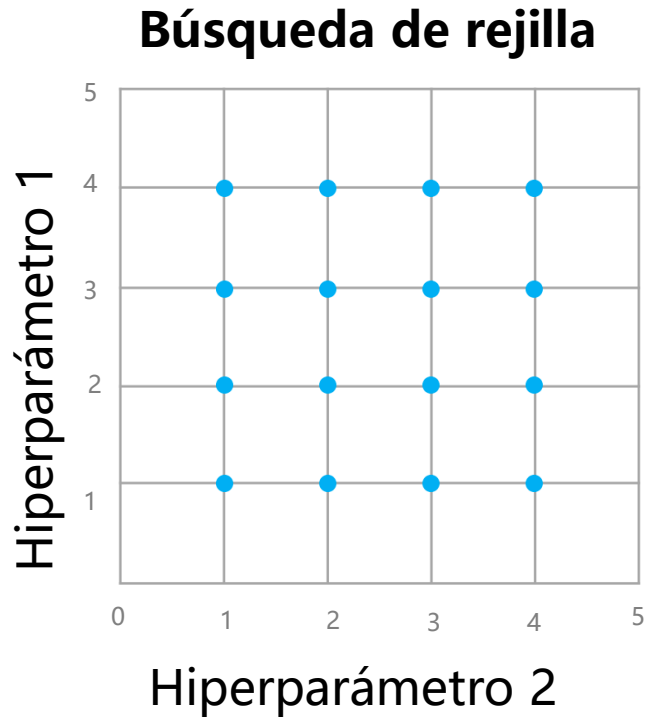
1. Dividir un conjunto de datos en un conjunto de entrenamiento, conjunto de validación y conjunto de pruebas.
2. Optimización de los parámetros del modelo utilizando el conjunto de entrenamiento basado en los indicadores de rendimiento del modelo.
3. Buscar los hiperparámetros del modelo utilizando el conjunto de validación basado en los indicadores de rendimiento del modelo.
4. Realizar los pasos 2 y 3 alternadamente. Finalmente, determinar los parámetros e hiperparámetros del modelo y evaluar el modelo utilizando el conjunto de pruebas.

Algoritmo de búsqueda (paso 3)

- **Búsqueda de rejilla**
- **Búsqueda aleatoria**
- Búsqueda inteligente heurística
- Búsqueda bayesiana

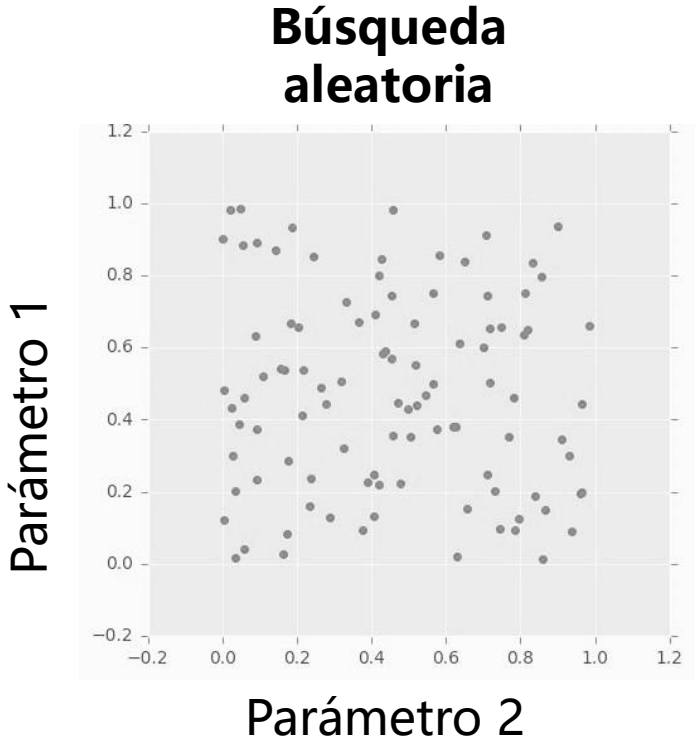
Método de búsqueda de hiperparámetros - Búsqueda de rejilla

- La búsqueda de la rejilla intenta **buscar exhaustivamente** todas las combinaciones de hiperparámetros posibles para formar una rejilla de valores de hiperparámetros.
- En la práctica, el rango de valores de hiperparámetros a buscar se especifica manualmente.
- La búsqueda en la rejilla es un método costoso y que requiere mucho tiempo.
  - Este método funciona bien cuando el número de hiperparámetros es relativamente pequeño. Por lo tanto, se aplica generalmente a los algoritmos de Machine Learning, pero no a las redes neuronales (Ver la parte de Aprendizaje Profundo - Deep Learning).



Método de búsqueda de hiperparámetros - Búsqueda aleatoria

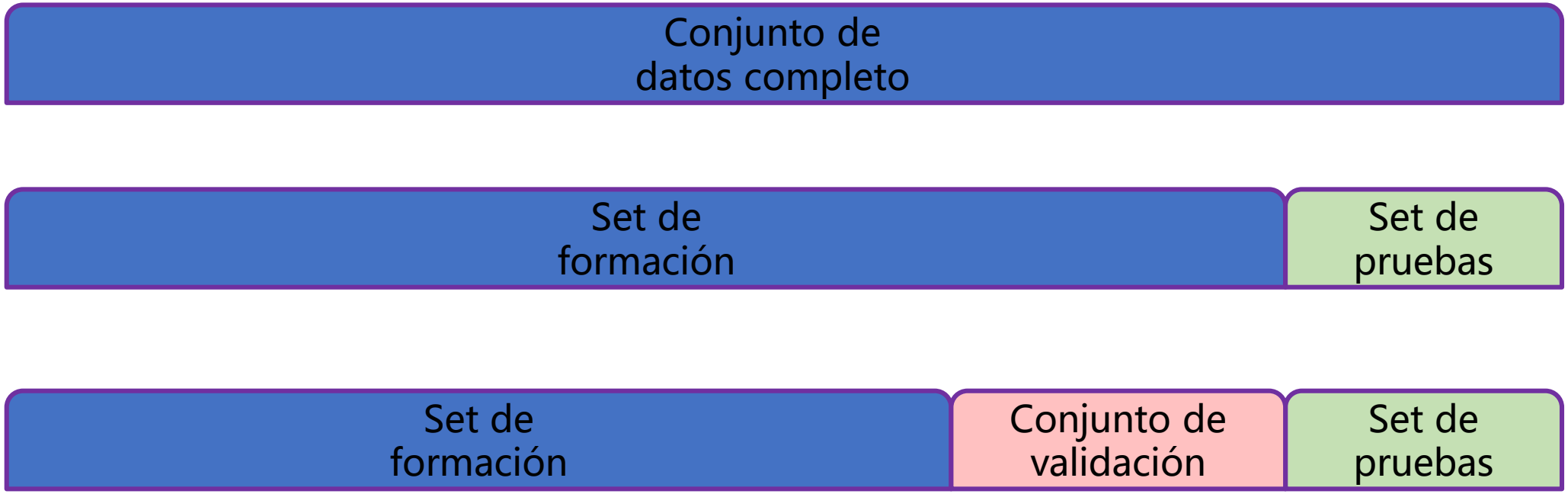
- Cuando el espacio de búsqueda de hiperparámetros es grande, la **búsqueda aleatoria** es mejor que la búsqueda en la rejilla.
- En la búsqueda aleatoria, cada configuración se muestrea de la distribución de los posibles valores de parámetros, en un intento de encontrar el mejor subconjunto de hiperparámetros.
- Notas:
  - La búsqueda se realiza dentro de un rango amplio, que se reducirá en función del lugar donde aparece el mejor resultado.
  - Algunos hiperparámetros son más importantes que otros, y la desviación de búsqueda se verá afectada durante la búsqueda aleatoria.



Validación cruzada (1)

- **Cross validation (Validación Cruzada)**: Es un método de análisis estadístico que se utiliza para validar el desempeño de un clasificador. La idea básica es dividir el conjunto de datos original en dos partes: conjunto de entrenamiento y conjunto de validación. Entrene al clasificador usando el conjunto de entrenamiento y pruebe el modelo usando el conjunto de validación para verificar el rendimiento del clasificador
- **k-fold cross validation ($K-CV$):**
  - Divida los datos brutos en k grupos (generalmente divididos de manera uniforme).
  - Use cada subconjunto como un conjunto de validación y use los otros subconjuntos k − 1 como el conjunto de entrenamiento. Se pueden obtener un total de k modelos.
  - Utilice la precisión de clasificación media de los conjuntos de validación final de los modelos k como indicador de rendimiento del clasificador $K-CV$.

Validación cruzada (2)

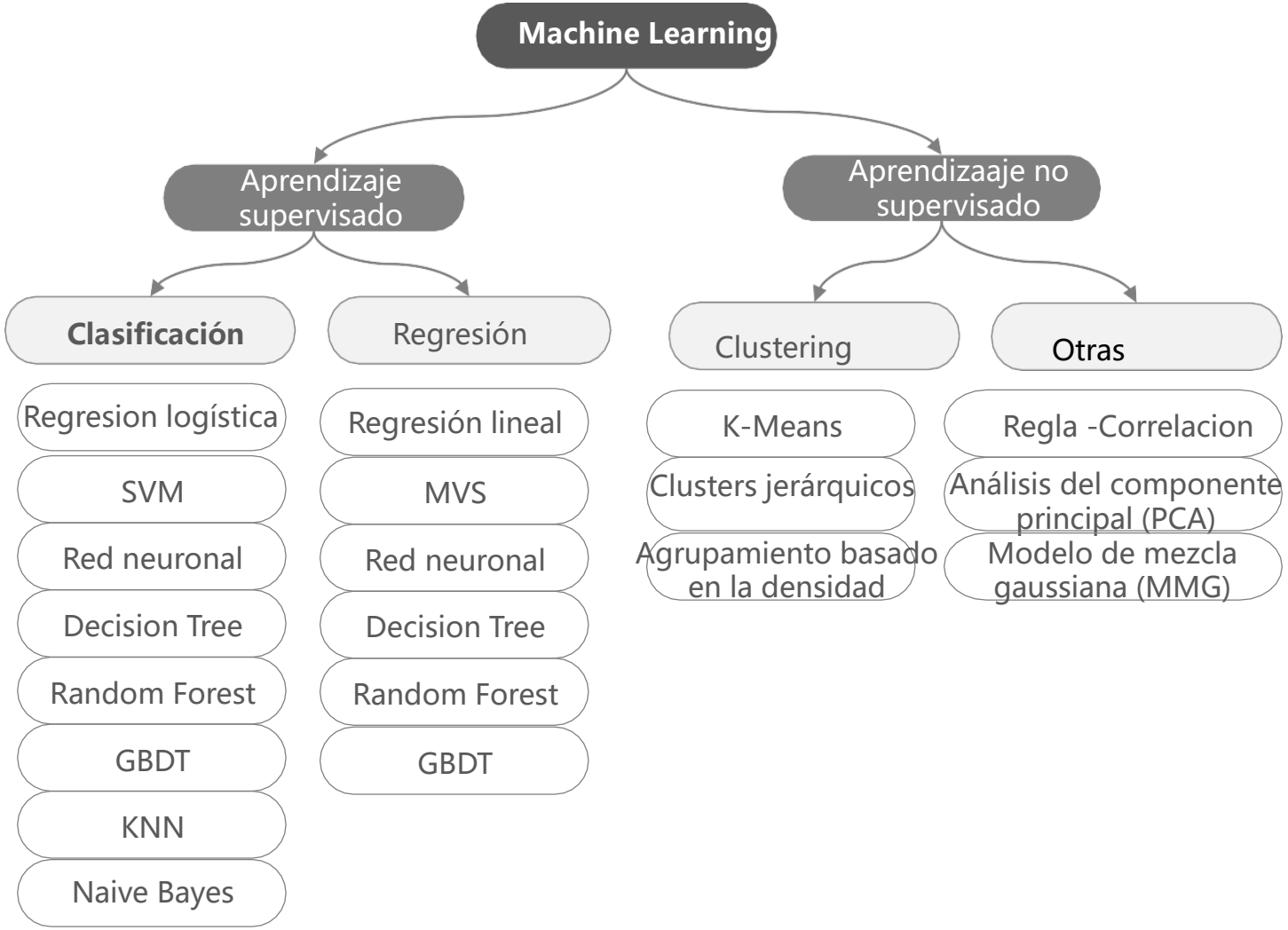


- Nota: El valor K en la validación cruzada por K es también un hiperparámetro.

Contenidos

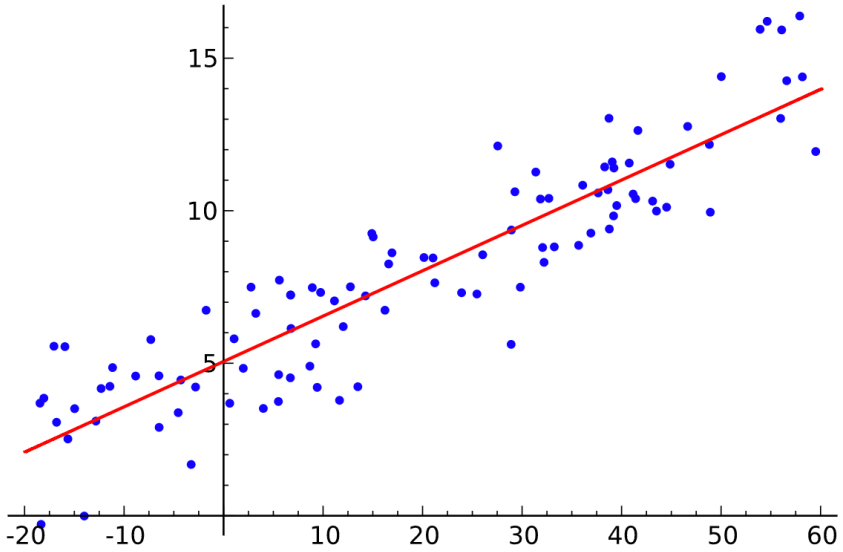
1. Definición de Machine Learning
2. Tipos de Machine Learning
3. Proceso de Machine Learning
4. Otros métodos de Machine Learning clave
5. **Algoritmos comunes de Machine Learning**
6. Caso práctico

Visión general del algoritmo de Machine Learning

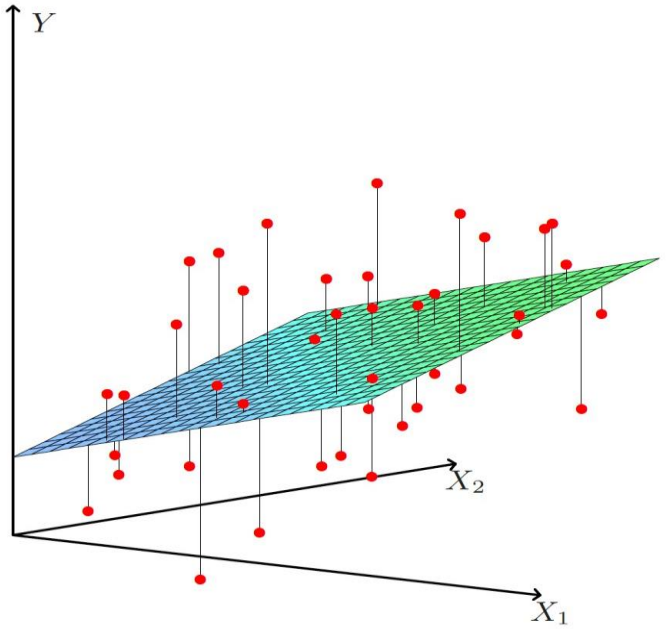


Regresión Lineal (1)

- Regresión lineal: un método de análisis estadístico para determinar las relaciones cuantitativas entre dos o más variables a través del análisis de regresión en estadísticas matemáticas.
- La regresión lineal es un tipo de aprendizaje supervisado.



Regresión lineal unaria



Regresión lineal multidimensional

Regresion Lineal (2)

- La función del modelo de regresión lineal es la siguiente, donde w indica el parámetro de ponderación, b indica el sesgo y x indica el atributo de la muestra.

$$h_w(x) = w^T x + b$$

- La relación entre el valor predicho por el modelo y el valor real es la siguiente, donde y indica el valor real y ε indica el error.

$$y = w^T x + b + \varepsilon$$

- El error ε está influenciado por muchos factores de forma independiente. Según el teorema del límite central, el error ε sigue una distribución normal. Según la función de distribución normal y la estimación de máxima verosimilitud, la función de pérdida de la regresión lineal es la siguiente:

$$J(w) = \frac{1}{2m} \sum \left(h_w(x) - y \right)^2$$

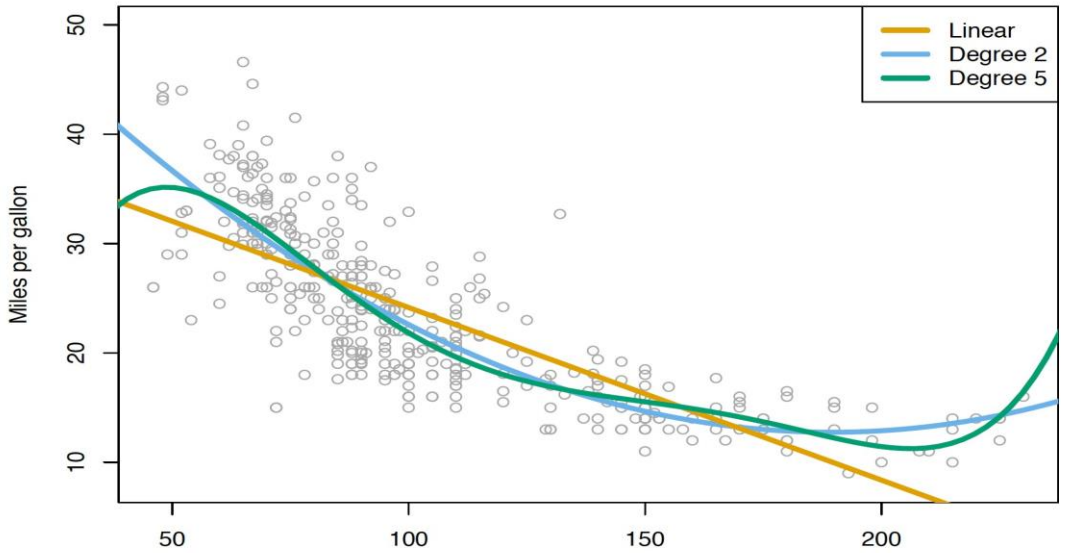
- Para que el valor previsto se acerque al valor real, debemos minimizar el valor de pérdida. Podemos usar el método de descenso de gradiente para calcular el parámetro de peso w cuando la función de pérdida alcanza el mínimo y luego completar la construcción del modelo.

Extensión de regresión lineal - Regresión polinómica

- La regresión polinómica es una extensión de la regresión lineal. En general, la complejidad de un conjunto de datos excede la posibilidad de que se pueda ajustar por una línea recta. Es decir, la falta de adecuación obvia ocurre si se utiliza el modelo de regresión lineal original. La solución es utilizar la regresión polinómica.

$$h_w(x) = w_1 x + w_2 x^2 + \mathrm{L} \ + w_n x^n + b$$

- donde, la enésima potencia es una dimensión de regresión polinomial (grado).
- La regresión polinomial pertenece a la regresión lineal ya que la relación entre sus parámetros de peso w sigue siendo lineal, mientras que su no linealidad se refleja en la dimensión de la característica.



Comparación entre regresión lineal y regresión polinómica

Regresión lineal y prevención de Sobreajuste

- Los términos de regularización se pueden utilizar para reducir el sobreajuste. El valor de *w* no puede ser demasiado grande o demasiado pequeño en el espacio de muestra. Puede agregar una pérdida de suma cuadrada en la función de destino.

$$J(w)=\frac{1}{2m}\sum\left(h_w(x)-y\right)^2+\lambda\sum\|w\|_2^2$$

- Términos de regularización (norma): El término de regularización aquí se denomina L2-norma. La regresión lineal que utiliza esta función de pérdida también se denomina regresión Ridge.

$$J(w)=\frac{1}{2m}\sum\left(h_w(x)-y\right)^2+\lambda\sum\|w\|_1$$

- La regresión lineal con pérdida absoluta se denomina regresión de Lasso.

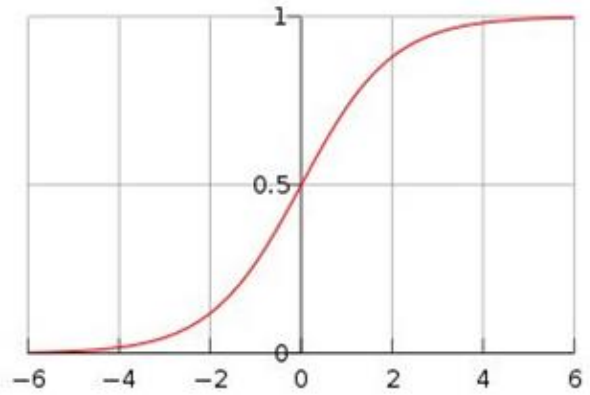
Regresión logistica(1)

- Regresión logística: el modelo de regresión logística se utiliza para resolver problemas de clasificación. El modelo se define de la siguiente manera:

$$P(Y = 1|x) = \frac{e^{wx+b}}{1 + e^{wx+b}}$$

$$P(Y = 0|x) = \frac{1}{1 + e^{wx+b}}$$

donde *w* indica el peso, *b* indica el sesgo, y *wx+b* se considera como la función lineal de *x*. Compare los dos valores de probabilidad anteriores. La clase con un valor de probabilidad más alto es la clase de *x*.



Regresión Logística (2)

- Tanto el modelo de regresión logística como el modelo de regresión lineal son modelos lineales generalizados. La regresión logística introduce factores no lineales (la función sigmoide) basados en la regresión lineal y establece umbrales, por lo que puede tratar los problemas de clasificación binaria.
- Según la función modelo de regresión logística, la función de pérdida de la regresión logística se puede estimar de la siguiente manera utilizando la estimación de máxima verosimilitud:

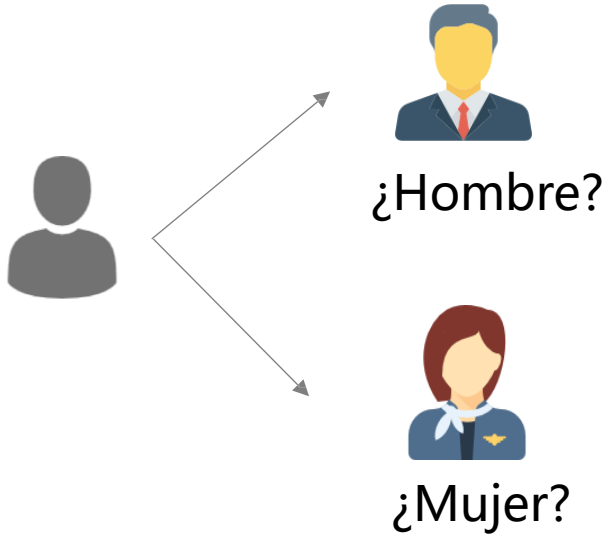
$$J(w) = -\frac{1}{m}\sum\left(y \ln h_w(x) + (1-y)\ln(1-h_w(x))\right)$$

- donde *w* indica el parámetro de peso, *m* indica el número de muestras, *x* indica la muestra e *y* indica el valor real. Los valores de todos los parámetros de peso *w* también se pueden obtener a través del algoritmo de descenso de gradiente.

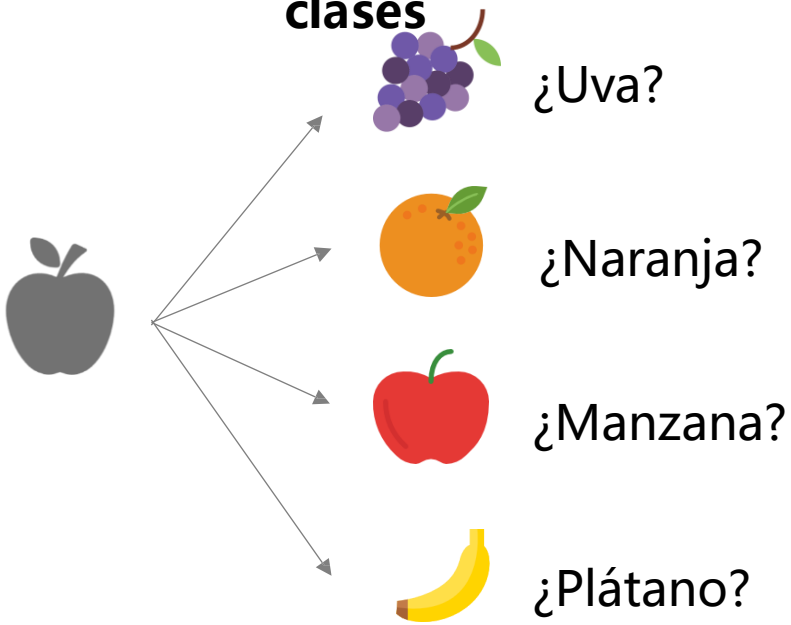
Extensión de regresión logística - Función Softmax (1)

- La regresión logística sólo se aplica a problemas de clasificación binaria. Para problemas de clasificación multi-clase, utilice la función Softmax.

Problema de clasificación binaria



Problema de clasificación de varias clases



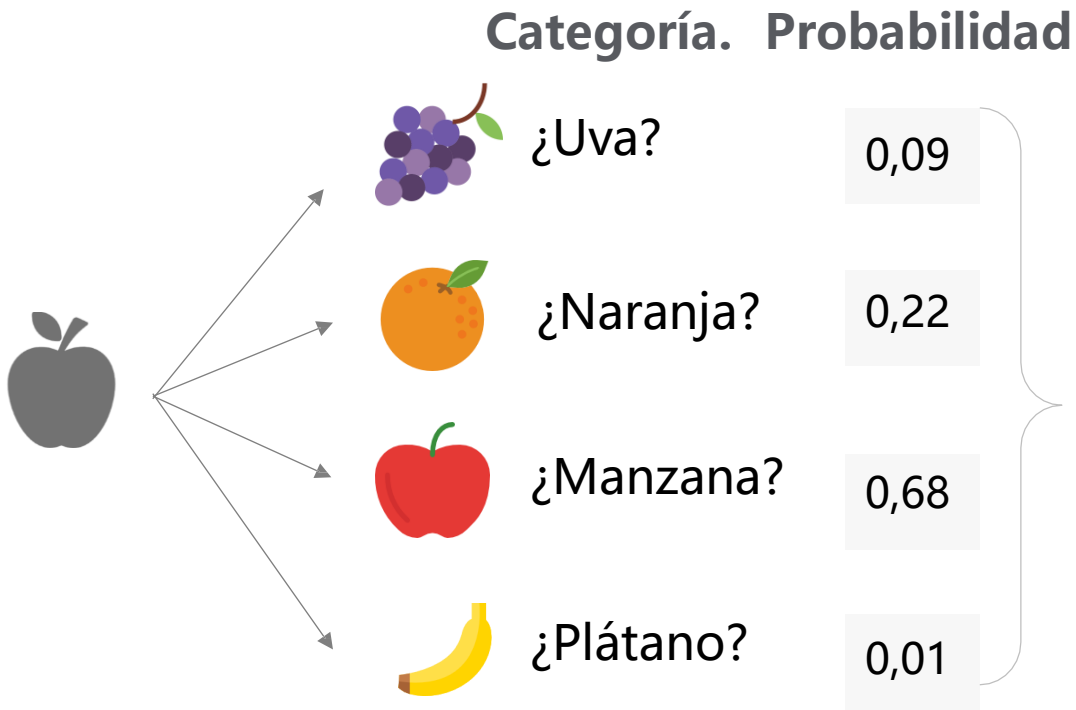
Extensión de regresión logística - Función Softmax (2)

- La regresión Softmax es una generalización de la regresión logística que podemos usar para la clasificación de clase K.
- La función Softmax se utiliza para asignar un vector K-dimensional de valores reales arbitrarios a otro vector K-dimensional de valores reales, donde cada elemento vectorial está en el intervalo (0, 1).
- La función de probabilidad de regresión de Softmax es la siguiente:

$$p(y=k \mid x;w)=\frac{e^{w_k^T x}}{\sum_{l=1}^{K} e^{w_l^T x}}, k=1,2\mathrm{L}\ \ ,K$$

Extensión de regresión logística - Función Softmax (3)

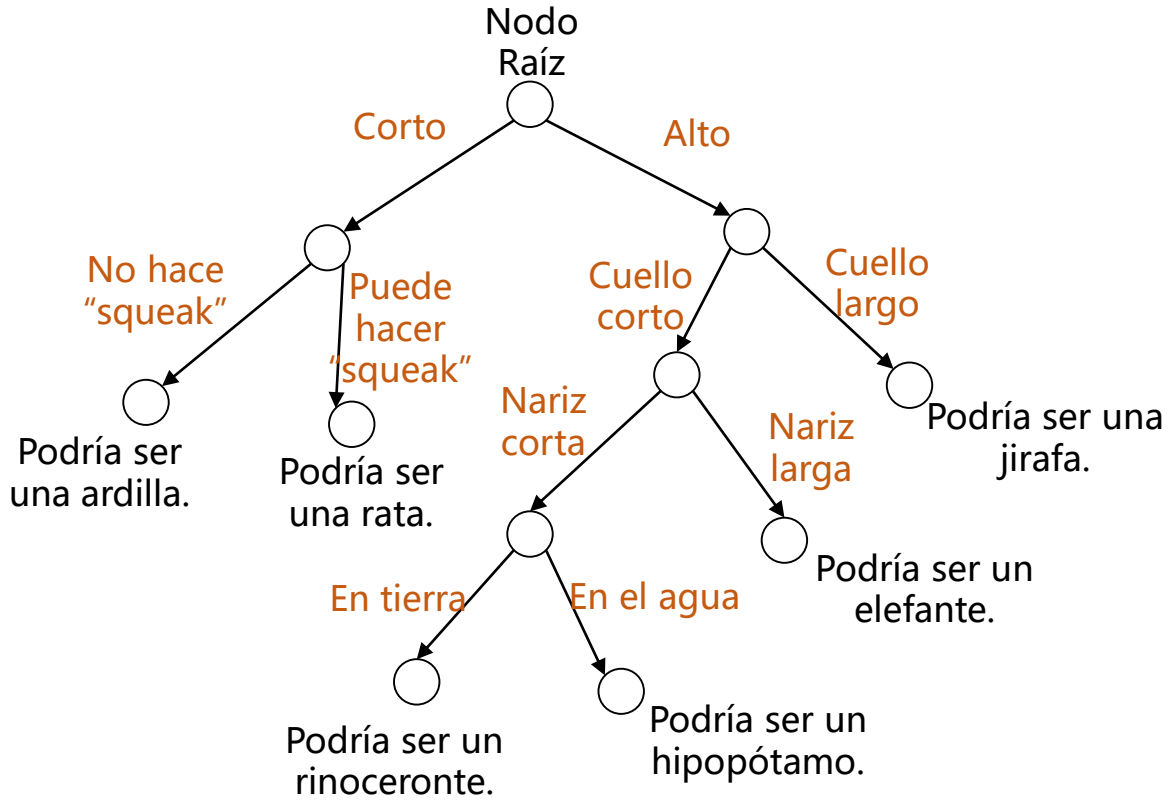
- Softmax asigna una probabilidad a cada clase en un problema de varias clases. Estas probabilidades deben sumar 1.
  - Softmax puede producir un formulario perteneciente a una clase particular. Ejemplo:



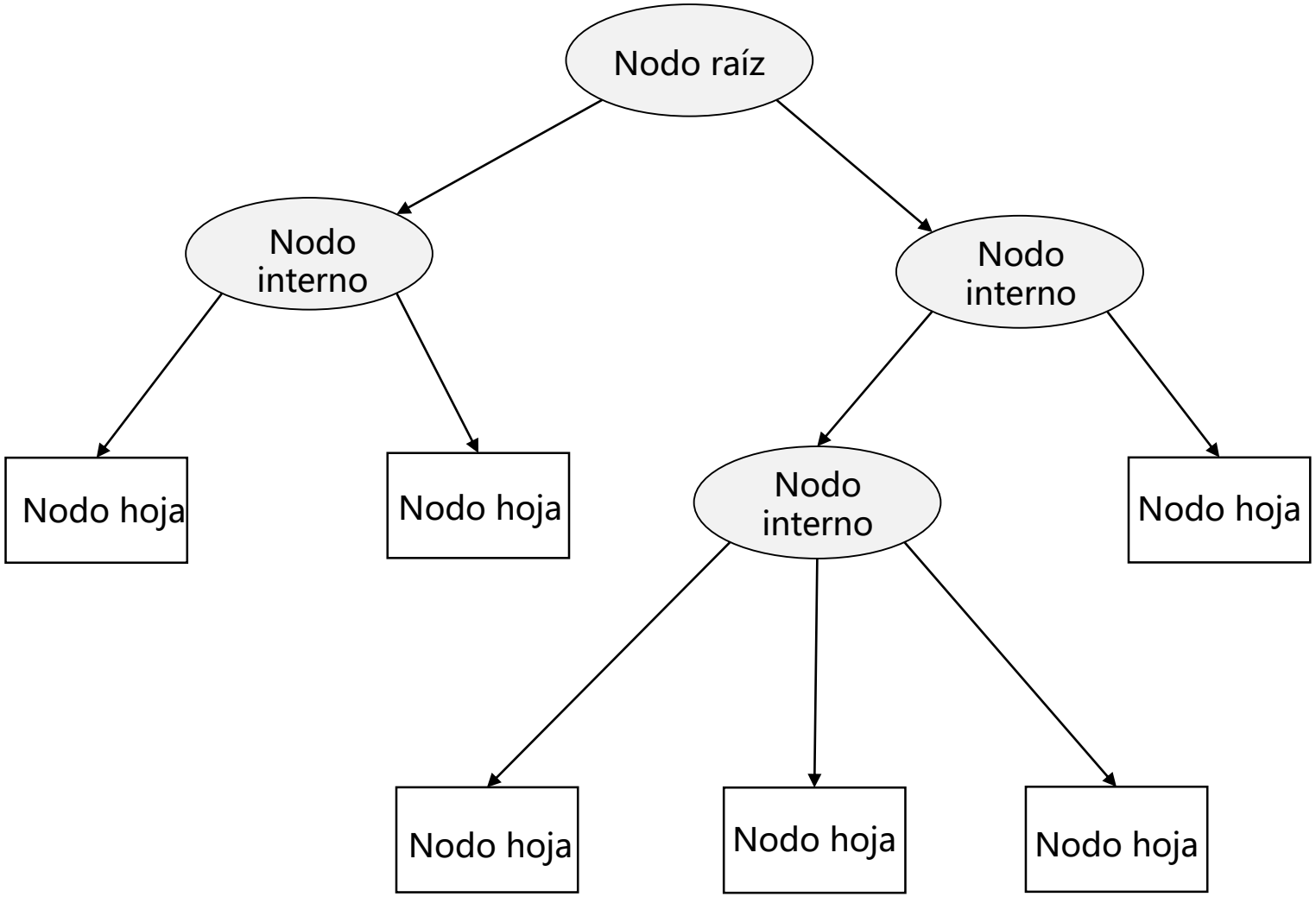
- **Suma de todas las probabilidades:**
  - 0.09 + 0.22 + 0.68 + 0.01 = 1
- Lo más probable es que esta foto sea una manzana.

Árbol de decisiones

- Un árbol de decisiones es una estructura de árbol (un árbol binario o un árbol no binario). Cada nodo no-hoja representa una prueba en un atributo de función. Cada rama representa la salida de un atributo de función en un cierto rango de valores, y cada nodo hoja almacena una categoría. Para utilizar el árbol de decisiones, inicie desde el nodo raíz, pruebe los atributos de las funciones de los elementos que se van a clasificar, seleccione las ramas de salida y utilice la categoría almacenada en el nodo hoja como resultado final.



Estructura del árbol de decisiones



Puntos clave de la construcción del Árbol de Decisiones

- Para crear un árbol de decisión, necesitamos seleccionar atributos y determinar la estructura de árbol entre los atributos de entidad. El paso clave de la construcción de un árbol de decisión es dividir los datos de todos los atributos de entidad, comparar los conjuntos de resultados en términos de 'pureza' y seleccionar el atributo con la "pureza" más alta como punto de datos para la división del conjunto de datos.
- Las métricas para cuantificar la "pureza" incluyen la entropía de información y el índice GINI. La fórmula es la siguiente:

$$H(X) = -\sum_{k=1}^{K} p_k \log_2(p_k) \qquad Gini = 1 - \sum_{k=1}^{K} p_k^2$$

- donde p_k indica la probabilidad de que la muestra pertenezca a la clase k (hay clases K en total). Una mayor diferencia entre la pureza antes de la segmentación y que después de la segmentación indica un mejor árbol de decisión
- Los algoritmos comunes del árbol de decisión incluyen ID3, C4.5 y CART.

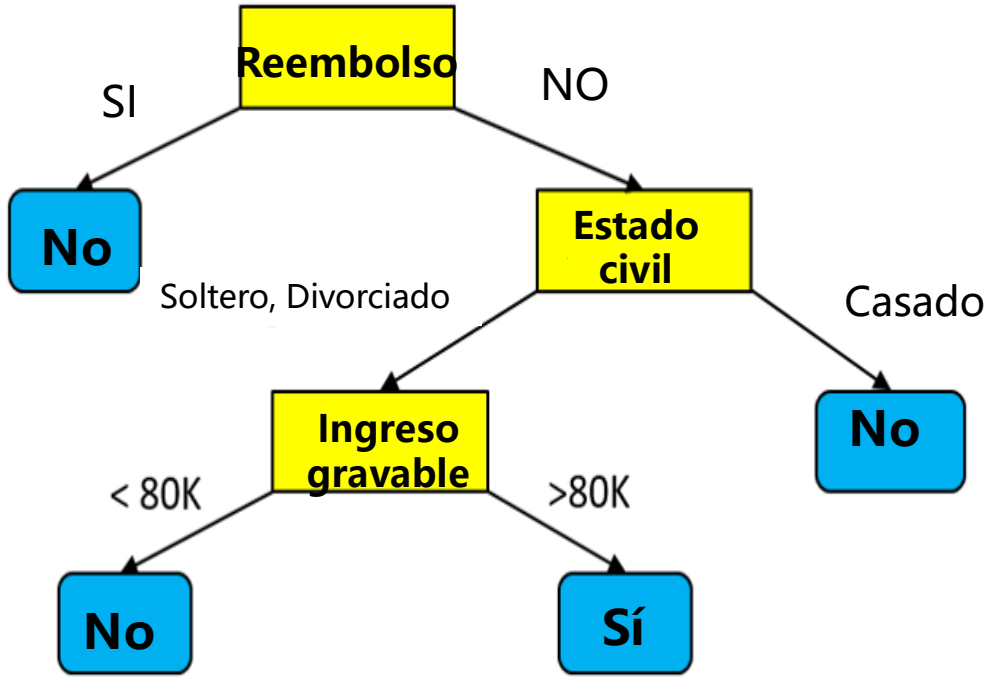
Proceso de construcción del Árbol de Decisiones

- **Selección de funciones:** Seleccione una característica/función de las características/funciones de los datos de formación como estándar dividido del nodo actual. (Diferentes estándares generan diferentes algoritmos de árbol de decisiones.)
- **Generación de árbol de decisiones:** Genere el nodo interno boca abajo en función de las características seleccionadas y deténgalo hasta que el conjunto de datos ya no pueda dividirse.
- **Poda:** El árbol de decisiones puede convertirse fácilmente en un exceso a menos que se realice la poda necesaria (incluyendo la prepoda y la poda posterior) para reducir el tamaño del árbol y optimizar su estructura de nodo.

Ejemplo del Árbol de Decisiones

- La siguiente figura muestra una clasificación cuando se utiliza un árbol de decisiones. El resultado de la clasificación se ve afectado por tres atributos: Reembolso, Estado civil e Ingresos gravables.

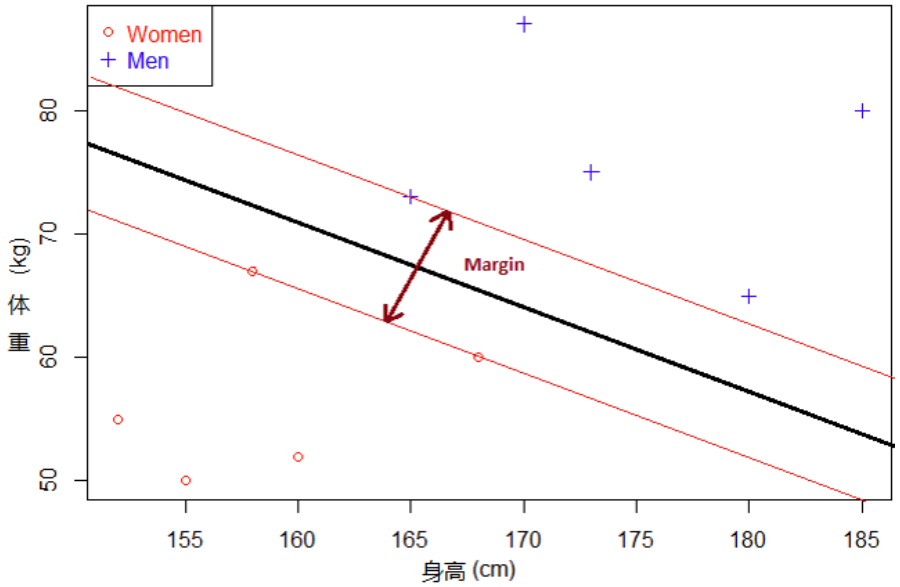
Tid.	Reembolso	Estado civil	Ingreso gravable	Engaño
1.	Si	Soltero	125.000.	No
3	No	Casado	100.000	No
3	No	Soltero	70.000	No
4	Sí	Casado	120.000	No
5	No	Divorciado	95.000	Sí
6	No	Casado	60.000	No
7	Sí	Divorciado	220.000	No
8	No	Soltero	85.000	Sí
9	No	Casado	75.000	No
10	No	Soltero	90.000	Sí

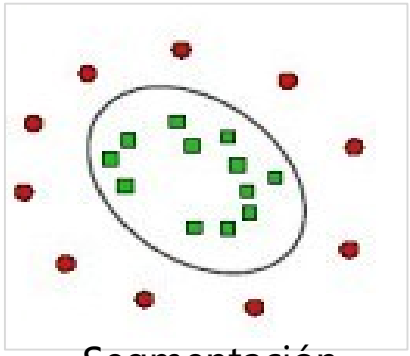


HUAWEI

MVS (Support Vector Machine – SVM)

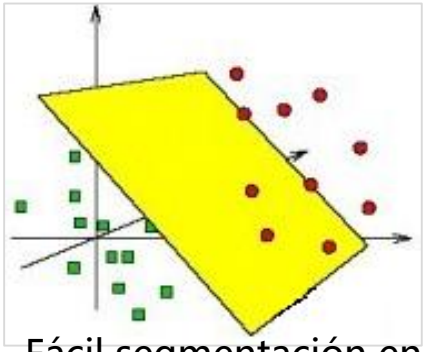
- Es un modelo de clasificación binaria cuyo modelo básico es un clasificador lineal definido en el espacio propio (eigenspace) con el intervalo más grande. Las MVS también incluyen trucos del kernel que las convierten en clasificadores no lineales. El algoritmo de aprendizaje de MVS es la solución óptima para la programación cuadrática convexa.





Segmentación compleja en un espacio de dimensiones bajas

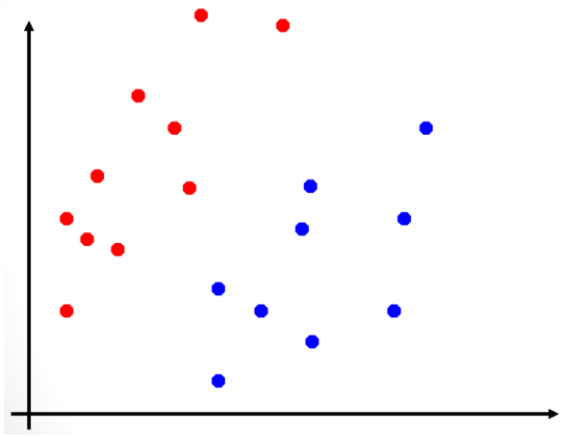
Proyección



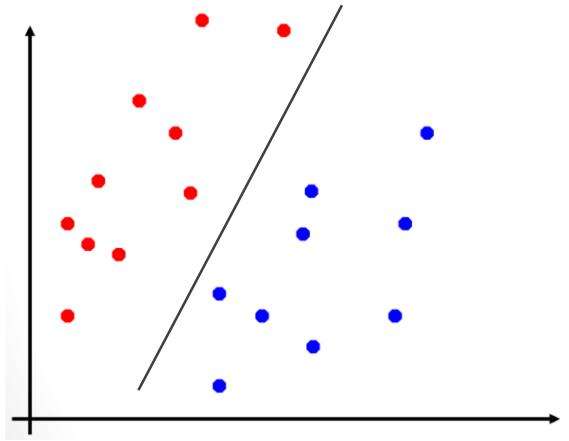
Fácil segmentación en espacio de alta dimensión

MVS Lineal (1)

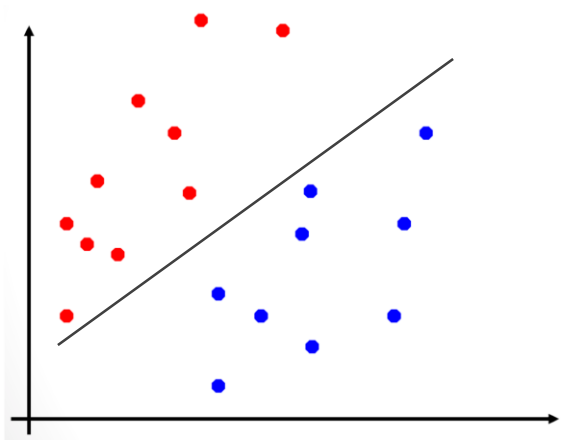
- ¿Cómo dividimos los conjuntos de datos rojo y azul por una línea recta?



Con clasificación binaria
Conjunto de datos
bidimensional



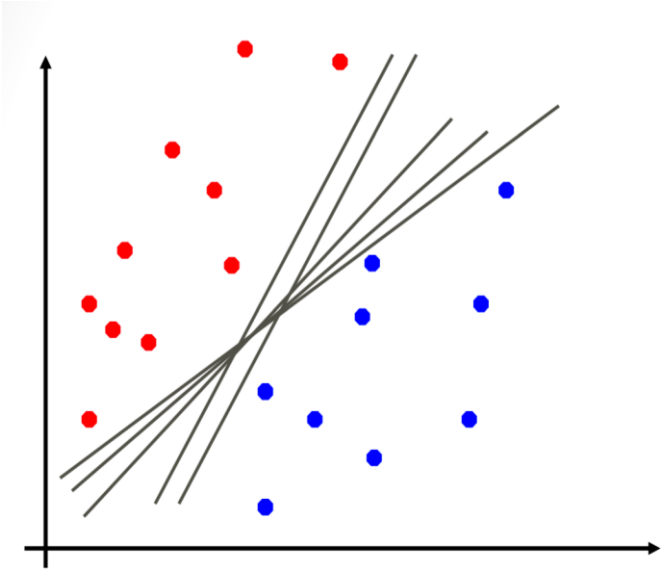
o

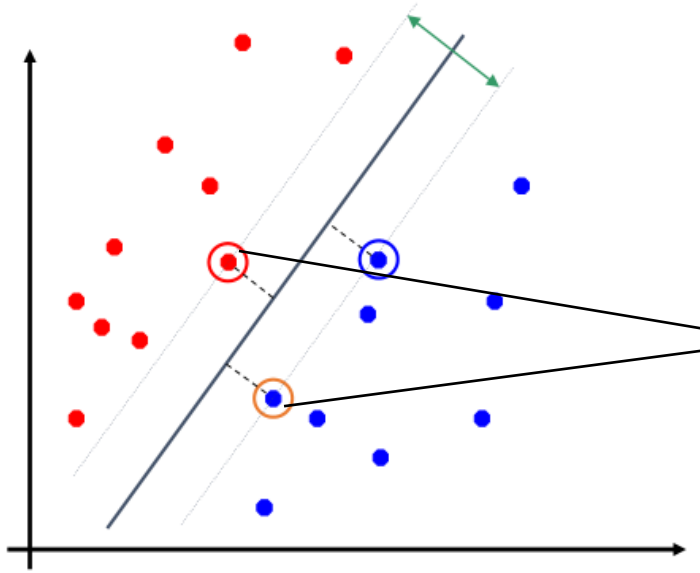


Tanto los métodos izquierdo como derecho se pueden utilizar para dividir los conjuntos de datos. ¿Cuál de ellos es el correcto?

MVS lineal (2)

- Las líneas rectas se utilizan para dividir los datos en diferentes clases. En realidad, podemos usar varias líneas rectas para dividir los datos. La idea central de la MVS es encontrar una línea recta y mantener el punto cerca de la línea recta lo más **lejos** posible de la línea recta. Esto puede permitir una fuerte capacidad de generalización del modelo. Estos puntos se llaman **vectores de apoyo**.
- En el espacio bidimensional, usamos líneas rectas para la segmentación. En el espacio de alta dimensión, usamos **hiperplanos** para la segmentación.

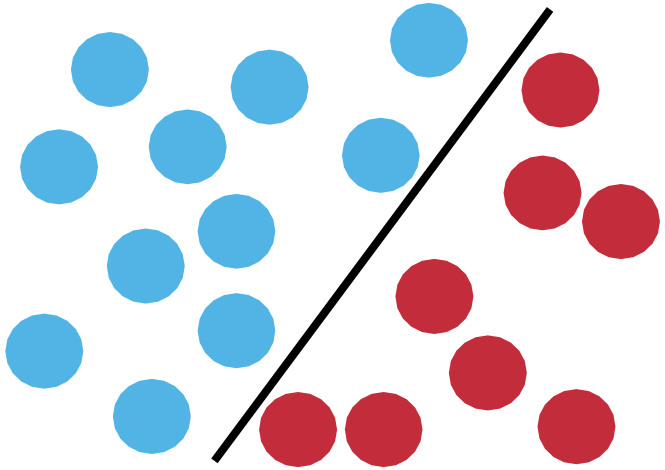




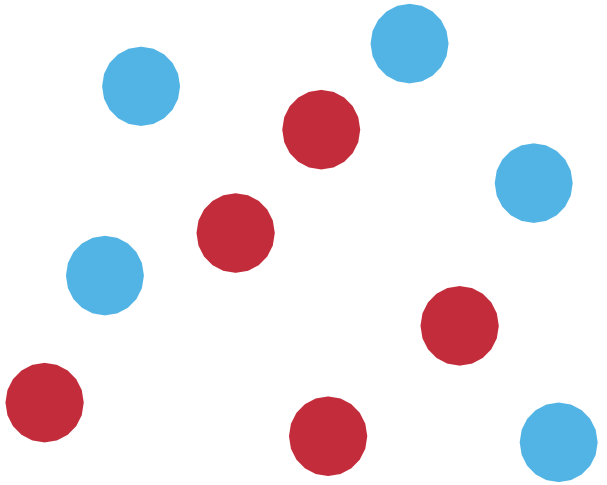
Distancia entre vectores de soporte es lo más lejos posible

MVS no lineal (1)

- ¿Cómo clasificamos un conjunto de datos separable no lineal?



La MVS lineal puede funcionar bien para conjuntos de datos lineales separables.

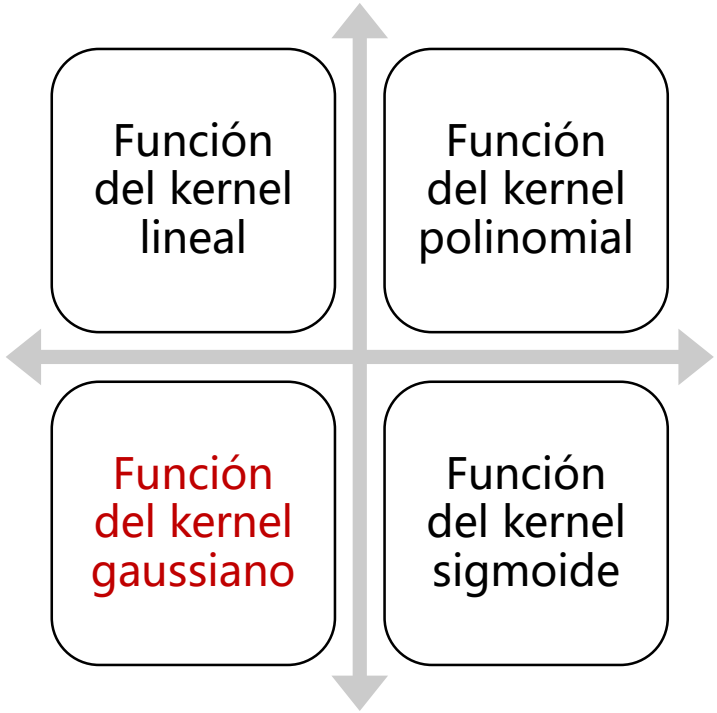


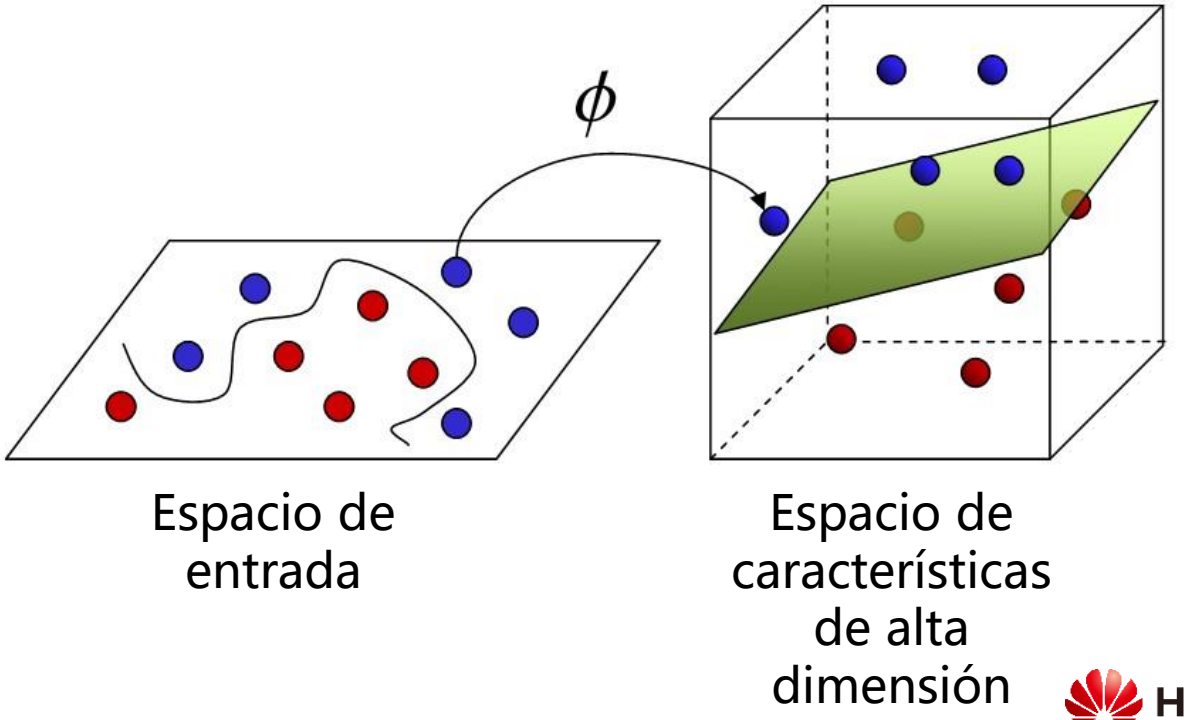
Los conjuntos de datos no lineales no se pueden dividir con líneas rectas.

MVS no lineal (2)

- Las funciones del kernel se utilizan para construir MVS no lineales.
- Las funciones del kernel permiten que los algoritmos se ajusten al hiperplano más grande en un espacio de características de alta dimensión transformado.

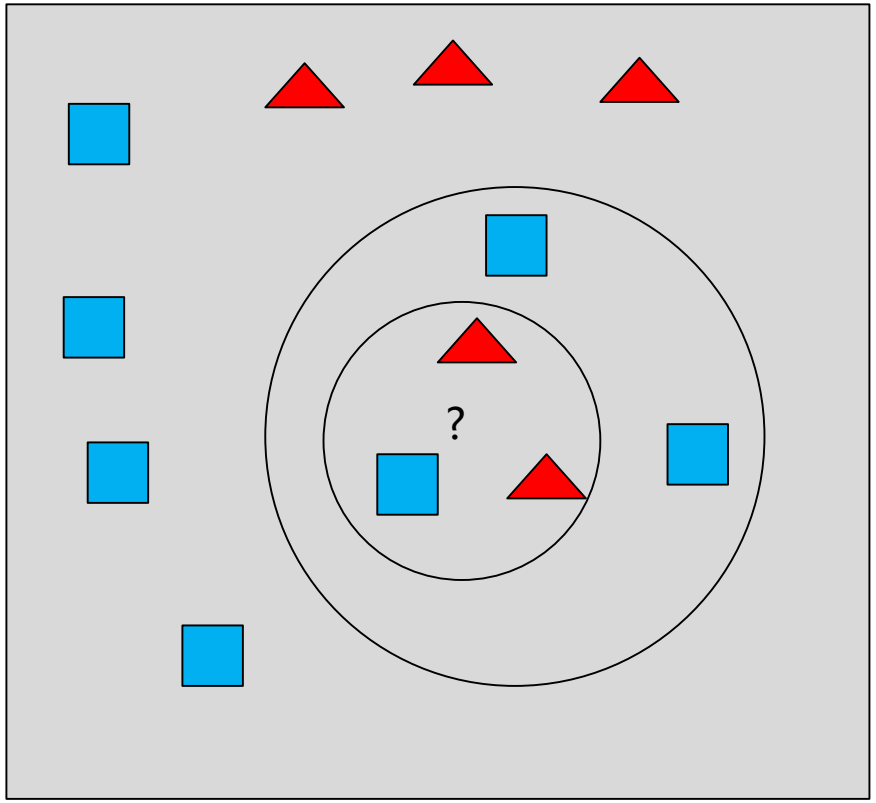
Funciones comunes del núcleo





Algoritmo KNN (1)

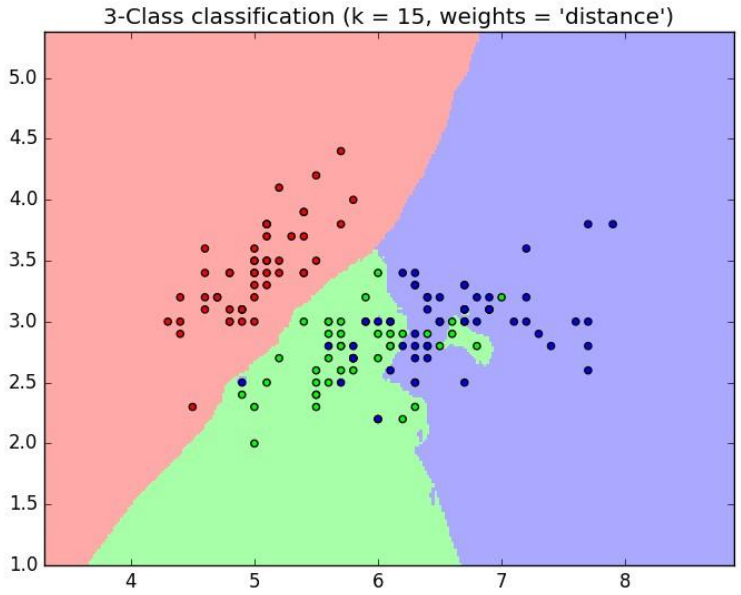
- El algoritmo de clasificación de KNN es un método teóricamente maduro y uno de los más simples algoritmos de Machine Learning.
- Según este método, si la mayoría de *k* muestras más similares a una muestra (los vecinos más cercanos en el espacio propio) pertenecen a una categoría específica, esta muestra también pertenece a esta categoría.



La categoría objetivo del punto ***?*** varía con el número de nodos más adyacentes.

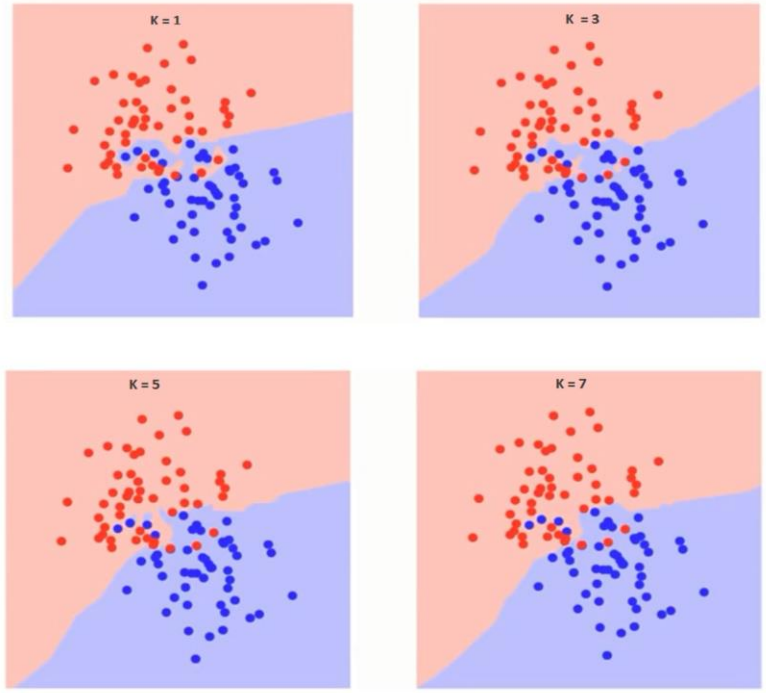
Algoritmo KNN (2)

- A medida que el resultado de predicción se determina en base al número y peso de los vecinos en el conjunto de entrenamiento, el algoritmo KNN tiene una lógica simple.
- KNN es un método no paramétrico que normalmente se utiliza en conjuntos de datos con límites de decisión irregulares.
  - El algoritmo KNN generalmente adopta el método de votación mayoritaria para la predicción de clasificación y el método de valor medio para la predicción de regresión.
- KNN requiere un gran número de cálculos.



Algoritmo KNN (3)

- Generalmente, un valor k mayor reduce el impacto del ruido en la clasificación, pero obscurece el límite entre las clases.
  - Un valor k más grande significa una mayor probabilidad de subajuste debido a que la segmentación es demasiado áspera. Un valor k menor significa una mayor probabilidad de sobreajuste debido a que la segmentación es demasiado refinada.



- El límite se vuelve más suave a medida que aumenta el valor de k.
- A medida que el valor de k aumenta hasta el infinito, todos los puntos de datos eventualmente se volverán azul o rojo.

Naive Bayes (1)

- Algoritmo Bayesiano Ingenuo: Un algoritmo simple de clasificación multi-clases basado en el teorema de Bayes. Asume que las características son independientes entre sí. Para una característica de muestra determinada *X*, la probabilidad de que una muestra pertenezca a una categoría *H* es:

$$P(C_k \mid X_1,\ldots,X_n) = \frac{P(X_1,\ldots,X_n \mid C_k)P(C_k)}{P(X_1,\ldots,X_n)}$$

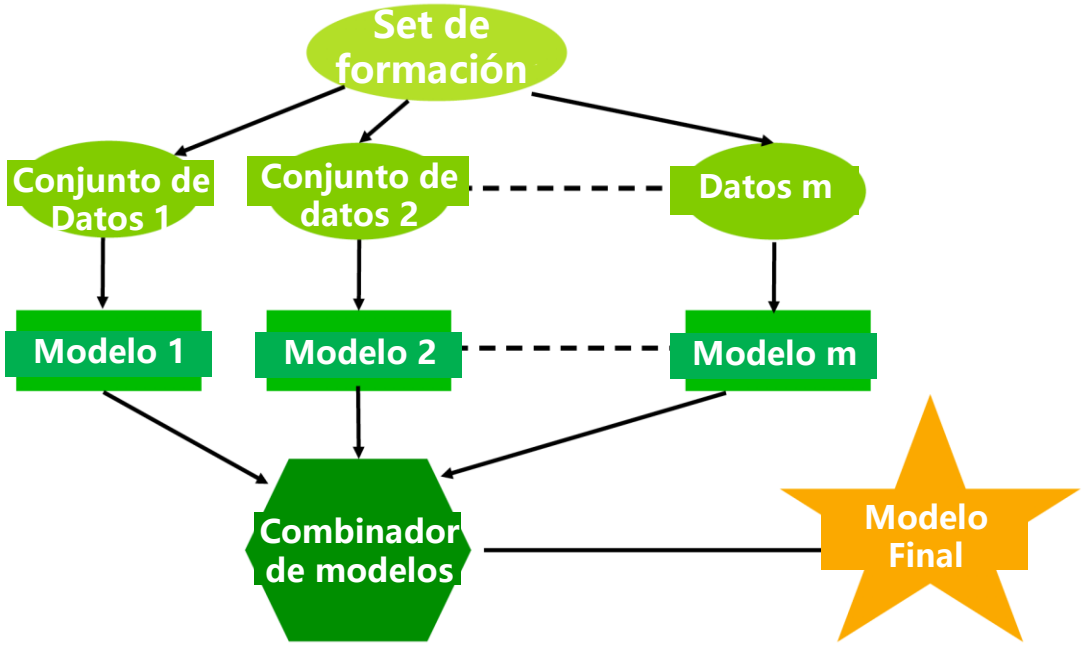
  - $X_1, \ldots, X_n$ son características de datos, que normalmente se describen mediante valores de medición de conjuntos de atributos m.
    - Por ejemplo, la característica de "color" puede tener tres atributos: rojo, amarillo y azul.
  - C_k indica que los datos pertenecen a una categoría específica *C*.
  - $P(C_k|X_1, \ldots, X_n)$es una probabilidad posterior, o una probabilidad posterior de condición C_k.
  - $P(C_k)$ es una probabilidad previa que es independiente de$X_1, \ldots, X_n$
  - $P(X_1, \ldots, X_n)$ es la probabilidad priori de X.

Naive Bayes (2)

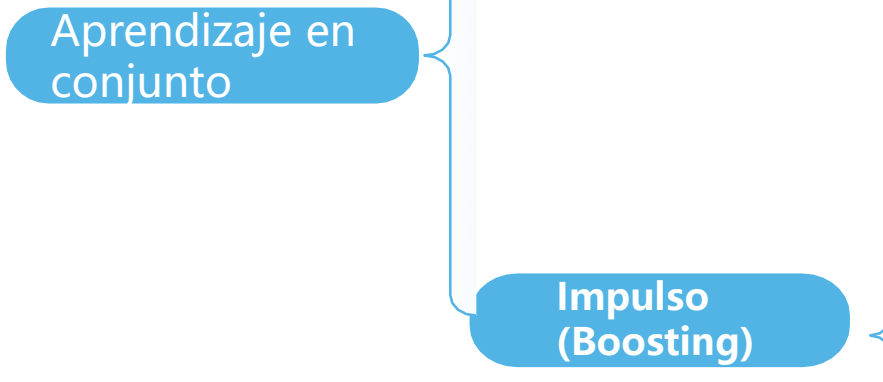
- Asunción independiente de las características.
  - Por ejemplo, si una fruta es roja, redonda y de unos 10 cm de diámetro, puede ser considerada una manzana.
  - Un clasificador de Naive Bayes considera que cada característica contribuye independientemente a la probabilidad de que el fruto sea una manzana, independientemente de cualquier posible correlación entre el color, redondez y diámetro.

Aprendizaje en Conjunto

- El aprendizaje en conjunto es un paradigma de Machine Learning en el que se entrenan y combinan múltiples aprendices/alumnos para resolver el mismo problema. Cuando se utilizan múltiples aprendices/alumnos, la capacidad de generalización integrada puede ser mucho más fuerte que la de un único aprendiz/alumno.
- Si usted hace una pregunta compleja a miles de personas al azar y luego resume sus respuestas, la respuesta resumida es mejor que la respuesta de un experto en la mayoría de los casos. Esta es la sabiduría de las masas.



Clasificación del Aprendizaje en Conjunto



Embalaje (Random Forest)

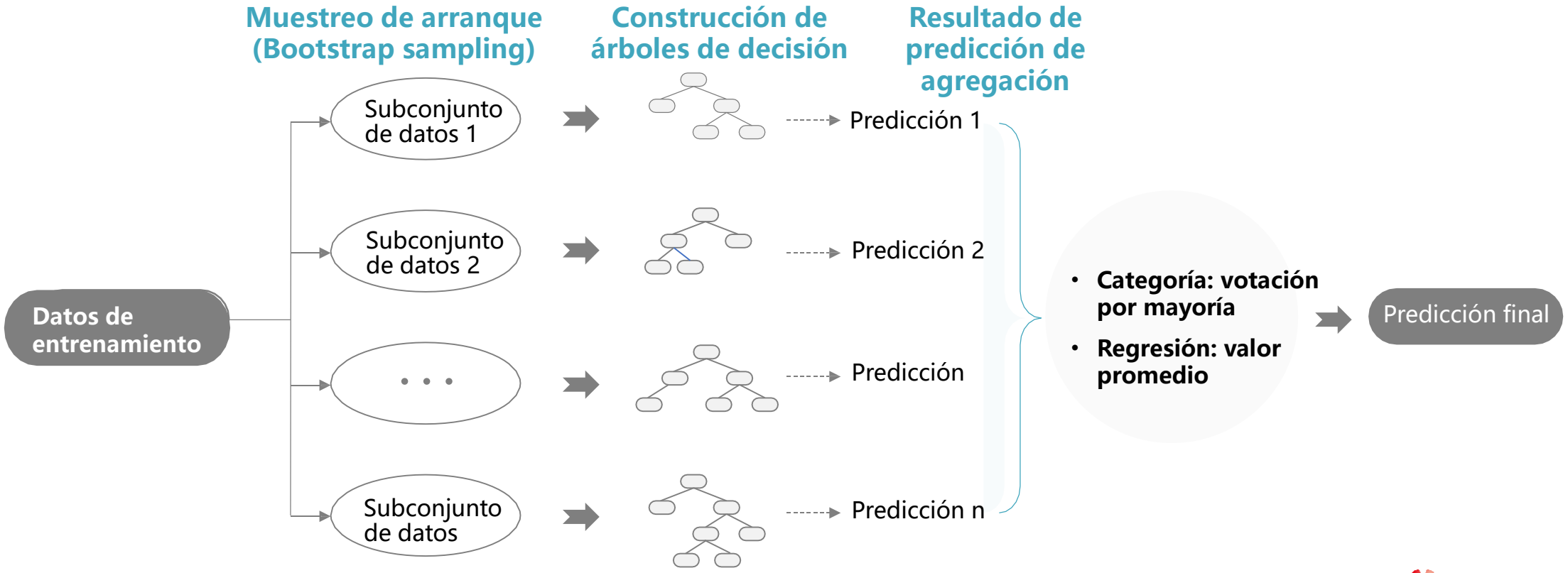
- Independientemente construye varios aprendices/alumnos básicos y luego promedia sus predicciones.
- En promedio, un aprendiz/alumno compuesto suele ser mejor que un aprendiz/alumno de base única debido a una menor variación.

Impulso (Adaboost, GBDT y XGboost)

Construye aprendices/alumnos básicos en secuencia para reducir gradualmente el sesgo de un aprendiz/alumno compuesto. El aprendiz/alumno compuesto puede ajustar bien los datos, lo que también puede causar sobreajustes.

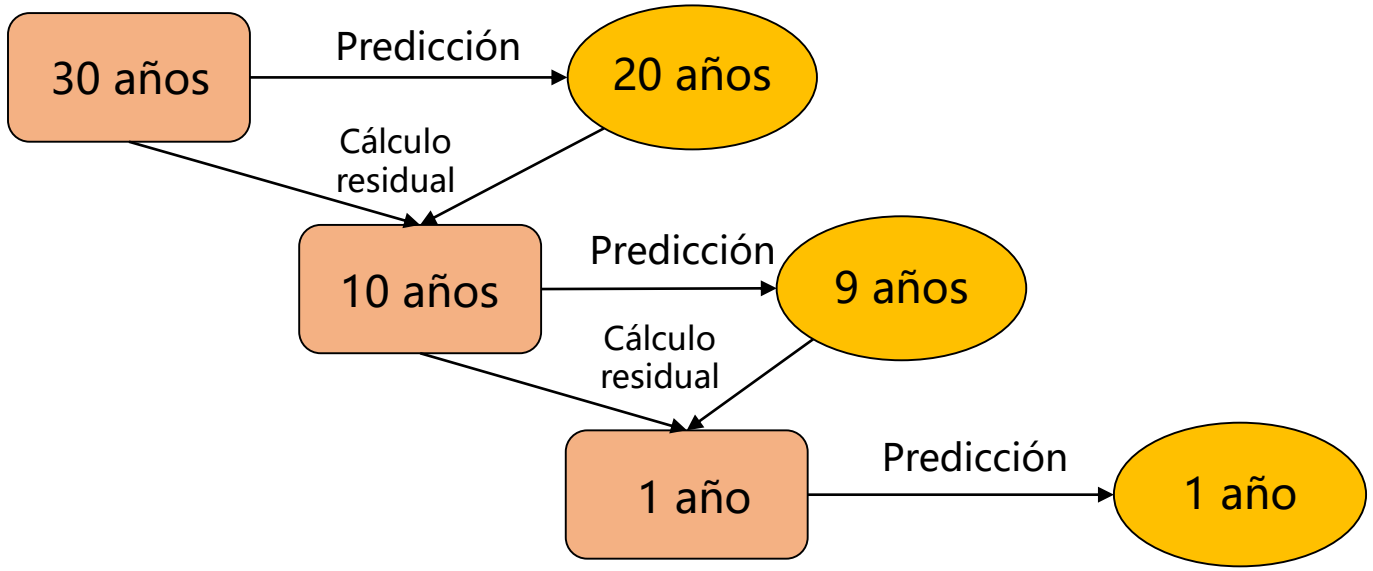
Métodos de conjunto en Machine Learning (1)

- Bosque aleatorio = Embalaje + árbol de decisiones CART
- Los bosques aleatorios construyen múltiples árboles de decisión y los fusionan para hacer las predicciones más precisas y estables.
  - Los bosques aleatorios pueden utilizarse para la clasificación y la regresión de problemas.



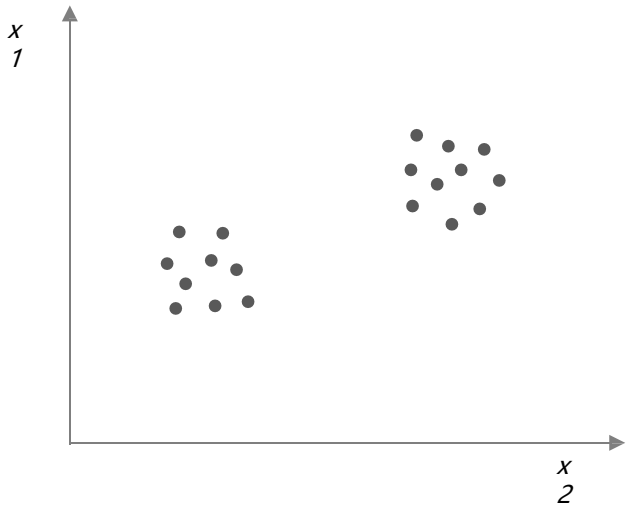
Métodos de conjunto en Machine Learning (2)

- GBDT es un tipo de algoritmo de impulse (Boosting).
- En un modo agregado, la suma de los resultados de todos los aprendices/alumnos básicos equivale al valor predicho. Esencialmente, el residual de la función de error al valor predicho se ajusta al siguiente aperndiz/alumno básico. - (El residual es el error entre el valor predicho y el valor real.)
- Durante el entrenamiento del modelo, el GBDT requiere que la pérdida de muestras para la predicción del modelo sea lo más pequeña posible.



Aprendizaje no supervisado K-Medias (K-Means)

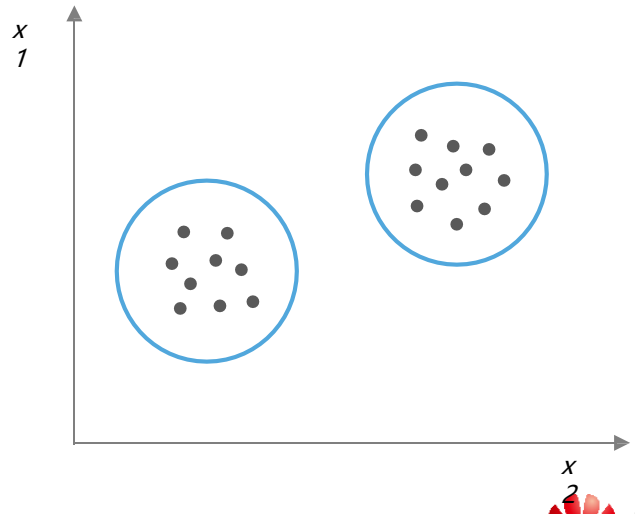
- Clusters de K-medias (K-Means clustering) tiene por objeto dividir n observaciones en k clusters en los que cada observación pertenece al cluster con la media más cercana, sirviendo como prototipo del cluster.
- Para el algoritmo de k-means, especifique el número final de clusters (k). Luego, divida n objetos de datos en k clusters. Los clusters obtenidos cumplen las siguientes condiciones: (1) Los objetos del mismo cluster son muy similares. (2) La similitud de los objetos en diferentes clusters es pequeña.



K-means clustering

Clusters de K-medias

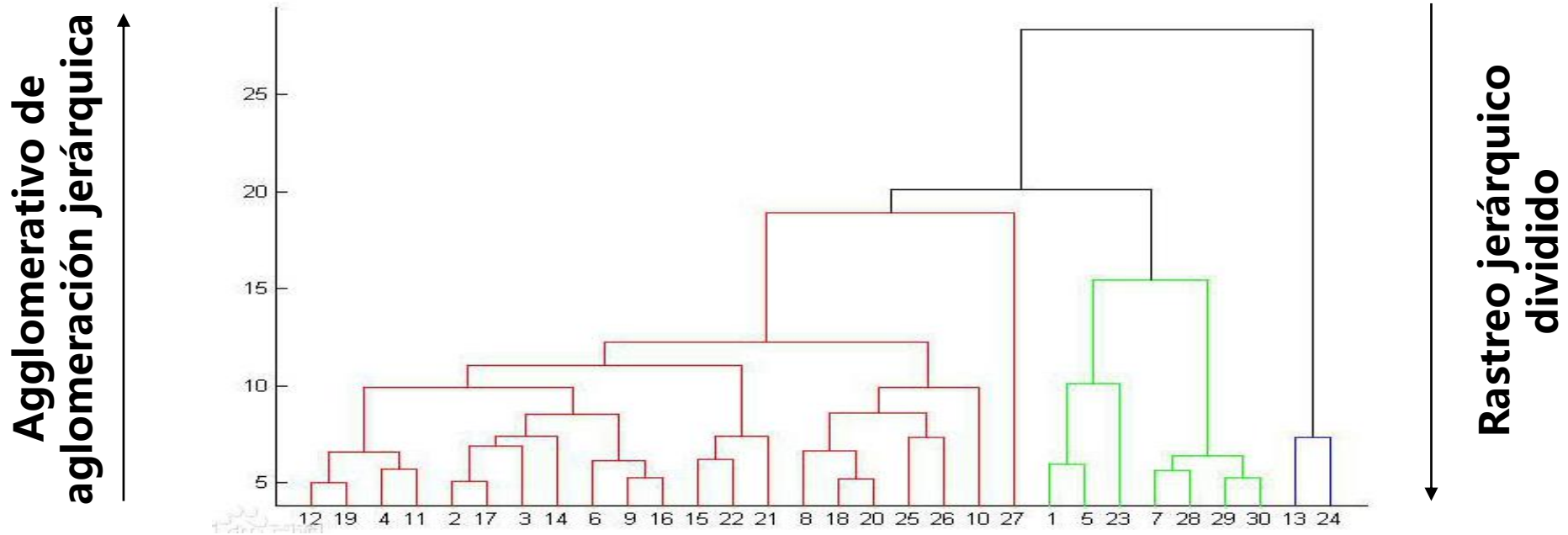
Los datos **no están etiquetados**. El clustering de K-Medias puede clasificar automáticamente los conjuntos de datos.



HUAWEI

Aprendizaje no supervisado - Clustering jerárquico

- El clustering jerárquico o agrupación jerárquica divide un conjunto de datos en diferentes capas y forma una estructura de agrupación similar a un árbol. La división del conjunto de datos puede utilizar una política de agregación "bottom-up" o una política de división "top-down". La jerarquía de agrupamiento se representa en un gráfico de árbol. La raíz es el cúmulo único de todas las muestras, y las hojas son el cúmulo de sólo una muestra.



Contenidos

1. Definición de Machine Learning
2. Tipos de Machine Learning
3. Proceso de Machine Learning
4. Otros métodos clave de Machine Learning
5. Algoritmos comunes de Machine Learning
6. **Caso práctico**

Caso Práctico

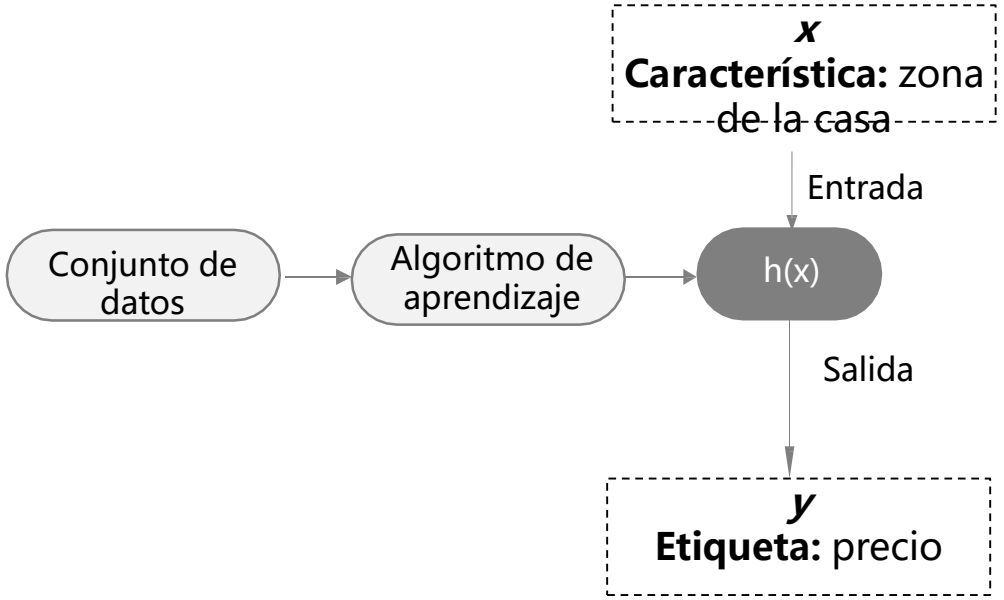
- Supongamos que hay un conjunto de datos que contiene las áreas de vivienda y los precios de 21,613 unidades de vivienda vendidas en una ciudad. A partir de estos datos, podemos predecir los precios de otras casas en la ciudad.

Área de la Casa	Precio
1,180	221,900
2,570	538,000
770	180,000
1,960	604,000
1,680	510,000
5,420	1,225,000
1,715	257,500
1,060	291,850
1,160	468,000
1,430	310,000
1,370	400,000
1,810	530,000
...	...

Conjunto de datos

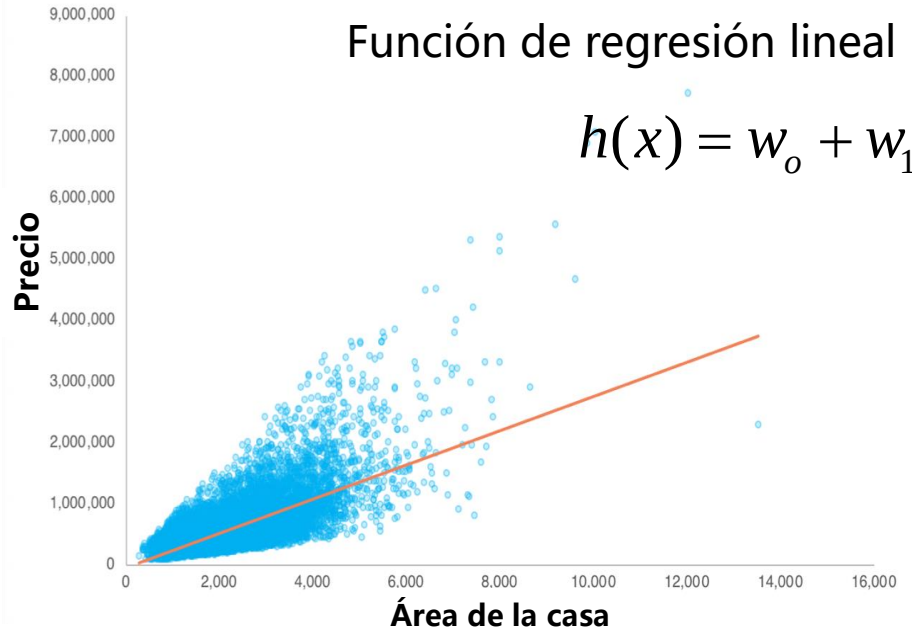
Análisis de problemas

- Este caso contiene una gran cantidad de datos, incluyendo la entrada ***x*** (área de la casa), y la salida ***y*** (precio), que es un valor continuo. Podemos usar la **regresión** del **aprendizaje supervisado**. Dibuje un gráfico de dispersión basado en los datos y utilice la **regresión lineal**.
- Nuestro objetivo es construir una función modelo h(x) que se aproxima infinitamente a la función que expresa la verdadera distribución del conjunto de datos.
- A continuación, utilice el modelo para predecir datos de precios desconocidos.



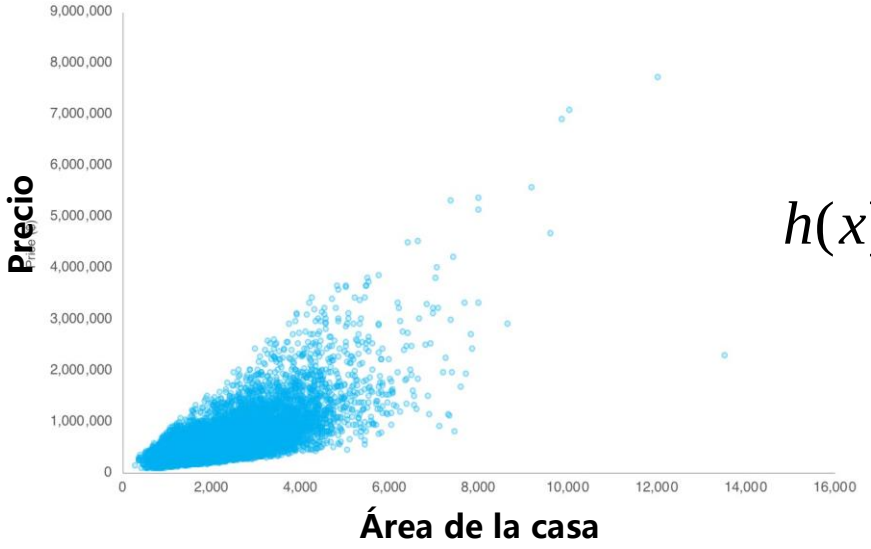
Función de regresión lineal unaria

$$h(x) = w_o + w_1 x$$

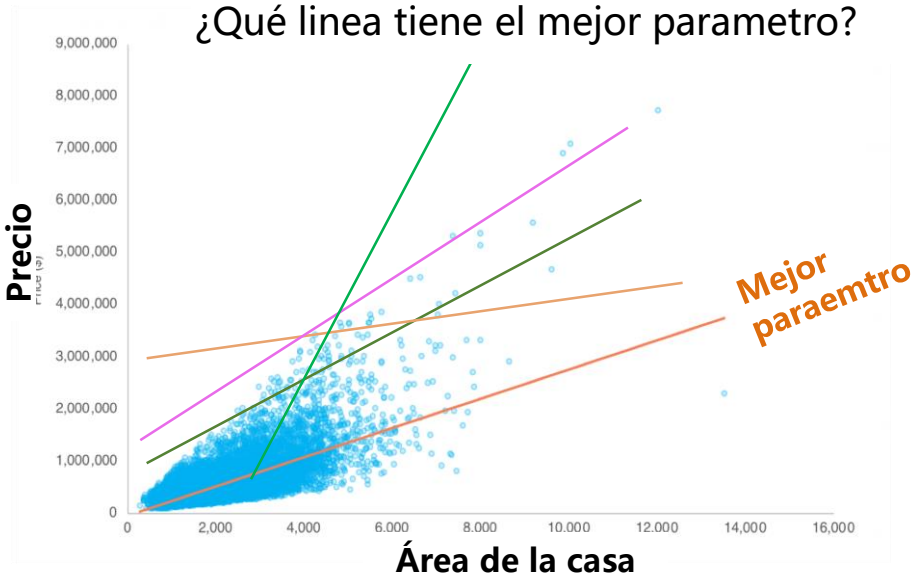


Objetivo de la regresión lineal

- La regresión lineal tiene como objetivo encontrar una línea recta que mejor se adapte al dataset.
- La regresión lineal es un modelo basado en parámetros. Aquí, necesitamos parámetros de aprendizaje w_0 y w_1. Cuando se encuentran estos dos parámetros, aparece el mejor modelo.

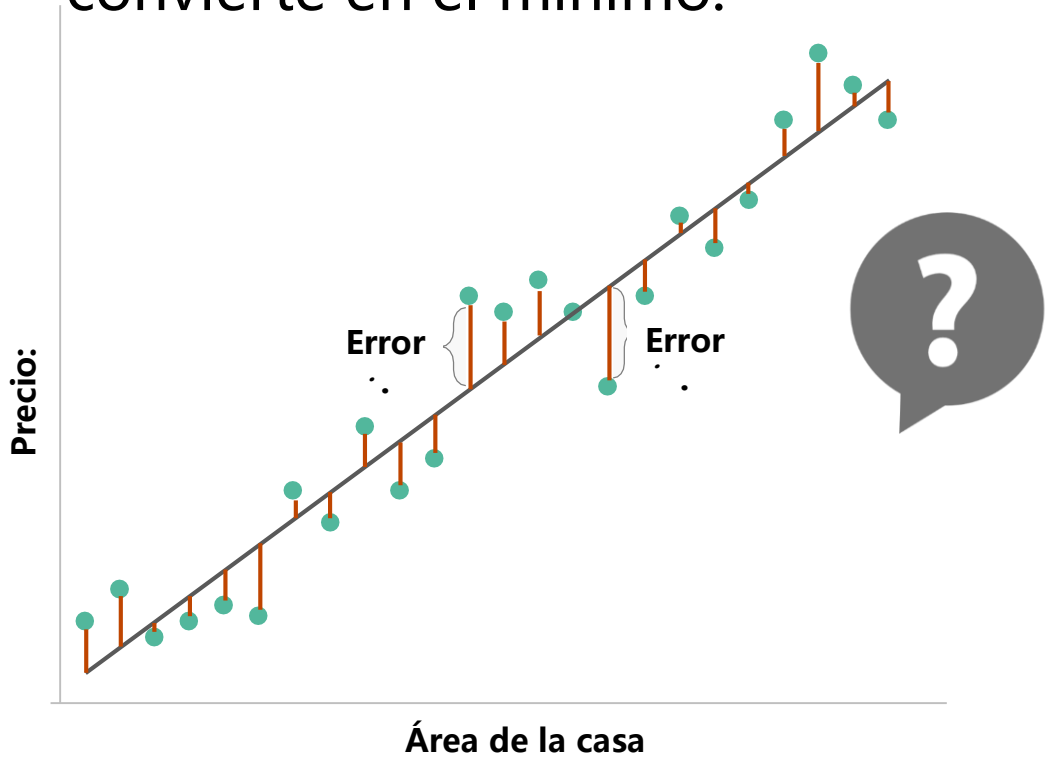


$$h(x) = w_o + w_1 x$$



Función de pérdida de regresión lineal

- Para encontrar el parámetro óptimo, construir una función de pérdida y encontrar los valores de los parámetros cuando la función de pérdida se convierte en el mínimo.



Función de pérdida de la regresión lineal: $$J(w)=\frac{1}{2m}\sum\left(h(x)-y\right)^2$$

Objetivo:

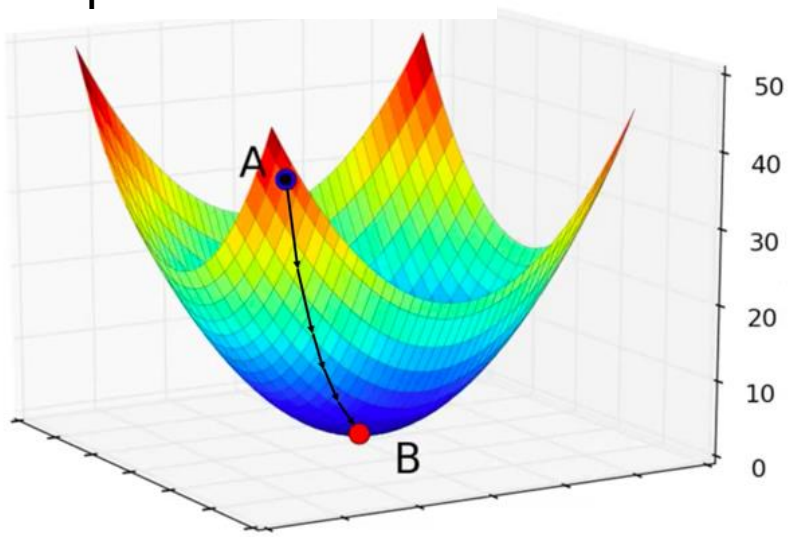
$$\arg\min_{w} J(w)=\frac{1}{2m}\sum\left(h(x)-y\right)^2$$

- donde, *m* indica el número de muestras,
- *H(x)* indica el valor predicho, e y indica el valor real.

Método de descenso de gradiente

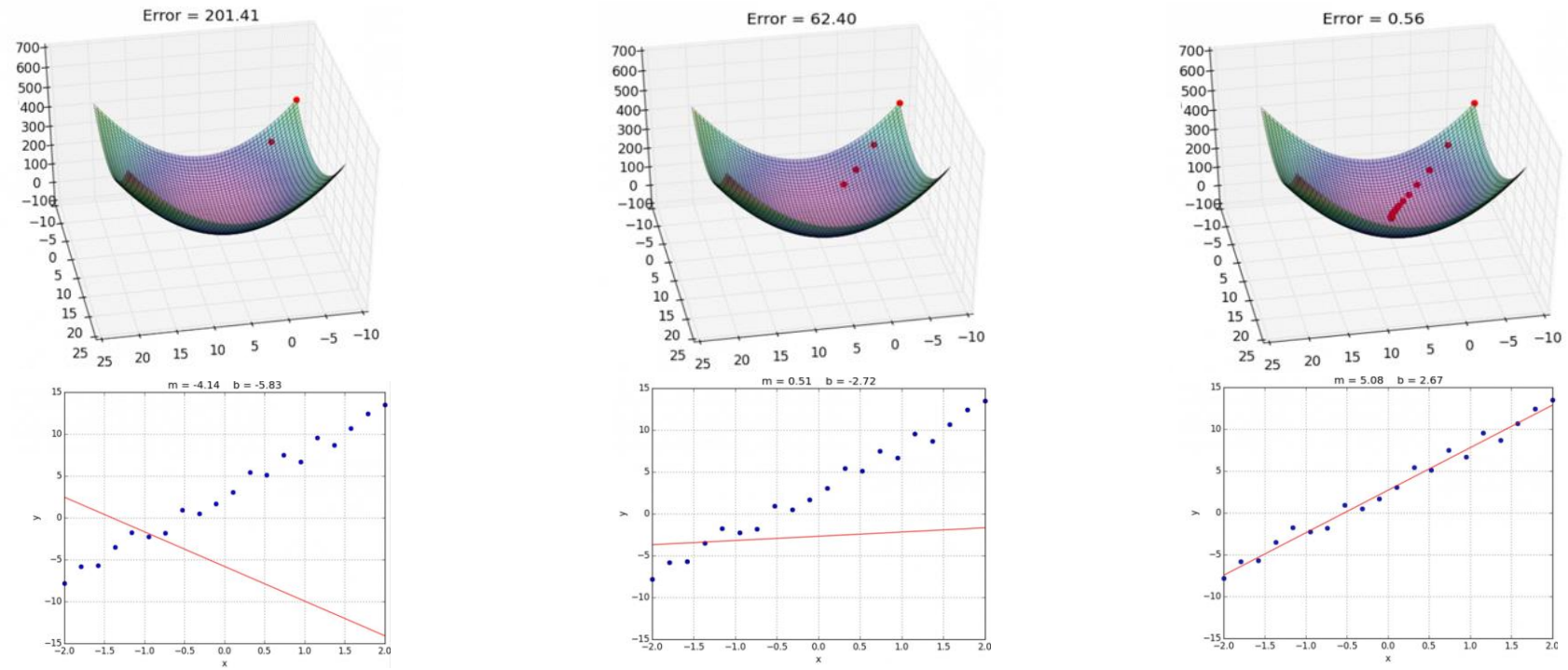
- El algoritmo de descenso de gradiente encuentra el valor mínimo de una función a través de la iteración.
- Su objetivo es aleatorizar un punto inicial sobre la función de pérdida, y luego encontrar el valor mínimo global de la función de pérdida basado en la dirección de gradiente negativo. Este valor de parámetro es el valor óptimo del parámetro.
  - **Punto A:** la posición de w_0 y w_1 después de la inicialización aleatoria.. w_0 y w_1 son **parametros** requeridos**.**
  - **Linea de conexion A-B:** un camino formado en base a descensos en una dirección de gradiente negativo. Tras cada descenso, los valores w_0 y w_1 **cambian**, y la línea de regresión también cambia.
  - **Punto B:** Valor minimo global de la función de perdida. Tambien se encuentran los valores finales de w_0 y w_1

Superficie de costo



Ejemplo de iteración

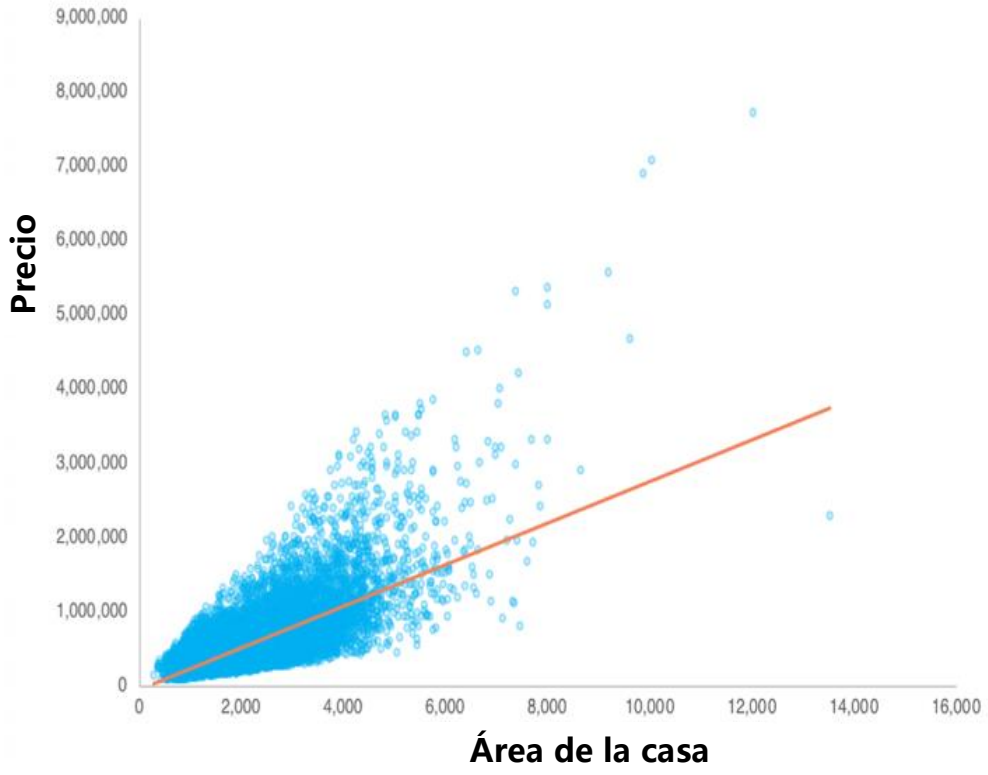
- El siguiente es un ejemplo de una iteración de descenso de gradiente. Podemos ver que a medida que los puntos rojos en la superficie de la función de pérdida se aproximan gradualmente a un punto más bajo, el ajuste de la línea roja de regresión lineal con los datos se hace mejor y mejor. En este momento, podemos obtener los mejores parámetros.



Depuración de modelos y aplicación

- Una vez entrenado el modelo, pruébelo con el conjunto de pruebas para garantizar la capacidad de generalización.
- Si se produce un sobreajuste, utilice la regresión de Lasso o la regresión de Ridge con términos de regularización y ajuste los hiperparámetros.
- Si se produce un subajuste, utilice un modelo de regresión más complejo, como el GBDT.
- Nota:
  - Prara datos reales, preste atención a las funciones de limpieza de datos e ingeniería de características.

El resultado final del modelo es el siguiente: $h(x) = 280.62x - 43581$



Resumen

- En primer lugar, este curso describe la definición y clasificación de Machine Learning, así como los problemas que el Machine Learning resuelve. A continuación, introduce los puntos clave del conocimiento de Machine Learning, incluido el procedimiento general . (recolección de datos, limpieza de datos, extracción de características, entrenamiento de modelos, entrenamiento y evaluación de modelos y despliegue de modelos) algoritmos comunes (regresión lineal, regresión logística, árbol de decisiones, SVM, Bayes ingenuo, KNN, aprendizaje en conjunto, K-media, etc.), algoritmo de descenso de gradiente, parámetros e hiper-parámetros.
- Por último, un proceso completo de Machine Learning se presenta mediante el caso de utilizar la regresión lineal para predecir los precios de la vivienda.

Quiz

1. (Verdadero o falso) La iteración de descenso gradual es el único método de los algoritmos de Machine Learning. ( )
   A. Verdadero
   B. Falso
2. (Pregunta de respuesta única) ¿Cuál de los siguientes algoritmos no es de aprendizaje supervisado? ( )
   A. Regresión lineal
   B. Árbol de decisiones
   C. KNN
   D. K-Medias

Recomendaciones

- Sitio web de aprendizaje en línea
  - https://e.huawei.com/en/talent/#/
- Base de conocimientos de Huawei
  - https://support.huawei.com/enterprise/en/knowledge?lang=es

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

HUAWEI

Información general sobre el aprendizaje profundo (Deep Learning)

HUAWEI

PREFACIO

- El capítulo describe el conocimiento básico del aprendizaje profundo, incluyendo la historia de desarrollo del aprendizaje profundo, los componentes y tipos de redes neuronales de aprendizaje profundo, y los problemas comunes en los proyectos de aprendizaje profundo.

OBJETIVOS

Al finalizar este curso, podrás:

- Describir la definición y el desarrollo de las redes neuronales.
- Conozca los componentes importantes de las redes neuronales de aprendizaje profundo.
- Comprender la capacitación y optimización de las redes neuronales.
- Describir los problemas comunes en el aprendizaje profundo.

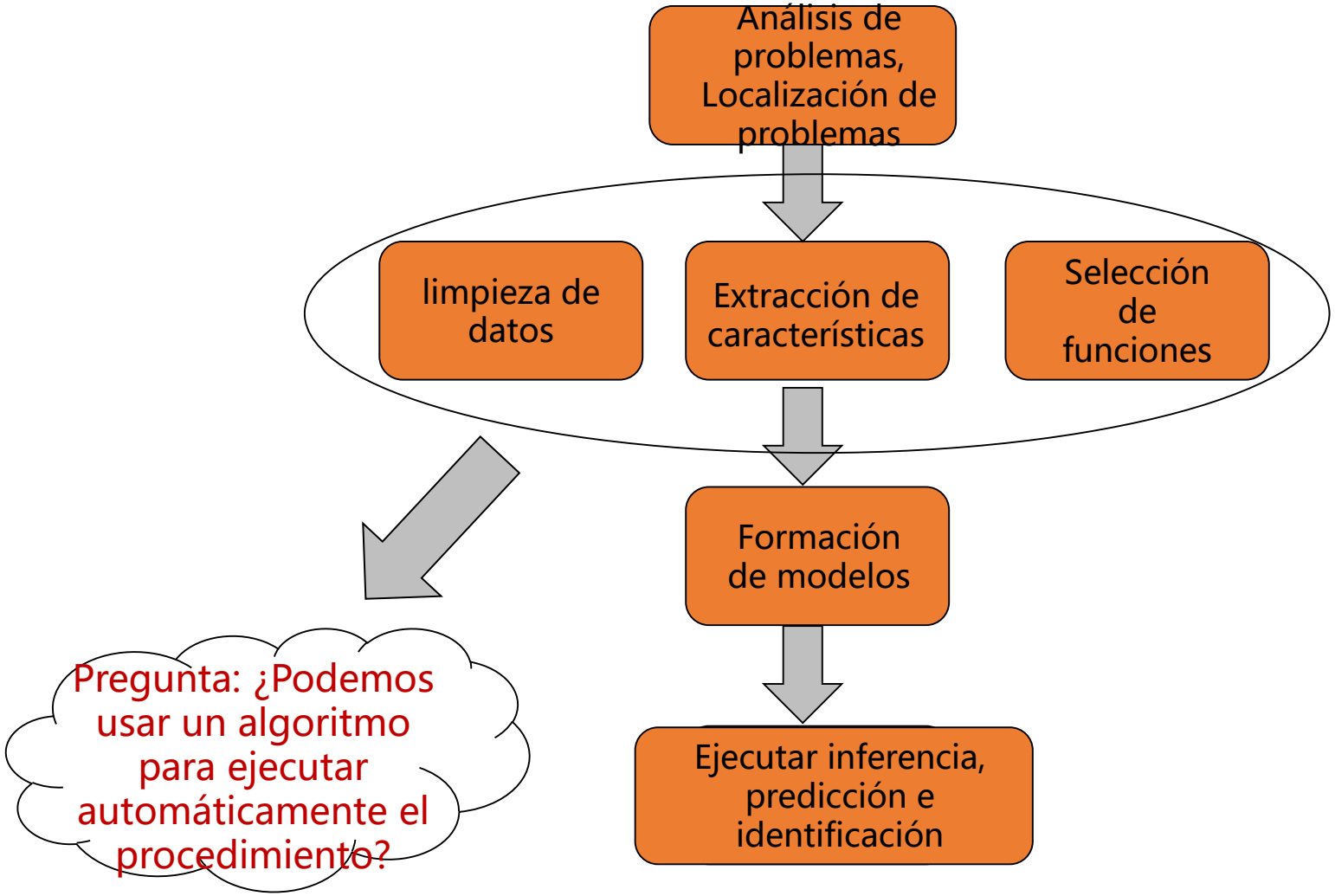
CONTENIDO

Machine Learning tradicional y aprendizaje profundo

- Como modelo basado en el aprendizaje no supervisado de características y de jerarquía de características, el aprendizaje profundo tiene grandes ventajas en campos como la visión por computadora, el reconocimiento de voz y el procesamiento de lenguaje natural.

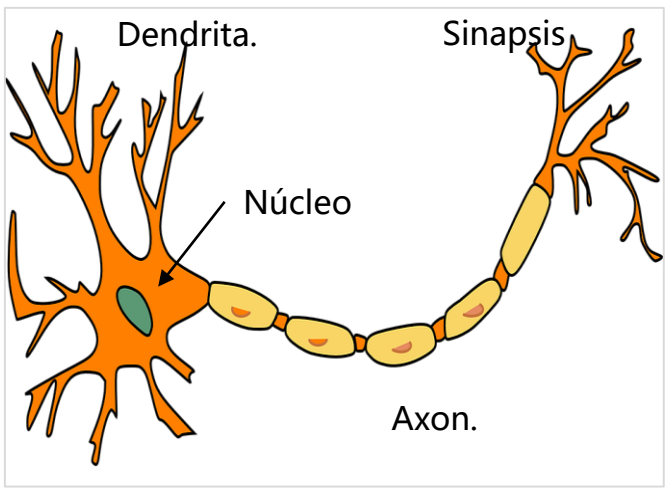
Machine Learning tradicional	Aprendizaje profundo
Bajos requisitos de hardware en el equipo: Dada la cantidad de computación limitada, el equipo no necesita una GPU para computación paralela en general.	Mayores requisitos de hardware en el ordenador: Para ejecutar operaciones matriciales en datos masivos, el ordenador necesita una GPU para realizar computación paralela.
Aplicable a entrenamiento con una pequeña cantidad de datos y cuyo rendimiento no se puede mejorar continuamente a medida que aumenta la cantidad de datos.	El rendimiento puede ser alto cuando se proporcionan parámetros de peso de alta dimensión y datos de entrenamiento masivos.
Desglose por niveles del problema	Aprendizaje de extremo a extremo (E2E)
Selección manual de funciones	Extracción automática de funciones basada en algoritmos
Funciones fáciles de explicar	Funciones difíciles de explicar

Aprendizaje automático tradicional

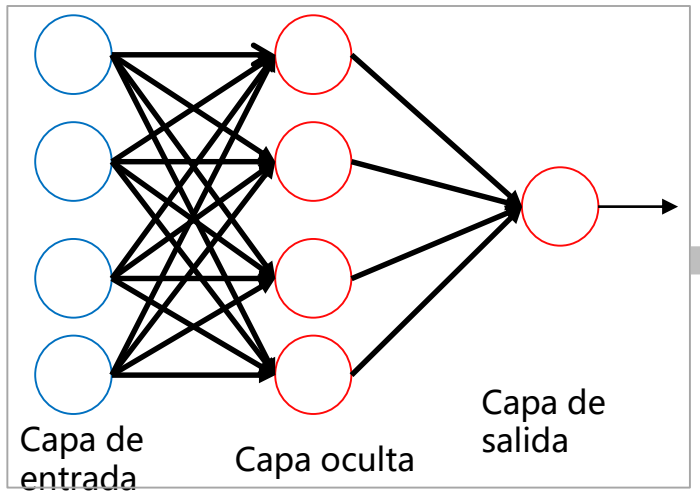


Aprendizaje profundo

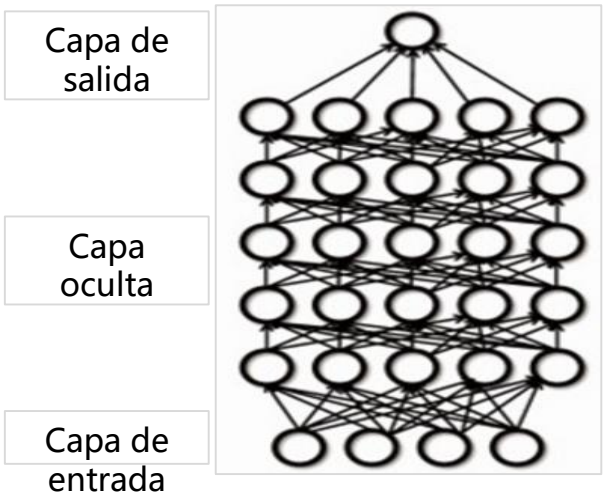
- Generalmente, la arquitectura de aprendizaje profundo es una red neural profunda. "Profundo" en "aprendizaje profundo" se refiere al número de capas de la red neural.



Red neuronal humana



Perceptrón

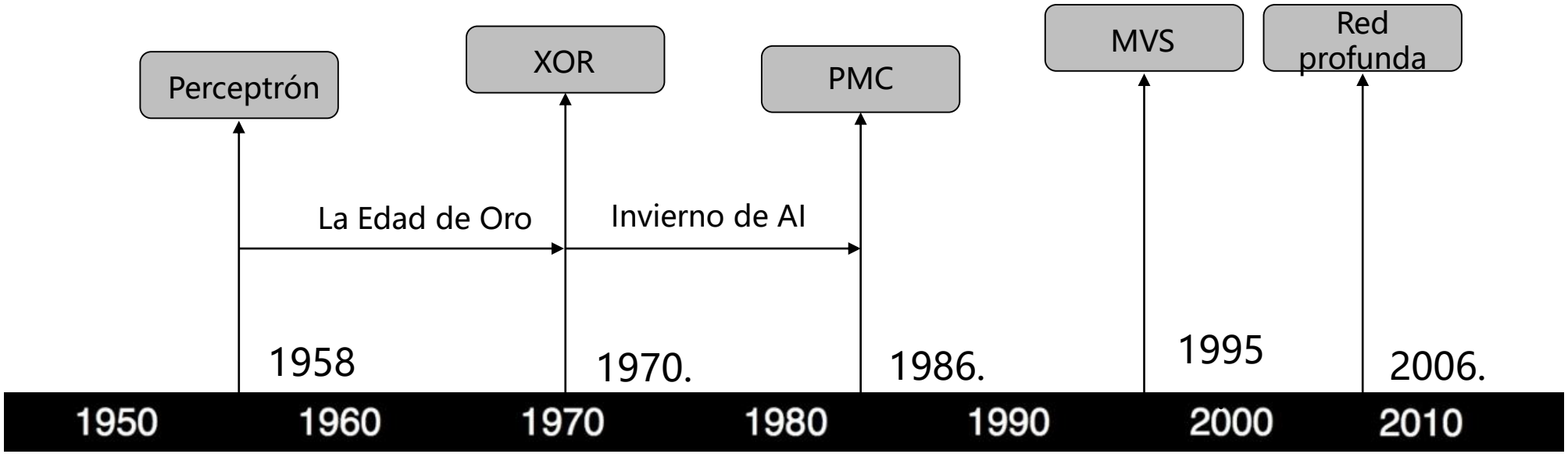


Red neural profunda

Red Neural

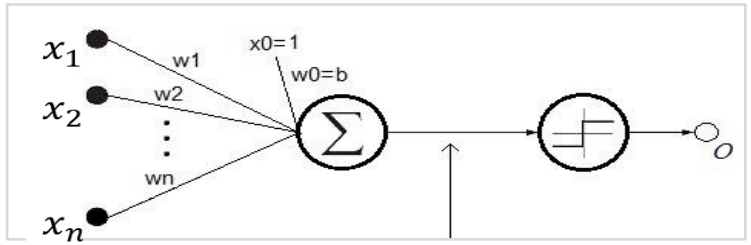
- En la actualidad, la definición de la red neural no se ha determinado todavía. Hecht Nielsen, investigador de redes neuronales en Estados Unidos, define una red neural como un sistema informático compuesto por elementos de procesamiento simples y altamente interconectados, que procesan la información mediante una respuesta dinámica a las entradas externas.
- Una red neural puede expresarse simplemente como un sistema de procesamiento de información diseñado para imitar la estructura y las funciones del cerebro humano basándose en su fuente, características y explicaciones.
- Red neuronal artificial (red neural): formada por neuronas artificiales conectadas entre sí, la red neural extrae y simplifica la microestructura y las funciones del cerebro humano. Es un enfoque importante para simular la inteligencia humana y reflejar varias características básicas de las funciones del cerebro humano, como el procesamiento simultáneo de información, el aprendizaje, la asociación, la clasificación de modelos y la memoria.

Historial de desarrollo de redes neuronales



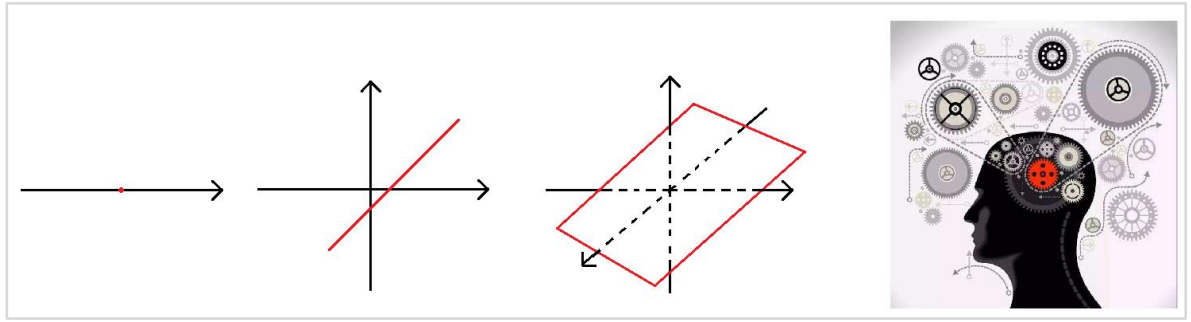
Perceptrón de una capa

- Vector de entrada: $X = [x_0, x_1, \ldots, x_n]^T$
- Peso: $W = [\omega_0, \omega_1, \ldots, \omega_n]^T$, donde ω_0 is the offset.
- Función de activación: $O = sign(net) = \begin{cases} 1, net > 0, \\ -1, otherwise. \end{cases}$



$$net = \sum_{i=0}^{n} \omega_i x_i = \boldsymbol{W^T X}$$

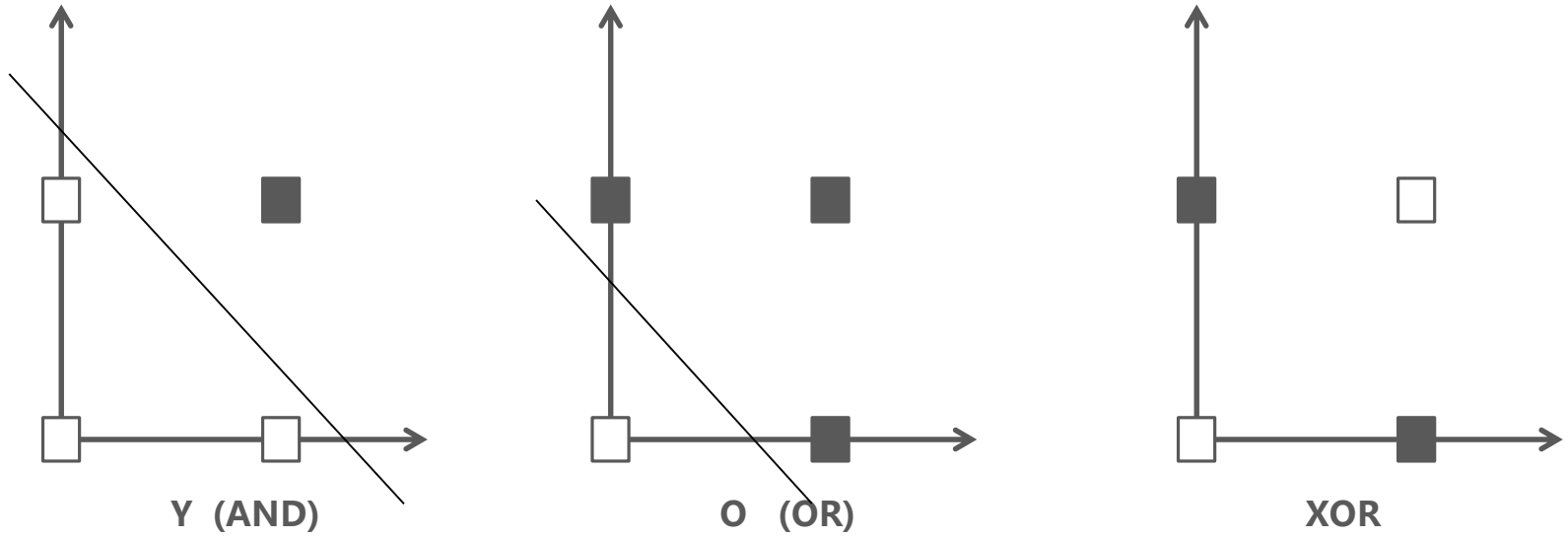
- El perceptrón anterior es equivalente a un clasificador. Utiliza el vector X de alta dimensión como entrada y realiza la clasificación binaria en muestras de entrada en el espacio de alta dimensión. Cuando $W^TX>0$, O=1. En este caso, las muestras se clasifican en un tipo. De lo contrario, O=-1. En este caso, las muestras se clasifican en el otro tipo. El límite de estos dos tipos es $W^TX=0$, que es un hiperplano de alta dimensión.



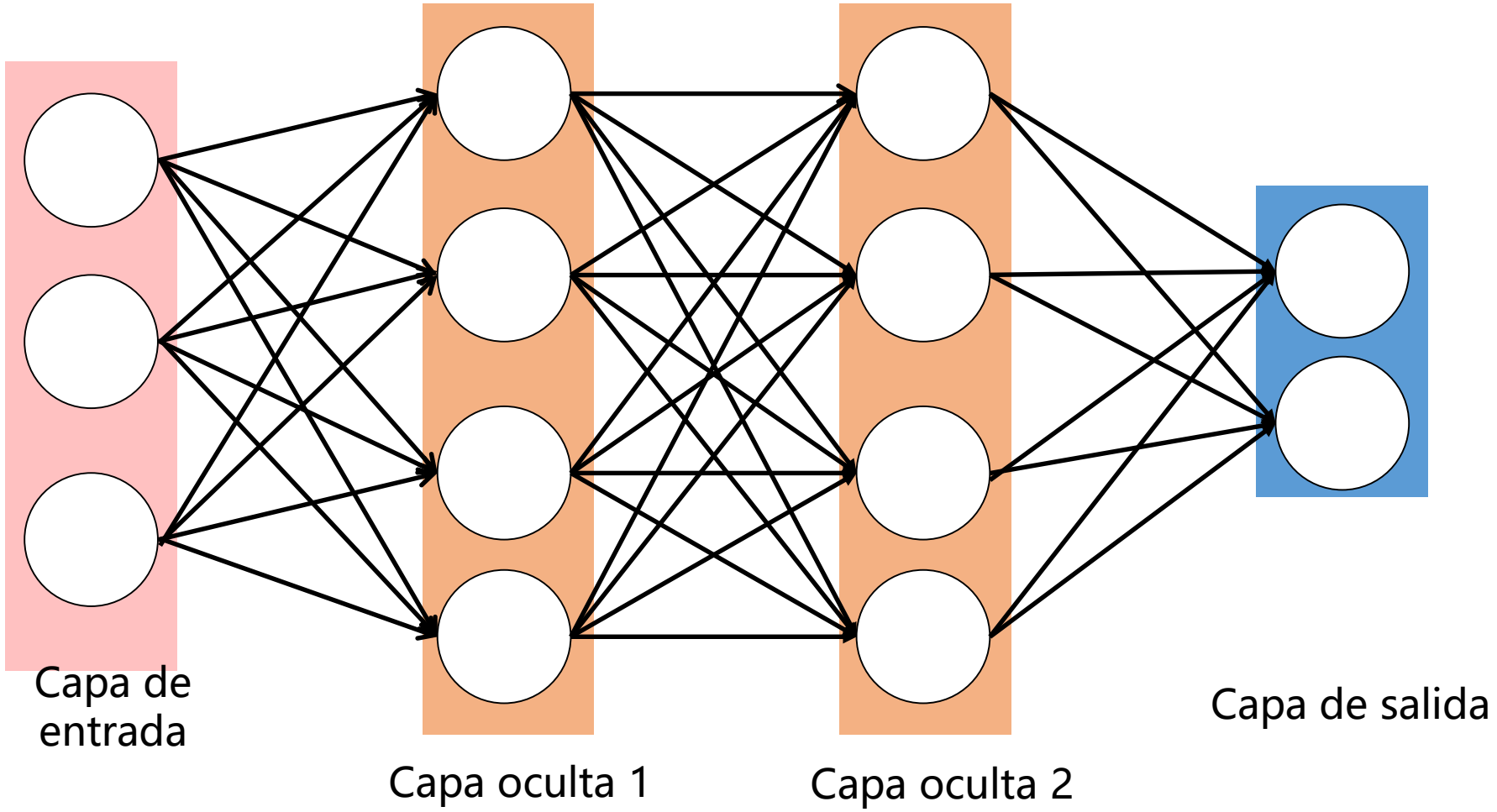
Classification point	Classification line	Classification plane	Classification hyperplane
$Ax + B = 0$	$Ax + By + C = 0$	$Ax + By + Cz + D = 0$	$W^T\mathrm{X} + b = 0$

Problema de XOR

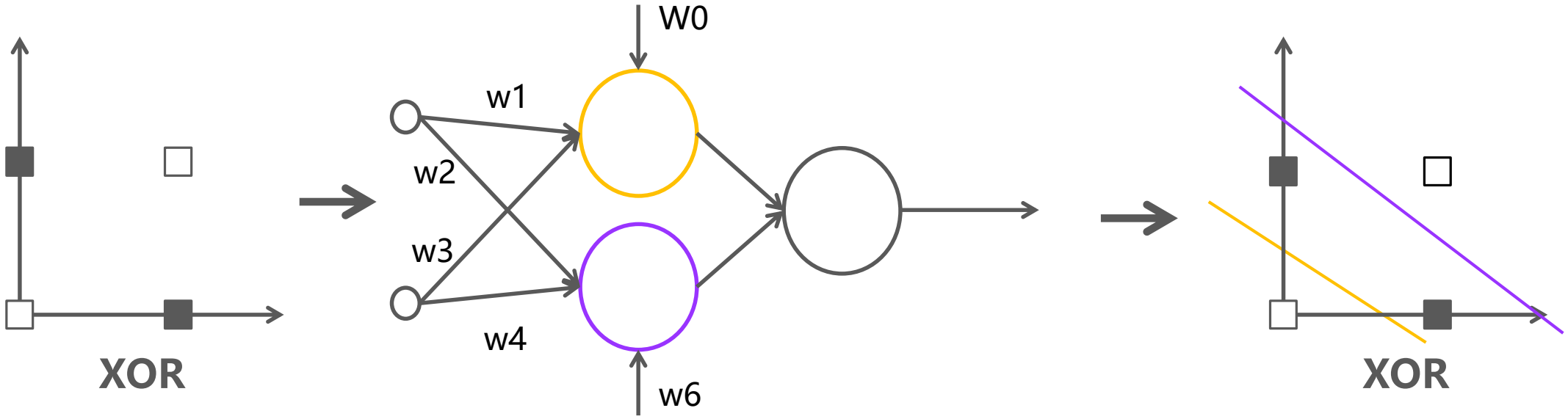
- En 1969, Minsky, un matemático estadounidense y pionero de la IA, demostró que un perceptrón es esencialmente un modelo lineal que sólo puede tratar problemas de clasificación lineal, pero no puede procesar datos no lineales.



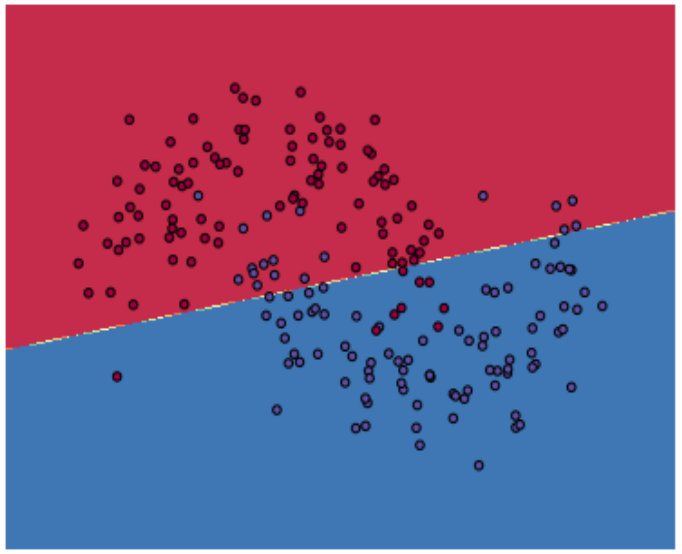
Red neural de feedforward



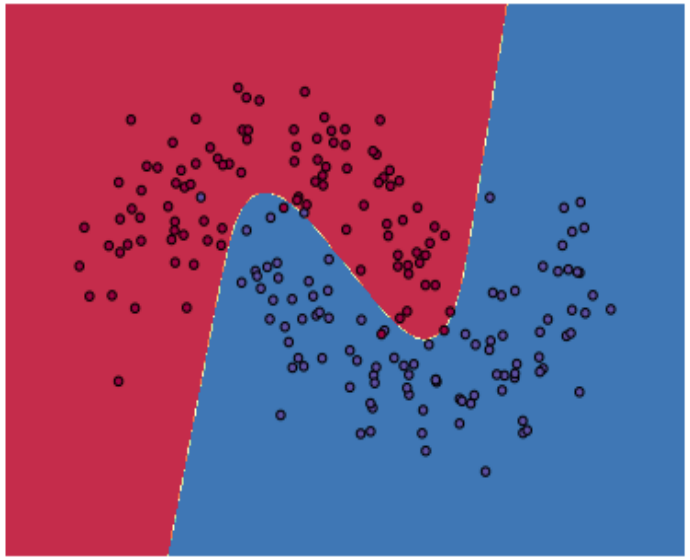
Solución de XOR



Impactos de capas ocultas en una red neural



Capas ocultas 0



Tres capas ocultas

Veinte capas ocultas

CONTENIDO

Función de pérdida y descenso de gradiente

- El gradiente de la función multivariante $o = f(x) = f(x_0, x_1, \dots, x_n)$ at $X' = [x_0', x_1', \dots, x_n']^T$ se muestra de esta forma:

$$\nabla f(x_0', x_1', \dots, x_n') = [\frac{\partial f}{\partial x_0}, \frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n}]^T |_{X=X'},$$

La dirección del vector de gradiente es la dirección de crecimiento más rápido de la función. Como resultado, la dirección del vector de gradiente negativo $-\nabla f$ es la dirección de descenso más rápida de la función.

Durante el entrenamiento de la red de aprendizaje profundo, se deben parametrizar los errores de clasificación de destino. Se utiliza una **función de pérdida (función de error),** que refleja el error entre la salida objetivo y la salida real del perceptrón. Para una sola muestra de entrenamiento x, la función de error más común es la **Función de costo cuadrático.**

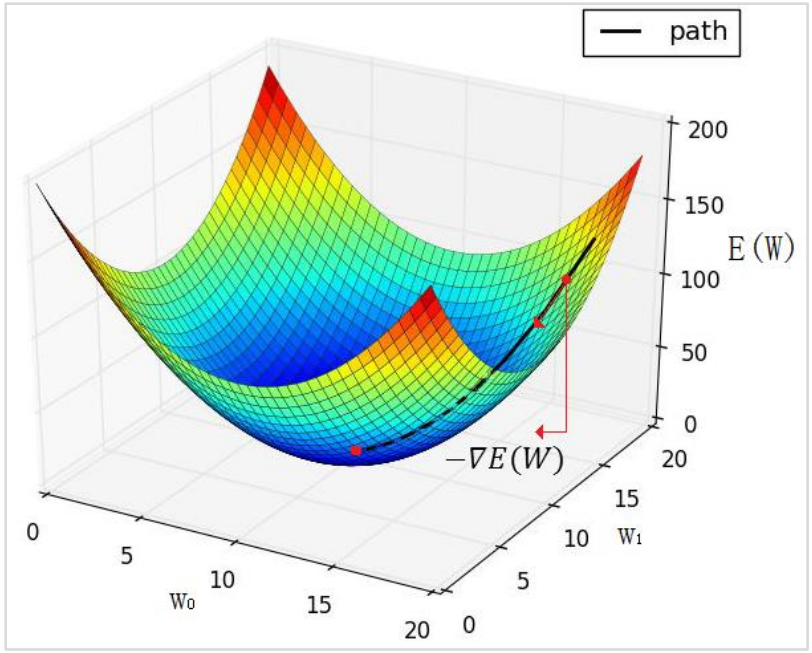
$$E(w) = \frac{1}{2}\sum_{d \in D}(t_d - o_d)^2,$$

En la función anterior, d es una neurona en la capa de salida, D son todas las neuronas en la capa de salida, t_d es la salida de destino y o_d es la salida real.

El método de descenso de gradiente permite que la función de pérdida busque a lo largo de la dirección del gradiente negativo y actualice los parámetros de forma iterativa, minimizando finalmente la función de pérdida.

Extrema de la función de pérdida

- Propósito: La función de pérdida E(W) se define en el espacio de peso. El objetivo es buscar el vector de peso W que puede minimizar E(W).
- Limitación: Ningún método eficaz puede resolver el extremo de las matemáticas en la compleja superficie de alta dimensión de: $E(W) = \frac{1}{2}\sum_{d \in D}(t_d - o_d)^2$.



Ejemplo de descenso de gradiente de paraboloide binario

Funciones de pérdida común en el aprendizaje profundo

- Función de coste cuadrática:

$$E(W) = \frac{1}{2}\sum_{d \in D}(t_d - o_d)^2$$

- Función de error de entropía cruzada:

$$E(W) = -\frac{1}{n}\sum_{x}\sum_{d \in D}[t_d \ln o_d + (1 - t_d)\ln(1 - o_d)]$$

- La función de error de entropía cruzada representa la distancia entre dos distribuciones de probabilidad, lo cual es una función de pérdida ampliamente utilizada para problemas de clasificación.
- Generalmente, la función de error cuadrado medio se utiliza para resolver el problema de regresión, mientras que la función de error de entropía cruzada se utiliza para resolver el problema de clasificación.

Algoritmo de Descenso de Gradiente por Lotes (BGD)

- En el conjunto de muestras de entrenamiento X, cada muestra se registra como $< x,t >$, en la que X es el vector de entrada, t la salida de destino, o la salida real y η la tasa de aprendizaje.
- Inicializa cada w_i en un valor aleatorio con un valor absoluto más pequeño.
- Antes que la condición final se cumpla:
  - Inicializa cada Δw_i a cero:
  - Por cada iteración:
    - Pasa todas las X a esta unidad y calcula el resultado. o_x
    - Por cada w_i en esta unidad: Δw_i $\Delta w_i += -\eta \frac{1}{n} \sum_{x \in X} \frac{\partial C(t_x, o_x)}{\partial w_i}$.
    - Por cada w_i en esta unidad : $w_i += \Delta w_i$.
- El algoritmo de descenso de gradiente de esta versión no se utiliza comúnmente porque:
- El proceso de convergencia es muy lento, ya que todas las muestras de entrenamiento deben calcularse cada vez que se actualiza el peso.

Algoritmo de Descenso de Gradiente Estocástico (SGD)

- Para abordar el defecto del algoritmo BGD, se utiliza una variante común llamada algoritmo de Descenso de Gradiante Incremental, que también se denomina algoritmo de Descenso de Gradiente Estocástico (SGD). Una implementación se llama Aprendizaje en Línea, que actualiza el gradiente en función de cada muestra:

$$\Delta w_i = -\eta \frac{1}{n} \sum_{x \in X} \frac{\partial C(t_x, o_x)}{\partial w_i} \Longrightarrow \Delta w_i = -\eta \frac{\partial C(t_x, o_x)}{\partial w_i}.$$

- ONLINE-GRADIENT-DESCENT
  - Inicializa cada w_i a un valor aleatorio con un valor absoluto más pequeño.
  - Antes de que se cumpla la condición final:
  - Genera un aleatorio <X, t> a partir de X y realiza el siguiente cálculo:
    - Introduzca X en esta unidad y calcule la salida o_x.
    - Por cada w_i en esta unidad : $w_i \mathrel{+}= -\eta \frac{\partial C(t_x, o_x)}{\partial w_i}$.

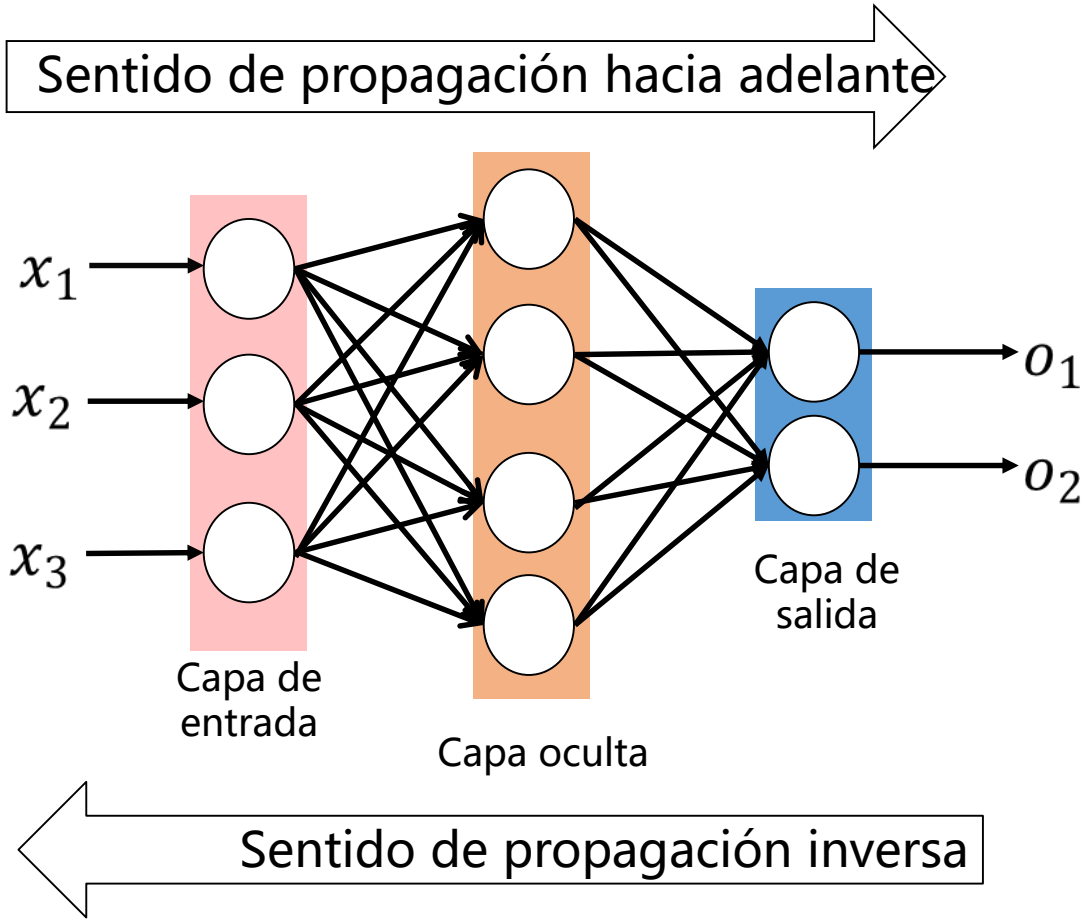
Algoritmo de Descenso de Gradiente Mini-Batch (MBGD)

- Para enfrentar los defectos de los dos algoritmos de Descenso de Gradiente anteriores, se propuso el algoritmo de descenso de Degradado de Mini-Batch (MBGD) y se ha utilizado más ampliamente. Un pequeño número de muestras de tamaño de lote (Batch Size - BS) se utilizan a la vez para calcular Δw_i ,y a continuación, el peso se actualiza en consecuencia.
- Descenso de Gradiente por Batch/lotes
  - Inicializa cada w_i en un valor aleatorio con un valor absoluto más pequeño.
  - Antes de que se cumpla la condición final:
    - Inicializa cada Δw_i a cero.
    - Por cada $<X, t>$ uno en las muestras de BS en el siguiente lote en B:
      - Introduzca X en esta unidad y calcule la salida o_x.
      - Por cada w_i en esta unidad $\Delta w_i += -\eta \frac{1}{n} \sum_{x \in B} \frac{\partial C(t_x o_x)}{\partial w_i}$
    - Por cada w_i en esta unidad: $w_i += \Delta w_i$
    - Para el último lote, las muestras de entrenamiento se mezclan en un orden aleatorio.

Algoritmo de Propagacion Inversa - BackPropagation(1)

- Las señales se propagan hacia adelante y los errores se propagan hacia atrás.
- En el conjunto de muestras de entrenamiento D, cada muestra se registra como < *X*,*t* >, en el que *X* es el vector de entrada, *t* la salida objetivo, *o* la salida real, y *w* el coeficiente de peso.
- Función de pérdida:

$$E(w)=\frac{1}{2}\sum_{(d\in D)}(t_d-o_d)^2$$



Algoritmo de Propagación Inversa - Backpropagation(2)

- De acuerdo con las siguientes fórmulas, los errores en las capas de entrada, ocultas y de salida se acumulan para generar el error en la función de pérdida.
- wc es el coeficiente de peso entre la capa oculta y la capa de salida, mientras que wb es el coeficiente de peso entre la capa de entrada y la capa oculta. *f* es la función de activación, D es el conjunto de capas de salida y C y B son el conjunto de capas ocultas y la capa de entrada, respectivamente. Supongamos que la función de pérdida es una función de coste cuadrático:

- Error de capa de salida: $$E = \frac{1}{2}\sum_{(d \in D)}(t_d - o_d)^2$$
- Error de capa oculta expandida: $$E = \frac{1}{2}\sum_{(d \in D)}\left[t_d - f(net_d)\right]^2 = \frac{1}{2}\sum_{(d \in D)}\left[t_d - f\left(\sum_{(c \in C)} w_c y_c\right)\right]^2$$
- Error de capa de entrada expandida: $$E = \frac{1}{2}\sum_{(d \in D)}\left[t_d - f\left(\sum_{(c \in C)} w_c f(net_c)\right)\right]^2 = \frac{1}{2}\sum_{(d \in D)}\left[t_d - f\left(\sum_{(c \in C)} w_c f\left(\sum_{b \in B} w_b x_b\right)\right)\right]^2$$

Algoritmo de propagación inversa - Backpropagation(3)

- Para minimizar el error E, el cálculo iterativo de descenso de gradiente se puede utilizar para resolver W_c y W_b, es decir, calcular W_c y W_b para minimizar el error E.
- Fórmula:

$$\Delta w_c = -\eta \frac{\partial E}{\partial w_c}, c \in C$$

$$\Delta w_b = -\eta \frac{\partial E}{\partial w_b}, b \in B$$

- Si hay varias capas ocultas, se utilizan reglas de cadena para tomar una derivada para cada capa para obtener los parámetros optimizados por iteración.

Algoritmo de propagación inversa - Backpropagation(4)

- Para una red neural con cualquier número de capas, la fórmula organizada para el entrenamiento es la siguiente:

$$\Delta w_{jk}^{l} = -\eta \delta_{k}^{l+1} f_{j}(z_{j}^{l})$$

$$\delta_{j}^{l} = \begin{cases} f_{j}^{'}(z_{j}^{l})(t_{j} - f_{j}(z_{j}^{l})), l \in outputs, (1) \\ \sum_{k} \delta_{k}^{l+1} w_{jk}^{l} f_{j}^{'}(z_{j}^{l}), otherwise, (2) \end{cases}$$

- El algoritmo PB se utiliza para entrenar la red de la siguiente manera:
  - Toma la siguiente muestra de capacitación < X, T >, ingresa X a la red y obtiene la salida real o.
  - Calcula la capa de salida δ de acuerdo con la fórmula de error de capa de salida (1).
  - Calcula δ de cada capa oculta de salida a entrada por iteración de acuerdo con la fórmula de propagación de errores de capa oculta (2).
  - De acuerdo con δ de cada capa, se actualizan los valores de peso de todas las capas.

CONTENIDO

Función de activación

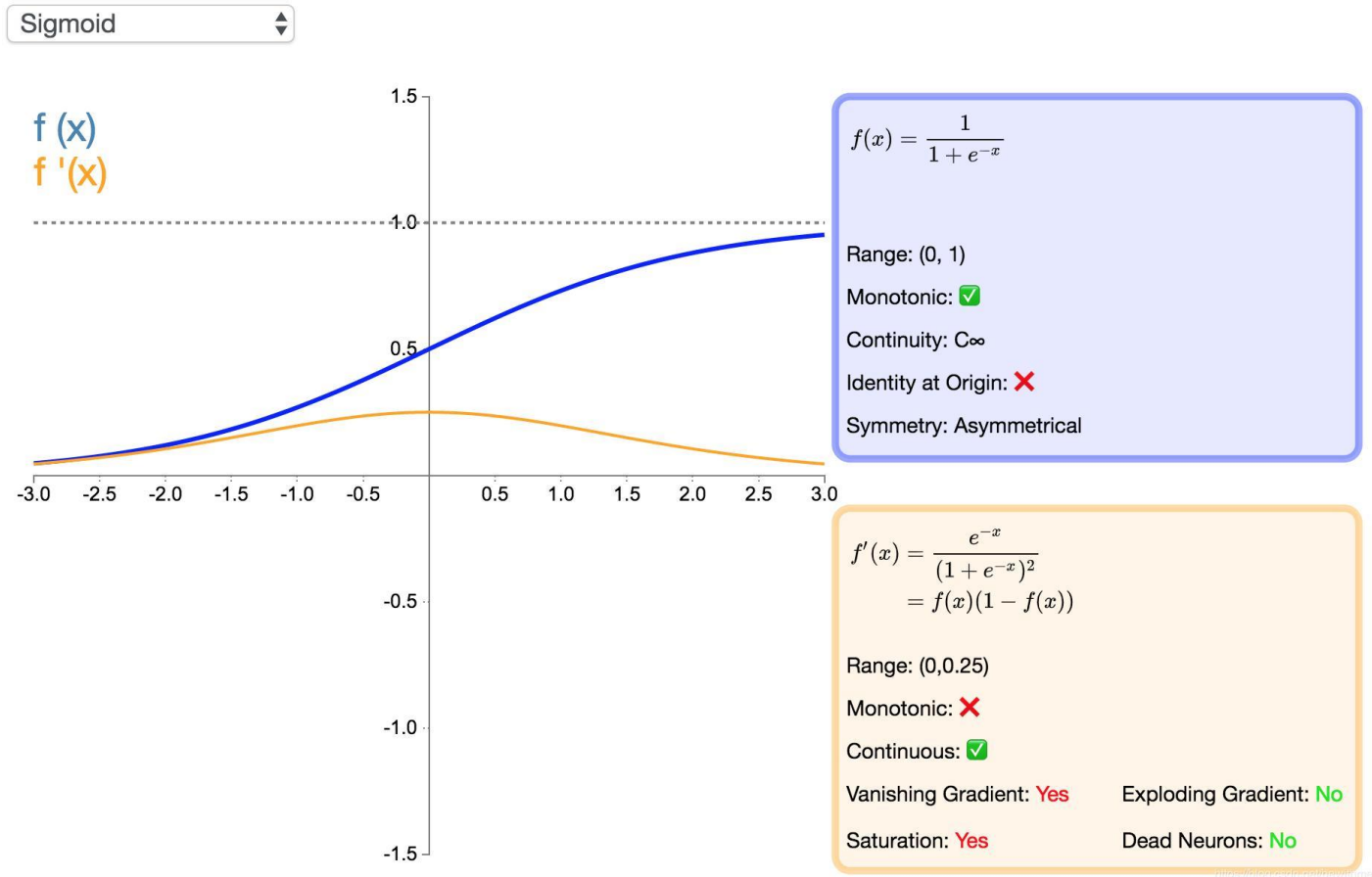
- Las funciones de activación son importantes para que el modelo de red neural aprenda y comprenda funciones complejas no lineales. Permiten la introducción de funcionalidades no lineales en la red.
- Sin funciones de activación, las señales de salida son sólo funciones lineales simples. La complejidad de las funciones lineales es limitada y la capacidad de aprender asignaciones de funciones complejas a partir de datos es baja.

Función de activación

$$output = f(w_1 x_1 + w_2 x_2 + w_3 x_3 \mathrm{K}) = f(W^t \bullet X)$$

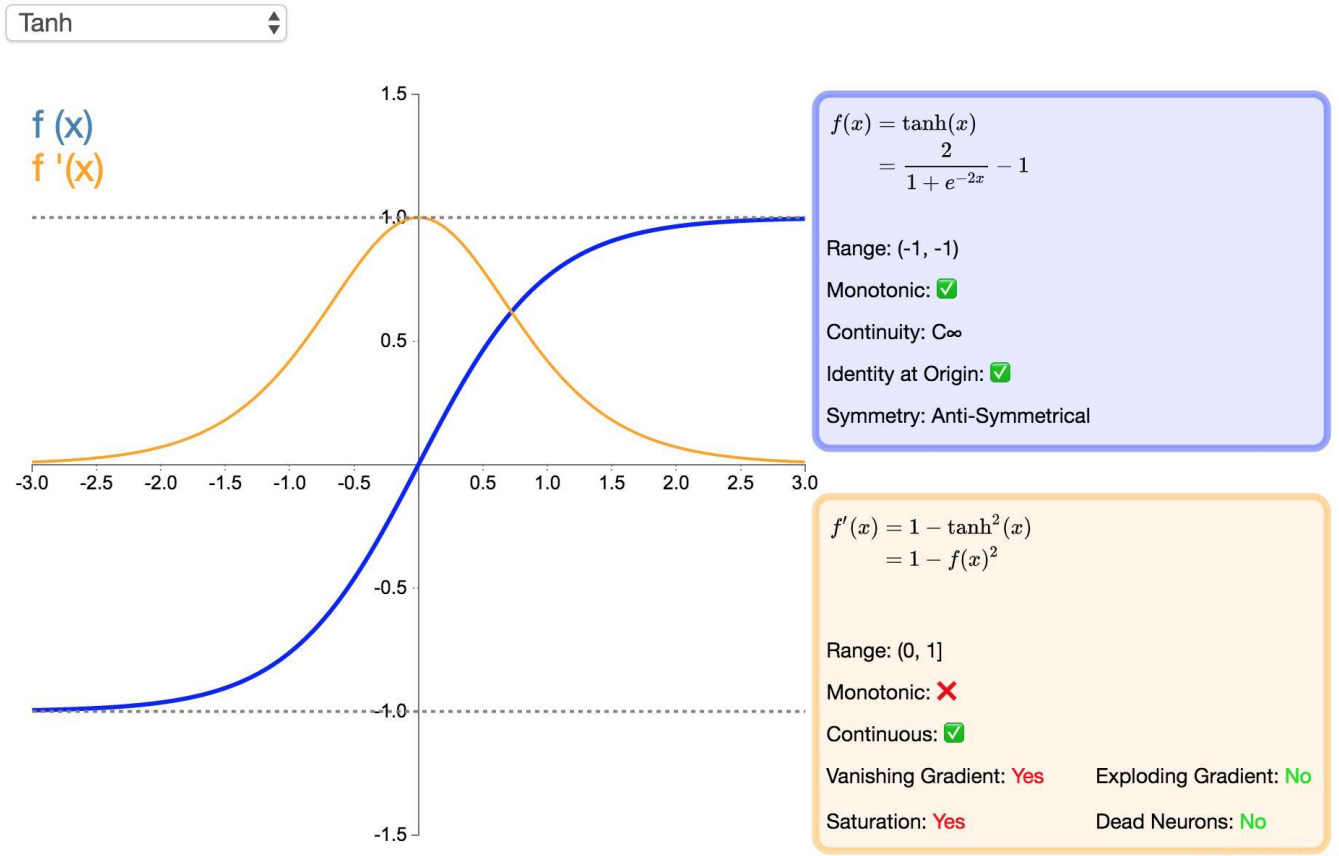
Sigmoide.

$$f(x) = \frac{1}{1 + e^{-x}}$$



Tanh.

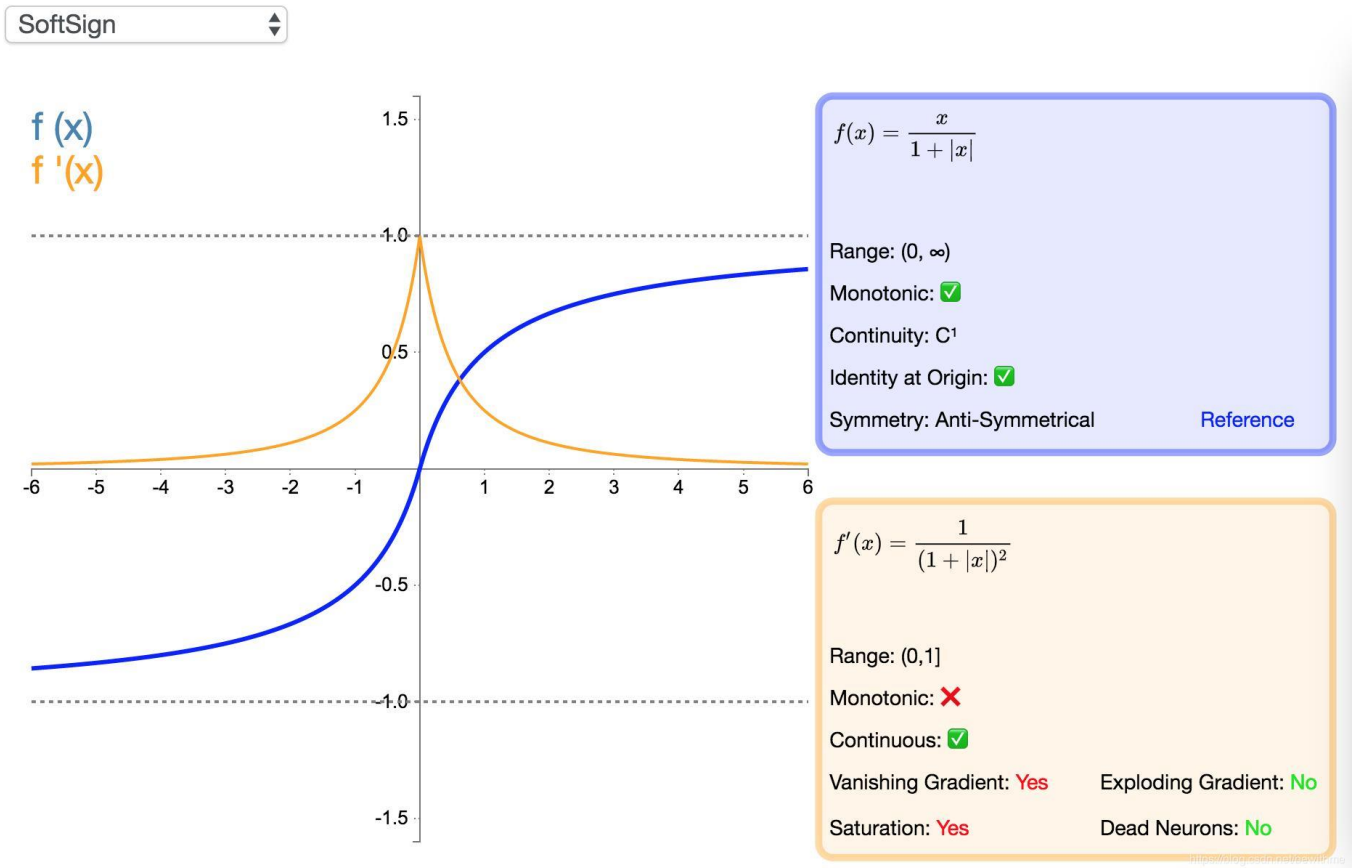
$$\tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$



Suavizar (softsign)

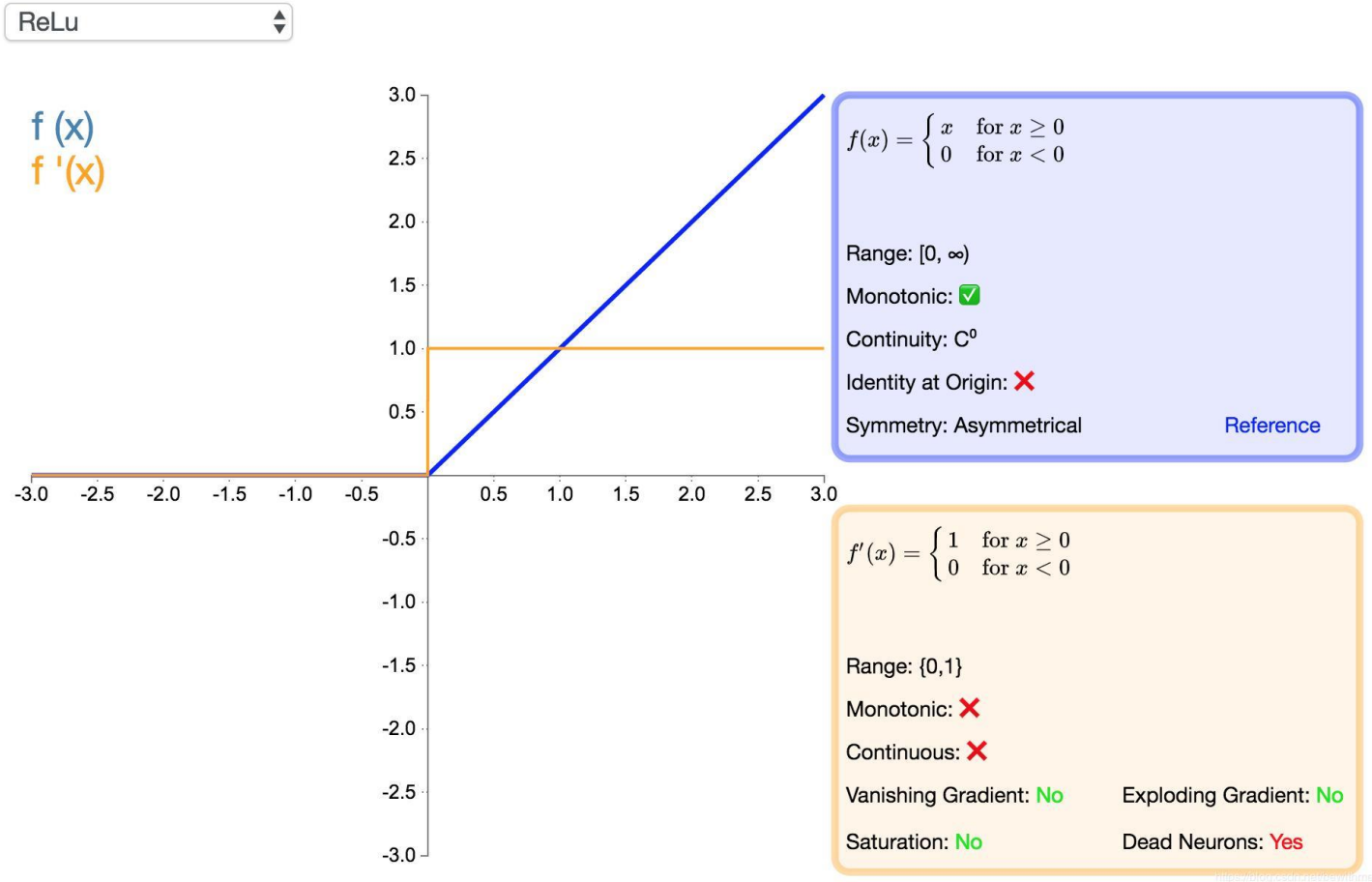
$$f(x) = \frac{x}{|x| + 1}$$

$$f\,(x) = \frac{\quad}{\quad}$$



Unidad lineal rectificada (ReLU)

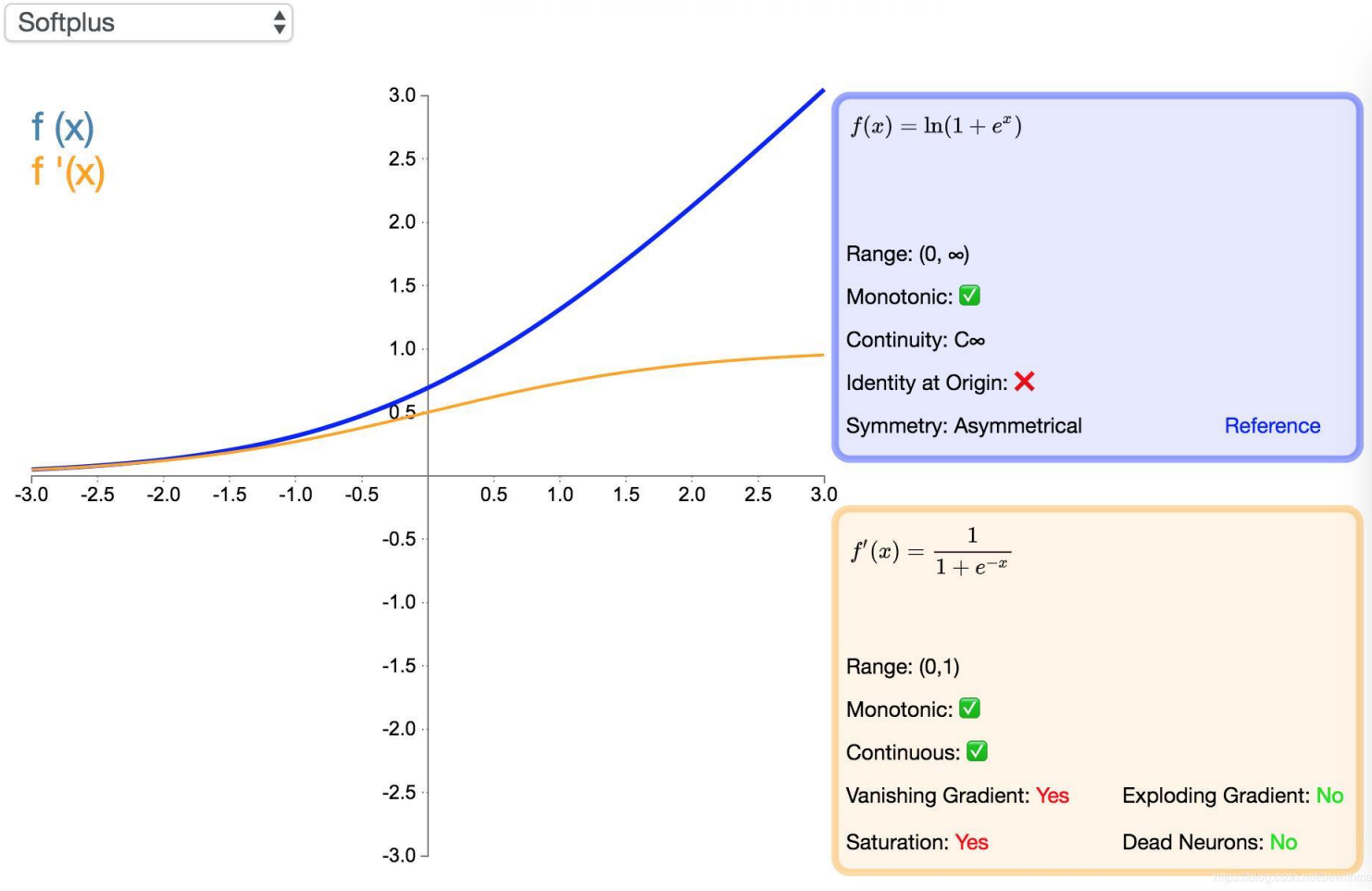
$$y = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases}$$



HUAWEI

Suave (Softplus)

$$f(x) = \ln(e^x + 1)$$



Softmax

- Softmax:

$$\sigma(\mathrm{z})_j = \frac{e^{z_j}}{\sum_k e^{z_k}}$$

- La función Softmax se utiliza para asignar un vector de dimensiones K de valores reales arbitrarios a otro vector de dimensiones K de valores reales, donde cada elemento vectorial está en el intervalo (0, 1). Todos los elementos suman 1.
- La función Softmax se utiliza a menudo como la capa de salida de una tarea de clasificación multiclase.

CONTENIDO

Normalizador

- La regularización es una tecnología importante y eficaz para reducir los errores de generalización en el aprendizaje automático. Es especialmente útil para los modelos de aprendizaje profundo que tienden a tener sobreajuste debido a un gran número de parámetros. Por lo tanto, los investigadores han propuesto muchas tecnologías eficaces para prevenir el sobreajuste, incluyendo:
- Adición de restricciones a parámetros, como normas L_1 y L_2
- Ampliar el conjunto de entrenamiento, como agregar ruido y transformar datos
- Dropout (Abandono)
- Early stopping (Paro temprano)

Parámetros de penalización

- Muchos métodos de regularización restringen la capacidad de aprendizaje de los modelos agregando un parámetro de penalización Ω (θ) a la función objetivo J. Suponga que la función objetivo después de la regularización es J ~.

$$\tilde{J}(\theta; X, y) = J(\theta; X, y) + \alpha\Omega(\theta),$$

- Donde $\alpha\epsilon$ [0, ∞) es un hiperparámetro que pondera la contribución relativa del término de penalización normal Ω y la función objetivo estándar J (X; θ). Si α se establece en 0, no se realiza ninguna regularización. La penalización en la regularización aumenta con α.

Regularización L_1

- Agregue una restricción de norma L_1 a los parámetros del modelo, es decir,

$$\tilde{J}(w; X, y) = J(w; X, y) + \alpha \|w\|_1,$$

- Si se utiliza un método de gradiente para resolver el valor, el gradiente del parámetro es.

$$\nabla \tilde{J}(w) = \propto sign(w) + \nabla J(w).$$

Regularización L_2

- Agrega la norma de penalizacion L2, para prevenir el overfitting (sobreajuste).

$$\tilde{J}(w; X, y) = J(w; X, y) + \frac{1}{2}\alpha\|w\|_2^2,$$

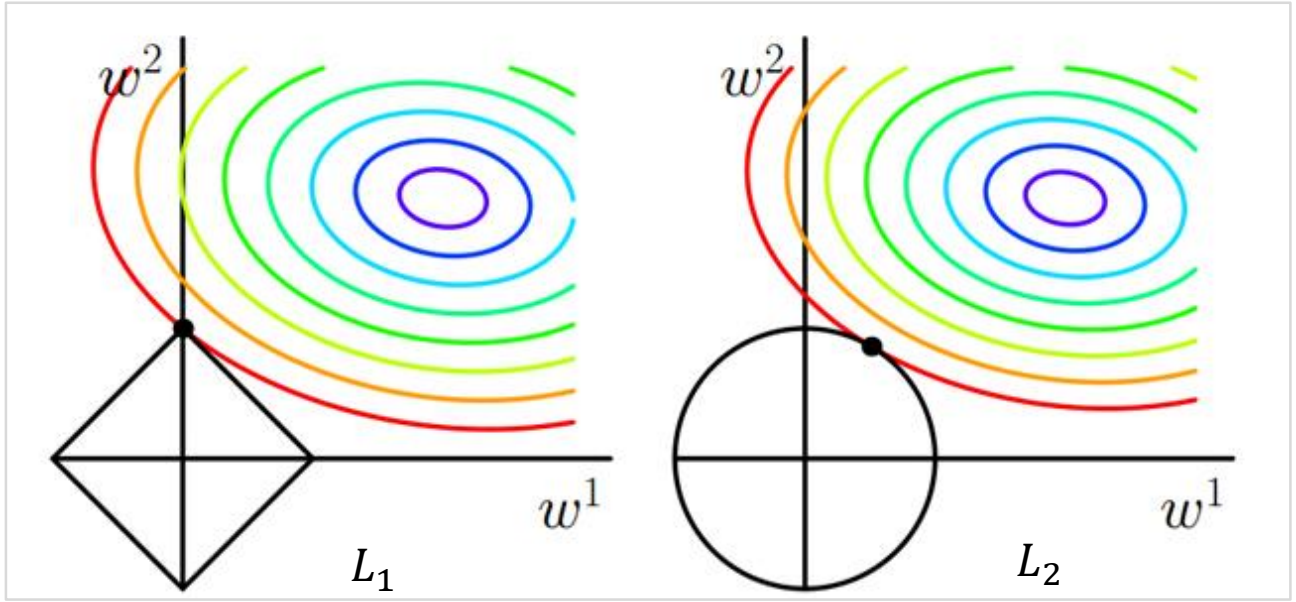
- Se puede inferir un método de optimización de parámetros utilizando una tecnología de optimización (como un método de gradiente):

$$w = (1 - \varepsilon\alpha)\omega - \varepsilon\nabla J(w),$$

- donde ε es la tasa de aprendizaje. En comparación con una fórmula de optimización de gradiente común, esta fórmula multiplica el parámetro por un factor de reducción.

L_1 v.s. L_2

- Las mayores diferencias entre L_2 y L_1:
  - Según el análisis anterior, L_1 puede generar un modelo más disperso que L_2. Cuando el valor del parámetro w es pequeño, la regularización L_1 puede reducir directamente el valor del parámetro a 0, que se puede utilizar para la selección de características.
  - Desde la perspectiva de la probabilidad, muchas restricciones normativas equivalen a agregar una distribución de probabilidad previa a los parámetros. En la regularización L_2, el valor del parámetro cumple con la regla de distribución gaussiana. En la regularización L_1, el valor del parámetro cumple con la regla de distribución de Laplace.

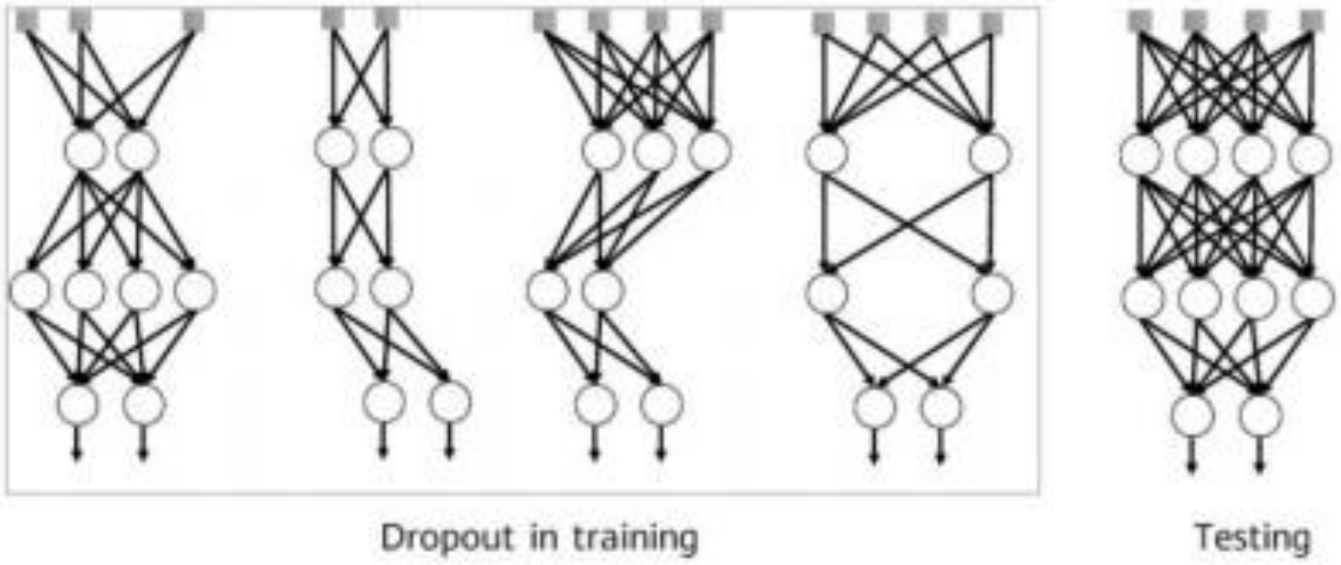


Expansión del conjunto de datos

- La forma más eficaz de evitar un sobreajuste es agregar un conjunto de entrenamiento. Un conjunto de entrenamiento más grande tiene una probabilidad de sobreajuste menor. La expansión del conjunto de datos es un método que ahorra tiempo, pero varía en diferentes campos.
  - Un método común en el campo del reconocimiento de objetos es rotar o escalar imágenes. (El requisito previo para la transformación de imágenes es que el tipo de imagen no se puede cambiar mediante la transformación. Por ejemplo, para el reconocimiento de dígitos escritos a mano, las categorías 6 y 9 se pueden cambiar fácilmente después de la rotación).
  - El ruido aleatorio se agrega a los datos de entrada en el reconocimiento de voz.
  - Una práctica común del procesamiento del lenguaje natural (PLN) es reemplazar palabras con sus sinónimos.
  - La inyección de ruido puede agregar ruido a la entrada o a la capa oculta o capa de salida. Por ejemplo, para la clasificación Softmax, se puede agregar ruido utilizando la tecnología de suavizado de etiquetas.
    Si se agrega ruido a las categorías 0 y 1, las probabilidades correspondientes se cambian a $\frac{\varepsilon}{k}$ y $1-\frac{k-1}{k}\varepsilon$ respectivamente.

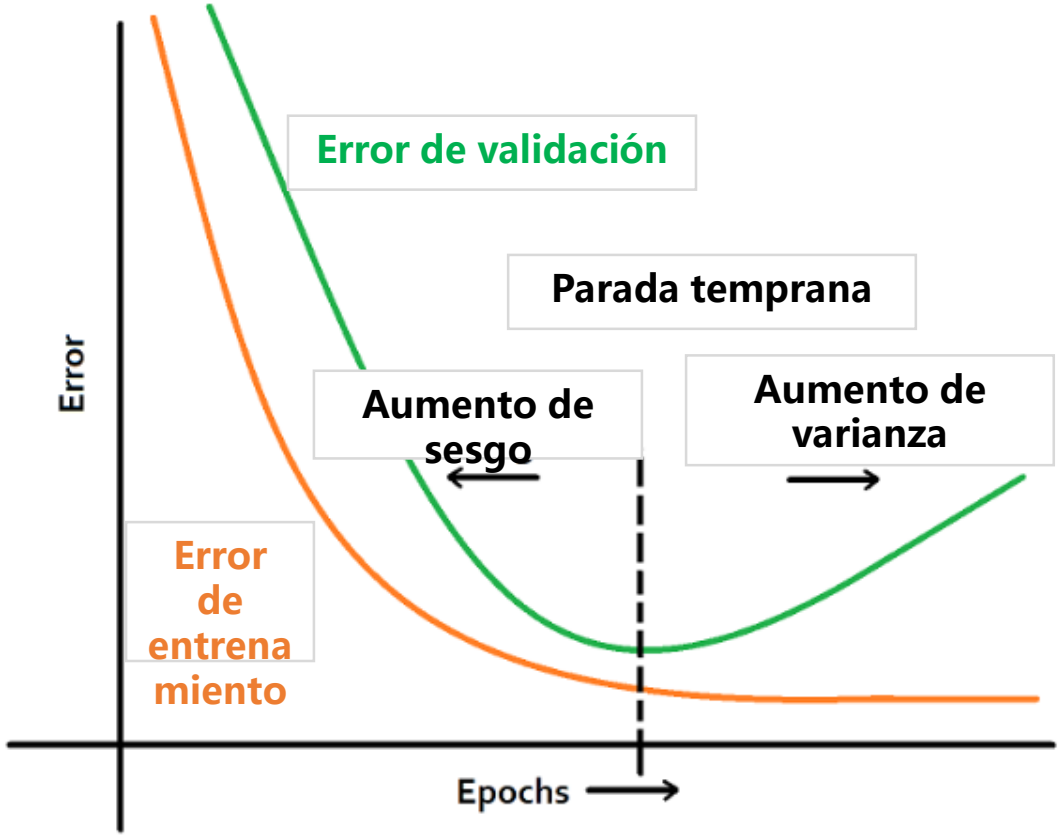
Dropout (Abandono)

- El abandono es un método común y sencillo de regularización, que se ha utilizado ampliamente desde 2014. En pocas palabras, el Dropout al azar descarta algunas entradas durante el proceso de entrenamiento. En este caso, los parámetros correspondientes a las entradas descartadas no se actualizan. Como método de integración, Dropout combina todos los resultados de la subred y obtiene subredes descartando entradas aleatoriamente. Vea las siguientes figuras:



Paro temprana (Early stopping)

- Durante el entrenamiento se puede introducir una prueba de los datos del conjunto de validación. Cuando aumente la pérdida de datos del conjunto de verificación, realice la detención temprana.



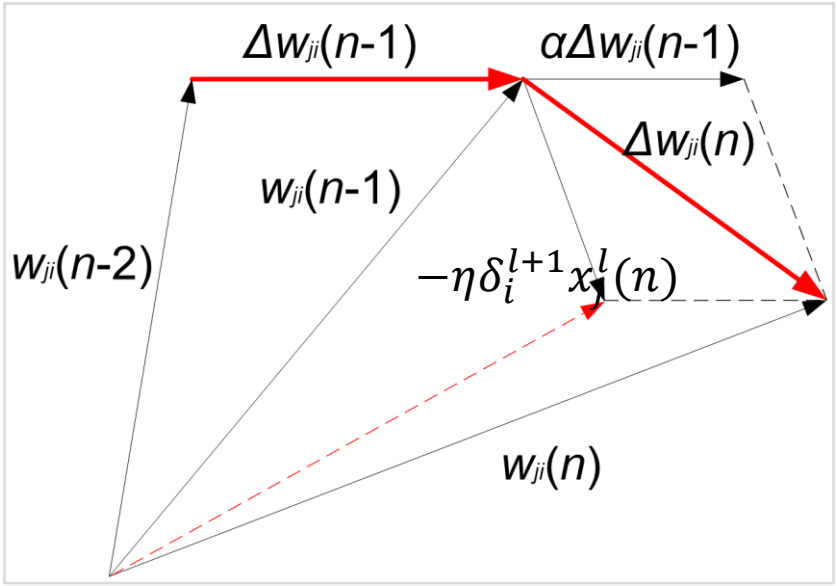
CONTENIDO

Optimizador

- Hay varias versiones optimizadas de algoritmos de descenso de gradiente. En la implementación de lenguaje orientado a objetos, los diferentes algoritmos de descenso de gradientes se suelen encapsular en objetos llamados optimizadores.
- Los objetivos de la optimización del algoritmo incluyen, entre otros:
  - Acelerar la convergencia de algoritmos.
  - Prevenir o saltarse valores extremos locales.
  - Simplificación de la configuración manual de parámetros, especialmente la tasa de aprendizaje TA (Learning Rate - LR).
- Optimizadores comunes: optimizador de GDs común, optimizador de momento, Nesterov, AdaGrad, AdaDelta, RMSProp, Adam, AdaMax y Nadam.

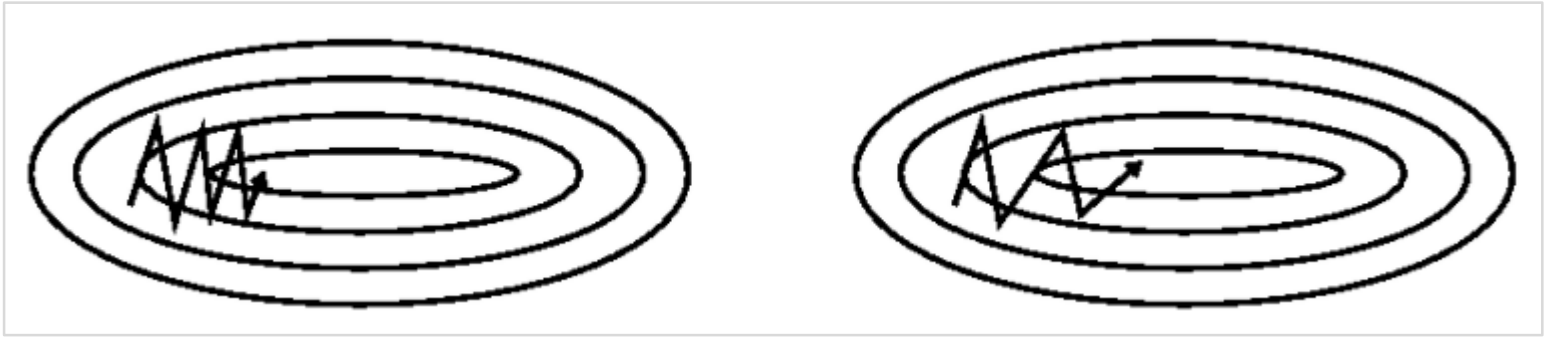
Optimizador de momento Momentum

- Una de las mejoras más básicas es agregar términos de momento para Δw_{ji}. Suponga que la corrección de peso de la n-ésima iteración es $\Delta w_{ji}(n)$. La regla de corrección de peso es:
- $\Delta w_{ji}^{l}(n) = -\eta \delta_i^{l+1} x_j^l(n) + \alpha \Delta w_{ji}^{l}(n-1)$
- donde α es una constante ($0 \leq \alpha < 1$) llamada Coeficiente de momento y $\alpha \Delta w_{ji}(n-1)$ es un término de momento.
- Imagina que una pequeña bola rueda hacia abajo desde un punto aleatorio en la superficie de error. La introducción del término de momento es equivalente a dar inercia a la bola pequeña.



Ventajas y desventajas del optimizador de momento

- Ventajas:
  - Mejora la estabilidad de la dirección de corrección del gradiente y reduce las mutaciones.
  - En áreas donde la dirección del gradiente es estable, la bola rueda cada vez más rápido (hay un límite superior de velocidad porque α <1), lo que ayuda a que la bola rebase rápidamente el área plana y acelera la convergencia.
  - Es más probable que una bola pequeña con inercia ruede sobre algunos extremos locales estrechos.
- Desventajas:
  - La tasa de aprendizaje η y el impulso α deben establecerse manualmente, lo que a menudo requiere más experimentos para determinar el valor apropiado.



Optimizador AdaGrad (1)

- La característica común del algoritmo de descenso de gradiente aleatorio (SGD), el algoritmo de descenso de gradiente de lotes pequeños (MBGD) y el optimizador de impulso es que cada parámetro se actualiza con el mismo LR..
- Según el enfoque de AdaGrad, es necesario establecer diferentes tasas de aprendizaje para diferentes parámetros.

$$g_t = \frac{\partial C(t,o)}{\partial w_t}$$ Calculo del gradiente

$$r_t = r_{t-1} + g_t^2$$ Acumulación de gradiente cuadrado

$$\Delta w_t = -\frac{\eta}{\varepsilon + \sqrt{r_t}} g_t$$ Actualizacion de computo

$$w_{t+1} = w_t + \Delta w_t$$ Actualización de aplicación

- g_t indica el gradiente t-ésimo, r es una variable de acumulación de gradiente y el valor inicial de r es 0, que **aumenta continuamente**. η indica el LR global, que debe configurarse manualmente. ε es una pequeña constante y se establece en aproximadamente 10-7 para estabilidad numérica..

Optimizador AdaGrad (2)

- El algoritmo de optimización de AdaGrad muestra que r continúa aumentando mientras que la tasa de aprendizaje general sigue disminuyendo a medida que el algoritmo itera. Esto se debe a que esperamos que LR disminuya a medida que aumenta el número de actualizaciones. En la fase de aprendizaje inicial, estamos muy lejos de la solución óptima para la función de pérdida. A medida que aumenta el número de actualizaciones, estamos más cerca de la solución óptima y, por lo tanto, LR puede disminuir..
- Pros:
  - La tasa de aprendizaje se actualiza automáticamente. A medida que aumenta el número de actualizaciones, la tasa de aprendizaje disminuye.
- Cons:
  - El denominador sigue acumulándose de modo que la tasa de aprendizaje eventualmente se volverá muy pequeña y el algoritmo se volverá ineficaz.

Optimizador de RMSProp

- El optimizador RMSProp es un optimizador AdaGrad mejorado. Introduce un coeficiente de atenuación para garantizar una cierta relación de atenuación para r en cada ronda.
- El optimizador RMSProp resuelve el problema de que el optimizador AdaGrad finaliza el proceso de optimización demasiado pronto. Es adecuado para el manejo de objetivos no estables y tiene buenos efectos en el RNN.

$$g_t = \frac{\partial C(t,o)}{\partial w_t}$$ Cálculo del gradiente

$$r_t = \beta r_{t-1} + (1-\beta) g_t^2$$ Acumulación de gradiente cuadrado

$$\Delta w_t = -\frac{\eta}{\varepsilon + \sqrt{r_t}} g_t$$ Actualización de cómputo

$$w_{t+1} = w_t + \Delta w_t$$ Actualización de aplicación

- g_t indica el gradiente t-ésimo, r es una variable de acumulación de gradiente y el valor inicial de r es 0, **que puede no aumentar y debe ajustarse mediante un parámetro**. β "es el factor de atenuación", η indica el LR global, que debe configurarse manualmente. ε es una pequeña constante y se establece en aproximadamente 10^{-7} para estabilidad numérica.

Optimizador Adam (1)

- Estimación de Momento Adaptativo (Adam): desarrollado en base a AdaGrad y AdaDelta, Adam mantiene dos variables adicionales m_t y v_t para cada variable que se va a entrenar:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$$
$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$$

- Donde t representa la t-ésima iteración y g_t es el gradiente calculado. m_t y v_t son promedios móviles del gradiente y del gradiente cuadrado. Desde la perspectiva estadística, m_t y v_t son estimaciones del primer momento (el valor promedio) y el segundo momento (la varianza no centrada) de los gradientes respectivamente, lo que también explica por qué el método se llama así.

Optimizador Adam (1)

- Si m_t y v_t se inicializan usando el vector cero, m_t y v_t están cerca de 0 durante las iteraciones iniciales, especialmente cuando β_1 y β_2 están cerca de 1. Para resolver este problema, usamos $\hat{m}_t$ y $\hat{v}_t$:

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}$$

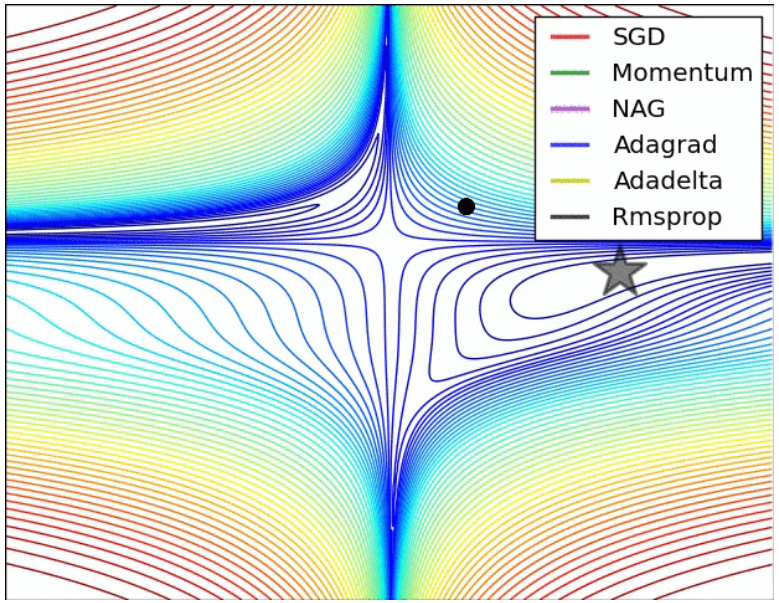
$$\hat{v}_t = \frac{v_t}{1-\beta_2^t}$$

- La regla de actualización de peso de Adam es la siguiente :

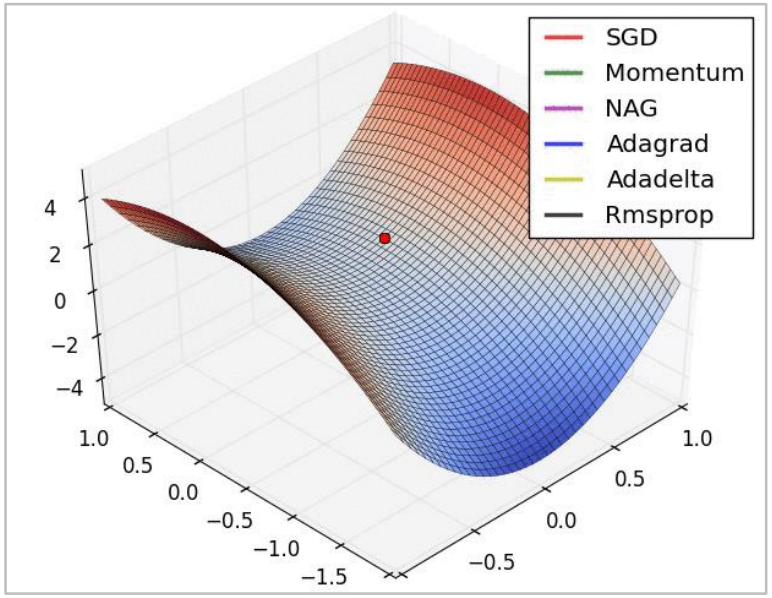
$$w_{t+1} = w_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon}\hat{m}_t$$

- Aunque la regla implica la configuración manual de η, β_1 y β_2, la configuración es mucho más simple. Según los experimentos, los ajustes predeterminados son β_1 = 0,9, β_2 = 0,999, ϵ = 10^{-8} y η = 0.001. En la práctica, Adam convergerá rápidamente. Cuando se alcanza la saturación de convergencia, se puede reducir xx. Después de varias veces de reducción, se obtendrá un extremo local satisfactorio. No es necesario ajustar otros parámetros.

Comparación de rendimiento del optimizador



Comparación de algoritmos de optimización en mapas de contorno de funciones de pérdida



Comparación de algoritmos de optimización en el punto de carga

CONTENIDO

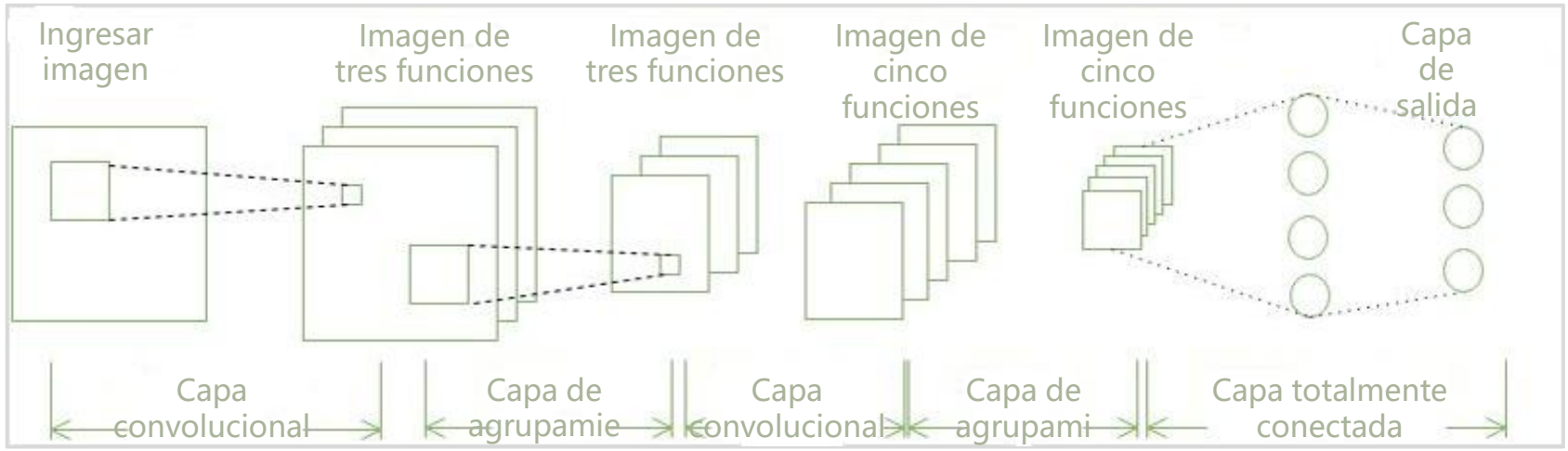
Red Neural Convolucional (Convolutional Neural Network CNN)

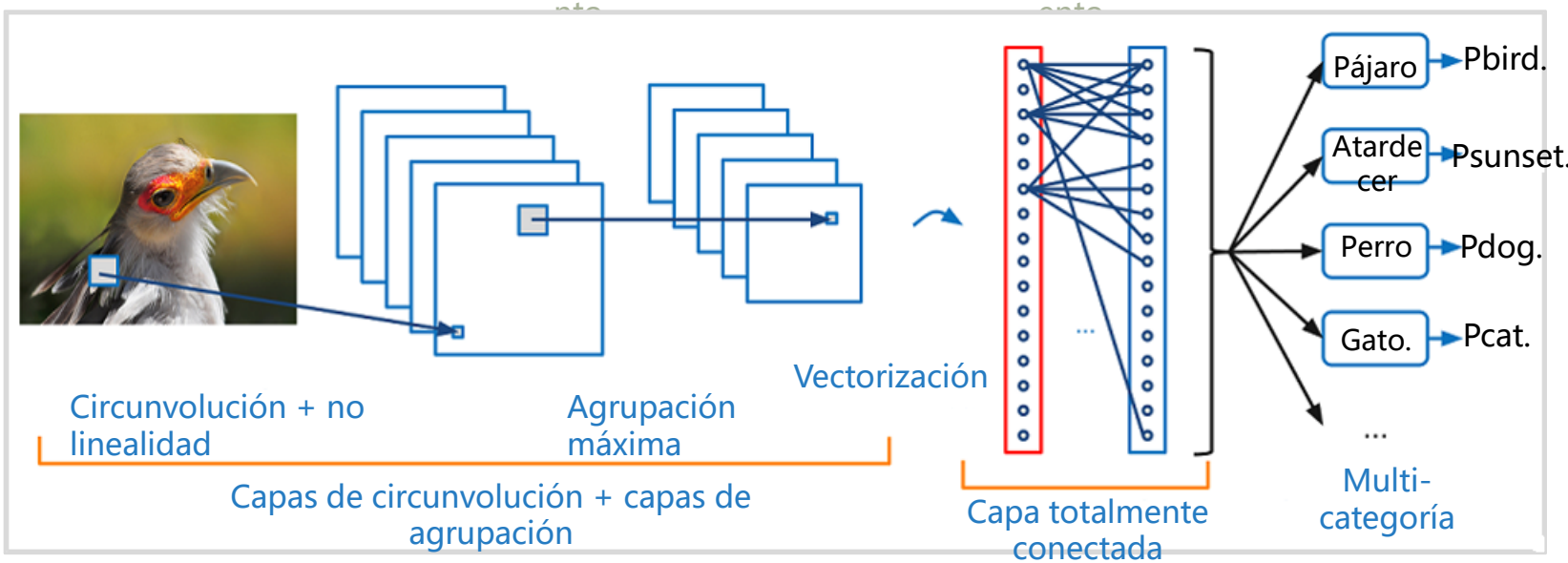
- Una red neuronal convolucional (CNN) es una red neural de prealimentación (feedforward). Sus neuronas artificiales pueden responder **a las unidades circundantes dentro del rango de cobertura**. La CNN destaca en el procesamiento de imágenes. Incluye **una capa convolucional, una capa de pooling y una capa totalmente conectada.**
- En los años 60, Hubel y Wiesel estudiaron las neuronas del cortex de los gatos utilizadas para la sensibilidad local y la selección de dirección, y descubrieron que su estructura de red única podía simplificar las redes neuronales de retroalimentación. Luego propusieron el CNN.
- En la actualidad, la CNN se ha convertido en uno de los puntos críticos de la investigación en muchos campos científicos, especialmente en el campo de la clasificación de patrones. La red es ampliamente utilizada porque puede evitar el pre-procesamiento previo complejo de imágenes y permite la entrada directa de imágenes originales.

Conceptos principales de la Red CNN

- **Campo receptivo local:** Se considera generalmente que la percepción humana del mundo exterior va de local a global. **Las correlaciones espaciales entre los píxeles locales de una imagen están más cerca que entre los píxeles distantes**. Por lo tanto, cada neurona no necesita conocer la imagen global. Sólo necesita conocer la imagen local. La información local se combina a un nivel superior para generar información mundial.
- **Compartir parámetros**: se pueden usar uno o más filtros/kernels para escanear las imágenes de entrada. Los parámetros transportados por los filtros son pesos. En una capa escaneada por filtros, cada filtro utiliza los mismos parámetros durante el cálculo ponderado. Compartir el peso significa que cuando cada filtro escanea una imagen entera, los parámetros del filtro son fijos.

Arquitectura de la red neuronal convolucional





Cálculo de un solo filtro (1)

- Descripción del cálculo de convolución

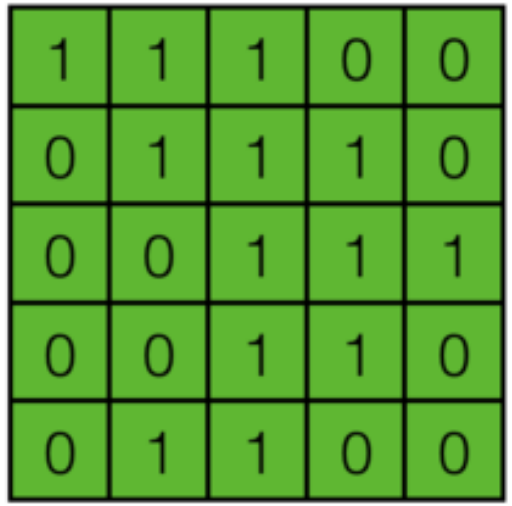
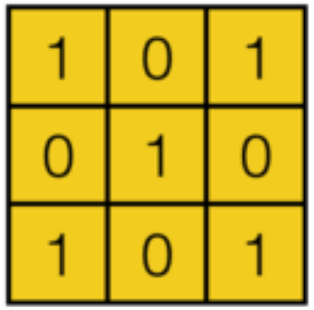
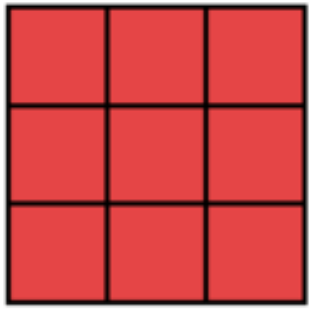


IMAGEN 5x5



SESGO 0

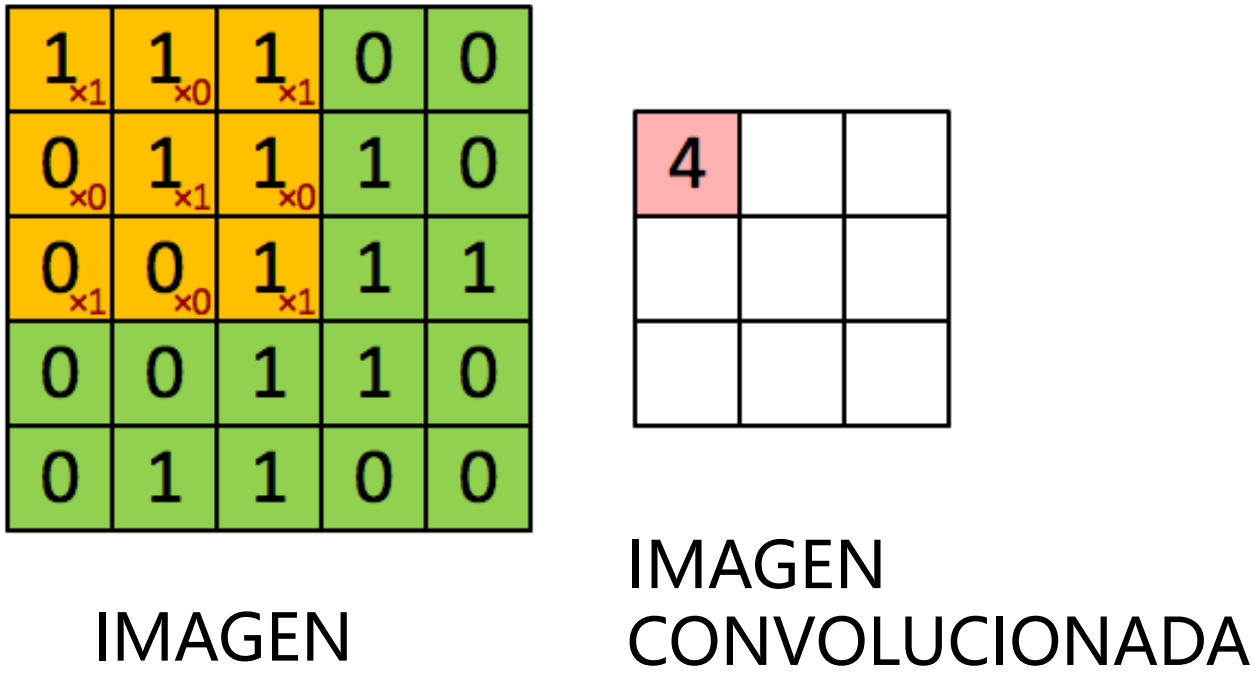
FILTRO 3x3



Mapa de características 3x3

Cálculo de un solo filtro (2)

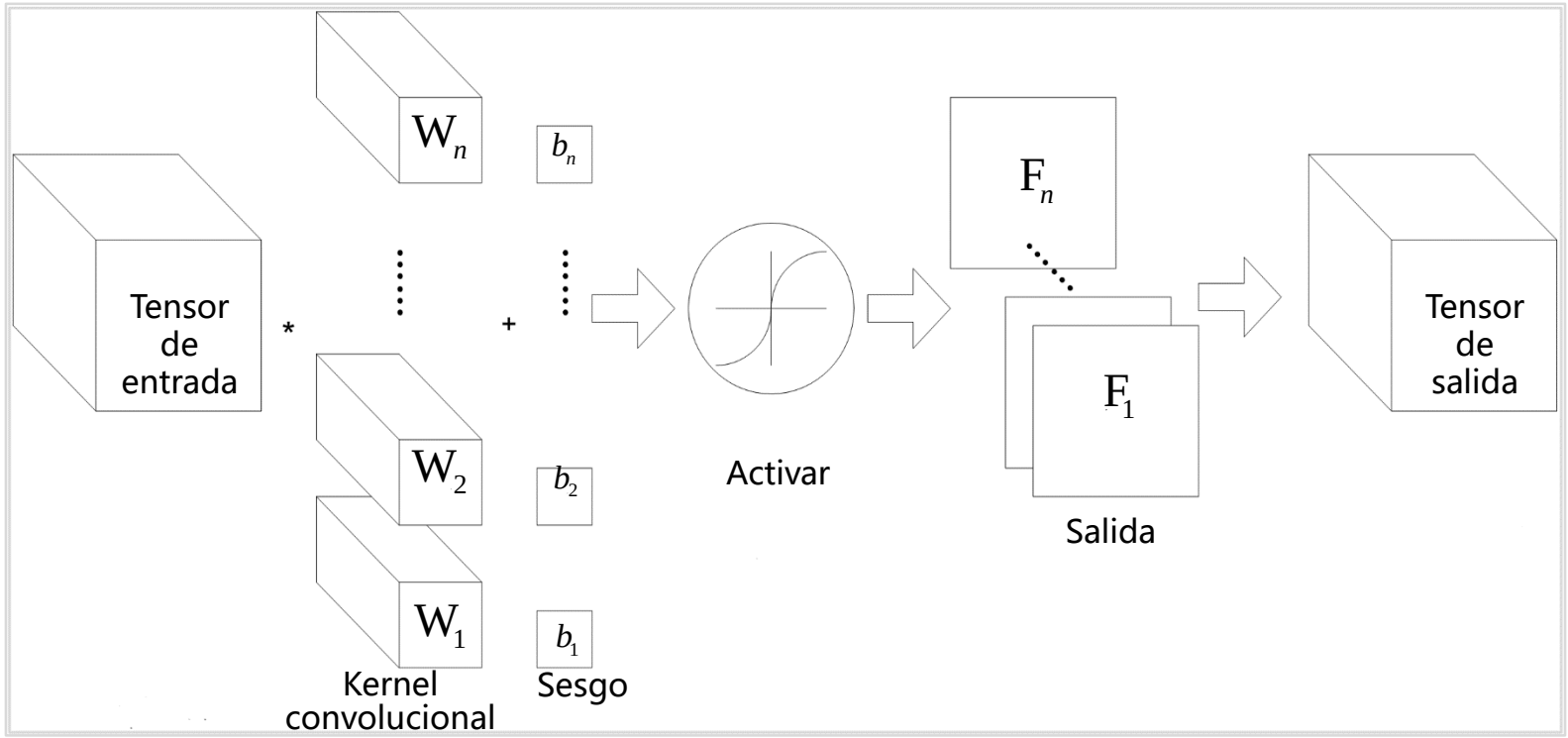
- Demostración del cálculo de la convolución



Han Bingtao, 2017, Red Neural Convolucional

Capa convolucional

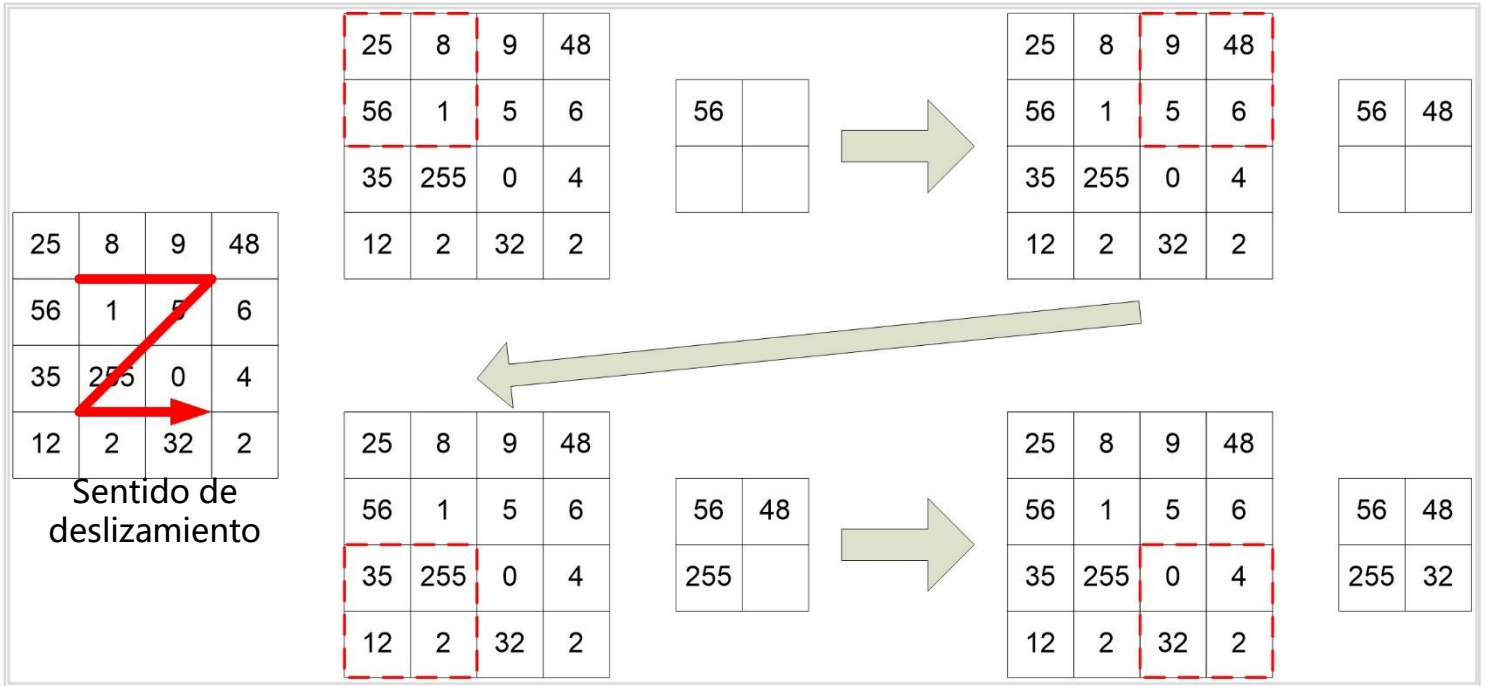
- La arquitectura básica de un CNN es la convolución multicanal que consiste en múltiples convoluciones individuales. La salida de la capa previa (o la imagen original de la primera capa) se utiliza como la entrada de la capa actual. Luego se convoluciona con el filtro en la capa y sirve como salida de esta capa. El núcleo de convolución de cada capa es el peso a aprender. Al igual que el FCN (Fully Convoluted Network), una vez finalizada la convolución, el resultado debe ser sesgado y activado a través de funciones de activación antes de ser introducido en la capa siguiente.



Capa de agrupamiento (Pooling layer)

- El agrupamiento (pooling) combina unidades cercanas para reducir el tamaño de la entrada en la capa siguiente, reduciendo las dimensiones. La agrupación común incluye la agrupación máxima (max pooling) y la agrupación promedio (average pooling). Cuando se utiliza la agrupación máxima, el valor máximo en un área cuadrada pequeña se selecciona como el representante de esta área, mientras que el valor medio se selecciona como el representativo cuando se utiliza la agrupación promedio. El lado de esta pequeña área es el tamaño de la ventana de grupo.

La siguiente figura muestra la operación de agrupación máxima cuyo tamaño de ventana de agrupación es 2.



Capa totalmente conectada

- La capa completamente conectada es esencialmente un clasificador. Las características extraídas en la capa convolucional y la capa de agrupación se enderezan y colocan en la capa completamente conectada para generar y clasificar los resultados.
- Generalmente, la función Softmax se utiliza como función de activación de la capa de salida final completamente conectada para combinar todas las características locales en características globales y calcular la puntuación de cada tipo.

$$\sigma(\mathrm{z})_j = \frac{e^{z_j}}{\sum_k e^{z_k}}$$

Red Neural Recurrente – Recurrent Neural Networl RNN

- La red neural recurrente es una red neural que captura información dinámica en datos secuenciales a través de conexiones periódicas de nodos de capa ocultos. Puede clasificar datos secuenciales.
- A diferencia de otras redes neuronales avanzadas, la RNN puede mantener un estado de contexto e incluso almacenar, aprender y expresar información relacionada en ventanas de contexto de cualquier longitud. A diferencia de las redes neuronales tradicionales, no se limita al límite del espacio, sino que también soporta secuencias de tiempo. En otras palabras, hay un lado entre la capa oculta del momento actual y la capa oculta del momento siguiente.
- La RNN es ampliamente utilizada en escenarios relacionados con secuencias, tales como videos que consisten en marcos de imágenes, audio que consiste en clips, y oraciones que consisten en palabras.

Arquitectura de Red Neural Recurrente (1)

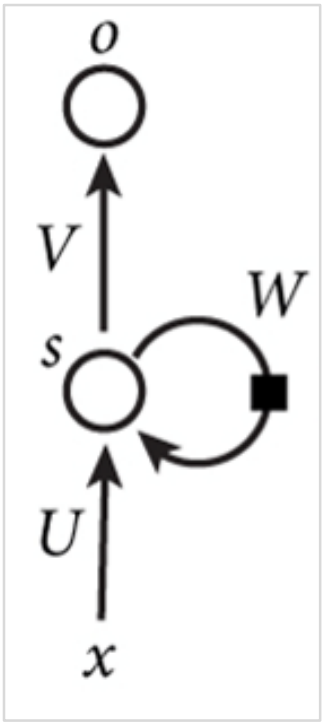
- X_t es la entrada de la secuencia de entrada en el tiempo t.
- S_t es la unidad de memoria de la secuencia en el tiempo t y almacena en caché la información anterior.

$$S_t = tanh(UX_t + WS_{t-1}).$$

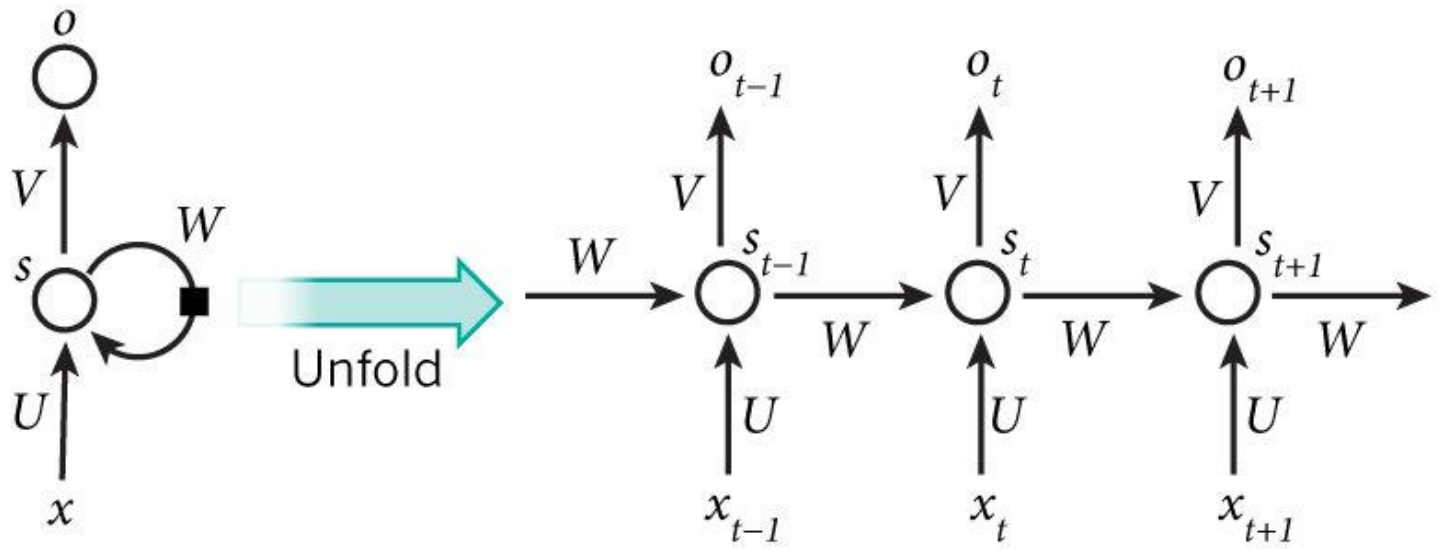
- O_t es la salida de la capa oculta de la secuencia en el tiempo t.

$$O_t = tanh(VS_t)$$

- O_t después de pasar por múltiples capas ocultas, puede obtener el resultado final de la secuencia en el tiempo t.

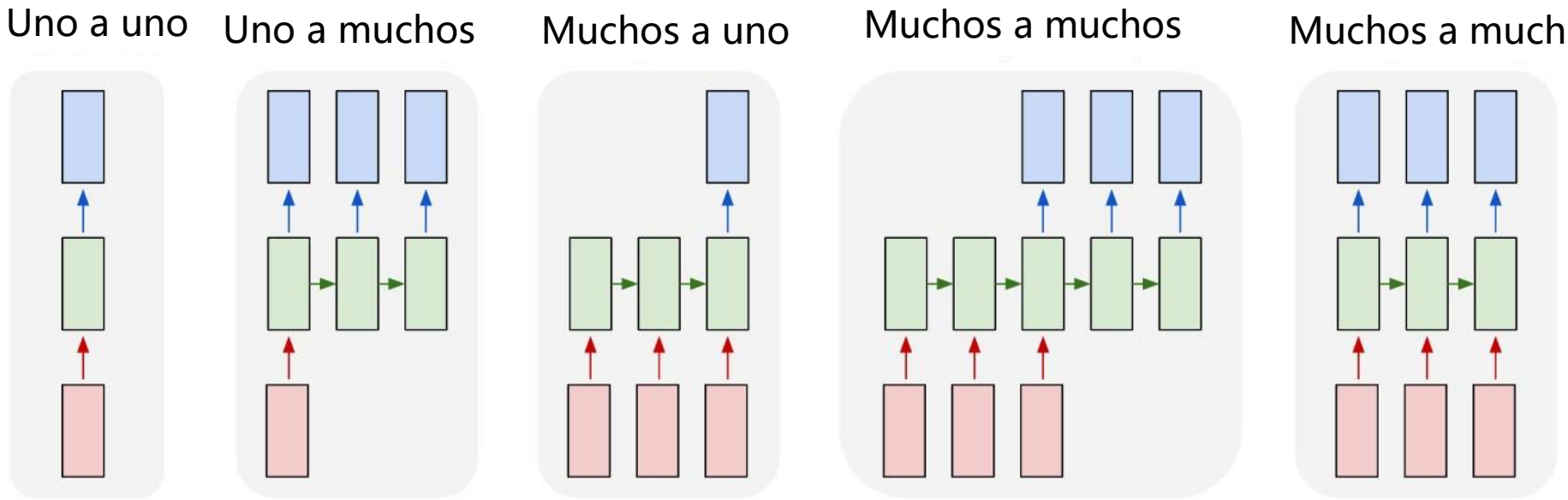


Arquitectura de red neural recurrente (2)



Lecond, Bengio, y G. Hinton, 2015, A Recurrent Neural Network y el despliegue en el tiempo del cómputo involucrado en su cálculo a futuro

Tipos de redes neuronales recurrentes



Andrej Karpathy, 2015, La eficacia irrazonable de las redes neuronales recurrentes

Retro propagación a través del tiempo - Backpropagation Through Time (BPTT)

- BPTT:
  - La retropropagación (propagación hacia atrás) tradicional es la extensión de la secuencia de tiempo.
  - Hay dos fuentes de errores en la secuencia en el momento de la unidad de memoria: la primera es del error de salida de la capa oculta en t secuencia de tiempo; el segundo es el error de la celda de memoria en la siguiente secuencia de tiempo t + 1.
  - Cuanto más larga sea la secuencia de tiempo, más probable es que la pérdida de la última secuencia de tiempo al gradiente de w en la primera secuencia de tiempo provoque el problema del gradiente de desaparición o explosión.
  - El gradiente total de peso w es la acumulación del gradiente del peso en toda la secuencia de tiempo..
- Tres pasos de BPTT:
  - Calcular el valor de salida de cada neurona mediante propagación directa.
  - Calcular el valor de error de cada neurona mediante retropropagación δ_j.
  - Calcular el gradiente de cada peso.
- Actualización de pesos mediante el algoritmo SGD.

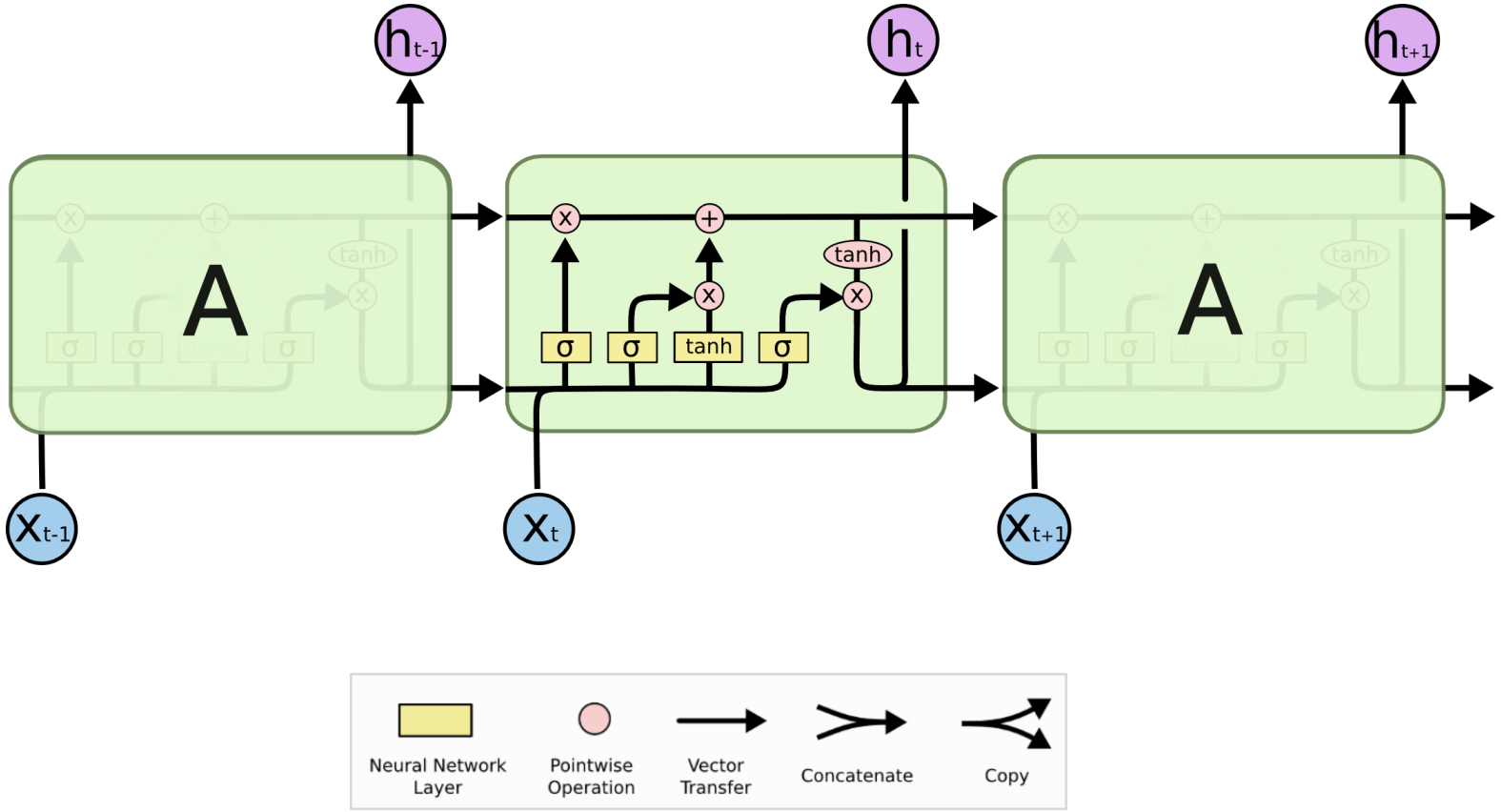
Problema de red neuronal recurrente

- $S_t = \sigma(UX_t + WS_{t-1})$ se exitende en la secuencia de tiempo.

- $S_t = \sigma\left(UX_t + W\left(\sigma\left(UX_{t-1} + W\left(\sigma\left(UX_{t-2} + W(\ldots)\right)\right)\right)\right)\right)$

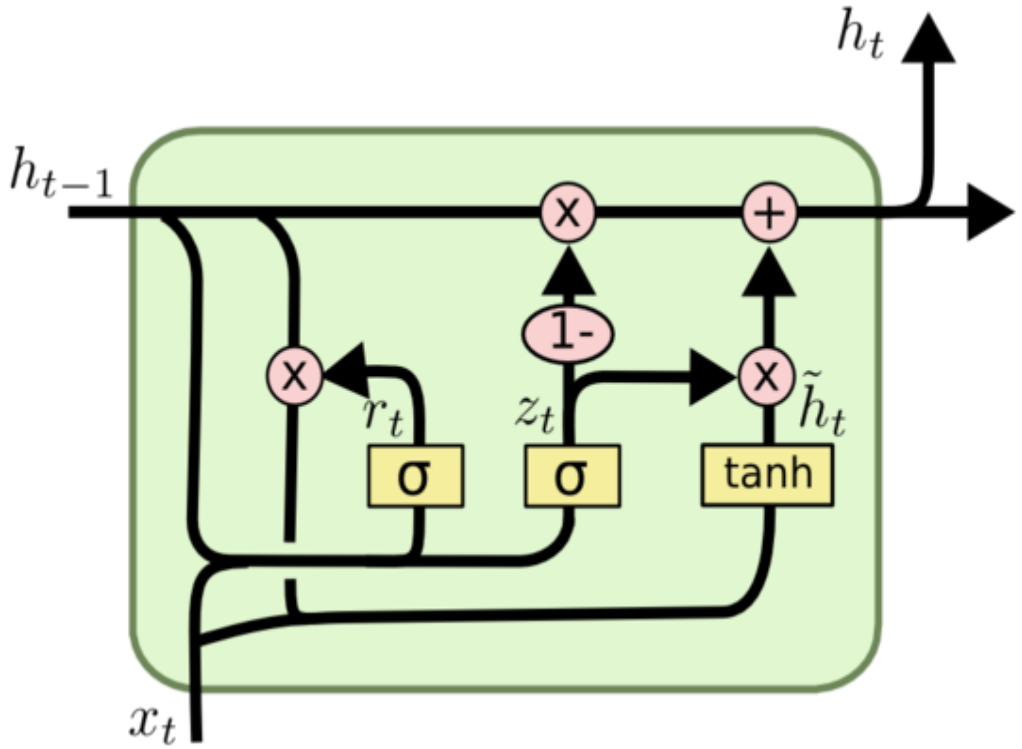
- A pesar de que la estructura estándar de RNN resuelve el problema de la memoria de información, la información se atenúa durante la memoria a largo plazo..
- La información debe guardarse durante mucho tiempo en muchas tareas. Por ejemplo, una pista al comienzo de una ficción especulativa puede no ser respondida hasta el final..
- Es posible que el RNN no pueda guardar información durante mucho tiempo debido a la capacidad limitada de la unidad de memoria.
- Esperamos que las unidades de memoria puedan recordar información clave-

Red de memoria a corto plazo (Long short-term Memory Network LSTM)



Colah, 2015, Descripción de las redes de LSTM

Unidad Puerteada Recurrente
Gated Recurrent Unit (GRU)



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$

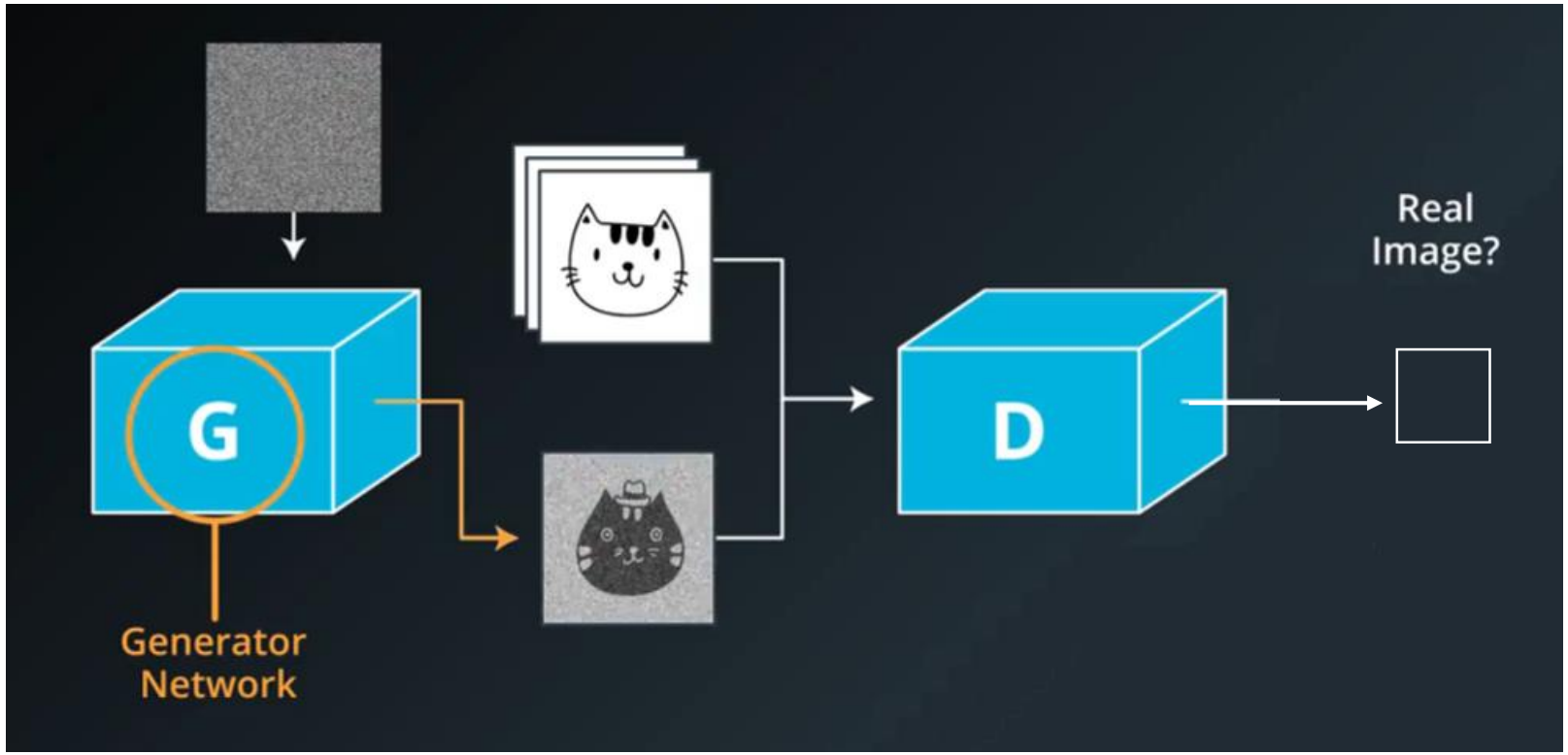
$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Red Generativa Adversarial – Generative Adversarial Network (GAN)

- Generative Adversarial Network es un marco que forma al generador G y al discriminador D a través del proceso adversarial. A través del proceso adversarial, el discriminador puede decir si la muestra del generador es falsa o real. La GAN adopta un algoritmo BP maduro.
- (1) Generador G: La entrada es ruido z, que cumple con la distribución de probabilidad previamente seleccionada manualmente, como distribución par y distribución gaussiana. El generador adopta la estructura de red del perceptrón multicapa (MLP), utiliza los parámetros de estimación de máxima verosimilitud (MLE) para representar el mapeo derivable G(z), y asigna el espacio de entrada al espacio de muestra.
- (2) Discriminador D: La entrada es la muestra real x y la muestra falsa G(z), que se etiquetan como real y falso respectivamente. La red del discriminador puede utilizar los parámetros de transporte de MLP. La salida es la probabilidad D(G(z)) que determina si la muestra es real o falsa.
- La GAN se puede aplicar a escenarios como la generación de imágenes, la generación de texto, la mejora de voz, la superresolución de imágenes.

Arquitectura de GAN

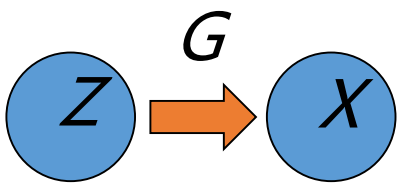
- Generador/Discriminador



Modelo Generador y Modelo Discriminador

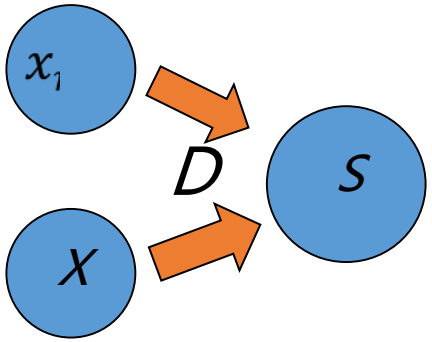
- Red generadora
  - Genera datos de muestra
    - Entrada: vector de ruido blanco gaussiano z
    - Salida: vector de datos de muestra x

$$x = G(z; \theta^G)$$



- Red discriminadora
  - Determina si los datos de muestra son reales
    - Input: datos de muestra real x_{real} y datos de muestra generada $x = G(z)$
    - Output: probabilidad que determina si la muestra es real o no

$$y = D(x; \theta^D)$$



Normas de entrenamiento de Red Generativa Adversarial GAN

- Objetivo de optimización:
  - función de valor

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}[logD(x)] + E_{z\sim p_{z(z)}}[log(1-D(G(z)))]$$

  - En la fase inicial de formación, cuando el resultado de G es muy pobre, D determina que la muestra generada es falsa con alta confianza, porque la muestra es obviamente diferente de los datos de entrenamiento. En este caso, log(1-D (G (z))) está saturado (donde el gradiente es 0 y no se puede realizar la iteración). Por lo tanto, elegimos entrenar a G solamente minimizando [-log(D(G(z))}].

CONTENIDO

1. Resumen de aprendizaje profundo
2. Reglas de entrenamiento
3. Función de activación
4. Normalizador
5. Optimizador
6. Tipos de redes neuronales
7. **Problemas comunes**

Desequilibrio de datos (1)

- Descripción del problema: En el conjunto de datos que consta de varias categorías de tareas, el número de muestras varía mucho de una categoría a otra. Una o más categorías de las categorías predichas contienen muy pocas muestras.
- Por ejemplo, en un experimento de reconocimiento de imágenes, más de 2.000 categorías de un total de 4251 imágenes de entrenamiento contienen sólo una imagen cada una. Algunos de los otros tienen 2-5 imágenes.
- Impactos:
  - Debido al número desequilibrado de muestras, no podemos obtener el resultado óptimo en tiempo real porque el modelo/algoritmo nunca examina adecuadamente las categorías con muy pocas muestras.
  - Dado que pocos objetos de observación pueden no ser representativos de una clase, es posible que no obtengamos muestras adecuadas para la verificación y el ensayo.

Desequilibrio de datos (2)

Submuestreo aleatorio

- Eliminación de muestras redundantes en una categoría

Sobremuestreo aleatorio

- Copiar muestras

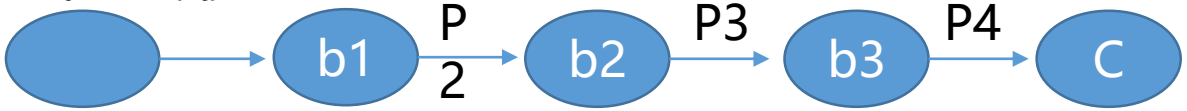
Técnica de sobremuestreo de la minoría sintética

- Muestreo
- Combinación de muestras

Problema de desvanecimiento de gradiente y de explosión de gradiente (1)

- Desvanecimiento de gradiente: A medida que las capas de red aumentan, el valor derivado de la retropropagación disminuye, lo que causa un problema de desaparición de gradiente.
- Explosión de gradiente: A medida que aumentan las capas de red, aumenta el valor derivado de la retropropagación, lo que causa un problema de explosion de gradiente.
- Causa:

$y_i = \sigma_{(z_i)} = \sigma(w_i x_i + b_i)$ Where σ is sigmoid function.

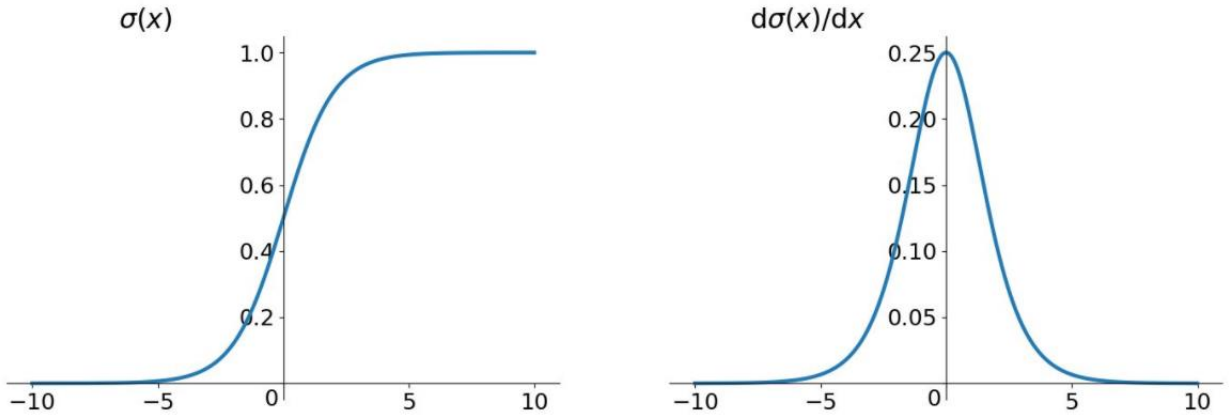


- La retropropagación se puede deducir de la siguiente manera:

$$\frac{\partial C}{\partial b_1} = \frac{\partial C}{\partial y_4}\frac{\partial y_4}{\partial z_4}\frac{\partial z_4}{\partial x_4}\frac{\partial x_4}{\partial z_3}\frac{\partial z_3}{\partial x_3}\frac{\partial x_3}{\partial z_2}\frac{\partial z_2}{\partial x_2}\frac{\partial x_2}{\partial z_1}\frac{\partial z_1}{\partial b_1}$$
$$= \frac{\partial C}{\partial y_4}\sigma'(z_4)w_4\sigma'(z_3)w_3\sigma'(z_2)w_2\sigma'(z_1)x$$

Problema de desvanecimiento de gradiente y de explosión de gradiente (1)

- El maximo valor de $\sigma'(x)$ is $\frac{1}{4}$:



- Sin embargo, el peso de la red | w | suele ser menor que 1. Por lo tanto, | $\sigma'(z)w$ | ≤1 / 4. Según la regla de la cadena, a medida que aumentan las capas, el resultado de la derivación $\partial C / (\partial b_1)$ disminuye, lo que da como resultado el problema del gradiente desvaneciente.
- Cuando el peso de la red | w | es grande, resultando en $|\sigma'(z)w|> 1$, se produce el problema del gradiente explosivo.
- Solución: Por ejemplo, el recorte de gradiente se usa para aliviar el problema del gradiente explosivo, la función de activación de ReLU y LSTM se usan para aliviar el problema del gradiente desvaneciente.

Sobreajuste

- Descripción del problema: El modelo funciona bien en el conjunto de entrenamiento, pero mal en el conjunto de pruebas.
- Causa raíz: Hay demasiadas dimensiones de características, supuestos de modelos y parámetros, demasiado ruido, pero muy pocos datos de entrenamiento. Como resultado, la función de ajuste predice perfectamente el conjunto de entrenamiento, mientras que el resultado de predicción del conjunto de pruebas de nuevos datos es pobre. Los datos de entrenamiento están sobreajustados sin considerar las capacidades de generalización.
- Solución: por ejemplo, aumento de datos, regularización, detención temprana y dropout (abandono)

RESUMEN

- Este capítulo describe la definición y el desarrollo de redes neuronales, perceptrones y sus reglas de entrenamiento, tipos comunes de redes neuronales (CNN, RNN y GAN), y los problemas comunes de redes neuronales en la ingeniería y soluciones de IA.

Quiz

1. (Verdadero o falso) Comparado con la red neural recurrente, la red neuronal convolucional es más adecuada para el reconocimiento de imágenes. ( )

   A. Cierto.

   B. Falso

2. (Verdadero o falso) La NGA es un modelo de aprendizaje profundo, que es uno de los métodos más prometedores para el aprendizaje no supervisado de la distribución compleja en los últimos años. ( )

   A. Cierto.

   B. Falso

Quiz

3. Hay muchos tipos de redes neuronales de aprendizaje profundo. ¿Cuál de las siguientes no es una red neural de aprendizaje profundo? (1)

   A. CNN

   B. RRN

   C. LSTM

   D. Logística

4. (Multi-choice) Hay muchos "componentes" importantes en la arquitectura de red neuronal convolucional. ¿Cuáles de los siguientes son los "componentes" de la red neural convolucional? (1)

   A. Función de activación

   B. Kernel convolucional

   C. Agrupamiento (pooling)

   D. Capa totalmente conectada

Recomendaciones

- Sitio web de aprendizaje en línea
  - https://e.huawei.com/en/talent/#/home
- Base de conocimientos de Huawei
  - https://support.huawei.com/enterprise/servicecenter?lang=zh

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

HUAWEI

Marcos de desarrollo más usados en la industria

HUAWEI

PREFACIO

- Este capítulo describe:
  - Definición del marco de aprendizaje profundo y sus ventajas, y dos marcos de aprendizaje profundo PyTorch y TensorFlow
  - Operaciones básicas y módulos comunes de TensorFlow 2.x (centrándose en código)
  - Experiencia de reconocimiento de dígitos manuscrita del MNIST realizada sobre la base de TensorFlow para comprender profundamente y familiarizarse con un proceso de modelado de aprendizaje profundo

Objetivos

Al finalizar este curso, podrás:

- Describir un marco de aprendizaje profundo (deep learning framework).
- Conocer los marcos generales de aprendizaje profundo.
- Conozca las características de PyTorch.
- Conozca las características de TensorFlow.
- Diferenciar entre TensorFlow 1.x y 2.x.
- Dominar la sintaxis básica y los módulos comunes de TensorFlow 2.x.
- Dominar el proceso de un experimento de reconocimiento de dígitos escrito a mano MNIST.

CONTENIDO

1. **Marcos generales de desarrollo**
   - Marco de aprendizaje profundo
     - PyTorch (PyTorch)
     - Tensorflow
2. Básicos de TensorFlow 2.x
3. Módulos comunes de TensorFlow 2.x
4. Los pasos básicos del desarrollo del aprendizaje profundo

Marco de aprendizaje profundo – Deep learning framework

- Un marco de aprendizaje profundo es una interfaz, una biblioteca o una herramienta que nos permite construir modelos de aprendizaje profundo más fácilmente y rápidamente, sin entrar en los detalles de los algoritmos subyacentes. Un marco de aprendizaje profundo puede considerarse un conjunto de elementos constitutivos. Cada componente en los bloques de construcción es un modelo o algoritmo. Por lo tanto, los desarrolladores pueden utilizar componentes para ensamblar modelos que cumplan con los requisitos, y no necesitan empezar desde cero.
- La aparición de marcos de aprendizaje profundo reduce los requisitos para los desarrolladores. Los desarrolladores ya no necesitan compilar código a partir de complejas redes neuronales y algoritmos de retropropagacióin. En su lugar, pueden utilizar modelos existentes para configurar los parámetros según se requiera, donde los parámetros del modelo se entrenan automáticamente. Además, pueden agregar capas de red autodefinidas a los modelos existentes, o seleccionar clasificadores y algoritmos de optimización necesarios directamente invocando código existente.

CONTENIDOS

1. **Marcos generales de desarrollo**
   - Marco de aprendizaje profundo
   - PyTorch
   - Tensorflow
2. Básicos de TensorFlow 2.x
3. Módulos comunes de TensorFlow 2.x
4. Los pasos básicos del desarrollo del aprendizaje profundo

PyTorch

- PyTorch es un framework de aprendizaje automático basado en Python desarrollado por Facebook. Es desarrollado sobre la base de Torch, un marco de computación científica apoyado por un gran número de algoritmos de Machine Learning. Torch es una biblioteca de operaciones de tensor similar a NumPy, caracterizada por una gran flexibilidad, pero es menos popular porque utiliza el lenguaje de programación Lua. Por eso PyTorch es desarrollado.
- Además de Facebook, institutos como Twitter, GMU y Salesforce también utilizan PyTorch.

Fuente de imagen: http://PyTorch123.com/FirstSection/PyTorchIntro/

Características de PyTorch

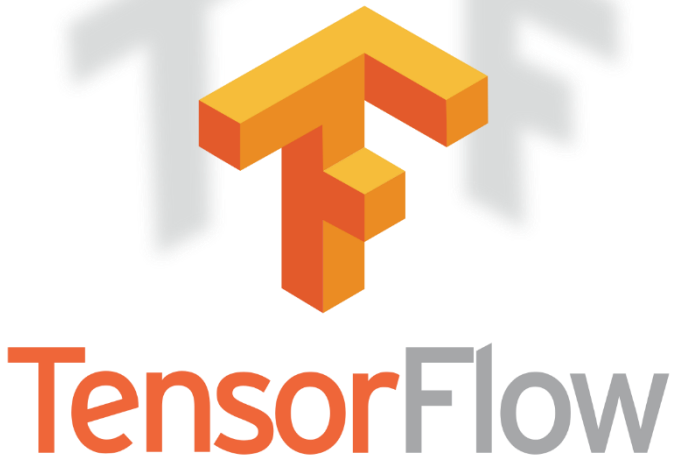
- **Python en primer lugar:** PyTorch no se limita a unir Python a un marco C++. PyTorch soporta directamente el acceso de Python a un nivel fino. Los desarrolladores pueden usar PyTorch tan fácilmente como el uso de NumPy o SciPy. Esto no sólo reduce el umbral para entender Python, sino que también asegura que el código es básicamente consistente con la implementación nativa de Python.
- **Red neuronal dinámica:** Muchos frameworks mainstream como TensorFlow 1.x no soportan esta función. Para ejecutar TensorFlow 1.x, los desarrolladores deben crear gráficos computacionales estáticos con antelación y ejecutar los comandos **feed** y **run** para ejecutar repetidamente los gráficos creados. En contraste, PyTorch con esta característica está libre de tal complejidad, y los programas de PyTorch pueden construir/ajustar dinámicamente gráficos computacionales durante la ejecución.
- **Fácil de depurar:** PyTorch puede generar gráficos dinámicos durante la ejecución. Los desarrolladores pueden detener un intérprete en un depurador y ver la salida de un nodo específico.
- PyTorch provee tensores que soportan CPUs y GPUs, que aceleran considerablemente la computación.

CONTENIDO

1. **Marcos generales de desarrollo**
   - Marco de aprendizaje profundo
   - PyTorch
   - Tensorflow
2. Básicos de TensorFlow 2.x
3. Módulos comunes de TensorFlow 2.x
4. Los pasos básicos del desarrollo del aprendizaje profundo

Flujo tensor

- TensorFlow software de código abierto de Google, de segunda generación, para la computación digital. El marco de computación TensorFlow soporta varios algoritmos de aprendizaje profundo y múltiples plataformas de computación, garantizando una alta estabilidad del sistema.



Fuente de imagen: https://www.TensorFlow.org/

Características de TensorFlow

Escalabilidad

Multi-lenguaje

GPU

Multiplataforma

Computabilidad potente

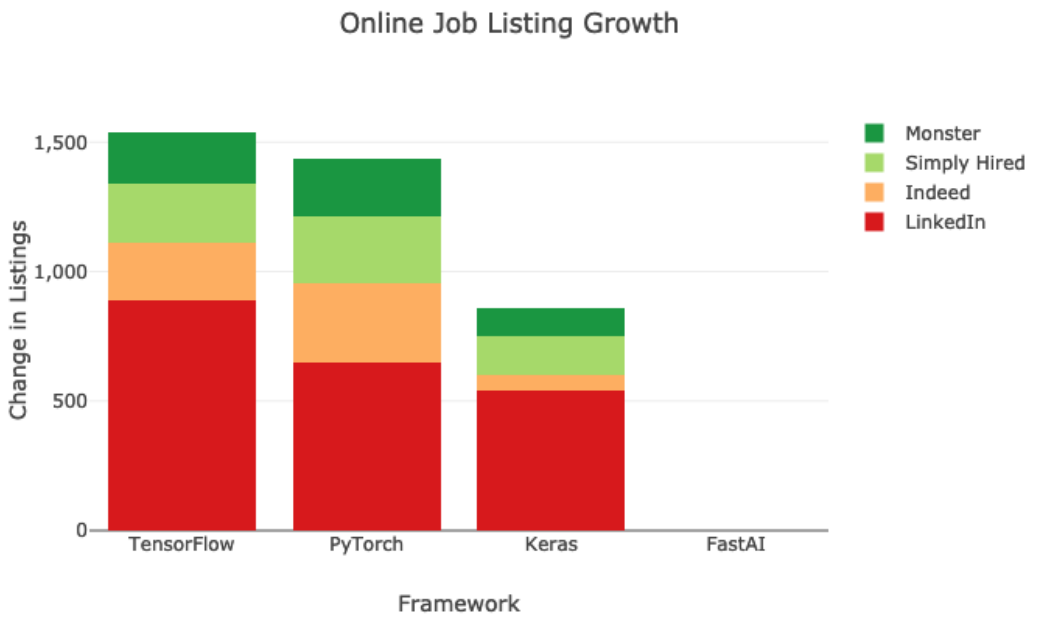
Distribuido

TensorFlow - Distribuido

- TensorFlow puede ejecutarse en diferentes ordenadores:
  - Desde smartphones a clústeres de computadoras, para generar los modelos de entrenamiento deseados.
- Actualmente, los marcos nativos de aprendizaje profundo distribuido soportados incluyen sólo TensorFlow, CNTK, Deeplearning4J y MXNet.
- Cuando se utiliza una GPU única, la mayoría de los marcos de aprendizaje profundo dependen de cuDNN, y por lo tanto soportan casi la misma velocidad de entrenamiento, siempre que las capacidades de computación de hardware o las memorias asignadas difieran ligeramente. Sin embargo, para el aprendizaje profundo a gran escala, los datos masivos hacen difícil para la GPU única completar la formación en un tiempo limitado. A fin de manejar estos casos, TensorFlow permite la capacitación distribuida.

¿Por qué TensorFlow?

- TensorFlow es considerada una de las mejores bibliotecas para redes neuronales, y puede reducir la dificultad en el desarrollo del aprendizaje profundo. Además, dado que es de fuente abierta (open source), puede mantenerse y actualizarse convenientemente, por lo que se puede mejorar la eficiencia del desarrollo.
- Keras, tercer lugar en el número de estrellas en GitHub, está envasado en una avanzada API de TensorFlow 2.0, lo que hace que TensorFlow 2.x sea más flexible y más fácil de depurar.



Demanda en el mercado de contratación

TensorFlow 2.x vs. TensorFlow 1.x

- Desventajas de TensorFlow 1.0:
  - Una vez creado un tensor en TensorFlow 1.0, el resultado no puede ser devuelto directamente. Para obtener el resultado, se debe crear el mecanismo de sesión, que incluye el concepto de gráfico, y el código no puede ejecutarse sin session.run. Este estilo es más como el lenguaje de programación de hardware VHDL.
  - Comparado con algunos marcos sencillos como PyTorch, TensorFlow 1.0 añade los conceptos de sesión y gráfico, que son inconvenientes para los usuarios.
  - Es complicado depurar TensorFlow 1.0, y sus APIs están desordenadas, lo que hace difícil para los principiantes. Los alumnos encontrarán muchas dificultades para utilizar TensorFlow 1.0 incluso después de adquirir los conocimientos básicos. Como resultado, muchos investigadores han recurrido a PyTorch.

TensorFlow 2.x vs. TensorFlow 1.x

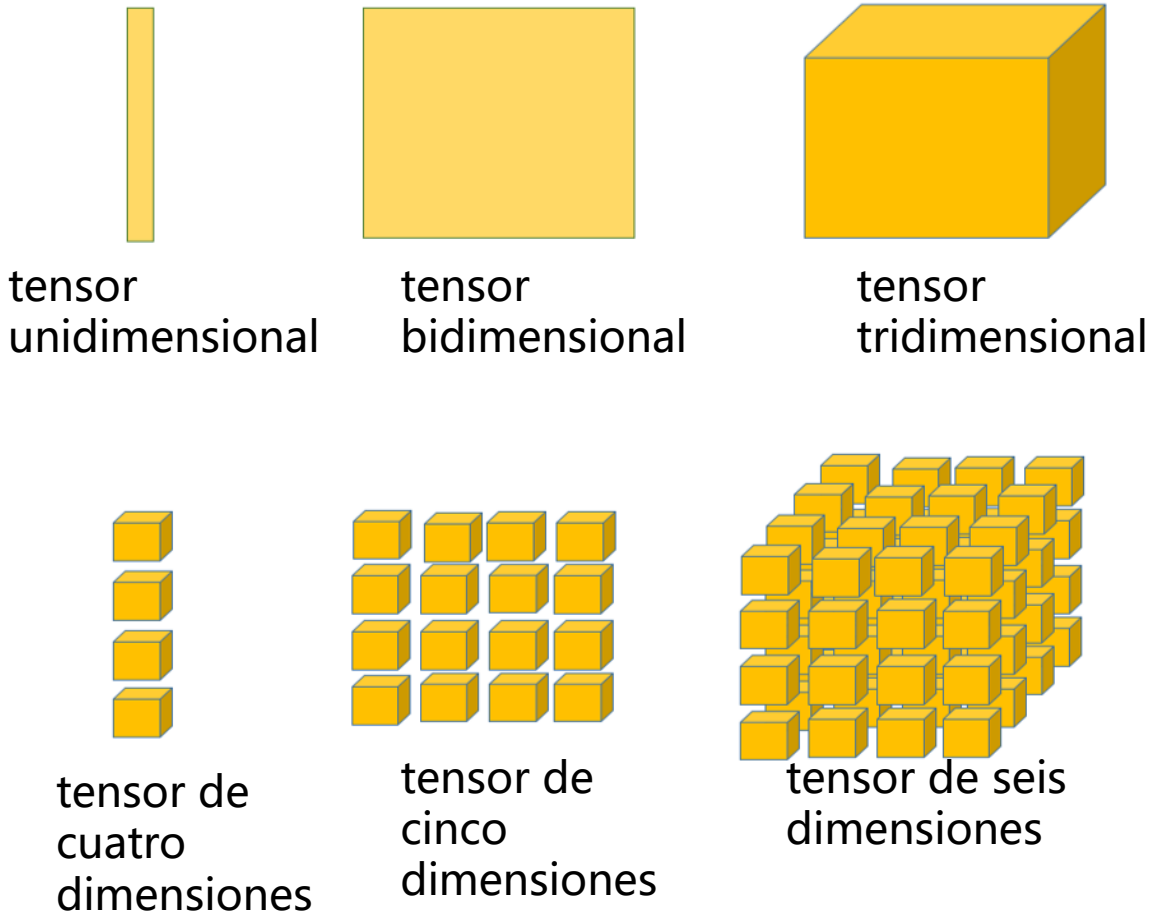
- Características de TensorFlow 2.x:
  - API avanzada Keras:
    - Fácil de usar: se eliminan los mecanismos de gráfico y sesión. Lo que ves es lo que obtienes, como Python y PyTorch.
  - Principales mejoras:
    - La función central de TensorFlow 2.x es el mecanismo de gráficos dinámicos llamado ejecución ansiosa. Permite a los usuarios compilar y depurar modelos como los programas normales, lo que hace que TensorFlow sea más fácil de aprender y usar.
    - Se soportan múltiples plataformas e idiomas y se puede mejorar la compatibilidad entre componentes mediante la estandarización de formatos de intercambio y la alineación de APIs.
    - Las API obsoletas se eliminan y las API duplicadas se reducen para evitar la confusión.
    - Compatibilidad y continuidad: TensorFlow 2.x proporciona un módulo que permite la compatibilidad con TensorFlow 1.x.
    - Se ha eliminado el módulo tf.contrib. Los módulos mantenidos se trasladan a repositorios separados. Se eliminan los módulos no utilizados y no mantenidos.

CONTENIDO

1. Marcos generales de desarrollo
2. **Básicos de TensorFlow 2.x**
3. Módulos comunes de TensorFlow 2.x
4. Los pasos básicos del desarrollo del aprendizaje profundo

Tensores

- Los tensores son las estructuras de datos más básicas en TensorFlow. Todos los datos están encapsulados en tensores.
- Tensor: una matriz multidimensional
  - Un escalar es un tensor de rango 0. Un vector es un tensor de rango 1. Una matriz es un tensor de rango 2.
- En TensorFlow, los tensores se clasifican en:
  - Tensores constantes
  - Tensores variables



Operaciones básicas de TensorFlow 2.x

- A continuación se describen las API comunes de TensorFlow centrándose en el código. El contenido principal es el siguiente:
  - Métodos para crear constantes y variables
  - Corte e indexación de tensores
  - Cambios en las dimensiones de los tensores
  - Operaciones aritméticas en tensores
  - Concatenación y división de tensores
  - Clasificación de tensores

Modo de ejecución temprana de Tensorflow 2.x

- Gráfico estático: TensorFlow 1.x, utilizando gráficos estáticos (modo gráfico), separa la definición y ejecución del cálculo mediante gráficos computacionales. Este es un modelo de programación declarativa. En el modo gráfico, los desarrolladores necesitan construir un gráfico computacional, iniciar una sesión y luego ingresar datos para obtener un resultado de ejecución.
- Los gráficos estáticos son ventajosos en la capacitación distribuida, la optimización del rendimiento y la implementación, pero inconvenientes para la depuración. La ejecución de un gráfico estático es similar a invocar un programa de lenguaje C compilado, y no se puede realizar la depuración interna en este caso. Por lo tanto, emerge una ejecución temprana (eager execution) basada en gráficos informáticos dinámicos.
- La ejecución temprana (Eager execution) es un método de programación basado en comandos, que es el mismo que Python nativo. Un resultado se devuelve inmediatamente después de una operación.

AutoGraph

- La ejecución temprana está habilitada de forma predeterminada en TensorFlow 2.x. La ejecución temprana es intuitiva y flexible para los usuarios (más fácil y rápida para ejecutar una operación de una sola vez), pero puede comprometer el rendimiento y la capacidad de implementación.
- Para lograr un rendimiento óptimo y hacer un modelo desplegable en cualquier lugar, puedes ejecutar @tf.function para añadir un decorador para construir un gráfico a partir de un programa, haciendo que el código Python sea más eficiente.
- tf.function puede construir una operación de TensorFlow en la función en un gráfico. De esta manera, esta función se puede ejecutar en modo gráfico. Esta práctica puede considerarse como encapsular la función como una operación de TensorFlow de un gráfico.

CONTENIDO

1. Marcos generales de desarrollo
2. Básicos de Tensorflow 2.x
3. **Módulos comunes de Tensorflow 2.x**
4. Pasos básicos para el desarrollo del aprendizaje profundo

Módulos comunes de TensorFlow 2.x (1)

- tf: Las funciones del módulo tf se utilizan para realizar operaciones aritméticas comunes, como tf.abs (cálculo de un valor absoluto), tf.add (adición de elementos uno por uno) y tf.concat (concatenación de tensores). La mayoría de las operaciones en este módulo pueden ser realizadas por Numpy.
- tf.errors: modulo de Tensorflow de tipo error.
- tf.data: implementa operaciones en conjuntos de datos.
  - Las tuberías de entrada creadas por tf.data se utilizan para leer datos de entrenamiento. Además, los datos pueden ser fácilmente introducidos desde las memorias (como Numpy).
- tf.distributions: implementa varias distribuciones estadísticas.
  - Las funciones de este módulo se utilizan para implementar diversas distribuciones estadísticas, como la distribución Bernoulli, la distribución uniforme y la distribución gaussiana.

Módulos comunes de flujo tensor 2.x (2)

- tf.io.gfile: implementa operaciones en archivos.
  - Las funciones de este módulo se pueden utilizar para realizar operaciones de archivo I/O, copiar archivos y renombrar archivos.
- tf.image: implementa operaciones en imágenes.
  - Las funciones de este módulo incluyen funciones de procesamiento de imágenes. Este módulo es similar a OpenCV, y proporciona funciones relacionadas con la luminancia de la imagen, la saturación, la inversión de fase, el recorte, el cambio de tamaño, la conversión de formato de imagen (de RGB a HSV, YUV, YIQ o gris), rotación, y detección de bordes de sobel. Este módulo es equivalente a un pequeño paquete de procesamiento de imágenes de OpenCV.
- tf.keras: una API de Python para invocar las herramientas de Keras.
  - Se trata de un módulo grande que permite varias operaciones de red.

Interfaz Keras

- TensorFlow 2.x recomienda Keras para la construcción de redes. Las redes neuronales comunes se incluyen en **Keras.layers**.
- Keras es una aplicación de alto nivel que se utiliza para construir y entrenar modelos de aprendizaje profundo. Se puede utilizar para el diseño rápido de prototipos, la investigación avanzada y la producción. Tiene las tres ventajas siguientes:
  - Fácil de usar
    - Keras proporciona GUI simples y consistentes optimizadas para casos comunes. Proporciona información práctica y clara sobre los errores del usuario.
  - Modular y componente
    - Puede crear modelos Keras conectando bloques de construcción configurables entre sí, con pocas restricciones.
  - Fácil de extender
    - Puede personalizar los bloques de construcción para expresar nuevas ideas de investigación, crear capas y funciones de pérdida, y desarrollar modelos avanzados.

Métodos e interfaces comunes de Keras

- A continuación se describen los métodos e interfaces comunes de tf.keras centrándose en el código. El contenido principal es el siguiente:
  - Procesamiento de conjuntos de datos: conjuntos de datos y preprocesamiento
  - Creación de modelos de red neuronal: secuencial, modelo, capas...
  - Compilación de red: Compilar, Pérdidas, Métricas y Optimizadores
  - Capacitación y evaluación de redes: ajuste, ajuste_generador y evaluación

CONTENIDO

1. Marcos generales de desarrollo
2. Básicos de Tensorflow 2.x
3. Módulos comunes de TensorFlow 2.x
4. **Pasos básicos para el desarrollo de aprendizaje profundo**

Configuración del entorno de flujo de tensor en Windows 10

- Configuración del entorno en Windows 10:
  - Sistema operativo: Windows 10
  - software pip construido en Anaconda 3 (adaptándose a Python 3)
  - Instalación de TensorFlow:
    - Abra Anaconda Prompt y ejecute el comando pip para instalar TensorFlow.
    - Ejecute pip install TensorFlow en la interfaz de línea de comandos.

```
(base) C:\Users\ThinkPad>pip install tensorflow
Requirement already satisfied: tensorflow in d:\vs\anaconda3_64\lib\site-packages (1.14.0)
Requirement already satisfied: astor>=0.6.0 in c:\users\thinkpad\appdata\roaming\python\python36\site-packages (from ten
sorflow) (0.8.0)
Requirement already satisfied: keras-preprocessing>=1.0.5 in c:\users\thinkpad\appdata\roaming\python\python36\site-pack
ages (from tensorflow) (1.1.0)
Requirement already satisfied: six>=1.10.0 in d:\vs\anaconda3_64\lib\site-packages (from tensorflow) (1.11.0)
Requirement already satisfied: keras-applications>=1.0.6 in c:\users\thinkpad\appdata\roaming\python\python36\site-packa
ges (from tensorflow) (1.0.8)
Requirement already satisfied: grpcio>=1.8.6 in d:\vs\anaconda3_64\lib\site-packages (from tensorflow) (1.23.0)
Requirement already satisfied: gast>=0.2.0 in d:\vs\anaconda3_64\lib\site-packages (from tensorflow) (0.3.2)
Requirement already satisfied: tensorflow-estimator<1.15.0rc0,>=1.14.0rc0 in c:\users\thinkpad\appdata\roaming\python\py
thon36\site-packages (from tensorflow) (1.14.0)
Requirement already satisfied: termcolor>=1.1.0 in c:\users\thinkpad\appdata\roaming\python\python36\site-packages (from
 tensorflow) (1.1.0)
```

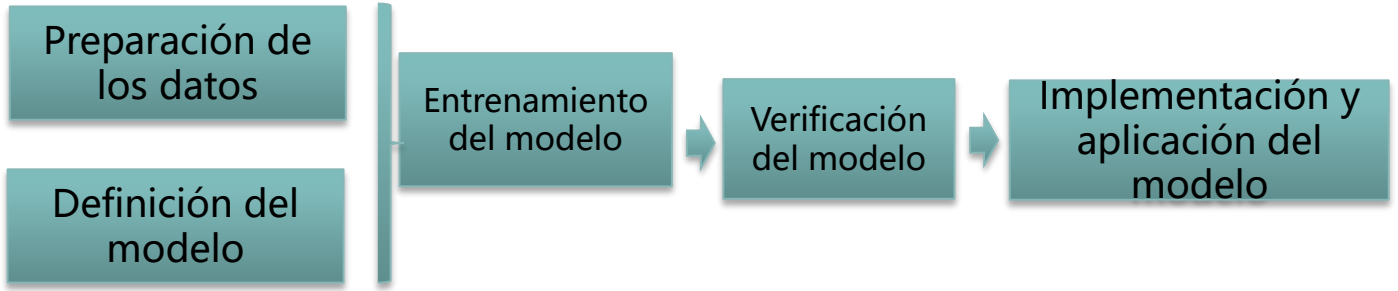
Configuración del entorno TensorFlow en Ubuntu/Linux

- La forma más sencilla de instalar TensorFlow en Linux es ejecutar el comando pip.

- comando pip: instalar TensorFlow==2.1.0

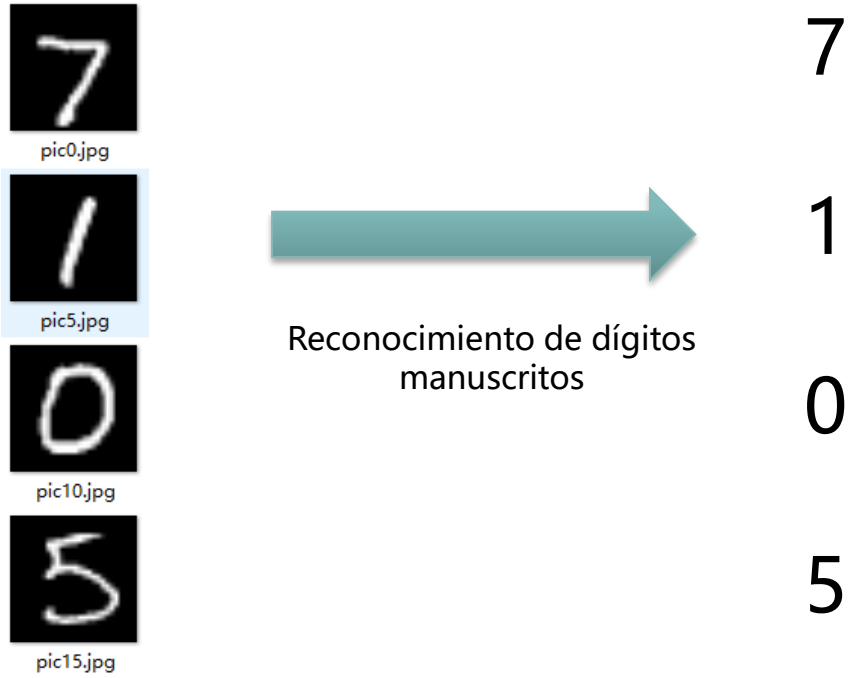
Proceso de desarrollo de TensorFlow

- Preparación de los datos
  - Exploración de datos
  - Procesamiento de datos
- Construcción de la red
  - Definición de una estructura de red.
  - Definición de funciones de pérdida, selección de optimizadores y definición de indicadores de evaluación de modelos.
- Modelo de entrenamiento y verificación
- Guardar modelos
- Restauración e invocación de modelos



Descripción del proyecto

- El reconocimiento de dígitos a mano es una tarea común de reconocimiento de imágenes donde las computadoras reconocen texto en imágenes de escritura a mano. Diferente de las fuentes impresas, la escritura a mano de diferentes personas tiene diferentes tamaños y estilos, lo que hace difícil que los ordenadores reconozcan la escritura a mano. Este proyecto aplica herramientas de aprendizaje profundo y TensorFlow para formar y construir modelos basados en el conjunto de datos de escritura a mano de MNIST.



Preparación de datos

- Conjuntos de datos MNIST
  - Descargue los conjuntos de datos MNIST de http://yann.lecun.com/exdb/mnist/
  - Los conjuntos de datos MNIST constan de un conjunto de entrenamiento y un conjunto de pruebas.
    - Conjunto de entrenamiento: 60,000 imágenes de escritura a mano y etiquetas correspondientes
    - Juego de pruebas: 10,000 imágenes de escritura a mano y etiquetas correspondientes

Ejemplos:

Etiquetas correspondientes	[0,0,0,0,0, 1,0,0,0,0]	[0,0,0,0,0, 0,0,0,0,1]	[0,0,0,0,0, 0,0,1,0,0]	[0,0,0,1,0, 0,0,0,0,0]	[0,0,0,0,1, 0,0,0,0,0]

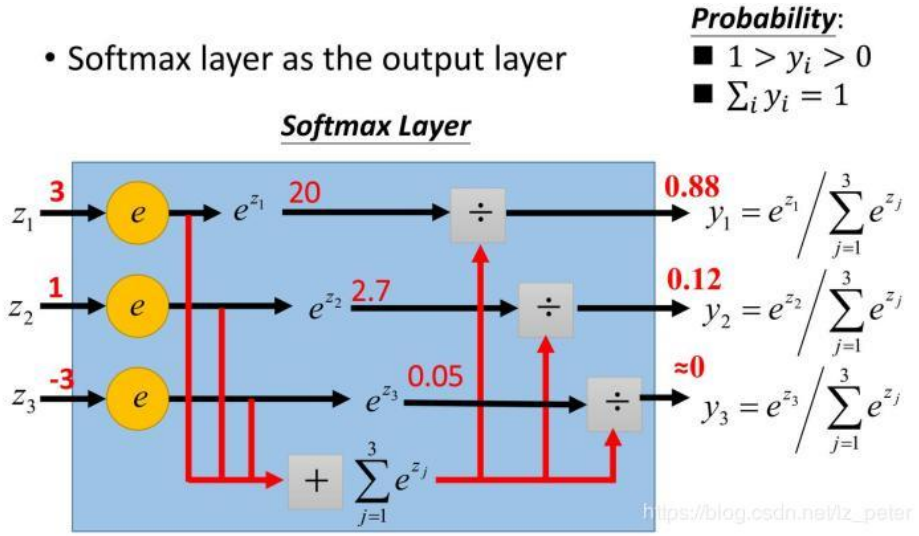
Definición de la estructura de red (1)

- Modelo de regresión Softmax

$$evidence_i = \sum_j W_{i,j} x_j + b_i$$

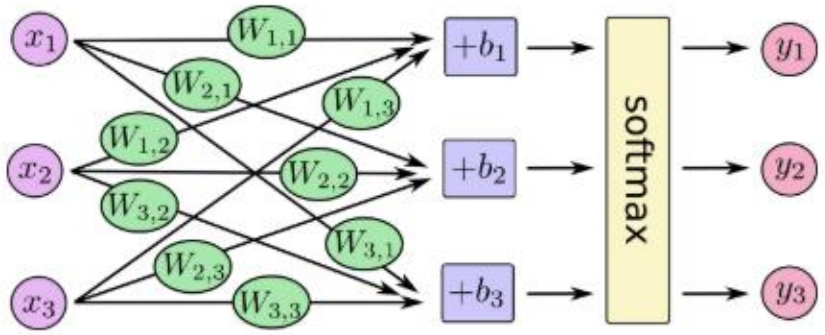
$$y = soft\max(evidence)$$

- La función softmax también se denomina función exponencial normalizada. Es una derivada de la función de clasificación binaria sigmoide en términos de clasificación multi-clase. La siguiente figura muestra el método de cálculo de softmax.



Definición de la estructura de red (2)

- El proceso de establecimiento de modelos es el proceso central de definición de la estructura de red.
- El proceso de operación de la red define cómo se calcula la salida del modelo en función de la entrada.



- La multiplicación de matrices y la adición de vectores se utilizan para expresar el proceso de cálculo de softmax.

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \text{softmax}\left(\begin{bmatrix} W_{1,1} & W_{1,2} & W_{1,3} \\ W_{2,1} & W_{2,2} & W_{2,3} \\ W_{3,1} & W_{3,2} & W_{3,3} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \right)$$

Definición de la estructura de red (3)

- Modelo de regresión softmax basado en TensorFlow

```
## import tensorflow
import tensorflow as tf
##define input variables with operator symbol variables.
‘” we use a variable to feed data into the graph through the placeholders X. Each input image is flattened into a 784-dimensional vector. In this case, the shape of the tensor is [None, 784], None indicates can be of any length. ”’
X = tf.placeholder(tf.float32,[None,784])
‘” The variable that can be modified is used to indicate the weight w and bias b. The initial values are set to 0. ”’
w = tf.Variable(tf.zeros([784,10]))
b = tf.Variable(tf.zeros([10]))
‘” If tf.matmul(x, w) is used to indicate that x is multiplied by w, the Soft regression equation is y = softmax(wx+b)'’’
y = tf.nn.softmax(tf.matmul(x,w)+b)
```

Compilación de red

- La compilación de modelos consta de las dos partes siguientes:
  - Selección de función de pérdida
- En machine learning/deep learning, se debe definir un indicador que indique si un modelo es adecuado. Este indicador se denomina costo o pérdida y se minimiza en la medida de lo posible. En este proyecto se utiliza la función de pérdida de entropía cruzada.
  - Método de descenso de gradiente
- Una función de pérdida se construye para un modelo original, necesita ser optimizado mediante el uso de un algoritmo de optimización, para encontrar parámetros óptimos y minimizar aún más un valor de la función de pérdida. Entre los algoritmos de optimización para la resolución de parámetros de aprendizaje automático, se utiliza generalmente el algoritmo de optimización basado en el descenso de gradiente (Gradient Descent).

```
model.compile(optimizer=tf.train.AdamOptimizer(),
              loss=tf.keras.losses.categorical_crossentropy,
              metrics=[tf.keras.metrics.categorical_accuracy])
```

Entrenamiento de modelos

- Proceso de entrenamiento:
  - Todos los datos del entrenamiento se entrenan a través de iteración por lotes o iteración completa. En el experimento, todos los datos son entrenados cinco veces.
  - En TensorFlow, model.fit se utiliza para el entrenamiento, donde epoch indica el número de iteraciones de entrenamiento.

```
model.fit(mnist.train.images, mnist.train.labels, epochs=5)
```

```
Epoch 1/5
55000/55000 [==============================] - 4s 74us/sample - loss: 0.3043 - categorical_accuracy: 0.9110
Epoch 2/5
55000/55000 [==============================] - 4s 73us/sample - loss: 0.1460 - categorical_accuracy: 0.9569
Epoch 3/5
55000/55000 [==============================] - 4s 79us/sample - loss: 0.1104 - categorical_accuracy: 0.9669
Epoch 4/5
55000/55000 [==============================] - 4s 74us/sample - loss: 0.0881 - categorical_accuracy: 0.9722
Epoch 5/5
55000/55000 [==============================] - 4s 73us/sample - loss: 0.0767 - categorical_accuracy: 0.9760
```

Evaluación del modelo

- Puede probar el modelo utilizando el conjunto de pruebas, comparar los resultados previstos con los reales y encontrar etiquetas correctamente predichas para calcular la precisión del conjunto de pruebas.

```
model.evaluate(mnist.test.images, mnist.test.labels)
```

```
10000/10000 [==============================] - 0s 42us/sample - loss: 0.0779 - categorical_accuracy: 0.9764
[0.07786676207473502, 0.9764]
```

Valor de pérdida

Precisión

Quiz

1. En TensorFlow 2.x, la ejecución temprana (eager execution) está habilitada por defecto. (1)
   A. Cierto.
   B. Falso
2. ¿Cuál de las siguientes afirmaciones acerca de tf.keras.Model y tf.keras.Sequential es incorrecta cuando se utiliza la interfaz tf.keras para construir un modelo de red? (1)
   A. tf.keras.Model admite modelos de red con varias entradas, mientras que tf.keras.Sequential no.
   B. tf.keras.Model admite modelos de red con varias salidas, mientras que tf.keras.Sequential no.
   C. tf.keras.Model se recomienda para la construcción de modelos cuando existe una capa de uso compartido en la red.
   D. tf.keras.Sequential se recomienda para la creación de modelos cuando existe una capa de uso compartido en la red.

RESUMEN

- Este capítulo describe el siguiente contenido enfocándose en el código: Características de los marcos de aprendizaje profundo comunes, incluyendo Pytorch y TensorFlow, sintaxis básica y módulos comunes de TensorFlow 2.x, Procedimiento de desarrollo de TensorFlow.

Más información

Página official de TensorFlow: https://tensorflow.google.cn

Página official de PyTorch : https://PyTorch.org/

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

HUAWEI

Framework de desarrollo de IA Mindspore de Huawei

HUAWEI

Prefacio

- Este capítulo presenta la estructura, el concepto de diseño y las características de Mindspore basadas en los problemas y dificultades que enfrenta el marco de computación de IA, y describe el proceso de desarrollo y aplicación en Mindspore.

Objetivos

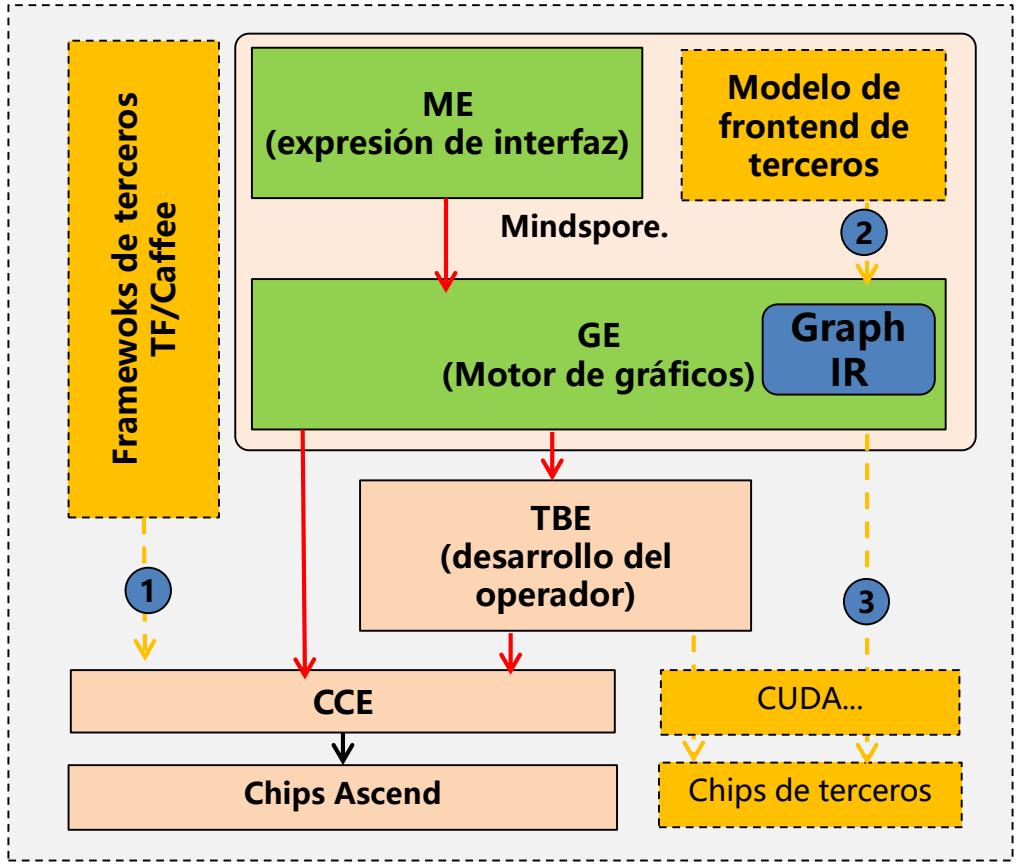
Al finalizar este curso, podrá:

- Aprender que es Mindspore
- Comprender el framework de Mindspore
- Comprender el concepto de diseño de Mindspore
- Aprender las características de Mindspore
- Entender el proceso de configuración del entorno y los casos de desarrollo

Contenido

1. **Marco de desarrollo**
   - Arquitectura.
     - Características principales
2. Desarrollo y aplicación

Arquitectura: Fácil desarrollo y ejecución eficiente



ME (Mind Expression): capa de interfaz (Python)

Usabilidad: programación diferencial automática y expresión matemática original

- Auto differ: diferencial automático a nivel de operador
- Automático paralelo: paralelismo automático
- Tensor automático: generación automática de operadores
- Etiquetado semiautomático: etiquetado de datos semiautomático

GE (Motor de gráficos): capa de compilación y ejecución de gráficos

Alto rendimiento: Cooptimización de hardware/software, y aplicación de escenario completo

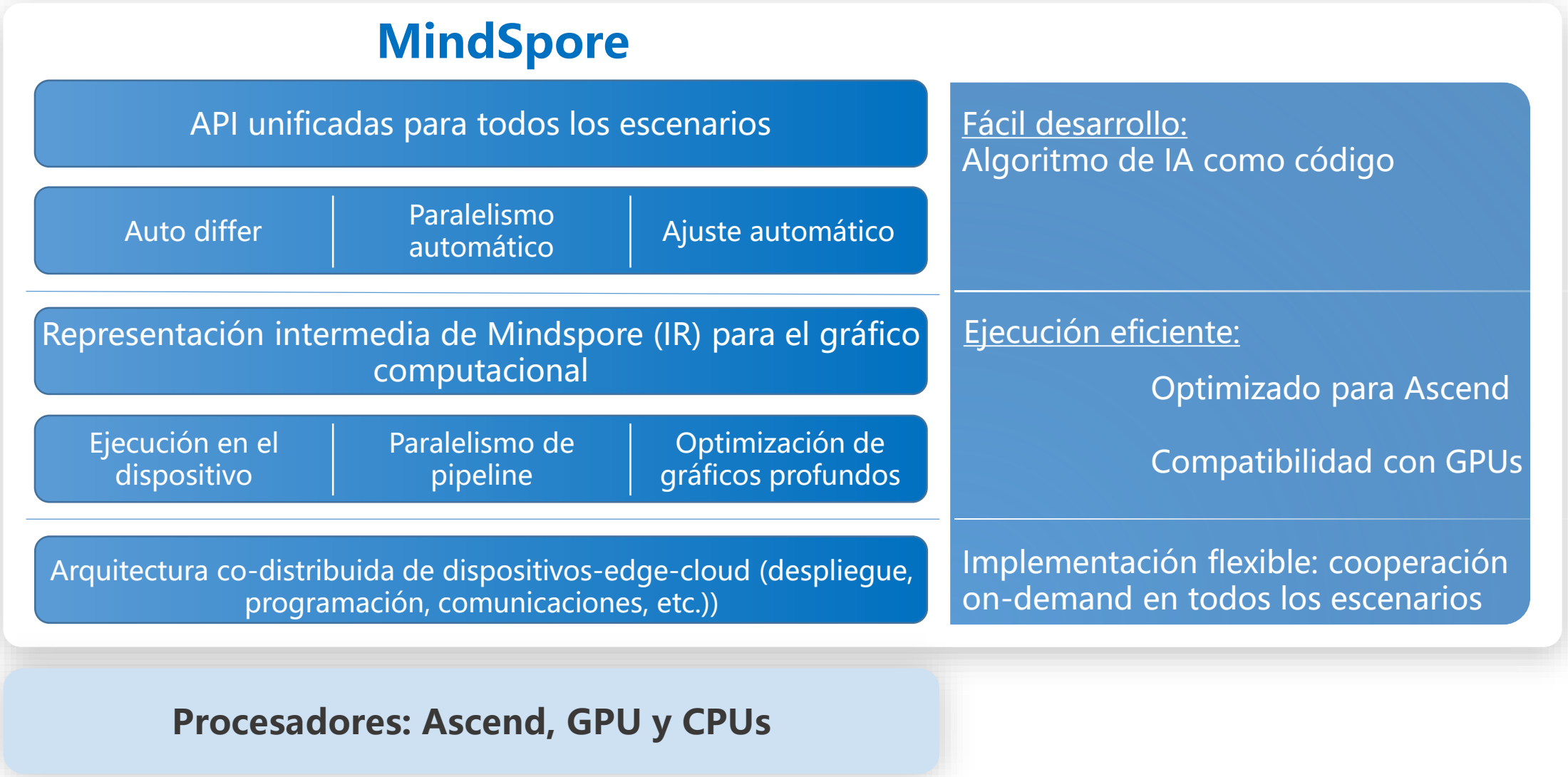
- Cross-layer overcommitment en memoria
- Optimización de graficas profundas
- Ejecución en el dispositivo
- Sinergia entre dispositivos y nubes (incluida la compilación en línea)

① Equivalente a los marcos de código abierto de la industria, Mindspore sirve preferentemente chips autodesarrollados y servicios en la nube.

② Soporta la interconexión ascendente con marcos de terceros y puede interconectarse con ecosistemas de terceros a través de Graph IR, incluyendo modelos de referencia y de formación. Los desarrolladores pueden ampliar la capacidad de Mindspore.

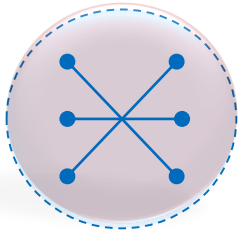
③ También es compatible con la interconexión con chips de terceros y ayuda a los desarrolladores a aumentar los escenarios de aplicación de Mindspore y ampliar el ecosistema de IA.

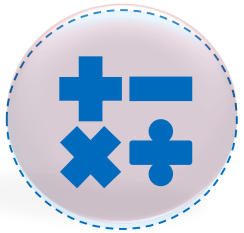
Solución general: Arquitectura Core



Diseño Mindspore: Auto Differ

Trayectoria técnica de diferencial automático





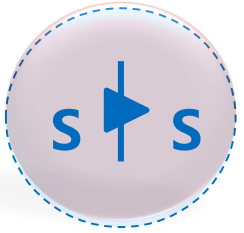


Gráfico: Tensorflow

- Programación no basada en Python basada en gráficos
- Representación compleja de flujos de control y derivados de orden superior

Sobrecarga del operador: Pytorch

- Encabezado de tiempo de ejecución
- Es difícil optimizar el rendimiento del proceso hacia atrás.

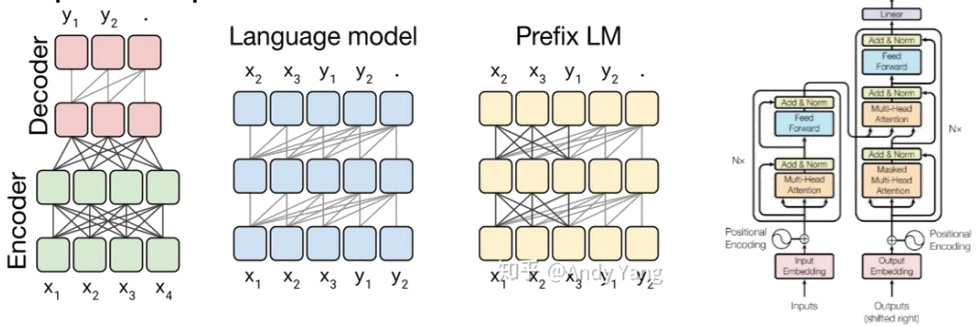
Transferencia de código fuente: Mindspore.

- API de Python para una mayor eficiencia
- Optimización de la compilación basada en IRS para obtener un mejor rendimiento

Paralelismo automático

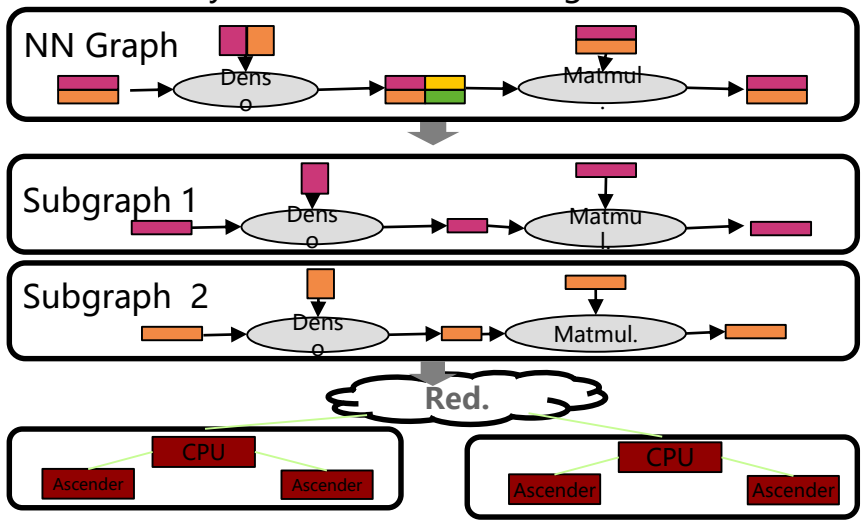
Desafíos

Los modelos ultragrandes realizan una capacitación distribuida eficiente:
A medida que los modelos de dominio NLP se hinchan, la sobrecarga de memoria para entrenar a modelos ultragrandes como Bert (340M)/GPT-2(1542M) ha excedido la capacidad de una sola tarjeta. Por lo tanto, los modelos deben dividirse en varias tarjetas antes de la ejecución.
Actualmente se utiliza el paralelismo de modelo manual. Es necesario diseñar la segmentación del modelo y comprender la topología del clúster. El desarrollo es extremadamente difícil. El rendimiento es mediocre y apenas se puede optimizar.



Tecnologías clave

Segmentación automática del gráfico: Puede segmentar todo el gráfico basándose en las dimensiones de datos de entrada y salida del operador e integrar el paralelismo de datos y modelos.
Programación de la conciencia de topología de clúster: puede percibir la topología de clúster, programar subgráficos automáticamente y minimizar la sobrecarga de comunicación.



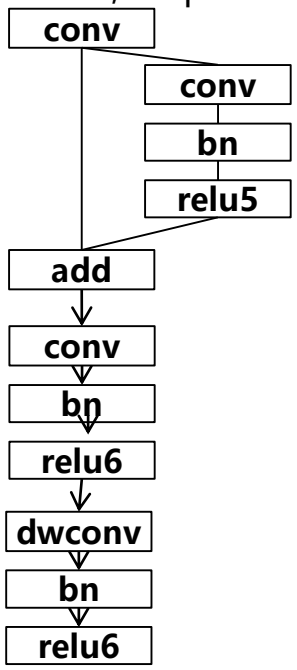
Efecto: Permite realizar el paralelismo del modelo basado en la lógica de código de nodo único existente, lo que mejora la eficiencia del desarrollo en diez veces en comparación con el paralelismo manual.

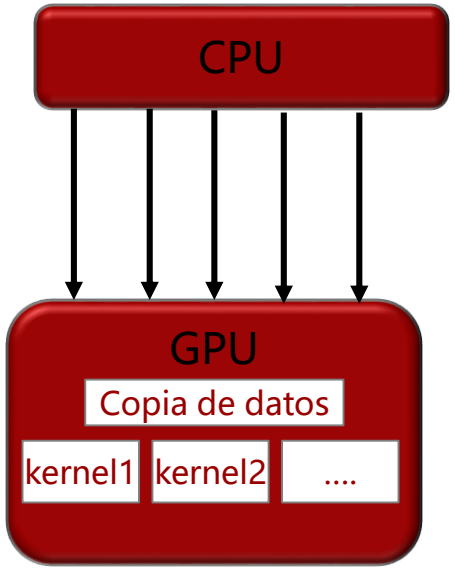
Ejecución en el dispositivo (1)

Desafíos

Desafíos para la ejecución de modelos con una potencia informática de chip suprema:

Pared de memoria, sobrecarga de interacción alta y dificultad de suministro de datos. Las operaciones parciales se realizan en el host, mientras que las demás se realizan en el dispositivo. La sobrecarga de interacción es mucho mayor que la sobrecarga de ejecución, lo que resulta en un bajo uso del acelerador.

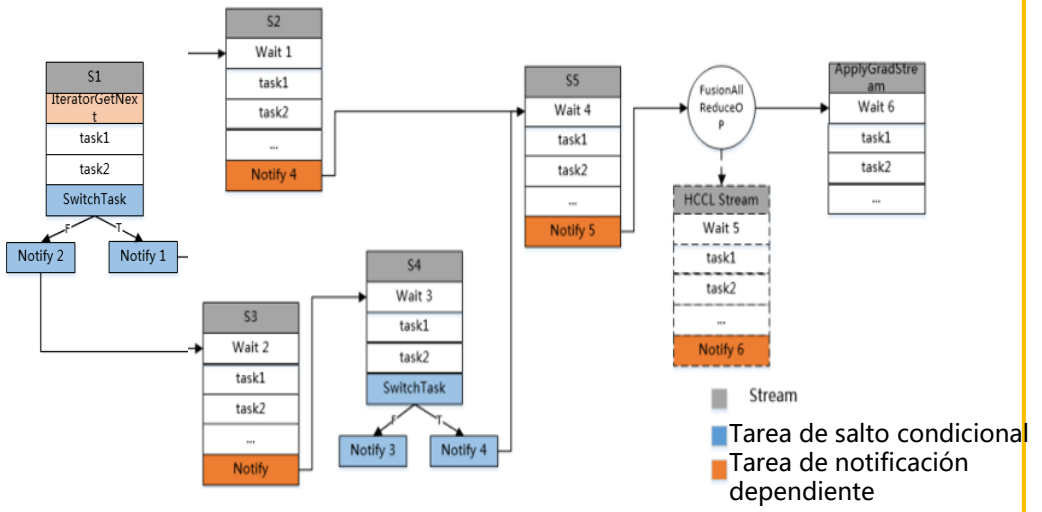




Gran carga de interacción de datos y difícil suministro de datos

Tecnologías clave

La optimización de gráficos profundos orientada a chips reduce el tiempo de espera de sincronización y maximiza el paralelismo de los datos, la informática y la comunicación. El preprocesamiento de datos y el cálculo se integran en el chip Ascend:



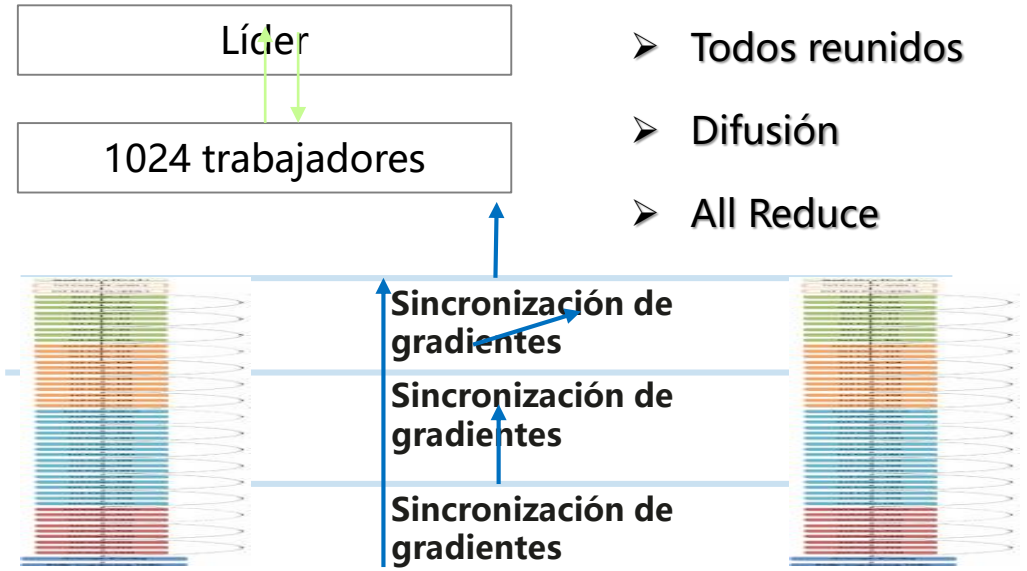
Efecto: Eleve el rendimiento del entrenamiento diez veces más que la programación gráfica en el host.

HUAWEI

Ejecución en el dispositivo (2)

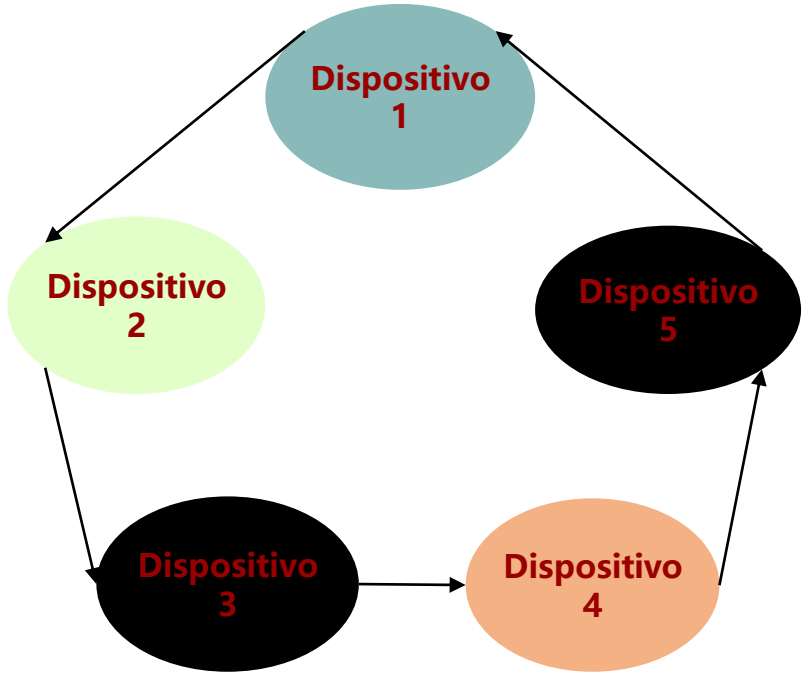
Desafíos

Desafíos para la agregación de gradientes distribuidos con una potencia computacional de chip suprema:
la sobrecarga de sincronización del control central y la sobrecarga de comunicación de sincronización frecuente de Resnet50 bajo la iteración única de 20 ms; el método tradicional sólo puede completar All Reduce después de tres veces de sincronización, mientras que el método de datos puede realizar de forma autónoma All Reduce sin causar sobrecarga de control.



Tecnologías clave

La optimización de la **segmentación gráfica adaptativa impulsada por los datos de gradiente** puede llevar a cabo la descentralización de All Reduce y sincronizar la agregación de gradientes, impulsando la eficiencia informática y de la comunicación.



Efecto: una sobrecarga difuminante de menos de 2 ms

Arquitectura de sinergia distribuida de Dispositivos-Edge-Cloud

Desafíos

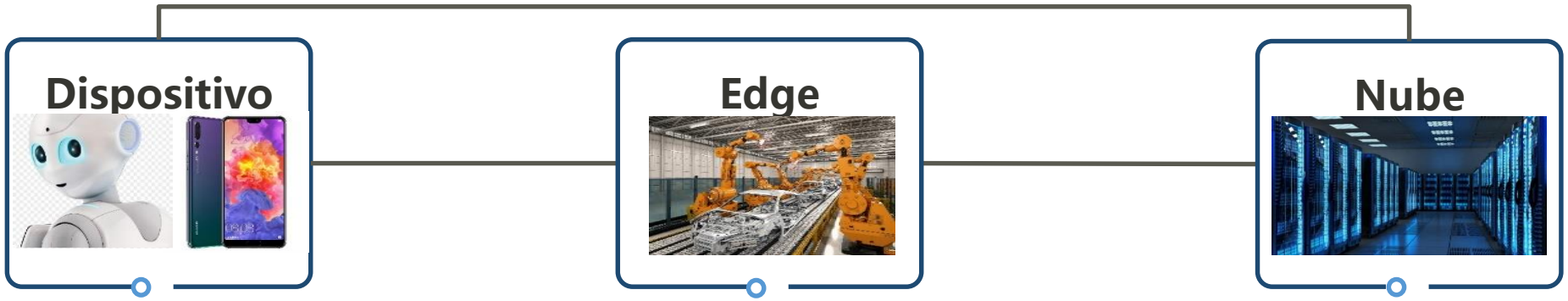
La diversidad de arquitecturas de hardware genera diferencias de implementación en escenarios completos e incertidumbres de rendimiento. La separación de la formación y la inferencia conduce al aislamiento de los modelos.

Tecnologías clave

- **El modelo unificado de IR** ofrece una experiencia de implementación coherente.
- **La tecnología de optimización de gráficos con colaboración de software y hardware** tiende puentes entre diferentes escenarios.
- Synergy Federal Meta Learning de dispositivo-nube, rompe el límite de dispositivos-nube y actualiza el modelo de colaboración multidispositivo en tiempo real.

Efecto: rendimiento de implementación de modelos consistente en todos los escenarios gracias a la arquitectura unificada y a la precisión mejorada de los modelos personalizados

Colaboración on-demand en todos los escenarios y experiencia de desarrollo consistente



HUAWEI

Contenido

1. **Marco de desarrollo**
   - Arquitectura.
   - Características principales
2. Desarrollo y aplicación

El marco de la computación en IA: retos

Desafíos del sector

Una gran brecha entre

Investigación industrial y aplicación de IA en todos los escenarios

- Grandes barreras de entrada
- Alto costo de ejecución
- Larga duración del despliegue

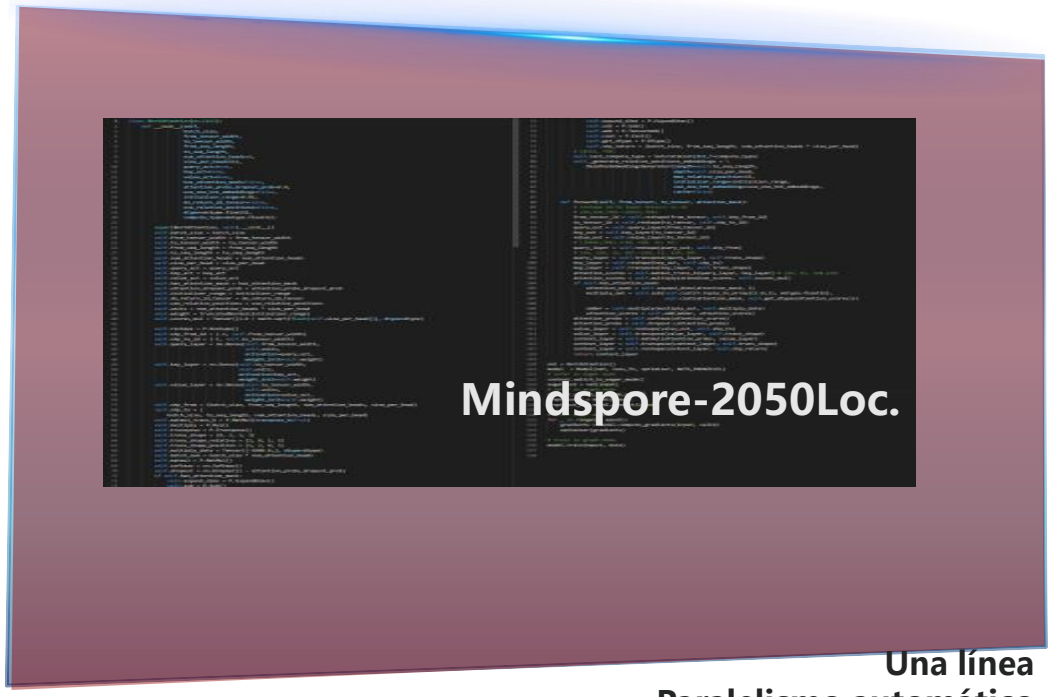
Innovación tecnológica

Mindspore facilita la IA inclusiva en todas las aplicaciones

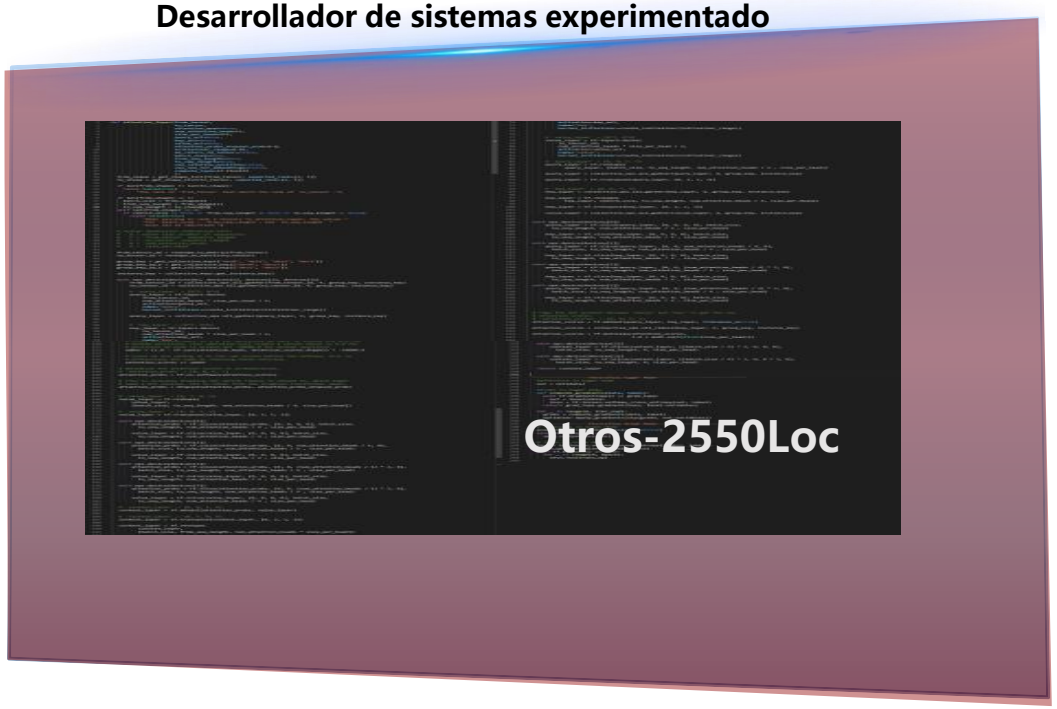
- Nuevo modo de programación
- Nuevo modo de ejecución
- Nuevo modo de colaboración

Nuevo paradigma de programación

Algoritmo científico



Algoritmo científico
+.
Desarrollador de sistemas experimentado



Una línea
Paralelismo automático

Diferencial automático eficiente

Conmutador de modo de depuración en una línea

Modelo de NLP: Transformer

HUAWEI

Ejemplo de código

Fragmento de código de tensorflow: líneas xx, paralelismo manual

```
import tensorflow as tf

model() {
    with tf.device("/device:0")
        token_type_table = tf.get_variable(
            name=token_type_embedding_name,
        shape=[token_type_vocab_size, width],
        initializer=create_initializer(initializer_range))
        flat_token_type_ids = tf.reshape(token_type_ids, [-1])
        one_hot_ids = tf.one_hot(flat_token_type_ids, depth=token_type_vocab_size)
        token_type_embeddings = tf.matmul(one_hot_ids, token_type_table)

    with tf.device("/device:1")
        query_layer = tf.layers.dense(
            from_tensor_2d,
            num_attention_heads * size_per_head,
            activation=query_act,
            name="query",
            kernel_initializer=create_initializer(initializer_range))

    with tf.device("/device:2")
        key_layer = tf.layers.dense(
            to_tensor_2d,
            num_attention_heads * size_per_head,
            activation=key_act,
            name="key",
            kernel_initializer=create_initializer(initializer_range))
```

Fragmento de código de Mindspore: dos líneas, paralelismo automático

```
class DenseMatMulNet(nn.Cell):
    def __init__(self):
        super(DenseMutMulNet, self).__init__()
        self.matmul1 = ops.MatMul.set_strategy({[4, 1], [1, 1]})
        self.matmul2 = ops.MatMul.set_strategy({[1, 1], [1, 4]})
    def construct(self, x, w, v):
        y = self.matmul1(x, w)
        z = self.matmul2(y, v)
        return s
```

Escenarios típicos: ReID

Nuevo modo de ejecución (1)

Desafíos de ejecución

Informática compleja de IA y diversas unidades informáticas

1. Núcleos de CPU, cubos y vectores
2. Cálculo escalar, vector y tensor
3. Cálculo de precisión mixta
4. Cálculo de matrices densas y matrices dispersas

Ejecución de varios dispositivos: alto coste del control en paralelo

El rendimiento no puede aumentar linealmente a medida que aumenta la cantidad de nodos.

Ejecución en el dispositivo

Descarga gráficos a dispositivos, maximizando la potencia de cálculo de Ascend

Optimización del marco

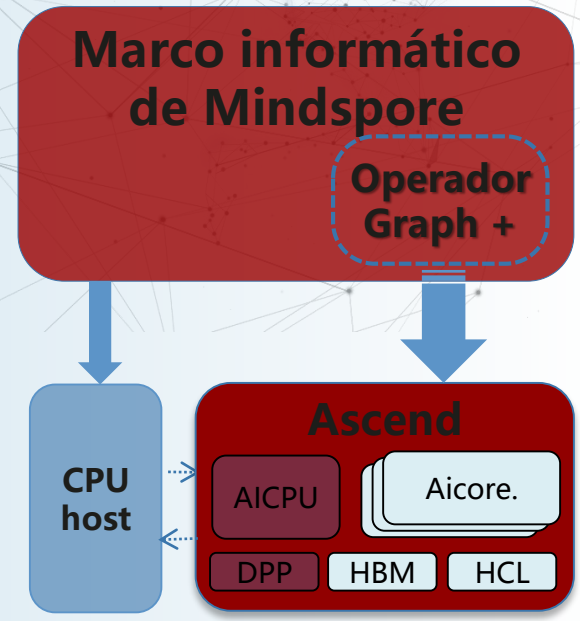
paralelismo de pipeline

Exceso de compromiso de memoria entre capas

Cooptimización de hardware/software

Ejecución en el dispositivo

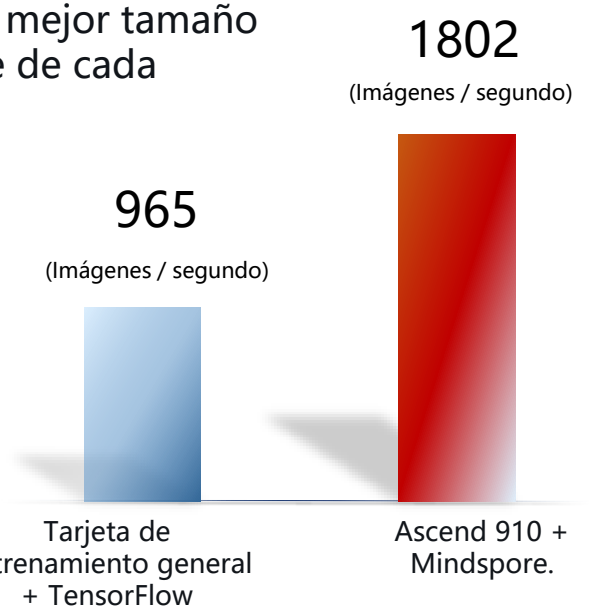
Optimización de gráficos profundos



HUAWEI

Nuevo modo de ejecución (2)

- ResNet 50 V1.5
- ImageNet 2012
- Con el mejor tamaño de lote de cada tarjeta



El rendimiento de ResNet-50 se duplica.
Una iteración:
58 ms (otros frameworks + V 100) v.s. unos 22 ms (Mindspore) (ResNet50 + ImageNet, un solo servidor, ocho dispositivos, tamaño de lote = 32)

Detección de objetos en 60 ms

Seguimiento de objetos en 5 ms

Reconocimiento multiobjeto en tiempo real
Implementación móvil basada en Mindspore, una experiencia sin problemas de detección de varios objetos

Nuevo modo de colaboración

Desafío de implementación

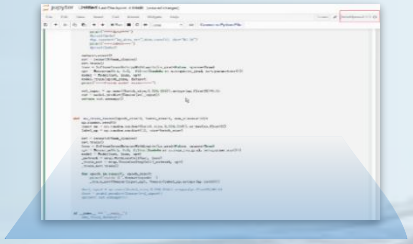
v.s.

- Diversos requisitos, objetivos y restricciones para los escenarios de aplicaciones de dispositivos, edge y nube

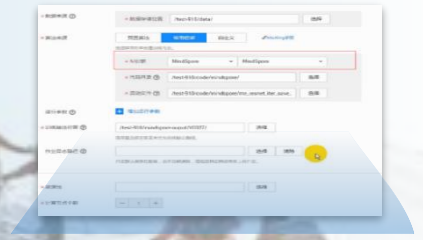
v.s.

- Distintas precisiones y velocidades de hardware

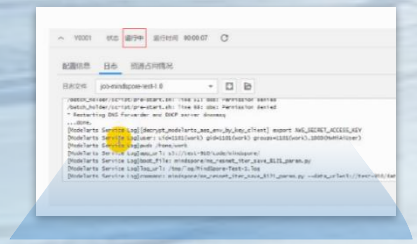
Desarrollo unificado; despliegue flexible; colaboración bajo demanda y alta seguridad y confiabilidad



Desarrollo



Despliegue



Ejecución

Guardando modelo

Desarrollo unificado e implementación flexible

HUAWEI

Alto Desempeño

Los retos de la informática de IA

Informática compleja

Cálculo escalar, vector y tensor

Informática de precisión mixta

Paralelismo entre agregación de gradientes y mini-batch computing

Diversas unidades de cálculo/procesadores

CPU, GPU y procesadores Ascend

Ecalar, vector y tensor

Mindspore

Optimización de framework

Paralelismo de pipeline

Exceso de compromiso de memoria entre capas

Software + Hardware co-optimización

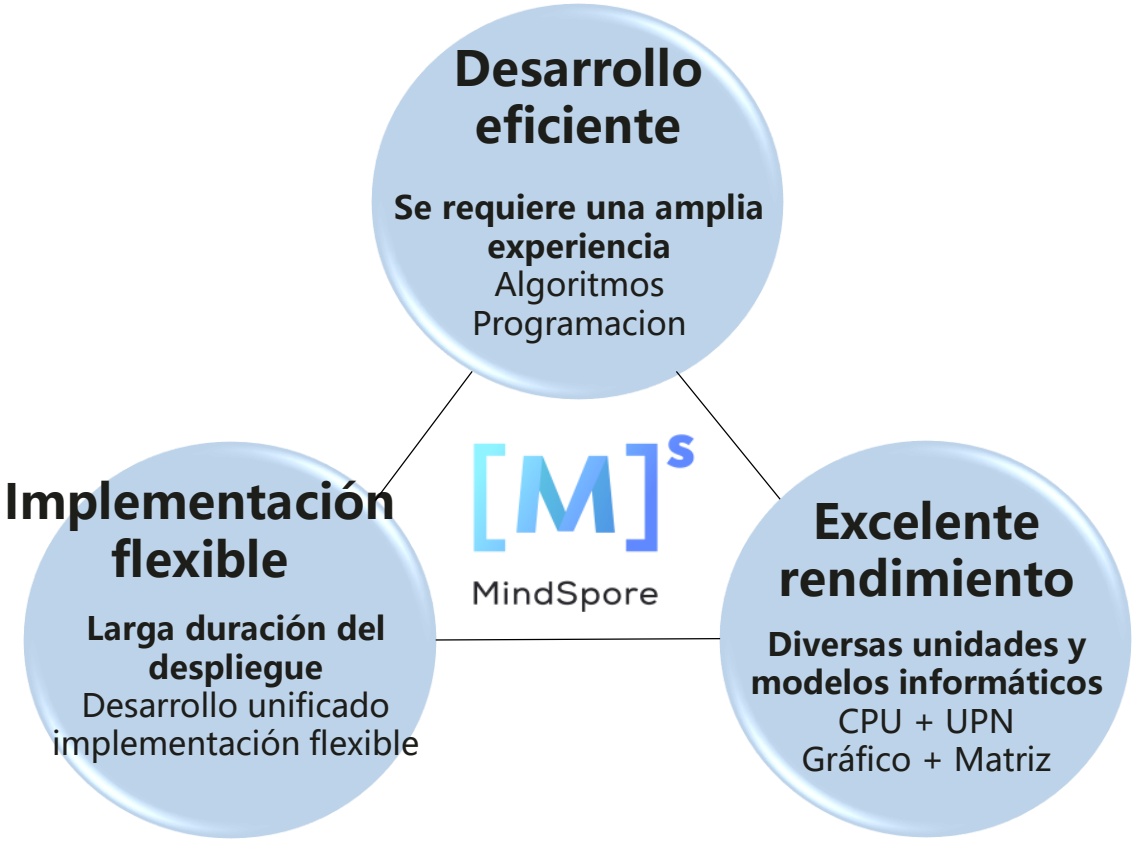
Ejecución en el dispositivo

Optimización de gráficos profundos

- RNet 50 V1.5
- ImageNet 2012
- Basado en tamaños de lote óptimos

HUAWEI

Visión y valor



Contenido

1. Marco de desarrollo
2. **Desarrollo y aplicación**
   - Configuración del entorno
   - Casos de desarrollo de aplicaciones

Instalación de Mindspore

Environment Requirements

System Requirements and Software Dependencies

Version	Operating System	Executable File Installation Dependencies	Source Code Compilation and Installation Dependencies
MindInsight 0.2.0-alpha	- Ubuntu 16.04 or later x86_64 - EulerOS 2.8 arrch64 - EulerOS 2.5 x86_64	- Python 3.7.5 - MindSpore 0.2.0-alpha - For details about other dependency items, see requirements.txt.	**Compilation dependencies:** - Python 3.7.5 - CMake >= 3.14.1 - GCC 7.3.0 - node.js >= 10.19.0 - wheel >= 0.32.0 - pybind11 >= 2.4.3 **Installation dependencies:** same as the executable file installation dependencies.

- When the network is connected, dependency items in the requirements.txt file are automatically downloaded during .whl package installation. In other cases, you need to manually install dependency items.

Installation Guide

Installing Using Executable Files

1. Download the .whl package from the MindSpore website. It is recommended to perform SHA-256 integrity verification first and run the following command to install MindInsight:

```
pip install mindinsight-{version}-cp37-cp37m-linux_{arch}.whl
```

2. Run the following command. If `web address: http://127.0.0.1:8080` is displayed, the installation is successful.

```
mindinsight start
```

Método 1: Compilación e instalación de Código Fuente

Dos entornos de instalación: Ascend y CPU

```
adding 'mindspore/transforms/validators.py'
adding 'mindspore-0.1.0.dist-info/METADATA'
adding 'mindspore-0.1.0.dist-info/WHEEL'
adding 'mindspore-0.1.0.dist-info/top_level.txt'
adding 'mindspore-0.1.0.dist-info/RECORD'
removing build/bdist.linux-x86_64/wheel
------Successfully created mindspore package------
---------------- mindspore: build test end ----------------
```

Método 2: instalación directa mediante el paquete de instalación

Dos entornos de instalación: Ascend y CPU

Comandos de instalación:

1. pip install –y mindspore-cpu
2. pip install –y mindspore-d

Introducción

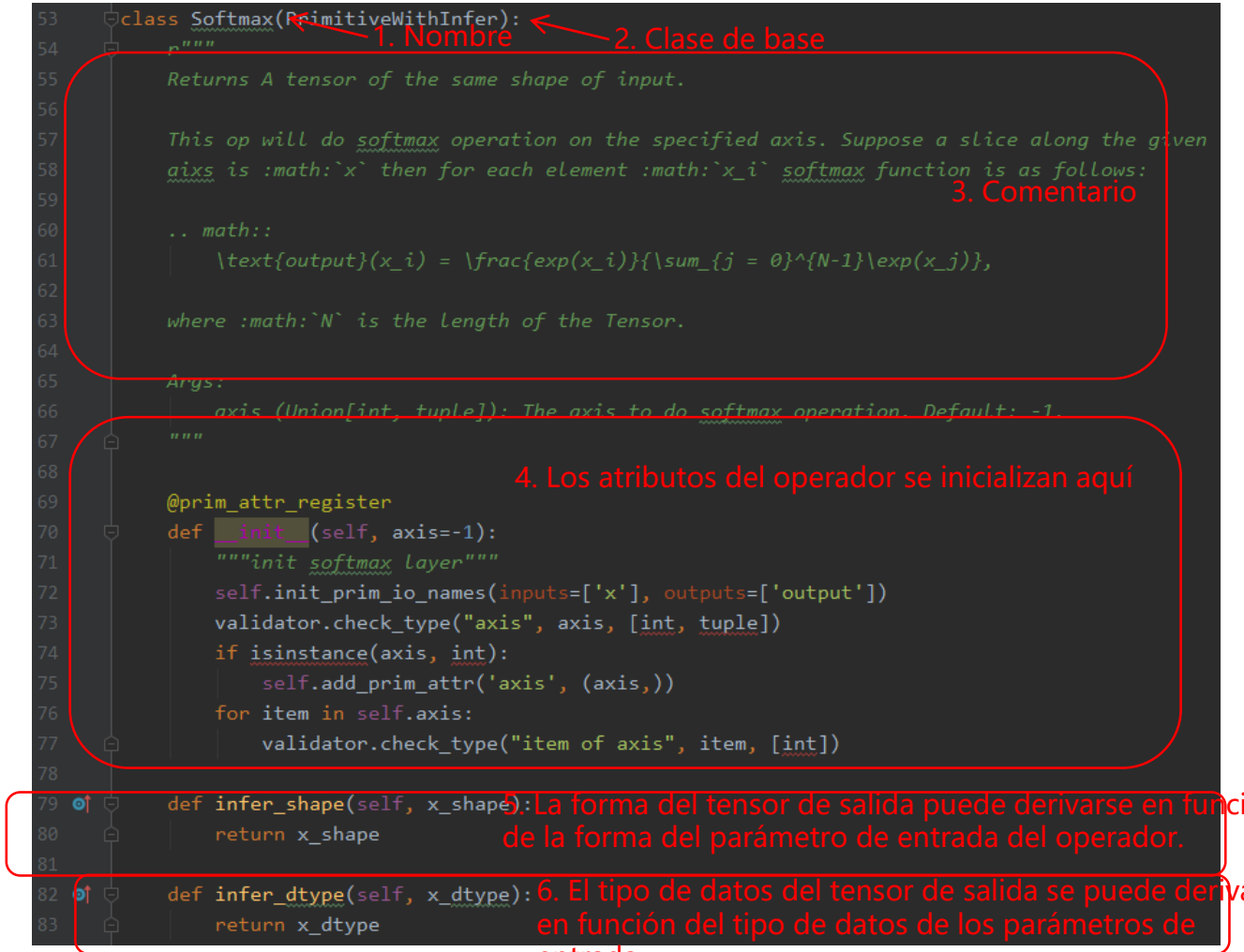
- En Mindspore, los datos se almacenan en tensores. Operaciones comunes del tensor:
  - asnumpy()
  - size()
  - dim()
  - dtype()
  - set_dtype()
  - tensor_add(other: Tensor)
  - tensor_mul(other: Tensor)
  - shape()
  - __Str__# (conversion a cadenas)

Módulo	Descripciùn
Model_zoo	Define modelos de red comunes
communication	Módulo de carga de datos, que define el dataloader y dataset y procesa datos como imágenes y textos.
dataset	Módulo de procesamiento de conjuntos de datos, que lee y procesa datos.
common	Define tensor, parámetro, dtype e inicializador.
contextr	Define la clase de contexto y establece los parámetros de ejecución del modelo, como los modos de conmutación de gráficos y PyNative.
akg	Biblioteca automática de operadores diferenciales y personalizados.
nn	Define células Mindspore (unidades de red neural), funciones de pérdida y optimizadores.
ops	Define los operadores básicos y registra los operadores inversos.
traing	Modelo de entrenamiento y módulos de funciones de resumen.
utils	Utilidades que verifican los parámetros. Este parámetro se utiliza en el framework.

Componentes de ME

Concepto de programación: Operación

Operador Softmax




```
class Softmax(PrimitiveWithInfer):
    """
    Returns A tensor of the same shape of input.

    This op will do softmax operation on the specified axis. Suppose a slice along the given
    aixs is :math:`x` then for each element :math:`x_i` softmax function is as follows:

    .. math::
        \text{output}(x_i) = \frac{exp(x_i)}{\sum_{j = 0}^{N-1}\exp(x_j)},

    where :math:`N` is the length of the Tensor.

    Args:
        axis (Union[int, tuple]): The axis to do softmax operation. Default: -1.
    """

    @prim_attr_register
    def __init__(self, axis=-1):
        """init softmax layer"""
        self.init_prim_io_names(inputs=['x'], outputs=['output'])
        validator.check_type("axis", axis, [int, tuple])
        if isinstance(axis, int):
            self.add_prim_attr('axis', (axis,))
        for item in self.axis:
            validator.check_type("item of axis", item, [int])

    def infer_shape(self, x_shape):
        return x_shape

    def infer_dtype(self, x_dtype):
        return x_dtype
```

Operaciones communes en MindSpore:

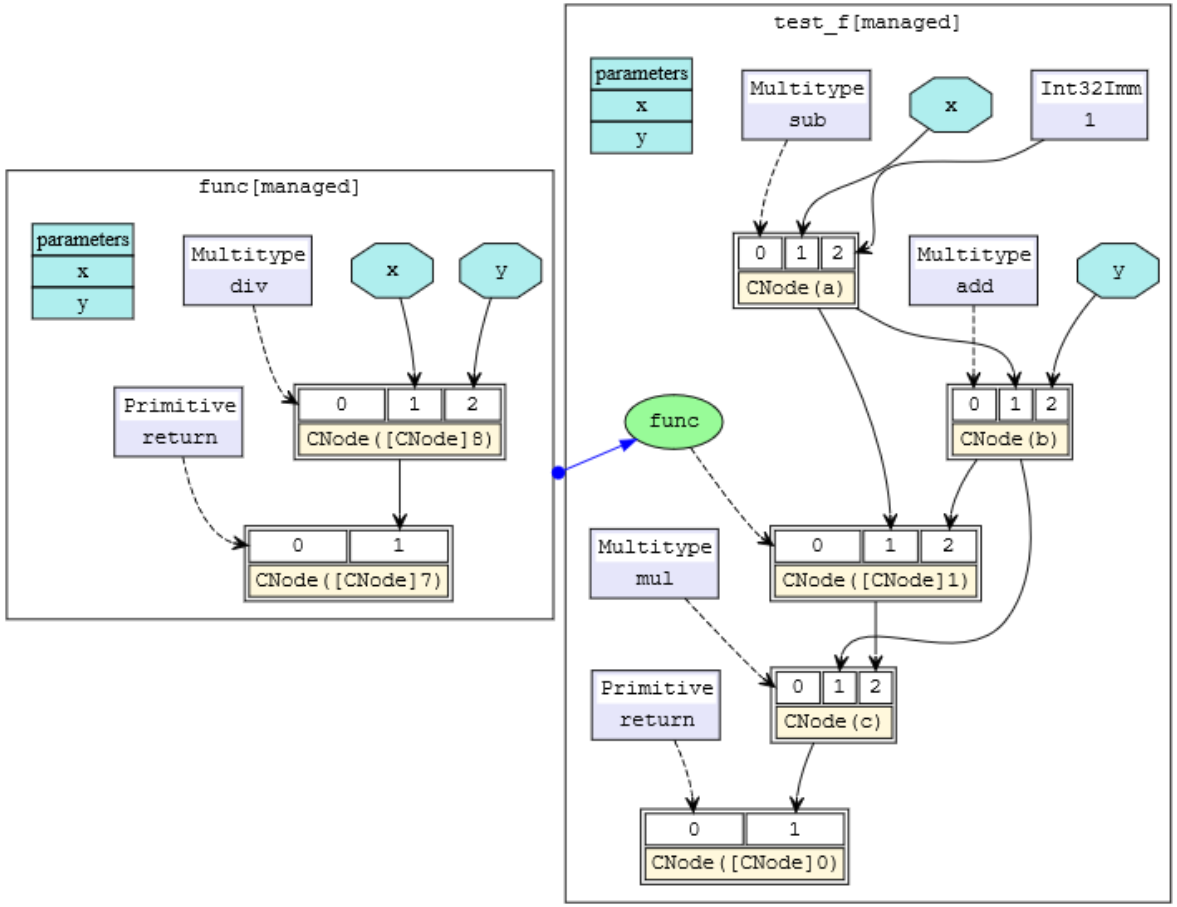
- **array: Array-related operators**
  - **ExpandDims**
  - **Concat**
  - **Select**
  - **ScatterNd...**
  - **Squeeze**
  - **OnesLike**
  - **StridedSlice**
- **math: Math-related operators**
  - **AddN**
  - **Sub**
  - **Mul**
  - **MatMul**
  - **RealDiv**
  - **ReduceMean**
  - **Cos**
  - **Sin**
  - **LogicalAnd**
  - **LogicalNot**
  - **Less**
  - **Greater...**
- **nn: Network operators**
  - **Conv2D**
  - **Flatten**
  - **Softmax**
  - **ReLU**
  - **Sigmoid**
  - **Pooling**
  - **BatchNorm**
  - **MaxPool**
  - **AvgPool**
  - **TopK**
  - **SoftmaxCrossEntropy**
  - **SmoothL1Loss**
  - **SGD**
  - **SigmoidCrossEntropy...**
- **control: Control operators**
  - **ControlDepend**
- **random: Random operators**

Concepto de programación: Cell (celda)

- Una cell define el módulo básico para el cálculo. Los objetos de la cell se pueden ejecutar directamente.
  - __init__: Inicia y verifica módulos como parámetros, celdas y primitivas.
  - Construcción: define el proceso de ejecución. En el modo gráfico, se compila un gráfico para su ejecución y está sujeto a restricciones de sintaxis específicas.
  - bprop (opcional): Es la dirección inversa de los módulos personalizados. Si esta función no está definida, se utiliza el diferencial automático para calcular la reversa de la parte de construcción.
- Las cell predefinidas en MindSpore incluyen principalmente: pérdida común (Softmax Cross Entropy With Logits and MSELoss), optimizadores comunes (Momentum, SGD, y Adam), y funciones comunes de embalaje de red, como el cálculo y actualización del gradiente de la red TrainOneStepCell y el cálculo del gradiente WithGradCell.

Concepto de programación: MindSporeIR

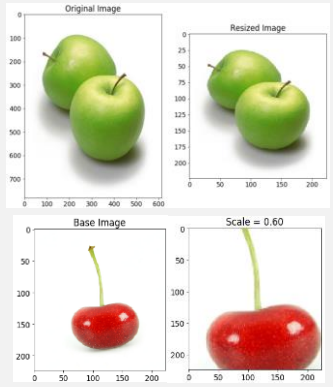
- MindSporeIR es un IR funcional basado en gráficos compacto, eficiente y flexible que puede representar la semántica funcional como variables libres, funciones de orden superior y recursividad. Es un portador de programas en proceso de optimización de compilación y AD.
- Cada gráfico representa un gráfico de definición de función y consta de ParameterNode, ValueNode y ComplexNode (CNode).
- La figura muestra la relación def-use.



Caso de desarrollo

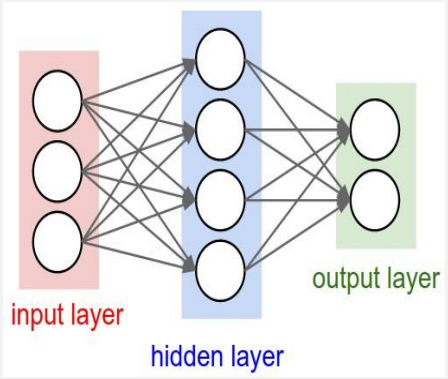
- Tomemos el reconocimiento de los dígitos manuscritos de MNIST como ejemplo para demostrar el proceso de modelado en MindSpore.

Datos



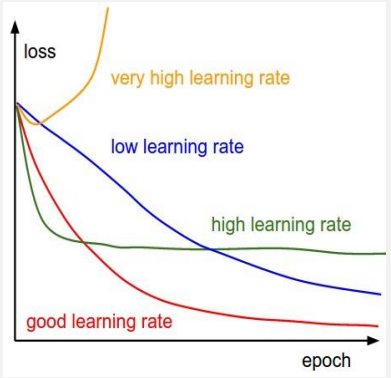
- **1. Carga de datos**
- **2. Mejora de los datos**

Red



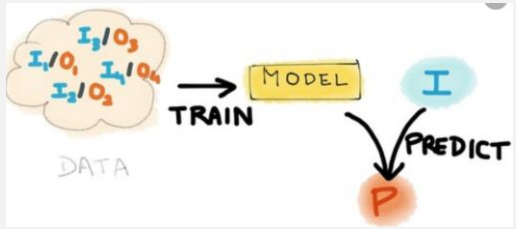
- **3. Definición de la red**
- **4. Inicialización del peso**
- **5. Ejecución de la red**

Modelo



- **6. Función de pérdida**
- **7. Optimizador**
- **8. Iteración de entrenamiento**
- **9. Evaluación de modelos**

Aplicación



- **10. Ahorro de modelos**
- **11. Predicción de carga**
- **12. Ajuste fino**

HUAWEI

Quiz

1. En Mindspore, ¿cuál de las siguientes operaciones es el tipo de nn? (1)
   A. Matemático
   B. Red
   C. Control
   D. Otros

Resumen

- Este capítulo describe el marco, el diseño, las funciones y el proceso de configuración del entorno y el procedimiento de desarrollo de Mindspore.

Más información

TensorFlow: https://www.tensorflow.org/

PyTorch: https://pytorch.org/

Mindspore: https://www.mindspore.cn/en

Comunidad de desarrolladores – Ascend : http://122.112.148.247/home

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

HUAWEI

Framework de desarrollo de IA Mindspore de Huawei

HUAWEI

Prefacio

- Este capítulo presenta la estructura, el concepto de diseño y las características de Mindspore basadas en los problemas y dificultades que enfrenta el marco de computación de IA, y describe el proceso de desarrollo y aplicación en Mindspore.

Objetivos

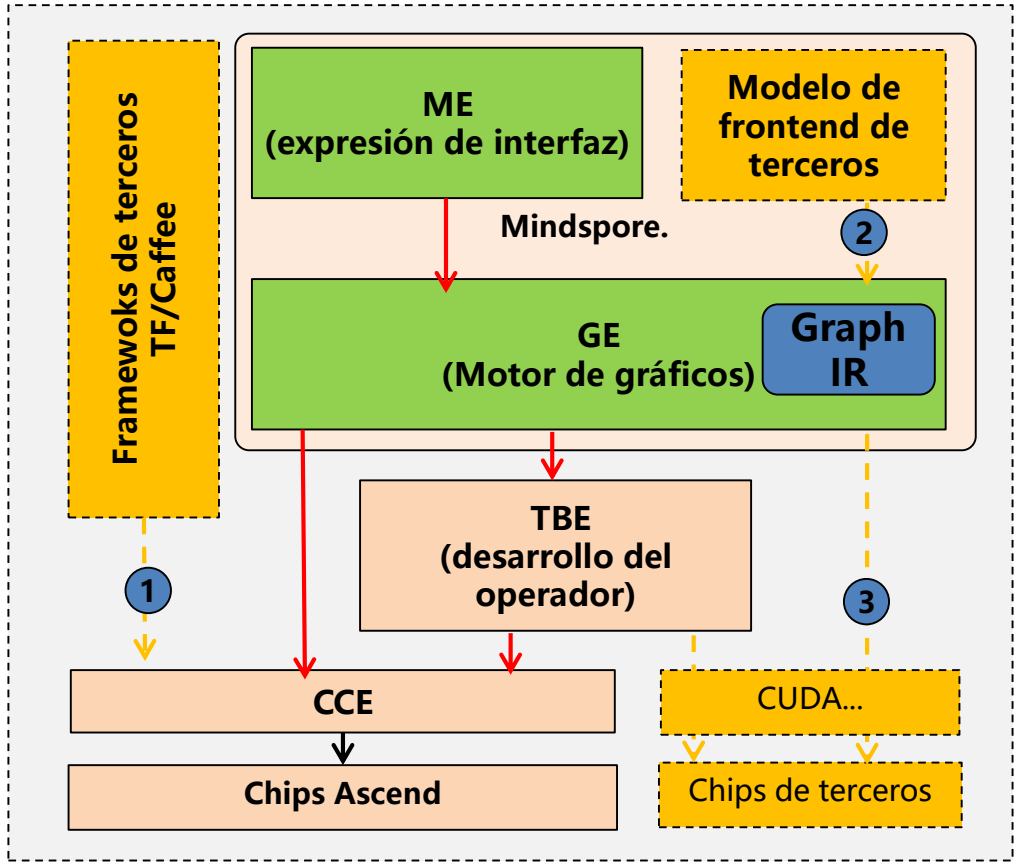
Al finalizar este curso, podrá:

- Aprender que es Mindspore
- Comprender el framework de Mindspore
- Comprender el concepto de diseño de Mindspore
- Aprender las características de Mindspore
- Entender el proceso de configuración del entorno y los casos de desarrollo

Contenido

1. **Marco de desarrollo**
   - Arquitectura.
     - Características principales
2. Desarrollo y aplicación

Arquitectura: Fácil desarrollo y ejecución eficiente



ME (Mind Expression): capa de interfaz (Python)

Usabilidad: programación diferencial automática y expresión matemática original

- Auto differ: diferencial automático a nivel de operador
- Automático paralelo: paralelismo automático
- Tensor automático: generación automática de operadores
- Etiquetado semiautomático: etiquetado de datos semiautomático

GE (Motor de gráficos): capa de compilación y ejecución de gráficos

Alto rendimiento: Cooptimización de hardware/software, y aplicación de escenario completo

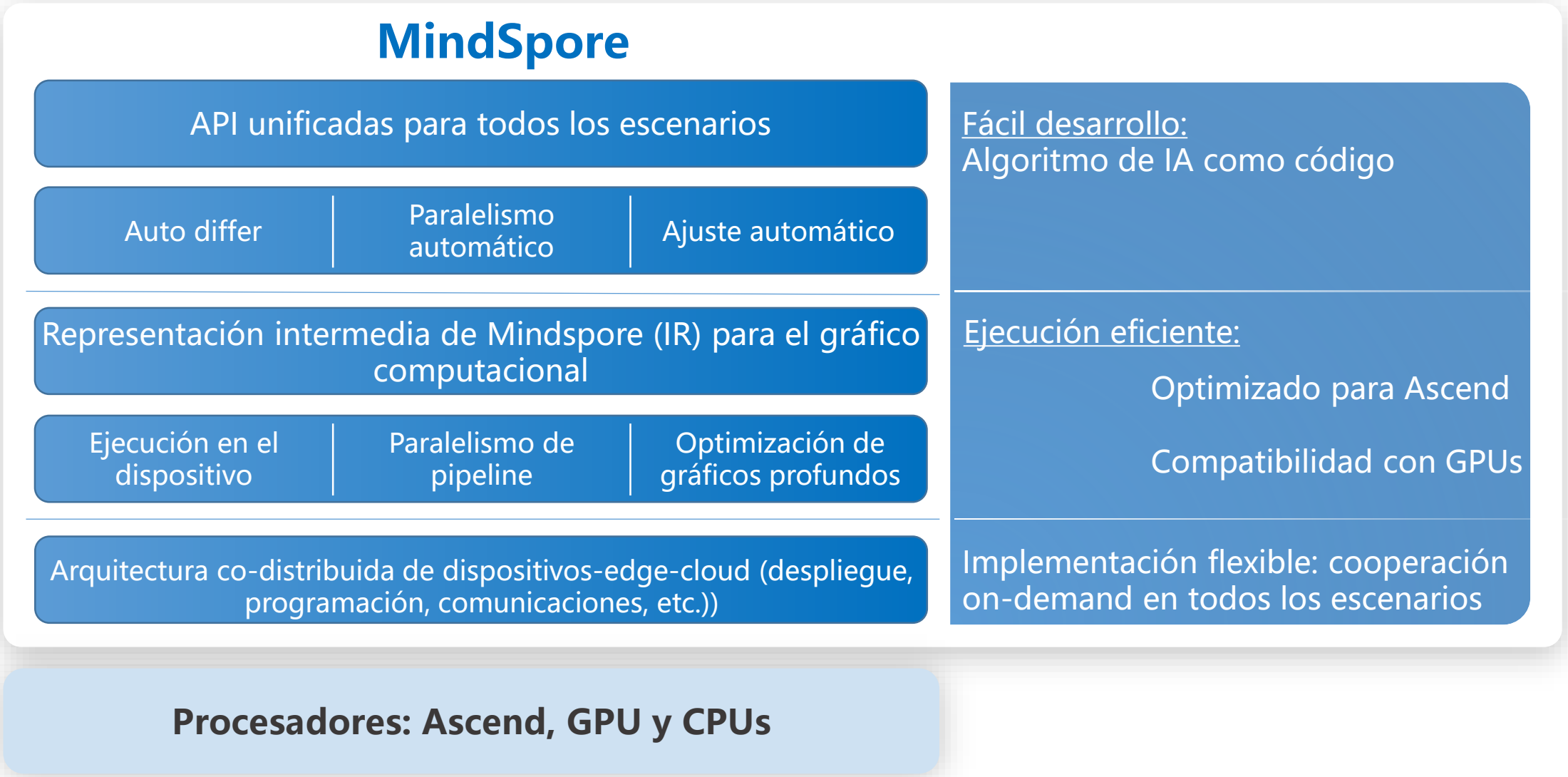
- Cross-layer overcommitment en memoria
- Optimización de graficas profundas
- Ejecución en el dispositivo
- Sinergia entre dispositivos y nubes (incluida la compilación en línea)

① Equivalente a los marcos de código abierto de la industria, Mindspore sirve preferentemente chips autodesarrollados y servicios en la nube.

② Soporta la interconexión ascendente con marcos de terceros y puede interconectarse con ecosistemas de terceros a través de Graph IR, incluyendo modelos de referencia y de formación. Los desarrolladores pueden ampliar la capacidad de Mindspore.

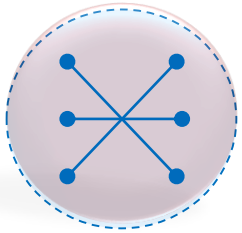
③ También es compatible con la interconexión con chips de terceros y ayuda a los desarrolladores a aumentar los escenarios de aplicación de Mindspore y ampliar el ecosistema de IA.

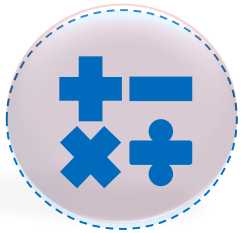
Solución general: Arquitectura Core



Diseño Mindspore: Auto Differ

Trayectoria técnica de diferencial automático





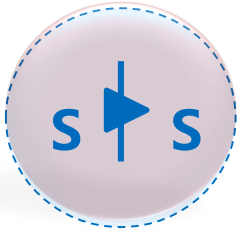


Gráfico: Tensorflow

- Programación no basada en Python basada en gráficos
- Representación compleja de flujos de control y derivados de orden superior

Sobrecarga del operador: Pytorch

- Encabezado de tiempo de ejecución
- Es difícil optimizar el rendimiento del proceso hacia atrás.

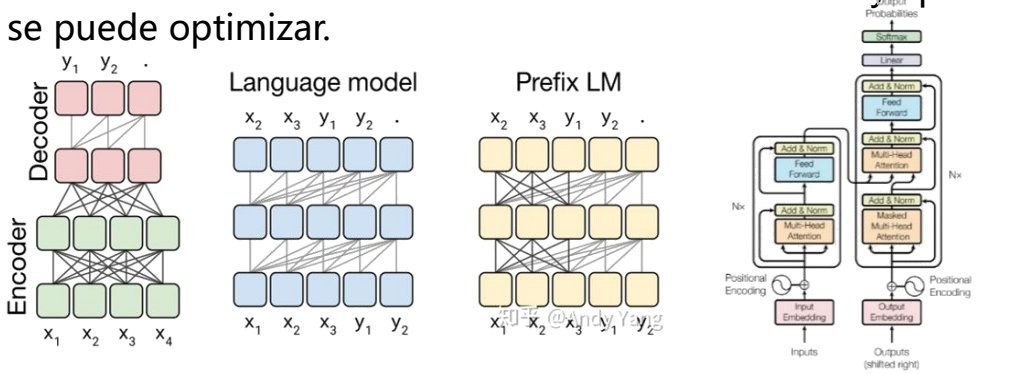
Transferencia de código fuente: Mindspore.

- API de Python para una mayor eficiencia
- Optimización de la compilación basada en IRS para obtener un mejor rendimiento

Paralelismo automático

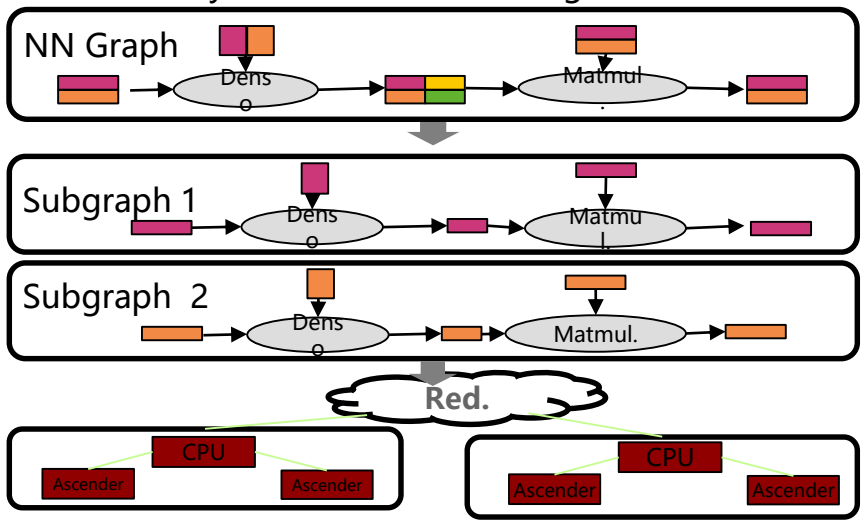
Desafíos

Los modelos ultragrandes realizan una capacitación distribuida eficiente:
A medida que los modelos de dominio NLP se hinchan, la sobrecarga de memoria para entrenar a modelos ultragrandes como Bert (340M)/GPT-2(1542M) ha excedido la capacidad de una sola tarjeta. Por lo tanto, los modelos deben dividirse en varias tarjetas antes de la ejecución.
Actualmente se utiliza el paralelismo de modelo manual. Es necesario diseñar la segmentación del modelo y comprender la topología del clúster. El desarrollo es extremadamente difícil. El rendimiento es mediocre y apenas se puede optimizar.



Tecnologías clave

Segmentación automática del gráfico: Puede segmentar todo el gráfico basándose en las dimensiones de datos de entrada y salida del operador e integrar el paralelismo de datos y modelos.
Programación de la conciencia de topología de clúster: puede percibir la topología de clúster, programar subgráficos automáticamente y minimizar la sobrecarga de comunicación.



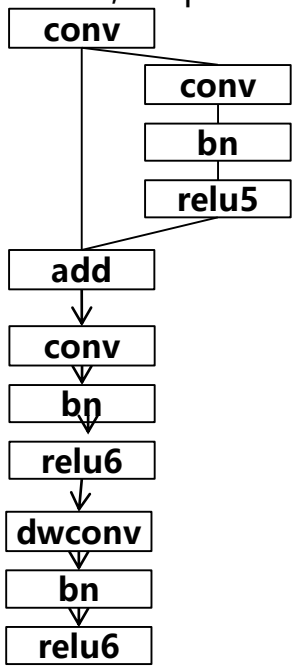
Efecto: Permite realizar el paralelismo del modelo basado en la lógica de código de nodo único existente, lo que mejora la eficiencia del desarrollo en diez veces en comparación con el paralelismo manual.

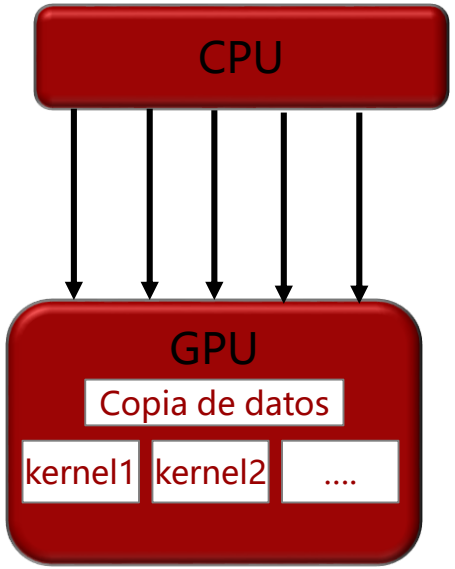
Ejecución en el dispositivo (1)

Desafíos

Desafíos para la ejecución de modelos con una potencia informática de chip suprema:

Pared de memoria, sobrecarga de interacción alta y dificultad de suministro de datos. Las operaciones parciales se realizan en el host, mientras que las demás se realizan en el dispositivo. La sobrecarga de interacción es mucho mayor que la sobrecarga de ejecución, lo que resulta en un bajo uso del acelerador.

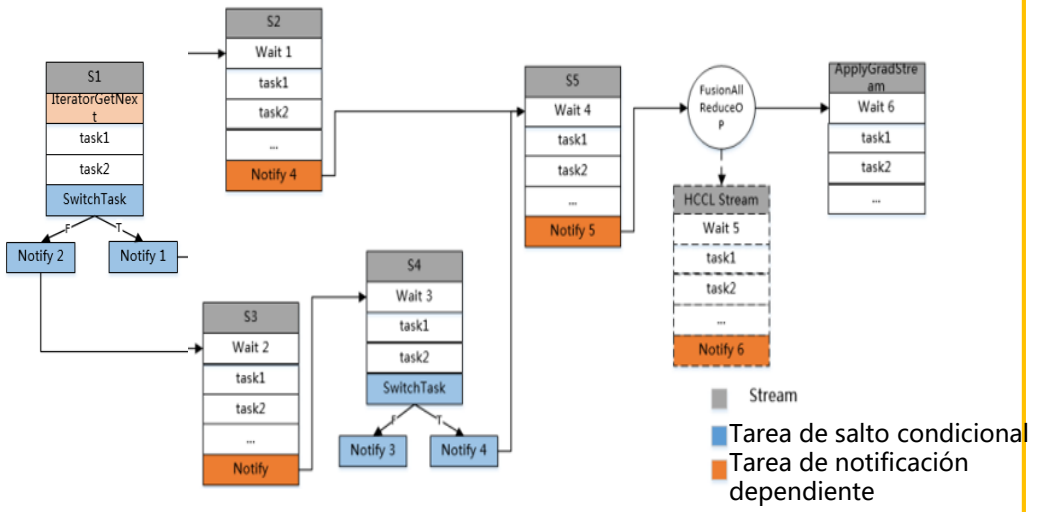




Gran carga de interacción de datos y difícil suministro de datos

Tecnologías clave

La optimización de gráficos profundos orientada a chips reduce el tiempo de espera de sincronización y maximiza el paralelismo de los datos, la informática y la comunicación. El preprocesamiento de datos y el cálculo se integran en el chip Ascend:



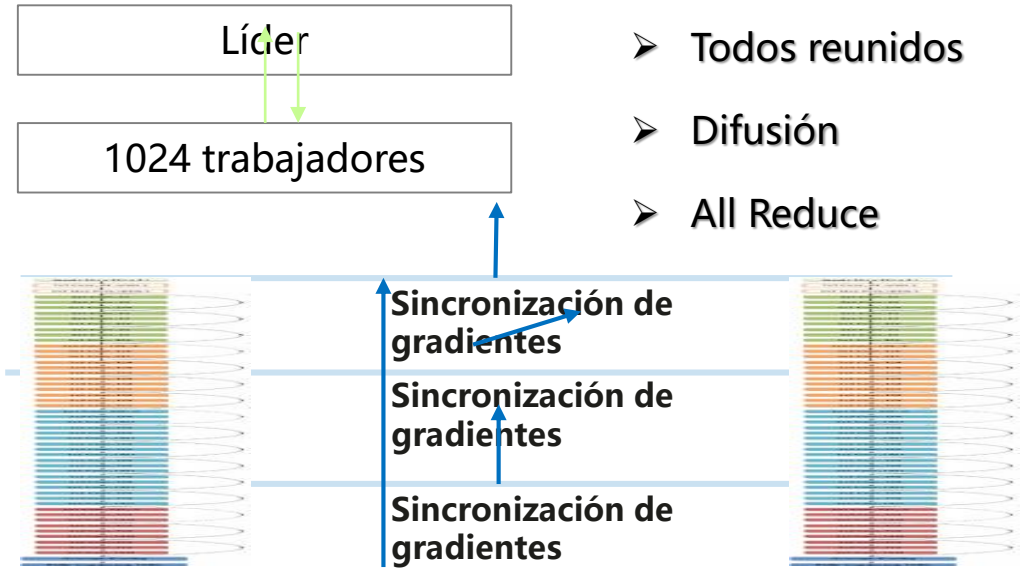
Efecto: Eleve el rendimiento del entrenamiento diez veces más que la programación gráfica en el host.

HUAWEI

Ejecución en el dispositivo (2)

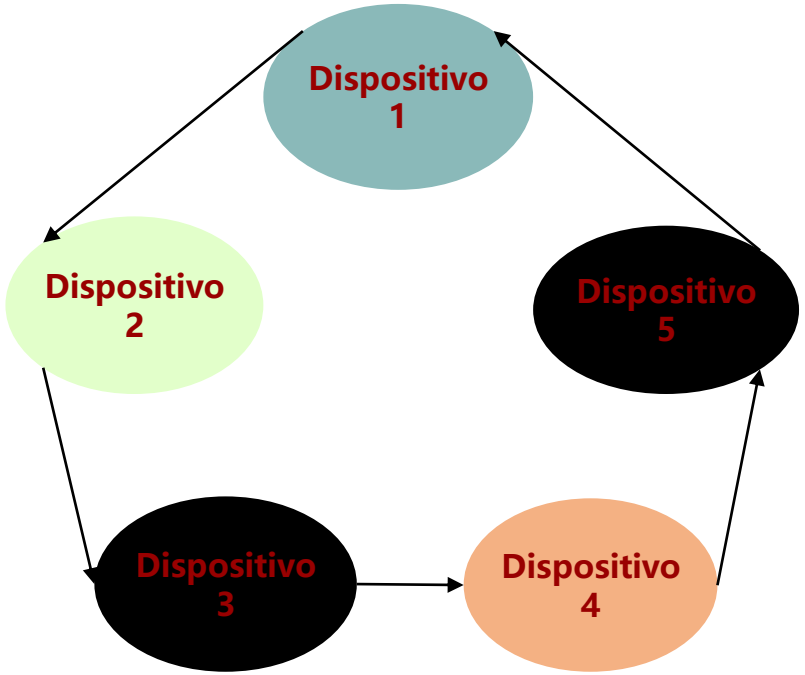
Desafíos

Desafíos para la agregación de gradientes distribuidos con una potencia computacional de chip suprema:
la sobrecarga de sincronización del control central y la sobrecarga de comunicación de sincronización frecuente de Resnet50 bajo la iteración única de 20 ms; el método tradicional sólo puede completar All Reduce después de tres veces de sincronización, mientras que el método de datos puede realizar de forma autónoma All Reduce sin causar sobrecarga de control.



Tecnologías clave

La optimización de la **segmentación gráfica adaptativa impulsada por los datos de gradiente** puede llevar a cabo la descentralización de All Reduce y sincronizar la agregación de gradientes, impulsando la eficiencia informática y de la comunicación.



Efecto: una sobrecarga difuminante de menos de 2 ms

Arquitectura de sinergia distribuida de Dispositivos-Edge-Cloud

Desafíos

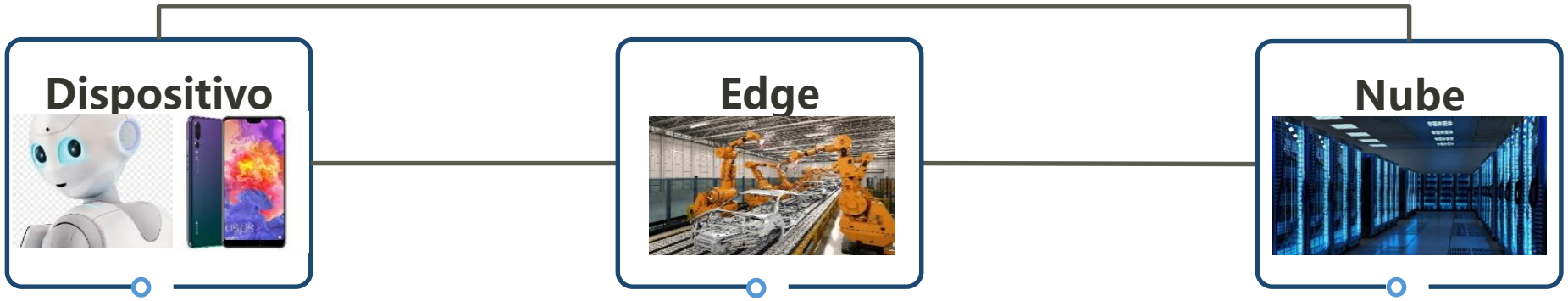
La diversidad de arquitecturas de hardware genera diferencias de implementación en escenarios completos e incertidumbres de rendimiento. La separación de la formación y la inferencia conduce al aislamiento de los modelos.

Tecnologías clave

- **El modelo unificado de IR** ofrece una experiencia de implementación coherente.
- **La tecnología de optimización de gráficos con colaboración de software y hardware** tiende puentes entre diferentes escenarios.
- Synergy Federal Meta Learning de dispositivo-nube, rompe el límite de dispositivos-nube y actualiza el modelo de colaboración multidispositivo en tiempo real.

Efecto: rendimiento de implementación de modelos consistente en todos los escenarios gracias a la arquitectura unificada y a la precisión mejorada de los modelos personalizados

Colaboración on-demand en todos los escenarios y experiencia de desarrollo consistente



HUAWEI

Contenido

1. **Marco de desarrollo**
   - Arquitectura.
   - Características principales
2. Desarrollo y aplicación

El marco de la computación en IA: retos

Desafíos del sector

Una gran brecha entre

Investigación industrial y aplicación de IA en todos los escenarios

- Grandes barreras de entrada
- Alto costo de ejecución
- Larga duración del despliegue

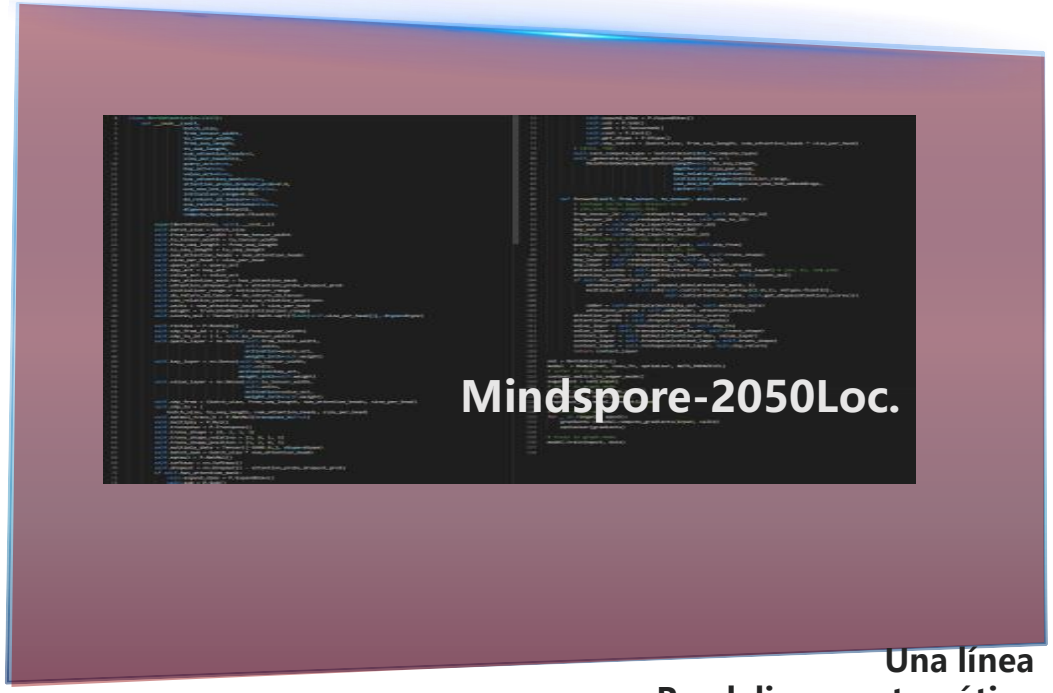
Innovación tecnológica

Mindspore facilita la IA inclusiva en todas las aplicaciones

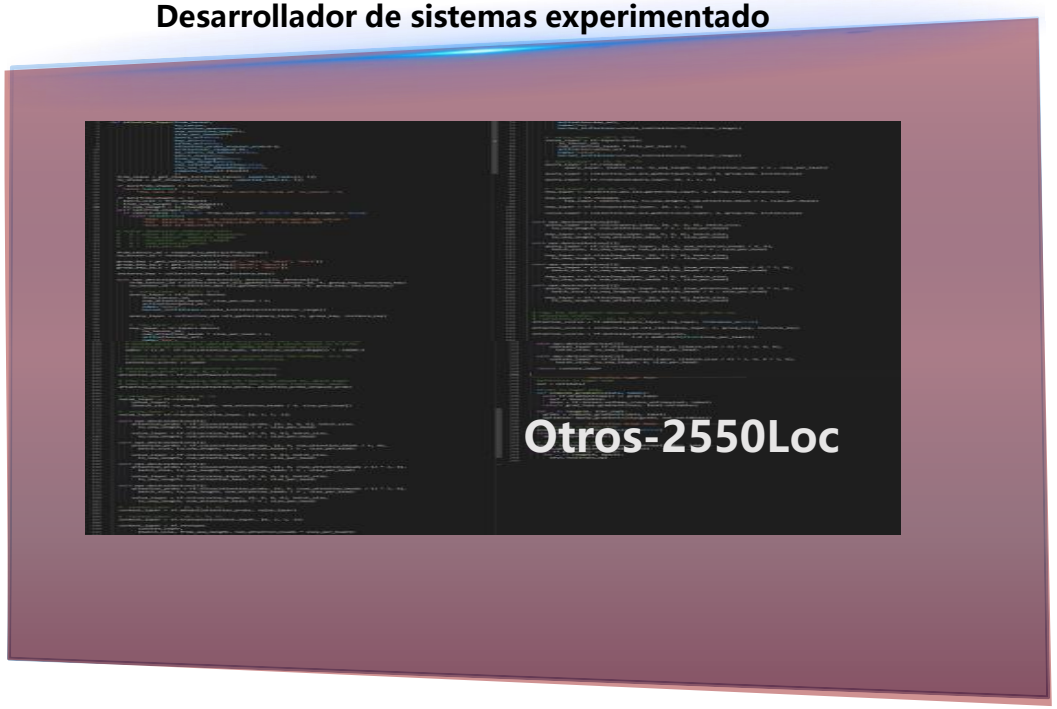
- Nuevo modo de programación
- Nuevo modo de ejecución
- Nuevo modo de colaboración

Nuevo paradigma de programación

Algoritmo científico



Algoritmo científico
+.
Desarrollador de sistemas experimentado



Una línea
Paralelismo automático

Diferencial automático eficiente

Conmutador de modo de depuración en una línea

Modelo de NLP: Transformer

Ejemplo de código

Fragmento de código de tensorflow: líneas xx, paralelismo manual

```
import tensorflow as tf

model() {
    with tf.device("/device:0")
        token_type_table = tf.get_variable(
            name=token_type_embedding_name,
        shape=[token_type_vocab_size, width],
        initializer=create_initializer(initializer_range))
        flat_token_type_ids = tf.reshape(token_type_ids, [-1])
        one_hot_ids = tf.one_hot(flat_token_type_ids, depth=token_type_vocab_size)
        token_type_embeddings = tf.matmul(one_hot_ids, token_type_table)

    with tf.device("/device:1")
        query_layer = tf.layers.dense(
            from_tensor_2d,
            num_attention_heads * size_per_head,
            activation=query_act,
            name="query",
            kernel_initializer=create_initializer(initializer_range))

    with tf.device("/device:2")
        key_layer = tf.layers.dense(
            to_tensor_2d,
            num_attention_heads * size_per_head,
            activation=key_act,
            name="key",
            kernel_initializer=create_initializer(initializer_range))
```

Fragmento de código de Mindspore: dos líneas, paralelismo automático

```
class DenseMatMulNet(nn.Cell):
    def __init__(self):
        super(DenseMutMulNet, self).__init__()
        self.matmul1 = ops.MatMul.set_strategy({[4, 1], [1, 1]})
        self.matmul2 = ops.MatMul.set_strategy({[1, 1], [1, 4]})
    def construct(self, x, w, v):
        y = self.matmul1(x, w)
        z = self.matmul2(y, v)
        return s
```

Escenarios típicos: ReID

Nuevo modo de ejecución (1)

Desafíos de ejecución

Informática compleja de IA y diversas unidades informáticas

1. Núcleos de CPU, cubos y vectores
2. Cálculo escalar, vector y tensor
3. Cálculo de precisión mixta
4. Cálculo de matrices densas y matrices dispersas

Ejecución de varios dispositivos: alto coste del control en paralelo

El rendimiento no puede aumentar linealmente a medida que aumenta la cantidad de nodos.

Ejecución en el dispositivo

Descarga gráficos a dispositivos, maximizando la potencia de cálculo de Ascend

Optimización del marco

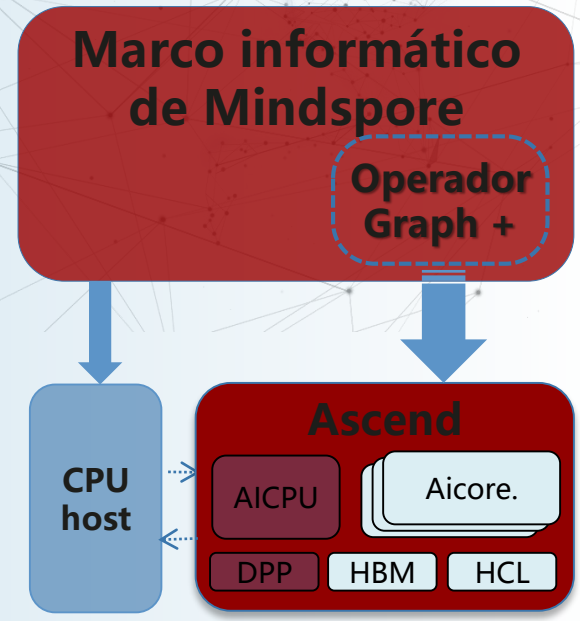
paralelismo de pipeline

Exceso de compromiso de memoria entre capas

Cooptimización de hardware/software

Ejecución en el dispositivo

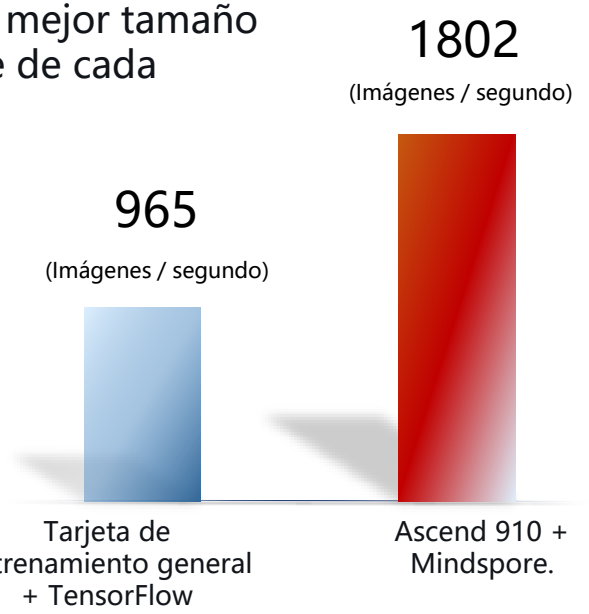
Optimización de gráficos profundos



HUAWEI

Nuevo modo de ejecución (2)

- ResNet 50 V1.5
- ImageNet 2012
- Con el mejor tamaño de lote de cada tarjeta



El rendimiento de ResNet-50 se duplica.
Una iteración:
58 ms (otros frameworks + V 100) v.s. unos 22 ms (Mindspore) (ResNet50 + ImageNet, un solo servidor, ocho dispositivos, tamaño de lote = 32)

Detección de objetos en 60 ms

Seguimiento de objetos en 5 ms

Reconocimiento multiobjeto en tiempo real
Implementación móvil basada en Mindspore, una experiencia sin problemas de detección de varios objetos

Nuevo modo de colaboración

Desafío de implementación

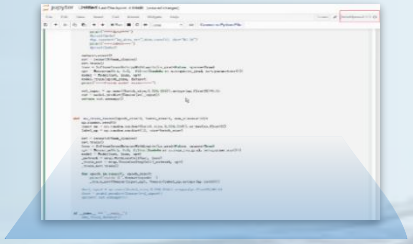
v.s.

- Diversos requisitos, objetivos y restricciones para los escenarios de aplicaciones de dispositivos, edge y nube

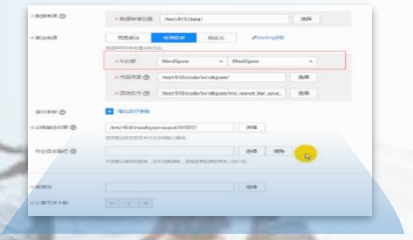
v.s.

- Distintas precisiones y velocidades de hardware

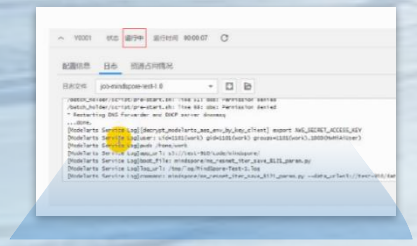
Desarrollo unificado; despliegue flexible; colaboración bajo demanda y alta seguridad y confiabilidad



Desarrollo



Despliegue



Ejecución

Guardando modelo

Desarrollo unificado e implementación flexible

Alto Desempeño

Los retos de la informática de IA

Informática compleja

Cálculo escalar, vector y tensor

Informática de precisión mixta

Paralelismo entre agregación de gradientes y mini-batch computing

Diversas unidades de cálculo/procesadores

CPU, GPU y procesadores Ascend

Ecalar, vector y tensor

Mindspore

Optimización de framework

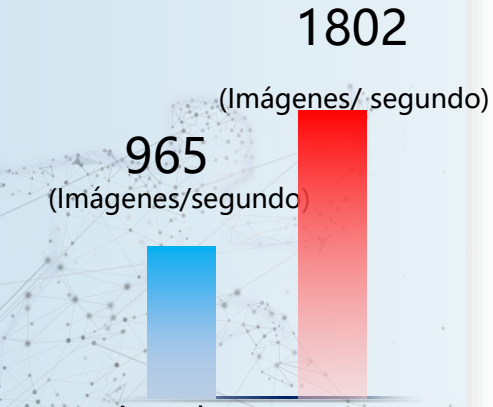
Paralelismo de pipeline

Exceso de compromiso de memoria entre capas

Software + Hardware co-optimización

Ejecución en el dispositivo

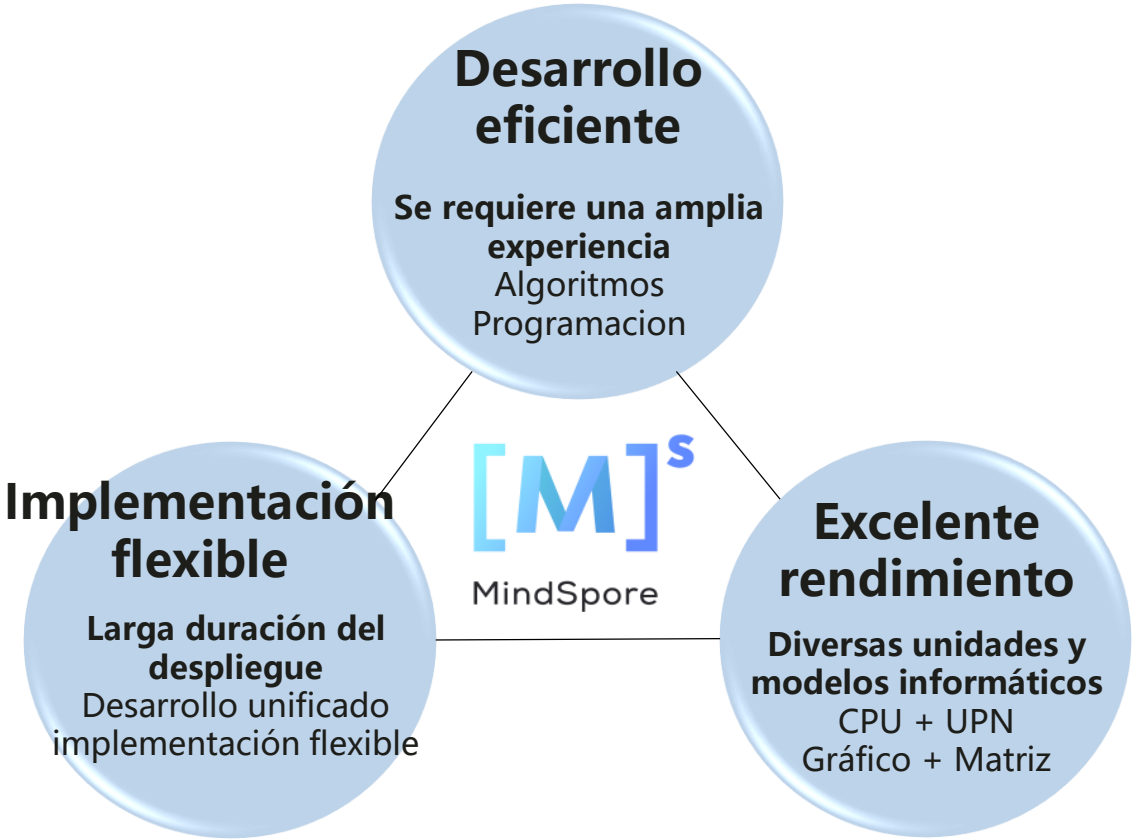
Optimización de gráficos profundos



- RNet 50 V1.5
- ImageNet 2012
- Basado en tamaños de lote óptimos

HUAWEI

Visión y valor



Contenido

1. Marco de desarrollo
2. **Desarrollo y aplicación**
   - Configuración del entorno
   - Casos de desarrollo de aplicaciones

Instalación de Mindspore

Environment Requirements

System Requirements and Software Dependencies

Version	Operating System	Executable File Installation Dependencies	Source Code Compilation and Installation Dependencies
MindInsight 0.2.0-alpha	- Ubuntu 16.04 or later x86_64 - EulerOS 2.8 arrch64 - EulerOS 2.5 x86_64	- Python 3.7.5 - MindSpore 0.2.0-alpha - For details about other dependency items, see requirements.txt.	**Compilation dependencies:** - Python 3.7.5 - CMake >= 3.14.1 - GCC 7.3.0 - node.js >= 10.19.0 - wheel >= 0.32.0 - pybind11 >= 2.4.3 **Installation dependencies:** same as the executable file installation dependencies.

- When the network is connected, dependency items in the requirements.txt file are automatically downloaded during .whl package installation. In other cases, you need to manually install dependency items.

Installation Guide

Installing Using Executable Files

1. Download the .whl package from the MindSpore website. It is recommended to perform SHA-256 integrity verification first and run the following command to install MindInsight:

```
pip install mindinsight-{version}-cp37-cp37m-linux_{arch}.whl
```

2. Run the following command. If `web address: http://127.0.0.1:8080` is displayed, the installation is successful.

```
mindinsight start
```

Método 1: Compilación e instalación de Código Fuente

Dos entornos de instalación: Ascend y CPU

```
adding 'mindspore/transforms/validators.py'
adding 'mindspore-0.1.0.dist-info/METADATA'
adding 'mindspore-0.1.0.dist-info/WHEEL'
adding 'mindspore-0.1.0.dist-info/top_level.txt'
adding 'mindspore-0.1.0.dist-info/RECORD'
removing build/bdist.linux-x86_64/wheel
------Successfully created mindspore package------
---------------- mindspore: build test end ----------------
```

Método 2: instalación directa mediante el paquete de instalación

Dos entornos de instalación: Ascend y CPU

Comandos de instalación:

1. pip install –y mindspore-cpu
2. pip install –y mindspore-d

Introducción

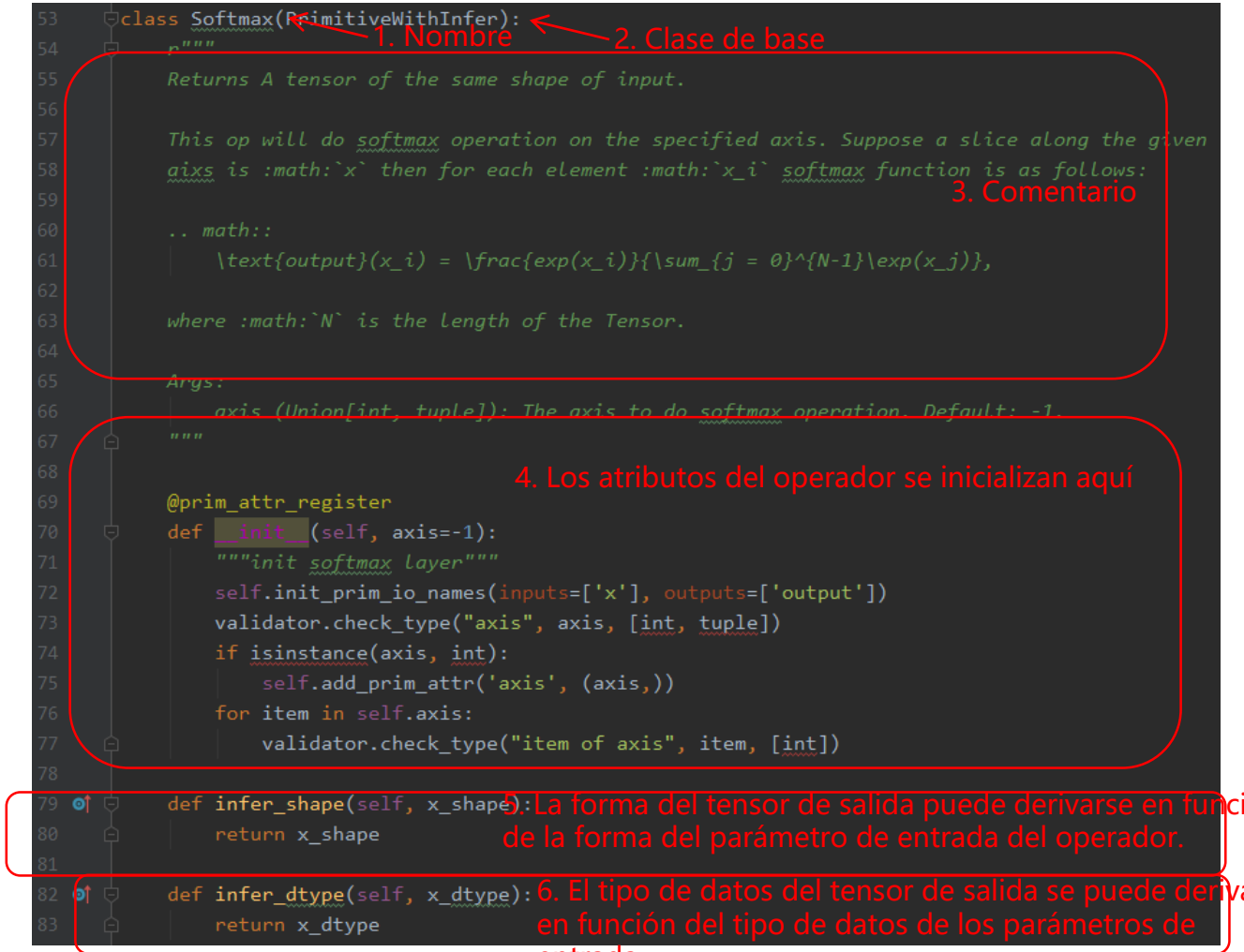
- En Mindspore, los datos se almacenan en tensores. Operaciones comunes del tensor:
  - asnumpy()
  - size()
  - dim()
  - dtype()
  - set_dtype()
  - tensor_add(other: Tensor)
  - tensor_mul(other: Tensor)
  - shape()
  - __Str__# (conversion a cadenas)

Módulo	Descripciùn
Model_zoo	Define modelos de red comunes
communication	Módulo de carga de datos, que define el dataloader y dataset y procesa datos como imágenes y textos.
dataset	Módulo de procesamiento de conjuntos de datos, que lee y procesa datos.
common	Define tensor, parámetro, dtype e inicializador.
contextr	Define la clase de contexto y establece los parámetros de ejecución del modelo, como los modos de conmutación de gráficos y PyNative.
akg	Biblioteca automática de operadores diferenciales y personalizados.
nn	Define células Mindspore (unidades de red neural), funciones de pérdida y optimizadores.
ops	Define los operadores básicos y registra los operadores inversos.
traing	Modelo de entrenamiento y módulos de funciones de resumen.
utils	Utilidades que verifican los parámetros. Este parámetro se utiliza en el framework.

Componentes de ME

Concepto de programación: Operación

Operador Softmax




```
class Softmax(PrimitiveWithInfer):
    """
    Returns A tensor of the same shape of input.

    This op will do softmax operation on the specified axis. Suppose a slice along the given
    aixs is :math:`x` then for each element :math:`x_i` softmax function is as follows:

    .. math::
        \text{output}(x_i) = \frac{exp(x_i)}{\sum_{j = 0}^{N-1}\exp(x_j)},

    where :math:`N` is the length of the Tensor.

    Args:
        axis (Union[int, tuple]): The axis to do softmax operation. Default: -1.
    """

    @prim_attr_register
    def __init__(self, axis=-1):
        """init softmax layer"""
        self.init_prim_io_names(inputs=['x'], outputs=['output'])
        validator.check_type("axis", axis, [int, tuple])
        if isinstance(axis, int):
            self.add_prim_attr('axis', (axis,))
        for item in self.axis:
            validator.check_type("item of axis", item, [int])

    def infer_shape(self, x_shape):
        return x_shape

    def infer_dtype(self, x_dtype):
        return x_dtype
```

Operaciones communes en MindSpore:

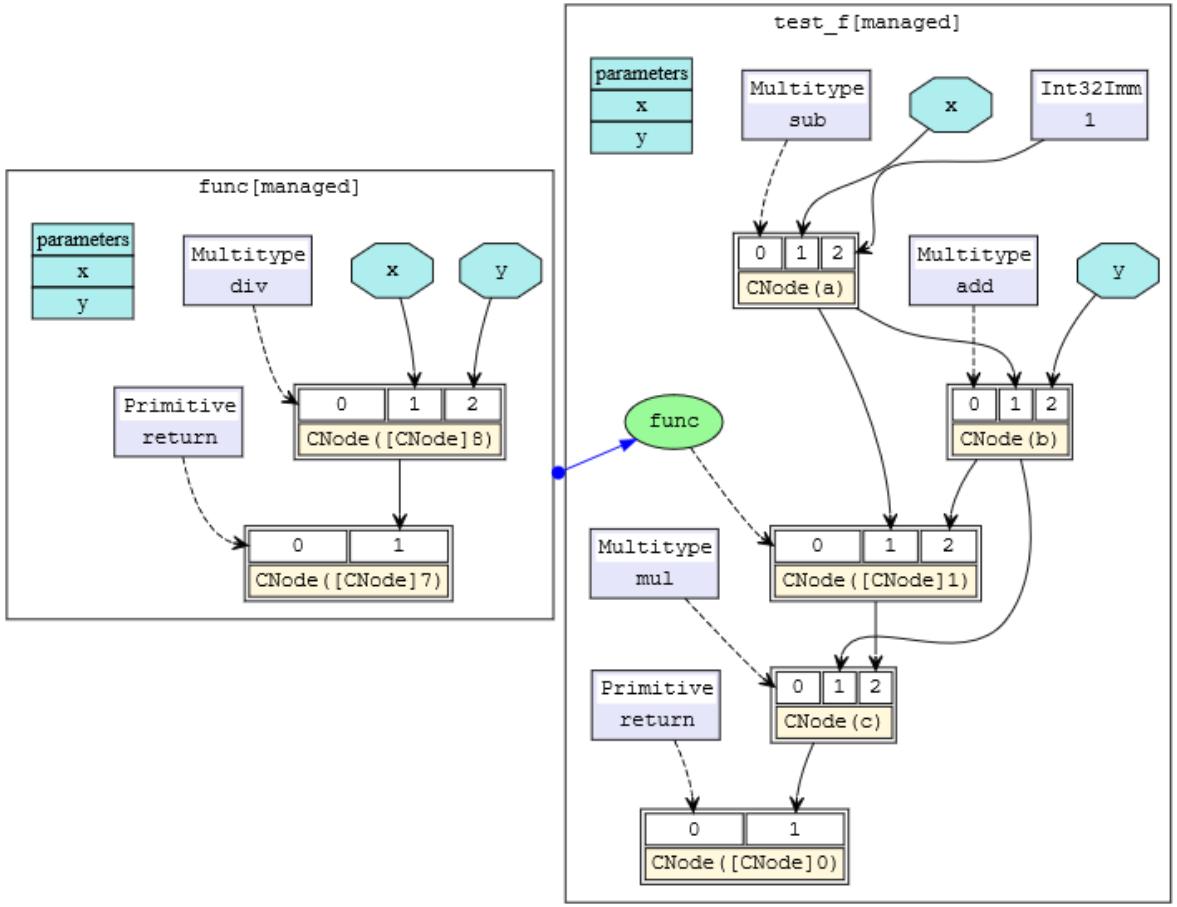
- **array: Array-related operators**
  - **ExpandDims**
  - **Concat**
  - **Select**
  - **ScatterNd...**
  - **Squeeze**
  - **OnesLike**
  - **StridedSlice**
- **math: Math-related operators**
  - **AddN**
  - **Sub**
  - **Mul**
  - **MatMul**
  - **RealDiv**
  - **ReduceMean**
  - **Cos**
  - **Sin**
  - **LogicalAnd**
  - **LogicalNot**
  - **Less**
  - **Greater...**
- **nn: Network operators**
  - **Conv2D**
  - **Flatten**
  - **Softmax**
  - **ReLU**
  - **Sigmoid**
  - **Pooling**
  - **BatchNorm**
  - **MaxPool**
  - **AvgPool**
  - **TopK**
  - **SoftmaxCrossEntropy**
  - **SmoothL1Loss**
  - **SGD**
  - **SigmoidCrossEntropy...**
- **control: Control operators**
  - **ControlDepend**
- **random: Random operators**

Concepto de programación: Cell (celda)

- Una cell define el módulo básico para el cálculo. Los objetos de la cell se pueden ejecutar directamente.
  - __init__: Inicia y verifica módulos como parámetros, celdas y primitivas.
  - Construcción: define el proceso de ejecución. En el modo gráfico, se compila un gráfico para su ejecución y está sujeto a restricciones de sintaxis específicas.
  - bprop (opcional): Es la dirección inversa de los módulos personalizados. Si esta función no está definida, se utiliza el diferencial automático para calcular la reversa de la parte de construcción.
- Las cell predefinidas en MindSpore incluyen principalmente: pérdida común (Softmax Cross Entropy With Logits and MSELoss), optimizadores comunes (Momentum, SGD, y Adam), y funciones comunes de embalaje de red, como el cálculo y actualización del gradiente de la red TrainOneStepCell y el cálculo del gradiente WithGradCell.

Concepto de programación: MindSporeIR

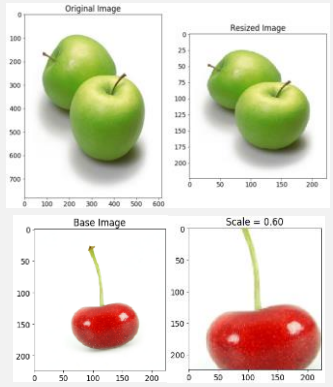
- MindSporeIR es un IR funcional basado en gráficos compacto, eficiente y flexible que puede representar la semántica funcional como variables libres, funciones de orden superior y recursividad. Es un portador de programas en proceso de optimización de compilación y AD.
- Cada gráfico representa un gráfico de definición de función y consta de ParameterNode, ValueNode y ComplexNode (CNode).
- La figura muestra la relación def-use.



Caso de desarrollo

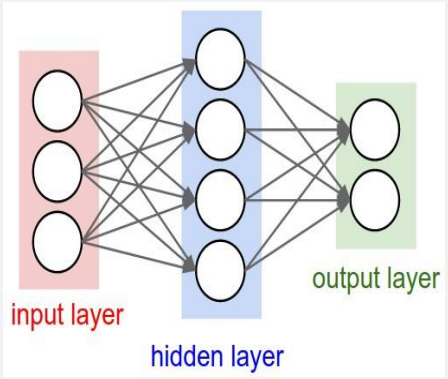
- Tomemos el reconocimiento de los dígitos manuscritos de MNIST como ejemplo para demostrar el proceso de modelado en MindSpore.

Datos



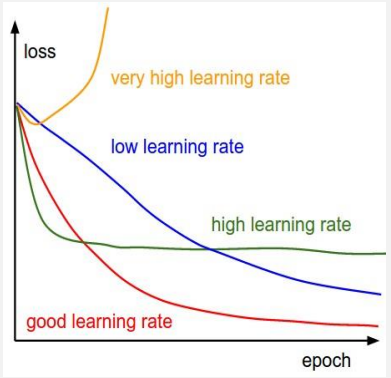
- **1. Carga de datos**
- **2. Mejora de los datos**

Red



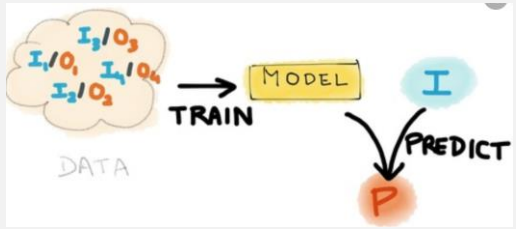
- **3. Definición de la red**
- **4. Inicialización del peso**
- **5. Ejecución de la red**

Modelo



- **6. Función de pérdida**
- **7. Optimizador**
- **8. Iteración de entrenamiento**
- **9. Evaluación de modelos**

Aplicación



- **10. Ahorro de modelos**
- **11. Predicción de carga**
- **12. Ajuste fino**

HUAWEI

Quiz

1. En Mindspore, ¿cuál de las siguientes operaciones es el tipo de nn? (1)
   A. Matemático
   B. Red
   C. Control
   D. Otros

Resumen

- Este capítulo describe el marco, el diseño, las funciones y el proceso de configuración del entorno y el procedimiento de desarrollo de Mindspore.

Más información

TensorFlow: https://www.tensorflow.org/

PyTorch: https://pytorch.org/

Mindspore: https://www.mindspore.cn/en

Comunidad de desarrolladores – Ascend : http://122.112.148.247/home

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

HUAWEI

Plataforma de computación Atlas AI

HUAWEI

PREFACIO

- Este capítulo describe los chips de IA Ascend de Huawei y las arquitecturas de hardware y software de los chips Ascend y las soluciones de full-stack, todo-escenario de los chips de IA Ascend.

Objetivos

Al finalizar el curso, podrás:

- Obtener una visión general de los chips de IA.
- Comprender las arquitecturas de hardware y software de los chips Ascend de Huawei.
- Conozca la plataforma de computación de Huawei, Atlas AI.
- Comprender las aplicaciones industriales de Atlas.

Contenido

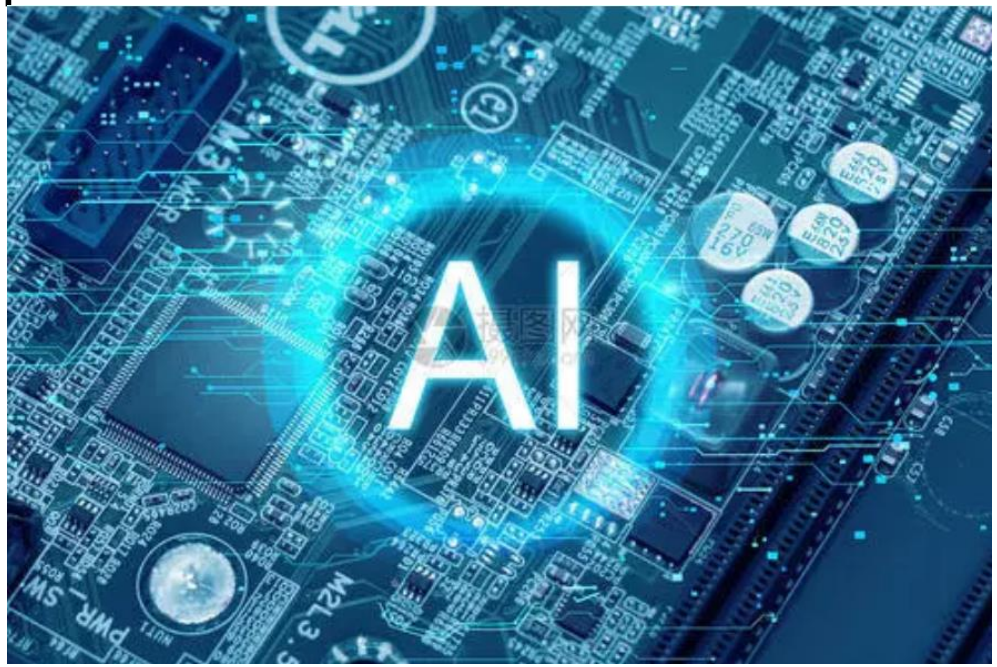
1. **Visión general de los chips de IA**
   - Resumen
     - Clasificación de los chips de IA
     - Situación actual de los chips de IA
     - Comparación de diseño GPU, TPU y Ascend 310
     - Procesadores de IA Ascend
2. Arquitectura de hardware de los Chips Ascend
3. ...

Visión general y objetivos

- Esta sección ofrece una visión general de los chips de IA, incluyendo la introducción, clasificación y estado de los chips de IA, la comparación entre las GPU y las CPU y la introducción de los procesadores de IA Ascend.

Definición

- Cuatro elementos de la IA: datos, algoritmo, escenario y potencia informática (computing power)
- Los chips de IA, también conocidos como aceleradores de IA, son módulos de funciones que procesan tareas informáticas masivas en aplicaciones de IA.



Contenido

1. **Visión general de los chips de IA**
   - Resumen
   - Clasificación de los chips de IA
   - Situación actual de chips de IA
   - Comparación de diseño de GPU y CPU
   - Procesadores de IA Ascend
2. Arquitectura de hardware de chips Ascend
3. ...

Clasificación de chips de IA (1)

- Los chips AI se pueden dividir en cuatro tipos por su arquitectura técnica:
  - Una unidad central de procesamiento (CPU): un circuito integrado a super-gran-escala, que es el núcleo de computación y la unidad de control de un ordenador. Puede interpretar las instrucciones de la computadora y procesar los datos del software de la computadora.
  - Una unidad de procesamiento gráfico (GPU): un núcleo de pantalla, un procesador visual y un chip de pantalla. Es un microprocesador que procesa imágenes en computadoras personales, estaciones de trabajo, consolas de juegos y dispositivos móviles, como tablets y teléfonos inteligentes.
  - Un circuito integrado específico de una aplicación (ASIC): un circuito integrado diseñado para un propósito específico.
  - Una matriz de puertas programable de campo (FPGA): diseñada para implementar funciones de un chip semi-personalizado. La estructura del hardware puede configurarse y cambiarse en tiempo real de acuerdo a los requerimientos.

Clasificación de chips de IA (2)

- Los chips de IA se pueden dividir en formación e inferencia por aplicación empresarial.
  - Durante la fase de formación, es necesario capacitar a un complejo modelo de red neuronal profunda a través de un gran número de entradas de datos o de un método de aprendizaje no supervisado como el aprendizaje mejorado. El proceso de formación requiere datos masivos de formación y una compleja estructura de red neural profunda. La enorme cantidad de computación requiere un rendimiento ultra-high incluyendo potencia de computación, precisión y escalabilidad de los procesadores. El clúster de GPU de Nvidia y las TPU de Google se utilizan comúnmente en el entrenamiento de IA.
  - Las inferencias se hacen utilizando modelos entrenados y nuevos datos. Por ejemplo, un dispositivo de videovigilancia utiliza el modelo de red neural profunda de fondo para reconocer una cara capturada. A pesar de que la cantidad de cálculo de la inferencia es mucho menor que la del entrenamiento, se trata de un gran número de operaciones matriz. GPU, FPGA y ASIC también se utilizan en el proceso de inferencia.

Contenido

1. **Visión general de los chips de IA**
   - Resumen
   - Clasificación de los chips de IA
   - Situación actual de chips de IA
   - Comparación de diseño de GPU y CPU
   - Procesadores de IA Ascend
2. Arquitectura de hardware de chips Ascend
3. ...

Situación actual de los chips IA - CPU

- Unidad central de procesamiento (CPU)
  - El rendimiento de la computadora ha mejorado constantemente con base en la Ley de Moore.
  - Los núcleos de CPU añadidos para mejorar el rendimiento también aumentan el consumo de energía y el costo.
  - Se han introducido instrucciones adicionales y se ha modificado la arquitectura para mejorar el rendimiento de la IA.
    - Instrucciones, como AVX512, se han introducido en los procesadores Intel (arquitectura de CISC) y módulos de computación vectorial, como FMA, en el módulo de computación ALU.
    - Se han introducido conjuntos de instrucciones incluyendo Cortex A en ARM (arquitectura RISC), que se actualizará continuamente.
  - A pesar de que el aumento de la frecuencia del procesador puede elevar el rendimiento, la alta frecuencia causará un enorme consumo de energía y sobrecalentamiento del chip a medida que la frecuencia alcanza el techo.

Estado actual de los chips AI - GPU

- Unidad de procesamiento de gráficos (GPU)
  - GPU realiza notablemente en la computación matricial y en la computación paralela y desempeña un papel clave en la computación heterogénea. Fue introducido por primera vez en el campo de la IA como un chip de aceleración para el aprendizaje profundo. Actualmente, el ecosistema de la GPU ha madurado.
  - Con la arquitectura GPU, NVIDIA se centra en los dos aspectos siguientes del aprendizaje profundo:
    - Diversificación del ecosistema: ha puesto en marcha la biblioteca de optimización cuDNN para redes neuronales para mejorar la usabilidad y optimizar la arquitectura subyacente de GPU.
    - Mejorar la personalización: Soporta varios tipos de datos, incluyendo int8 además de float32; introduce módulos dedicados al aprendizaje profundo. Por ejemplo, se ha introducido la arquitectura optimizada de los núcleos Tensor, como el TensorCore de V100.
  - Los problemas existentes incluyen altos costes y latencia y baja eficiencia energética.

Situación actual de chips AI - TPU

- Unidad de Procesamiento de Tensores (TPU)
  - Desde 2006, Google ha tratado de aplicar el concepto de diseño de ASICs al campo de la red neuronal y ha lanzado TPU, un chip AI personalizado que soporta TensorFlow, que es un marco de código abierto para aprendizaje profundo.
  - Las matrices sistólicas masivas y el almacenamiento en chip de gran capacidad se adoptan para acelerar las operaciones de convolución más comunes en redes neurales profundas.
    - Las matrices sistólicas optimizan las operaciones de multiplicación y convolución de matriz para aumentar la potencia de computación y reducir el consumo de energía.

Situación actual de los chips de IA - FPGA

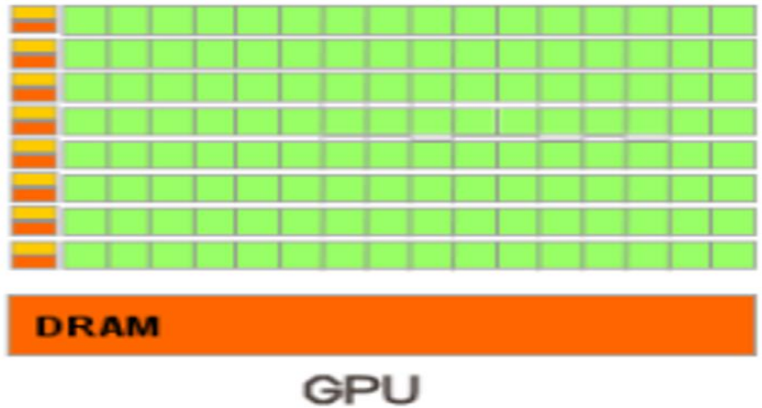
- Matriz de puertas programable de campo (FPGA)
  - Con el modo programable HDL, los FPGA son muy flexibles, reconfigurables y reprogramables, y personalizables.
  - Múltiples FPGA se pueden utilizar para cargar el modelo DNN en los chips para reducir la latencia de computación. Los FPGA superan a las GPU en términos de rendimiento informático. Sin embargo, el rendimiento óptimo no se puede lograr debido al borrado y programación continuos. Además de los transistores y cables redundantes, los circuitos lógicos con las mismas funciones ocupan un área de chip más grande.
  - La estructura reconfigurable reduce los riesgos de la oferta y la I+D. El coste es relativamente flexible dependiendo de la cantidad de compra.
  - Los procesos de diseño y grabación se desacoplaron. El período de desarrollo es largo, generalmente medio año. La barrera de entrada es alta.

Contenido

1. **Resumen de los chips de IA**
   - Resumen:
   - Clasificación de los chips de IA
   - Estado actual de los chips de IA
   - Comparación de diseño de GPU y CPU
   - Procesadores de IA Ascend
2. Arquitectura de hardware de los chips Ascend
3. Arquitectura de software de los chips Ascend
4. ....

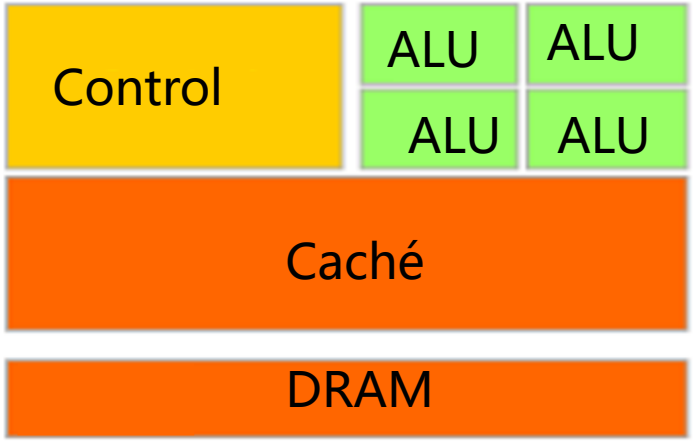
Comparación de diseño de GPU y CPU

- Las GPU están diseñadas para datos masivos del mismo tipo independientes entre sí y entornos informáticos puros que no necesitan ser interrumpidos.
  - Cada GPU comprende varias arquitecturas informáticas paralelas de gran tamaño con miles de núcleos más pequeños diseñados para manejar múltiples tareas simultáneamente.
  - Diseño orientado a rendimiento
    - Con muchas ALUs (Unidad Lógica Aritmética) y pocas cachés, lo que mejora los servicios para los hilos, a diferencia de los de la CPU. La caché fusiona el acceso a DRAM, lo que causa latencia.
    - La unidad de control realiza el acceso combinado.
    - Un gran número de ALUs procesan numerosos subprocesos al mismo tiempo para cubrir la latencia.
  - Especializado en programas informáticos intensivos y fáciles de usar en paralelo



Comparación de diseño de GPU y CPU

- Las CPU deben procesar diferentes tipos de datos de manera universal, realizar el juicio lógico e introducir saltos de ramificación masivos y procesamiento interrumpido.
  - Compuesta de varios núcleos optimizados para el procesamiento en serie secuencial
  - Diseño de baja latencia
    - La potente unidad ALU puede completar el cálculo en un ciclo de reloj corto.
    - La caché grande reduce la latencia.
    - Alta frecuencia de reloj
    - Unidad de control lógica compleja, los programas de varias sucursales pueden reducir la latencia a través de la predicción de ramas (branch prediction).
    - Para instrucciones que dependen del resultado de la instrucción anterior, la unidad lógica determina la ubicación de las instrucciones en la tubería (pipeline) para acelerar el reenvío de datos.
  - Especializado en el control lógico y el funcionamiento en serie



Contenido

1. **Aspectos generales de los chips de IA**
   - Resumen:
   - Clasificación de los chips de IA
   - Estado actual de los chips de IA
   - Comparación de diseño de GPU y CPU
   - Procesadores de IA Ascend
2. Arquitectura de hardware de los chips Ascend
3. Arquitectura de software de los chips Ascend
4. ....

Procesadores de IA Ascend

- Unidad de procesamiento de red neural UPN (Neural Network Processing Unit NPU): utiliza un conjunto de instrucciones de aprendizaje profundo para procesar un gran número de neuronas humanas y sinapsis simuladas en la capa de circuitos. Una instrucción se utiliza para procesar un grupo de neuronas.
- NPU típicas:Cchips de IA Ascend de Huawei, chips Cambricon e IBM TrueNorth

- Ascend-Mini.
- Arquitectura: Da Vinci
- Media precisión (FP16): 8 TERAFLOPS
- Precisión entera (INT 8): 16 Tera-OPS
- 16-channel full-HD video decoder: H.264/H.265
- 1-channel full-HD video decoder: H.264/H.265
- **Potencia Max.: 8 W**
- CFF de 12 nm

- Ascend-Max
- Arquitectura: Da Vinci
- Media precisión (FP16): 256 TERAFLOPS
- Precisión entera (INT 8): 512 Tera-OPS
- 128-channel full-HD video decoder: H.264/H.265
- Potencia Max. : 350 W
- 7 nm

Resumen de la sección

- Esta sección describe los chips de IA, incluyendo la clasificación de chips de IA por tecnologías y funciones, el ecosistema de chips de IA y la comparación entre GPU y CPU.

Contenido

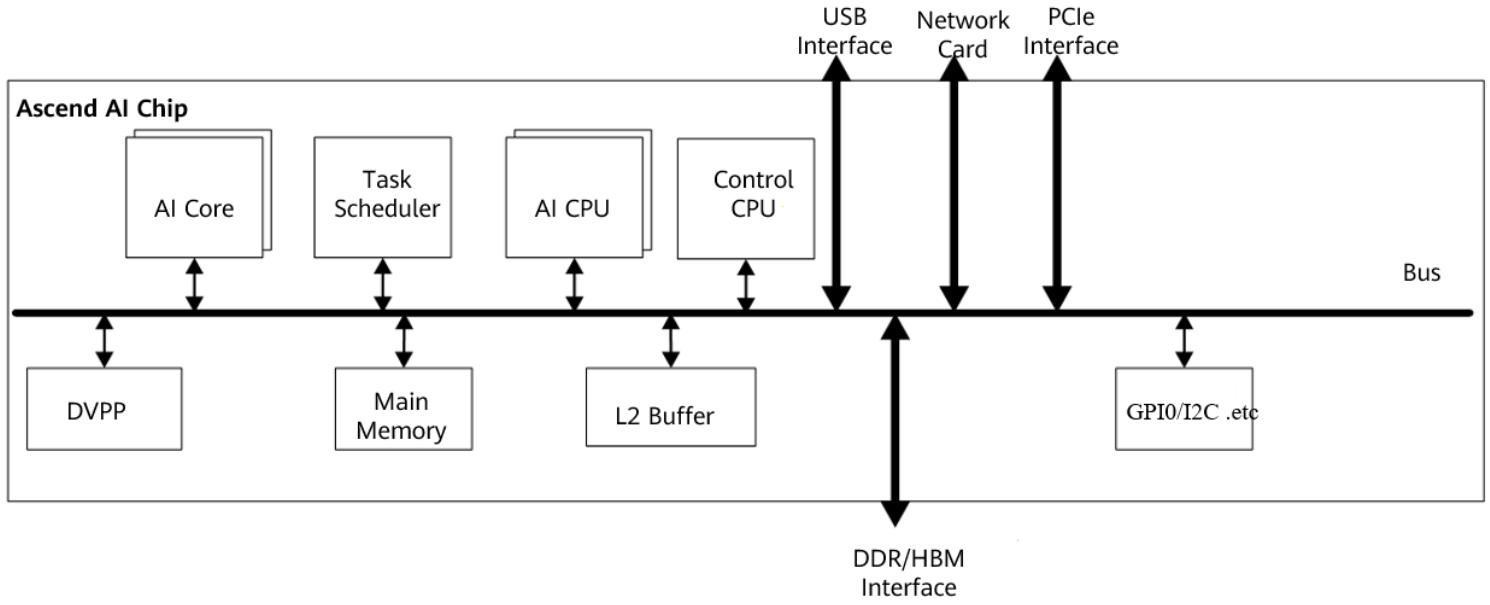
1. Aspectos generales de los chips de IA
2. **Arquitectura de hardware de los chips Ascend**
   - Arquitectura lógica de los procesadores Ascend AI
     - Arquitectura Da Vinci
3. Arquitectura de software de los chips Ascend
4. Plataforma informática Huawei Atlas AI
5. Aplicaciones industriales de Atlas

Visión general y objetivos

- Esta sección describe la arquitectura de hardware de los chips Ascend, incluyendo la arquitectura lógica de los procesadores Ascend Al y la arquitectura Da Vinci.

Arquitectura lógica de los procesadores Ascend AI

- El procesador Ascend AI consta de:
  - Control de CPU
  - Motor de computación de IA, incluyendo núcleo de IA y CPU de IA
  - Cachés o búferes de sistema en chip (SoC) de capas multiples
  - Módulo de preprocesamiento de visión digital (DVPP)



Contenido

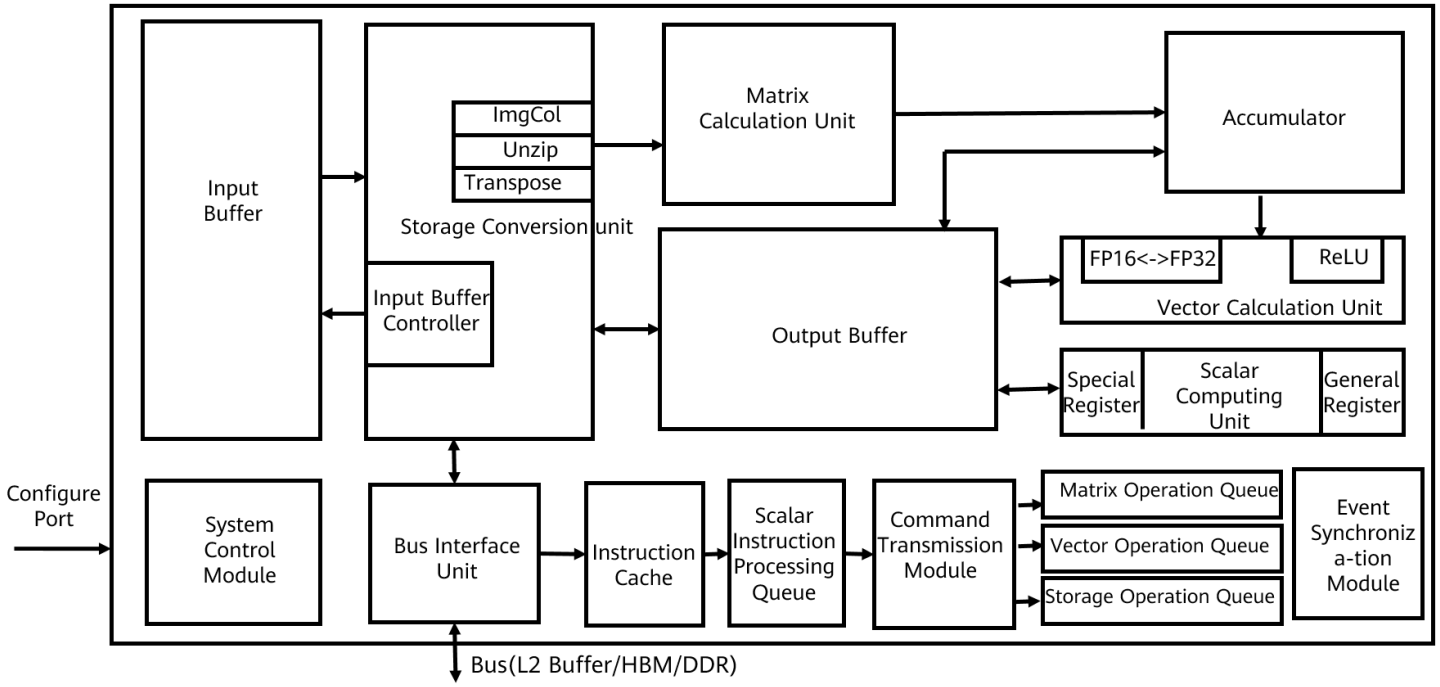
1. Aspectos generales de los chips de IA
2. **Arquitectura de hardware de los chips Ascend**
   - Arquitectura lógica de los procesadores Ascend AI
   - Arquitectura Da Vinci
3. Arquitectura de software de los chips Ascend
4. Plataforma informática Huawei Atlas AI
5. Aplicaciones industriales de Atlas

Ascend AI Computing Engine - Da Vinci Architecture

- Una de las cuatro arquitecturas principales de los procesadores Ascend es el motor de computación de IA, que consiste en el núcleo de IA (arquitectura Da Vinci) y la unidad central (CPU) de IA. La arquitectura Da Vinci desarrollada para mejorar la potencia de computación de IA sirve como núcleo del motor de computación de IA Ascend y del procesador de IA.

Arquitectura Da Vinci (núcleo de IA)

- Componentes principales de la arquitectura Da Vinci:
  - Unidad de cálculo: consta de la unidad de cubo, la unidad vectorial y la unidad escalar.
  - Sistema de almacenamiento: consiste en la unidad de almacenamiento en el chip del núcleo de IA y los canales de datos.
  - La unidad de control proporciona el control de instrucciones para todo el proceso de computación. Es equivalente al centro de comando del núcleo de IA y es responsable del funcionamiento de todo el núcleo de IA.



Arquitectura Da Vinci (Core AI) - Unidad de cálculo

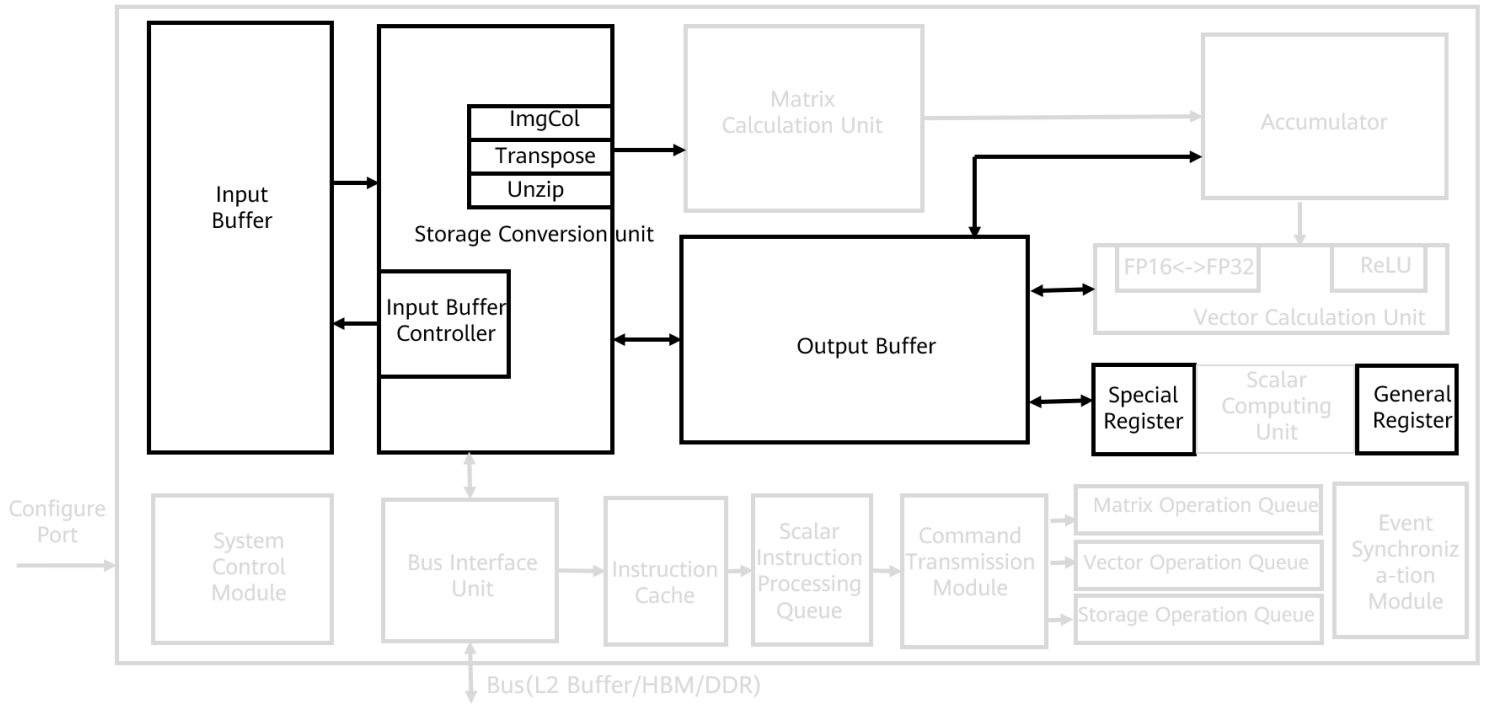
- Tres tipos de unidades de computación básicas: cubo, vector y unidades escalares, que corresponden a los modos de computación de matriz, vector y escalar respectivamente.
- Cube computing unit: La unidad de computación de matriz y el acumulador se utilizan para realizar operaciones relacionadas con matrices. Completa una matriz (4096) de 16x16 multiplicada por 16x16 para FP16, o una matriz (8192) de 16x32 multiplicada por 32x16 para la entrada INT8 en una toma.
- Unidad de computación de vectores: Implementa computación entre vectores y escalares o entre vectores. Esta función cubre varios tipos de computación básica y muchos tipos de computación personalizados, incluyendo computación de tipos de datos como FP16, FP32, INT32 y INT8.
- Unidad de computación escalar: Equivalente a una micro CPU, la unidad escalar controla el funcionamiento de todo el núcleo AI. Implementa el control de bucles y el juicio de ramas para todo el programa, y proporciona el cálculo de direcciones de datos y parámetros relacionados para cubos o vectores, así como operaciones aritméticas básicas.

Arquitectura Da Vinci (Núcelo IA) - Sistema de Almacenamiento (1)

- El sistema de almacenamiento del núcleo de IA está compuesto por la unidad de almacenamiento y el canal de datos correspondiente.
- La unidad de almacenamiento consta de la unidad de control de almacenamiento, buffer y registros:
- Unidad de control de almacenamiento: La caché a un nivel inferior al núcleo AI se puede acceder directamente a través de la interfaz del bus. La memoria también se puede acceder directamente a través del DDR o HBM. Una unidad de conversión de almacenamiento se configura como controlador de transmisión del canal de datos interno del núcleo AI para implementar la gestión de lectura/escritura de datos internos del Núcleo AI entre diferentes búferes. También completa una serie de operaciones de conversión de formato, como cero padding, Img2Col, transposing y descompresión.
- Buffer de entrada: El buffer almacena temporalmente los datos que se necesitan utilizar con frecuencia para que los datos no necesiten ser leídos desde el núcleo AI a través de la interfaz de bus cada vez. Este modo reduce la frecuencia de acceso a datos en el bus y el riesgo de congestión del mismo, reduciendo así el consumo de energía y mejorando el rendimiento.
- Buffer de salida: El buffer almacena los resultados intermedios de la computación en cada capa de la red neuronal, de modo que los datos se pueden obtener fácilmente para la computación de la siguiente capa. La lectura de datos a través del bus implica un bajo ancho de banda y una larga latencia, mientras que el uso del buffer de salida mejora considerablemente la eficiencia de la computación.
- Registro: Varios registros en el núcleo de IA son utilizados principalmente por la unidad escalar.

Arquitectura Da Vinci (Core AI) - Sistema de almacenamiento (2)

- Canal de datos: ruta para el flujo de datos en el núcleo de la IA durante la ejecución de las tareas de computación
  - Un canal de datos de la arquitectura Da Vinci se caracteriza por una entrada múltiple y una salida única. Considerando diversos tipos y una gran cantidad de datos de entrada en el proceso de computación en la red neuronal, las entradas paralelas pueden mejorar la eficiencia de entrada de datos. Por el contrario, sólo se genera una matriz de características de salida después de procesar varios tipos de datos de entrada. El canal de datos con una sola salida de datos reduce el uso de recursos de hardware de chip.

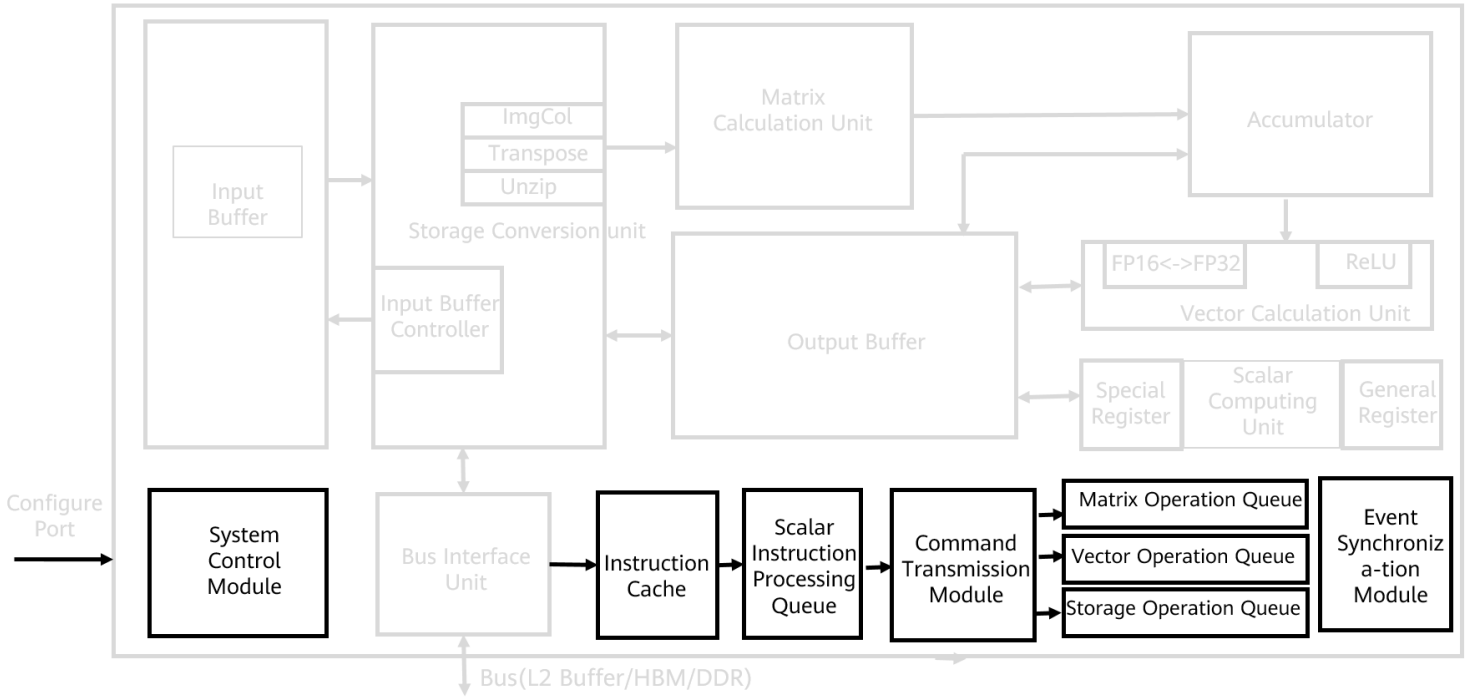


Arquitectura Da Vinci (Núcleo IA) - Unidad de Control (1)

- La unidad de control está compuesta por el módulo de control del sistema, la caché de instrucciones, la cola escalar de procesamiento de instrucciones, el módulo de transmisión de instrucciones, la cola de operaciones de matrices y la de vectores, la cola de conversión de almacenamiento y el módulo de sincronización de eventos.
  - Módulo de control del sistema: Controla el proceso de ejecución de un bloque de tareas (granularidad informática de tareas mínima para el núcleo de IA) - Una vez ejecutado el bloque de tareas, el módulo de control del sistema procesa la interrupción y reporta el estado. Si se produce un error durante la ejecución, el estado de error se informa al programador de tareas.
  - Caché de instrucciones: Obtiene las instrucciones subsiguientes de antemano durante la ejecución de las instrucciones y lee varias instrucciones en la caché al mismo tiempo, lo que mejora la eficiencia de la ejecución de las instrucciones.
  - Cola de procesión de instrucciones escalares: Después de ser decodificadas, las instrucciones se importan a una cola escalar para implementar la decodificación de direcciones y el control de operaciones. Las instrucciones incluyen instrucciones de cálculo matricial, instrucciones de cálculo vectorial e instrucciones de conversión de almacenamiento.
  - Módulo de transmisión de instrucciones: Lee las direcciones de instrucciones configuradas y los parámetros decodificados en la cola de instrucciones escalares y los envía a la cola de ejecución de instrucciones correspondiente según el tipo de instrucción. Las instrucciones escalares residen en la cola de procesamiento de instrucciones escalares para su ejecución posterior.

Arquitectura Da Vinci (Core IA) - Unidad de Control (2)

- Cola de ejecución de instrucciones: Incluye una cola de operación matricial, una cola de operación vectorial y una cola de conversión de almacenamiento. Diferentes instrucciones ingresan las colas de operación correspondientes, y las instrucciones en las colas se ejecutan de acuerdo con la secuencia de entrada.
- Módulo de sincronización de eventos: Controla el estado de ejecución de cada una de las tuberías de instrucción en tiempo real y analiza las relaciones de dependencia entre las diferentes tuberías para resolver problemas de dependencia de datos y sincronización entre las tuberías de instrucción.



Resumen de la sección

- Esta sección describe la arquitectura de hardware de los chips Ascend, incluyendo la unidad de computación, la unidad de almacenamiento y la unidad de control de la arquitectura principal Da Vinci.

Contenido

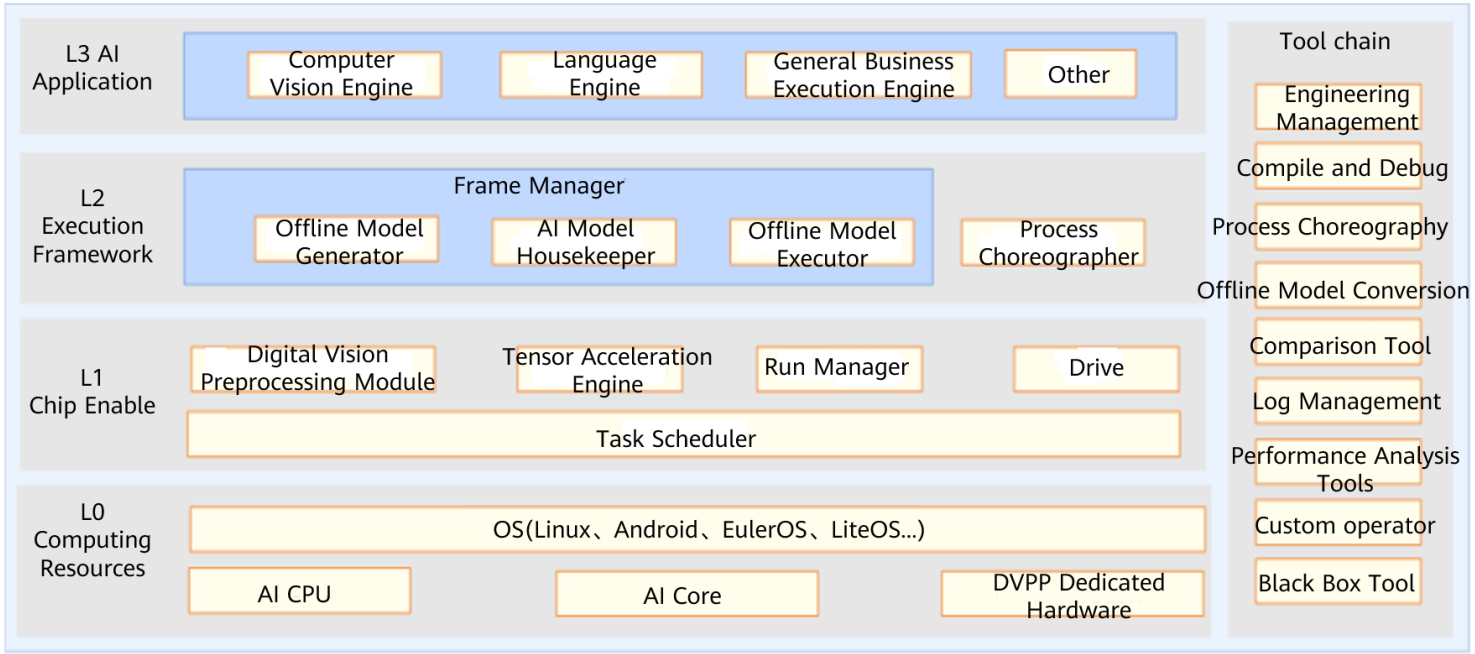
1. Panorámica de los chips de IA
2. Arquitectura de hardware de los chips Ascend
3. **Arquitectura de software de los chips Ascend**
   - ■ Arquitectura lógica de Ascend 310
     - ▫ Flujo de software de red neuronal de Ascend 310
     - ▫ Diagrama de flujo de datos de Ascend 310
4. Plataforma de computación de AI Atlas de Huawei
5. Aplicaciones industriales de Atlas

Visión general y objetivos

- Esta sección describe la arquitectura de software de los chips Ascend, incluyendo la arquitectura lógica y el flujo de software de red neuronal de los procesadores Ascend AI.

Arquitectura lógica de Ascend AI Processor Software Stack (1)

- Capa activadora de aplicación L3: Es una capa de encapsulación a nivel de aplicación que proporciona diferentes algoritmos de procesamiento para campos de aplicación específicos. L3 proporciona varios campos con motores de computación y procesamiento. Puede utilizar directamente la capacidad de programación de frameworks proporcionada por L2 para generar redes neuronales (Neural Networks) NNs correspondientes e implementar funciones específicas de motor.
  - Motor genérico: proporciona la capacidad genérica de inferencia de red neural.
  - Motor de visión de computadora: encapsula algoritmos de procesamiento de vídeo o de imagen.
  - Motor de lenguaje y texto: encapsula los algoritmos de procesamiento básicos para datos de voz y texto.



Arquitectura lógica de la pila de software del procesador Ascend AI (2)

- Capa de estructura de ejecución L2: encapsula la capacidad de llamada de la estructura y la capacidad de generación de modelos sin conexión. Una vez desarrollado y encapsulado el algoritmo de aplicación en un motor de capa 3, capa 2 llama al marco de aprendizaje profundo apropiado, como Cafe o TensorFlow, basado en las características del algoritmo para obtener la red neural de la función correspondiente. y genera un modelo offline a través del framework manager. Una vez que la capa 2 convierte el modelo de red neural original en un modelo sin conexión que se puede ejecutar en chips de IA Ascend, el ejecutor del modelo sin conexión transfiere el modelo sin conexión a la capa 1 para la asignación de tareas.
- L1 chip enabling layer : conecta el modelo sin conexión a chips Ascend. La capa 1 acelera el modelo offline para diferentes tareas de computación a través de bibliotecas. Más cerca de los recursos informáticos de capa inferior, la capa 1 muestra las tareas de la capa de operador al hardware.
- L0 computing resource layer : proporciona recursos de computación y ejecuta tareas de computación específicas. Es la base de computación de hardware del chip Ascend.

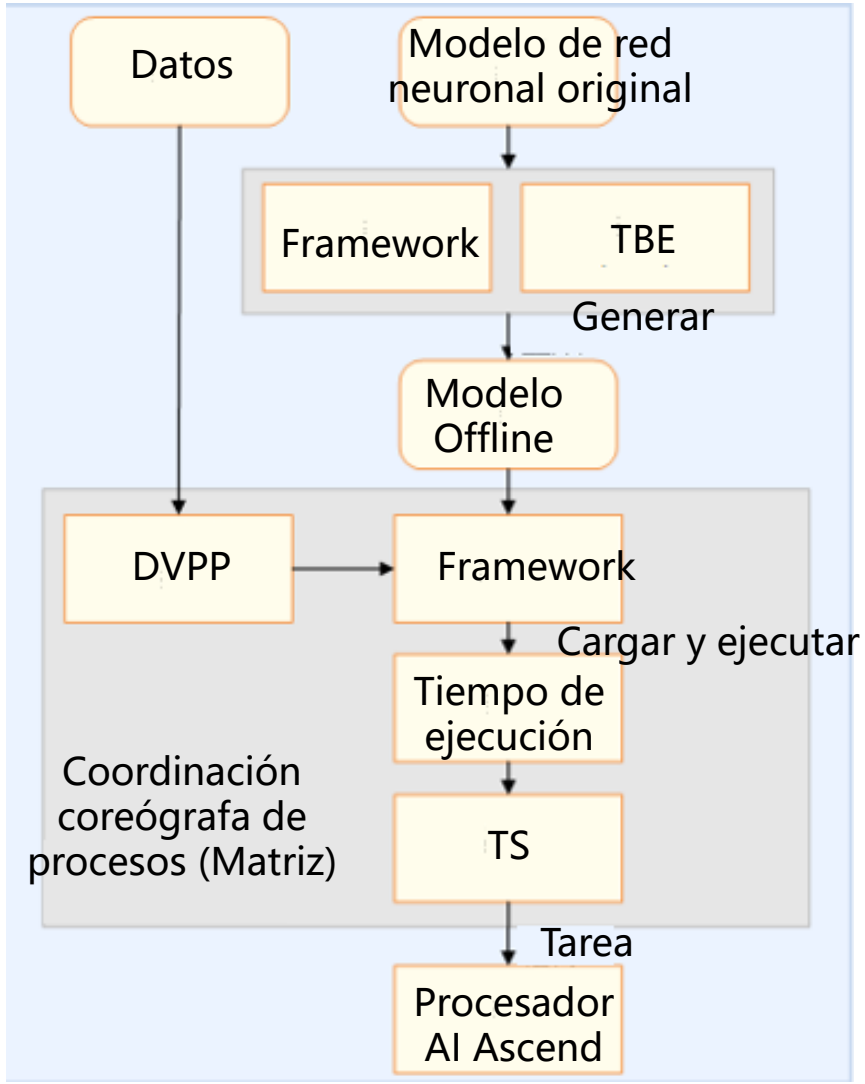
Contenido

1. Resumen de los chips de IA
2. Arquitectura de hardware de los chips Ascend
3. **Arquitectura de software de los chips Ascend**
   - Arquitectura lógica de Ascend 310
   - Flujo de software de red neuronal de Ascend 310
   - Diagrama de flujo de datos de Ascend 310
4. Plataforma de computación AI Atlas de Huawei
5. Aplicaciones industriales de Atlas

Flujo de software de red neuronal de procesadores Ascend IA

- El flujo de software de la red neuronal de los procesadores Ascend AI es un puente entre el marco (framework) de aprendizaje profundo y los chips Ascend AI. Realiza y ejecuta una aplicación de red neuronal e integra los siguientes módulos funcionales.
- Orquestador de procesos: implementa la red neuronal en chips Ascend AI, coordina todo el proceso de realización de la red neuronal y controla la carga y ejecución de modelos offline.
- Modulo de preprocesamiento de visión digital (DVPP): realiza el procesamiento y limpieza de datos antes de la entrada para satisfacer los requisitos de formato para la computación.
- Motor de impulsión de tensores (TBE): funciona como una fábrica de operadores de redes neuronales que proporciona potentes operadores de computación para modelos de redes neuronales.
- Framework manager: construye un modelo de red neuronal original en una forma soportada por los chips de Ascend AI, e integra el nuevo modelo en los chips de Ascend AI para asegurar el funcionamiento eficiente de la red neuronal.
- Runtime Manager: proporciona varias rutas de gestión de recursos para la entrega de tareas y asignación de la red neural.
- Planificador de tareas: Como controlador de tareas para la ejecución de hardware, proporciona tareas específicas específicas para los chips Ascend AI. El gestor de operaciones y el programador de tareas trabajan juntos para formar un sistema de presas para el flujo de tareas de red neural a los recursos de hardware, y monitorear y distribuir diferentes tipos de tareas de ejecución en tiempo real.

Flujo de software de red neuronal de procesadores Ascend AI

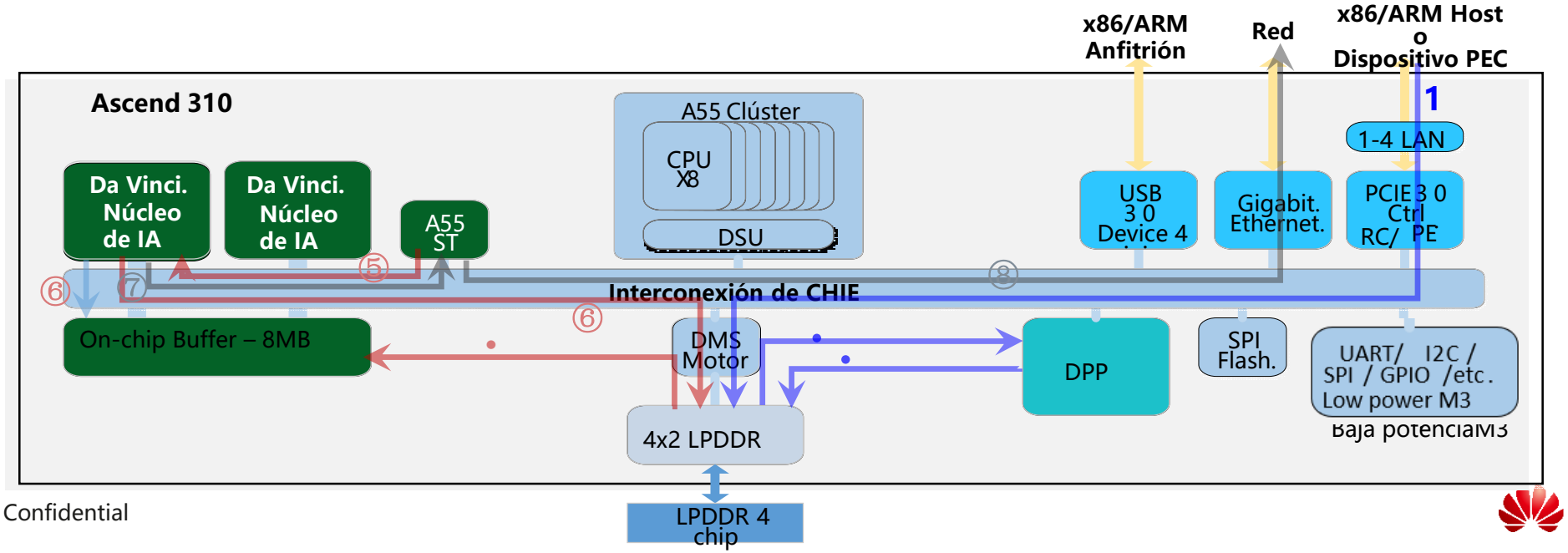


Contenido

1. Aspectos generales de los chips de IA
2. Arquitectura de hardware de los chips Ascend
3. **Arquitectura de software de los chips Ascend**
   - Arquitectura lógica de Ascend 310
   - Flujo de software de red neuronal de Ascend 310
   - Diagrama de flujo de datos de Ascend 310
4. Plataforma informática Huawei Atlas AI
5. Aplicaciones industriales de Atlas

Diagrama de flujo de datos del procesador Ascend AI - Aplicación de inferencia de reconocimiento facial (1)

- Recopilación y procesamiento de datos de cámaras
  - Los flujos de video comprimidos se transmiten desde la cámara a la memoria de DDR a través de PCIE.
  - DVPP lee los flujos de vídeo comprimidos en la caché.
  - Después del preprocesamiento, DvPP escribe tramas descomprimidas en la memoria del DDR.



HUAWEI

Diagrama de flujo de datos del procesador Ascend AI - Aplicación de inferencia de reconocimiento facial (2)

- Inferencia de datos
  - El programador de tareas envía una instrucción al motor de AMD para que precargue los recursos de IA desde el DDR al búfer en chip.
  - El ST configura el core de IA para ejecutar tareas.
  - El núcleo de IA lee el mapa de características y el peso, y escribe el resultado en el DDR o en el búfer en el chip.
- Resultado de reconocimiento facial
  - Después del procesamiento, el núcleo de IA envía las señales a la DT, que verifica el resultado. Si se debe asignar otra tarea, se realiza la operación en el paso 4.
  - Una vez finalizada la última tarea de IA, el ST reporta el resultado al host.

Resumen de la sección

- Esta sección describe la arquitectura de software de los chips Ascend, incluidas las arquitecturas de software L0-L3 y el flujo de software de red neural del procesador Ascend I.A.

Contenido

1. Aspectos generales de los chips de IA
2. Arquitectura de hardware de los chips Ascend
3. Arquitectura de software de los chips Ascend
4. **Plataforma informática Huawei Atlas AI**
5. Aplicaciones industriales de Atlas

Visión general y objetivos

- Esta sección describe los principales productos de la plataforma de computación de Huawei Atlas IA, incluyendo inferencia y entrenamiento.

Cartera de plataformas informáticas Atlas AI

Internet, seguridad, finanzas, transporte, energía, etc.

Habilitar Aplicacion

Industry SDK/container engine/
Repositorio de servicio basico

Administración de cluster/model management/data pre-processing

AI Framework

Mindspore.

TensorFlow/PyTorch/Caffe/MxNet

Adaptador de framework

CANN

AscendCL

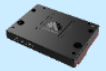
Motor de optimización de gráficos

Operator/Acceleration/Communication libraries (BLAS, FFT, DNN, Rand, Solver, Sparse, HCCL)

Tiempo de ejecución

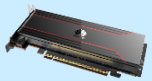
Controlador

Chips y hardware

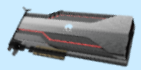


Atlas 200
16 TOPS INT8

Kit para desarrolladores de Atlas 200



Tarjeta aceleradora de inferencia Atlas 300
64 TOPS INT



Tarjeta de capacitación Atlas 300
256 TFLOPS FP16

Atlas 500.
16 TOPS INT8

Servidor de inferencias Atlas 800 IA
512 TOPS INT8

Servidor de capacitación en IA Atlas 800
2 PFLOPS FP16

Atlas 900.
256-1024 PFLOPS FP16

Ascend 310

Arquitectura Da Vinci

Ascend 910

Componentes públicos

Subsistema unificado de gestión de configuración y # induction # M

Subsistema de seguridad

AXE toolchain (log/profiling/Mind Studio)

Atlas acelera la inferencia de IA

Procesador IA
Ascend 310

Rendimiento mejorado 7 veces para dispositivos terminales

Kit para desarrolladores de Atlas 200
Modelo: 3000

Módulo acelerador de IA Atlas 200
Modelo: 3000

Mayor densidad en la industria (64 canales) para inferencia de vídeo

Tarjeta aceleradora Atlas 300 IA
Modelo: 3000

Inteligencia de vanguardia y colaboración de vanguardia

Estación Edge Atlas 500 IA
Modelo: 3000

Poderosa plataforma informática para la inferencia de IA

Servidor Atlas 800 IA
Modelo 3000/3010

Atlas 200DK: gran capacidad de computación y facilidad de uso

- **16 TTOPS INT8 24 W**
- **1 puerto USB tipo C, 2 puertos de cámara, 1 puerto GE, 1 ranura para tarjeta SD**
- **Memoria de 8 GB**
- **Temperatura de funcionamiento: 0° C a 45 ° C**
- **Dimensiones (altura x ancho x profundidad): 24 mm x 125 mm x 80 mm**

Desarrollo de IA Full Stack dentro y fuera de la nube

Desarrolladores

Configurar un entorno dev con un portátil
Coste ultra bajo para entornos locales independientes, con múltiples funciones e interfaces para cumplir con los requisitos básicos

Investigadores

Colaboración local dev + cloud training
Misma pila de protocolos para la nube de Huawei y el kit del desarrollador; capacitación en la nube e implementación local; no se requieren modificaciones

Startups

Demostración a nivel de código
Implementación de la función de algoritmo mediante la modificación de un código del 10 % basado en la arquitectura de referencia; interacción con la Comunidad de Desarrolladores; migración fluida de productos comerciales

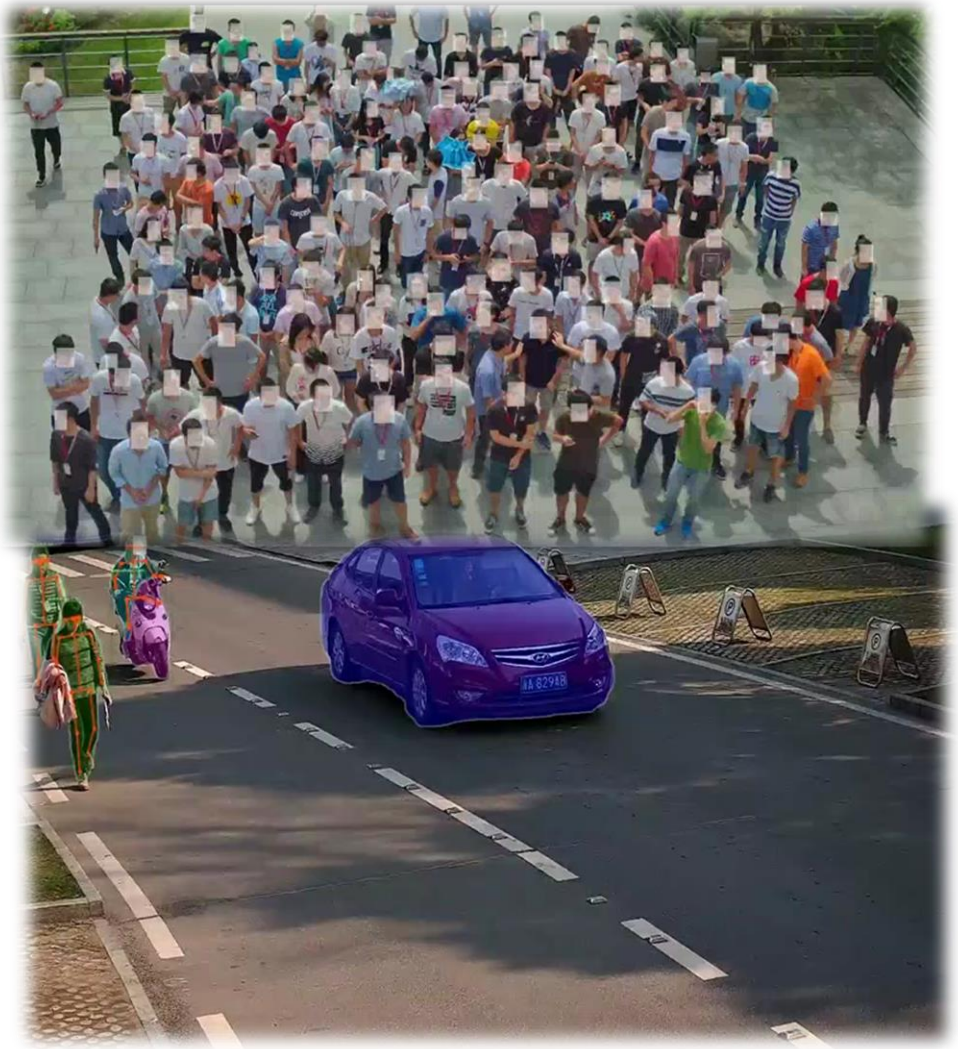
Atlas 200DK: gran capacidad de computación y facilidad de uso

- **16 TOPS INT8, 9,5 w**
- **Análisis en tiempo real de vídeo de alta definición de 16 canales, decodificación JPEGName**
- **Memoria de 4 Gb/8Gb, cuatro interfaces PCIe 3.0**
- **Temperatura de funcionamiento - 25° C to +80° C**

30 200

Cantidad de caras reconocidas por trama

- **Ejecución simultánea de múltiples algoritmos en la cámara de IA**

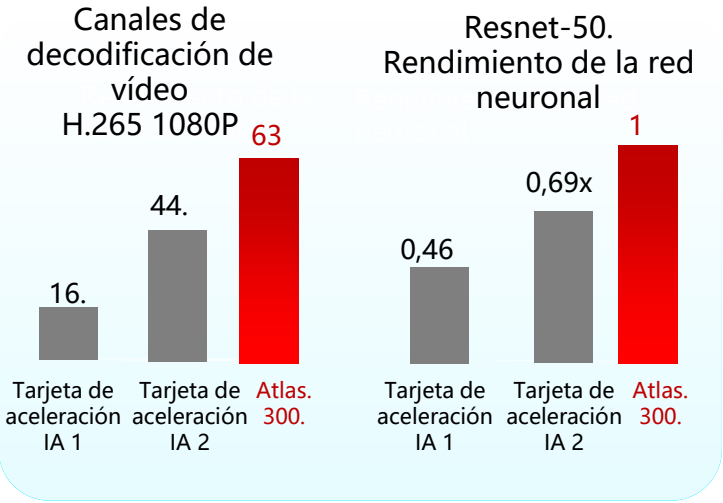


HUAWEI

Atlas 300: tarjeta aceleradora de interferencia de vídeo de 64 canales y densidad más alta

- **INT8 DE 64 TOPS, 67 w**
- **Memoria de 32 GB**
- **Análisis de vídeo de alta definición de 64 canales en tiempo real**

Modelo: 3000

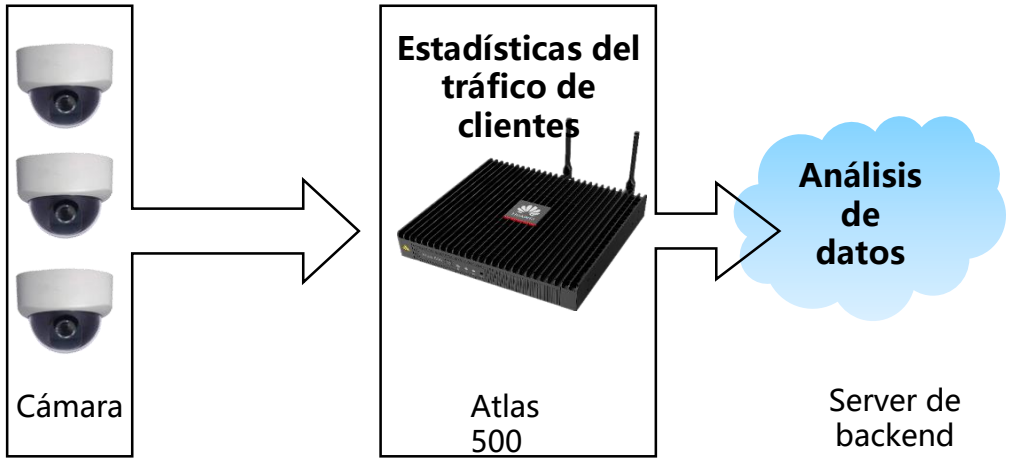


Análisis de vídeo | reconocimiento de voz | Marketing de precisión | Análisis de imagen médica

HUAWEI

Atlas 500 AI Edge Station

- **16 TOPS INT8**
- **25 w a 40 w**
- **Wi-Fi y LTE**
- **Análisis de vídeo HD de 64 canales en tiempo real**
- **Diseño sin ventiladores, estable de -40° C a +70° C**



HUAWEI

Servidor AI Atlas 800

Modelo: 3000

- **Una plataforma de inferencia eficiente alimentada por Kunpeng**
- **Funciones clave:**
  - 2 procesadores Kunpeng 920 en un espacio de 2U
  - 8 ranuras PCIe, que soportan hasta 8 tarjetas de acelerador Atlas 300 AI
  - Análisis de vídeo HD de hasta 512 canales en tiempo real
  - Refrigerado por aire, estable entre 5 ° C y 40 ° C

Modelo: 3010

- **Una plataforma de inferencia flexible alimentada por Intel**
- **Funciones clave:**
  - 2 procesadores Intel Xeon SP Skylake o Cascade Lake en un espacio de 2U
  - 8 ranuras PCIe, que soportan hasta 7 tarjetas de acelerador Atlas 300/NVIDIA T4 AI
  - Análisis de vídeo HD de hasta 448 canales en tiempo real
  - Refrigerado por aire, estable entre 5 ° C y 35 ° C

Atlas acelera el entrenamiento de IA

Ascend 910
Procesador AI

Tarjeta de entrenamiento con máxima potencia de computación

Tarjeta de acelerador Atlas 300 AI
Modelo: 9000

El servidor de entrenamiento más potente del mundo

Server AI Atlas 800
Modelo: 9000/9010

El clúster de entrenamiento de IA más rápido del mundo

cluster de AI Atlas 900

Tarjeta Aceleradora AI Atlas 300: Tarjeta aceleradora de mayor rendimiento para entrenamiento de AI

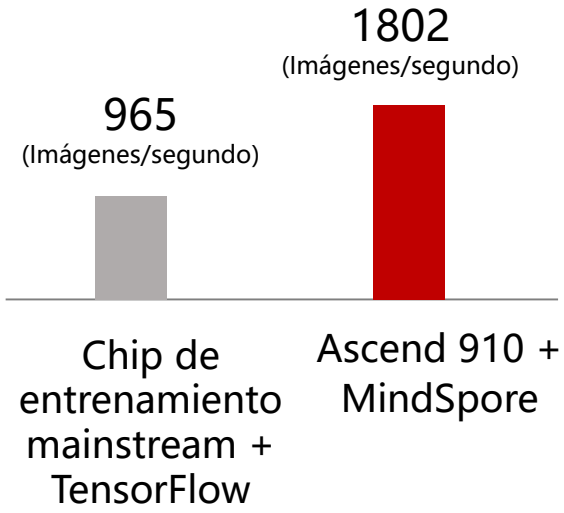
Atlas 300

Modelo: 9000

2 x↑

Potencia de cálculo por tarjeta

256 T FLOPS FP16



70%↓

Latencia de sincronización gradual

Directo 100G RoCE

- **Prueba de referencia:**
  - ResNet 50 V1.5
  - ImageNet 2012
  - Tamaño óptimo del lote, respectivamente

Server de entrenamiento Atlas 800: el servidor más potente de la industria para la formación en IA

Atlas 800

Modelo: 9000

2,5x ↑

Densidad de potencia de computación

2 P FLOPS/4U

25 ↑

Decodificador de hardware

16384 Imágenes / segundo

(decodificación de 1080 p)

1,8x ↑

Perf. / Watt

2 P FLOPS/5.6kW

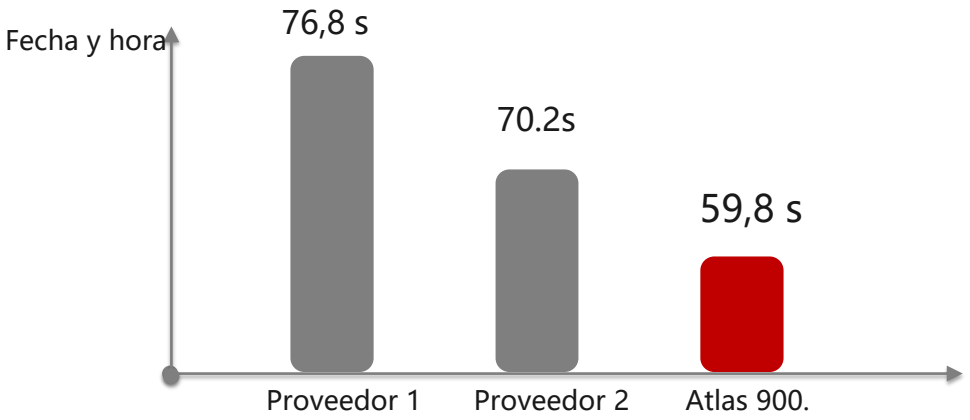
Cluster IA Atlas 900:
Cluster de IA más rápido para entrenamiento en IA

Atlas 900

256-1024 PFLOPS FP16

Potencia computarizada líder en el sector | Mejor red de clústers | Disipación térmica máxima

Consumo de tiempo más corto: 59.8s



- **Prueba de referencia:**
  - Análisis de rendimiento: modelo ResNet-50 V1.5, conjunto de datos ImageNet-1k
  - Cluster: 1024 procesadores Ascend 910 de IA
  - Exactitud: 75.9%

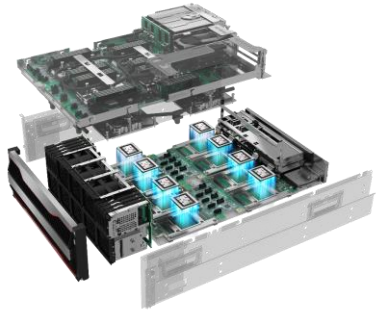
HUAWEI

El sistema de aprendizaje profundo de Atlas acelera la formación de modelos de IA y crea amplias aplicaciones

Formación modelo

Implementación del modelo

Atlas 300 tarjeta de aceleración de la capacitación
Modelo: 9000



Servidor de entrenamiento Atlas 800 AI
Modelo: 9000

cluster de IA Atlas 900

Análisis de video

Investigaciones sobre los genes

Conducción automatizada

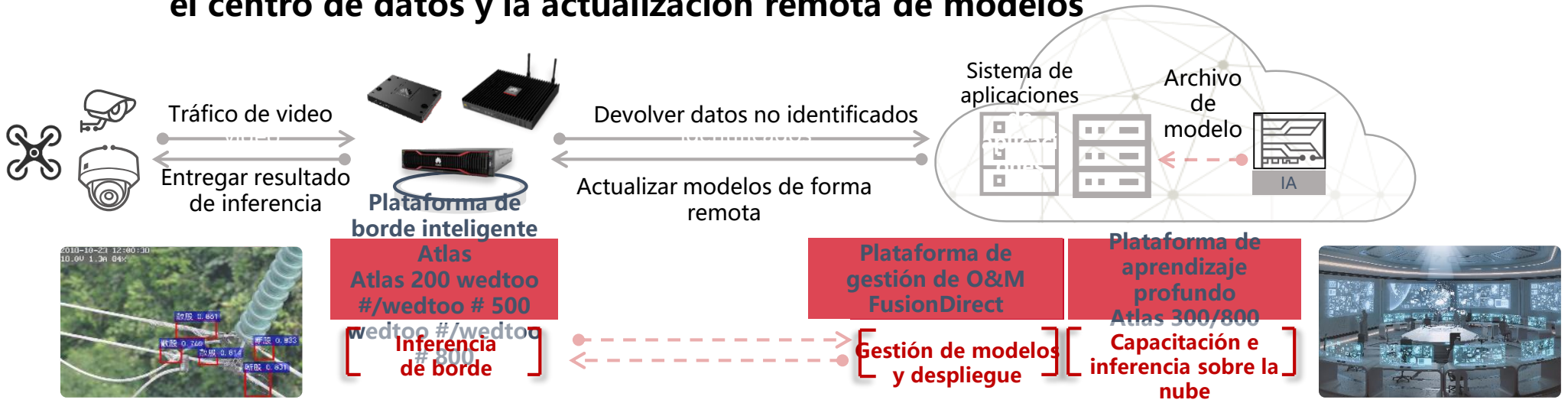
Pronóstico del tiempo

Exploración de petróleo

HUAWEI

La colaboración entre dispositivos y cloud permite el desarrollo y la experiencia de usuario finales

Colaboración entre dispositivos y cloud para la formación continua en el centro de datos y la actualización remota de modelos



	Desarrollo centralizado	Centralizado de O&M	Seguridad mejorada
Atlas	• Arquitectura de desarrollo centralizada basada en Da Vinci y CANN, **desarrollar una vez, implementar en todas partes**	• FusionDirector gestiona hasta 50.000 nodos, gestiona dispositivos centrales y de borde y, de forma remota, impulsa modelos y actualiza dispositivos	• Encriptación de canal de transmisión • **Cifrado de modelo, doble aseguramiento**
Solucion para industria	• Los centros de datos y de edge utilizan diferentes arquitecturas de desarrollo. **Los modelos no pueden transferirse libremente, lo que requiere un desarrollo secundario.**	• **No hay herramientas de gestión de O&M;** solo proporciona API, por lo que los clientes necesitan desarrollar API por sí mismos.	• **No hay motor de cifrado/descifrado;** los modelos no están cifrados.

Resumen de la sección

- Esta sección presenta los productos de la plataforma informática Atlas AI de Huawei, principalmente productos de inferencia, como Atlas 200 DK, el módulo acelerador Atlas 200 AI, la tarjeta aceleradora Atlas 300 AI, la estación Edge Atlas 500 AI y el servidor Atlas 800 AI. Las plataformas informáticas utilizadas para la capacitación incluyen la tarjeta aceleradora Atlas 300 AI, el servidor Atlas 800 AI y el Cluster Atlas 900 AI.

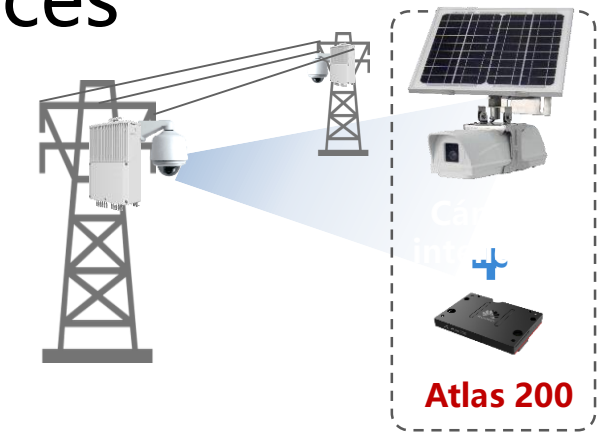
Contenido

1. Aspectos generales de los chips de IA
2. Arquitectura de hardware de los chips Ascend
3. Arquitectura de software de los chips Ascend
4. Plataforma informática Huawei Atlas Al
5. **Aplicaciones industriales de Atlas**

Visión general y objetivos

- En esta sección se presentan los escenarios de aplicación industrial de la plataforma informática de IA Atlas.

Energía eléctrica: la primera solución de inspección inteligente desatendida del sector, con una eficiencia de 5 veces



Alarmas de retorno y análisis front-end

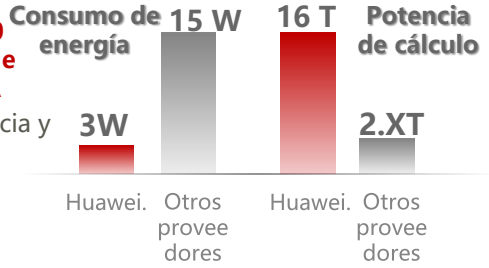
Optimización de modelos y actualización remota

Atlas 300/800 Capacitación de modelo de IA

Plataforma de vigilancia y gestión

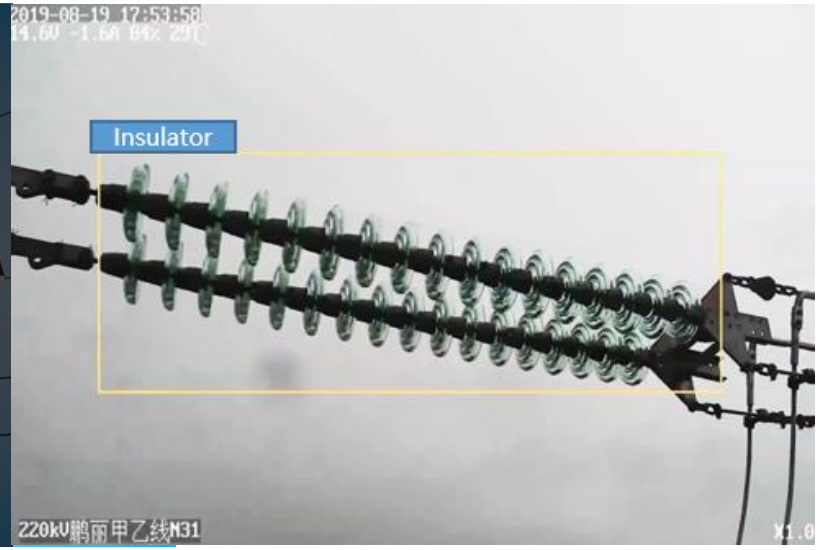
Consumo promedio del sistema	Análisis en tiempo real	Alta potencia informática
1W	Una vez por minuto	16 TOPS

Análisis inteligente en el borde con un consumo ultra bajo



Consumo de energía 80 %

Ideal para la informática de vanguardia

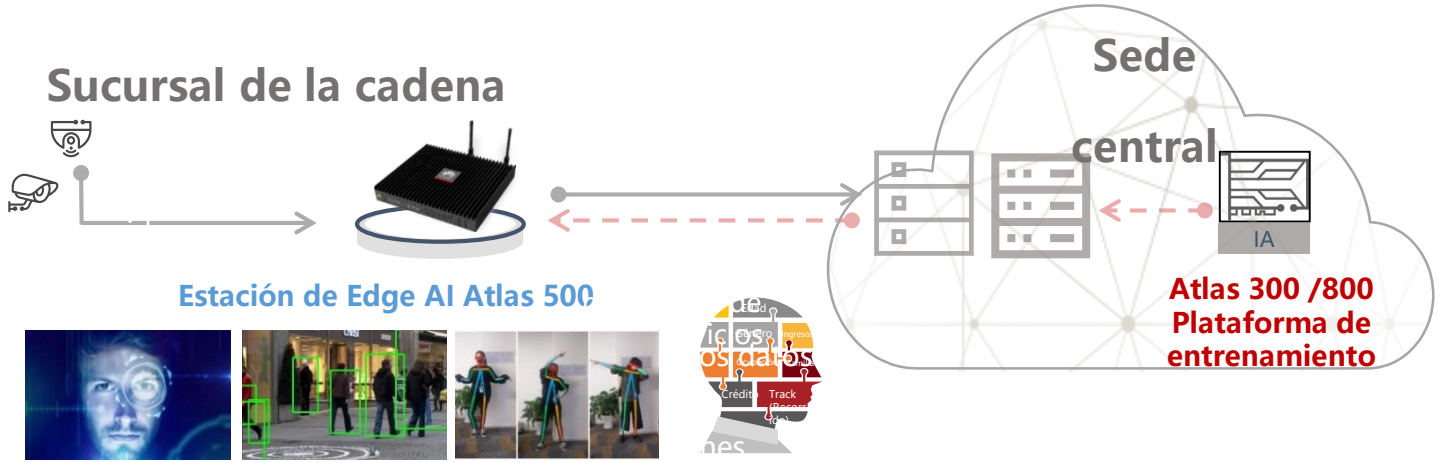


HUAWEI

Finanzas: AI permite sucursales inteligentes para los bancos

Pasado: La experiencia del cliente necesita mejorar

Ahora: colaboración de vanguardia, finanzas inteligentes



Identificación VIP precisa
Tasa de conversión del cliente potencial 60%

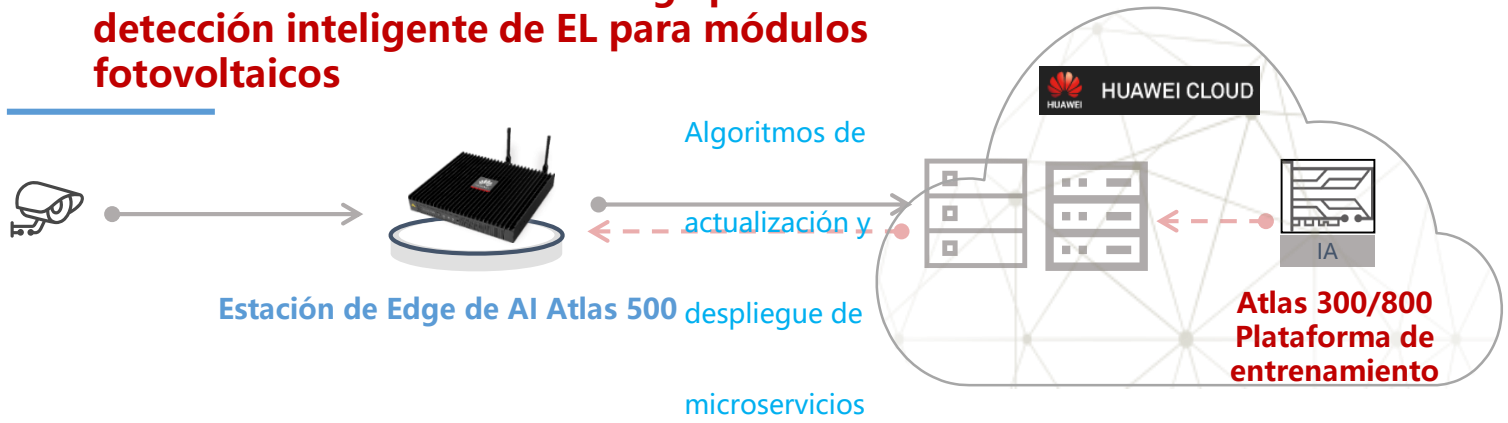
Autenticación basada en reconocimiento facial
Tiempo de procesamiento del servicio 70%

Análisis de la duración de la fila de clientes
Quejas del cliente 50%

HUAWEI

Fabricación: IA permite línea de producción inteligente con visión artificial

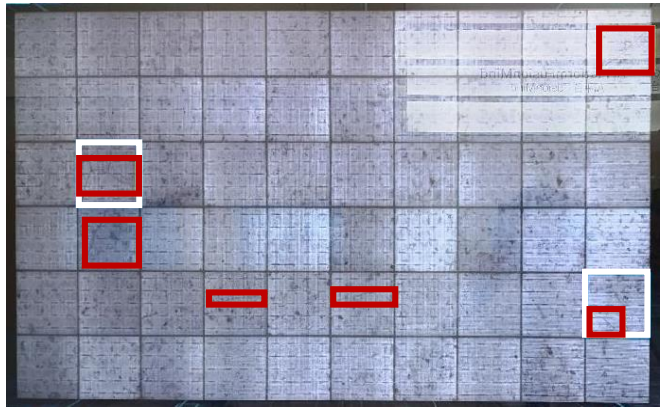
La colaboración con Cloud-Edge permite la detección inteligente de EL para módulos fotovoltaicos



Inspección manual

Resultados inestables	Bajo rendimiento	Proceso discontinuo	Altos costos de mano de obra

Cantidad de chips de batería defectuosos	*2*
Duración de la detección	*unos 5s*
Exactitud	*33,33%*



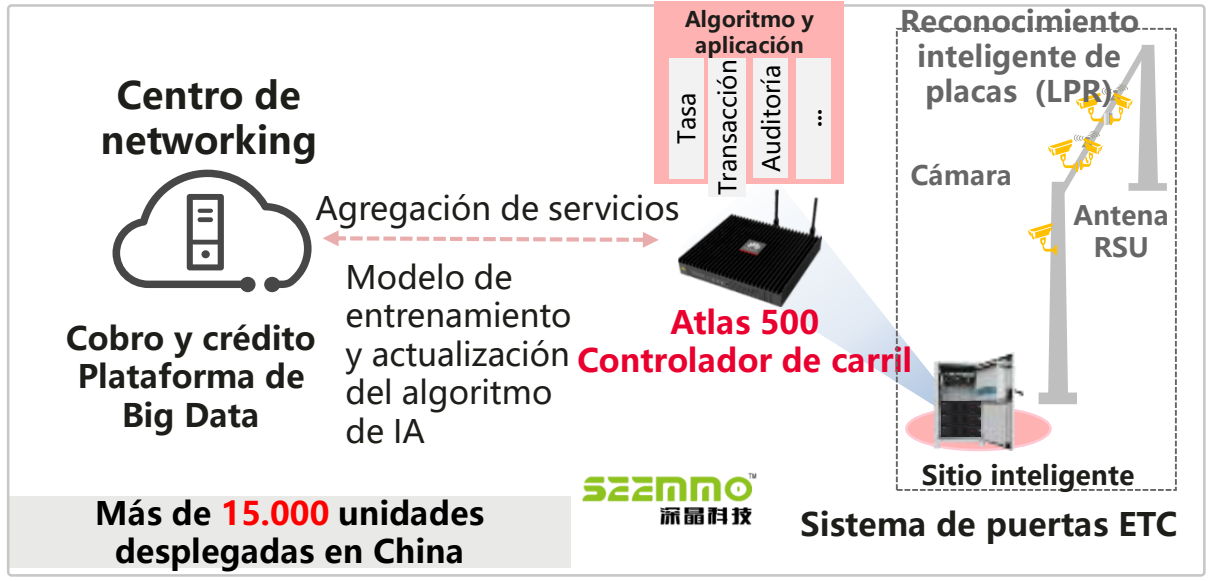
Inspección inteligente

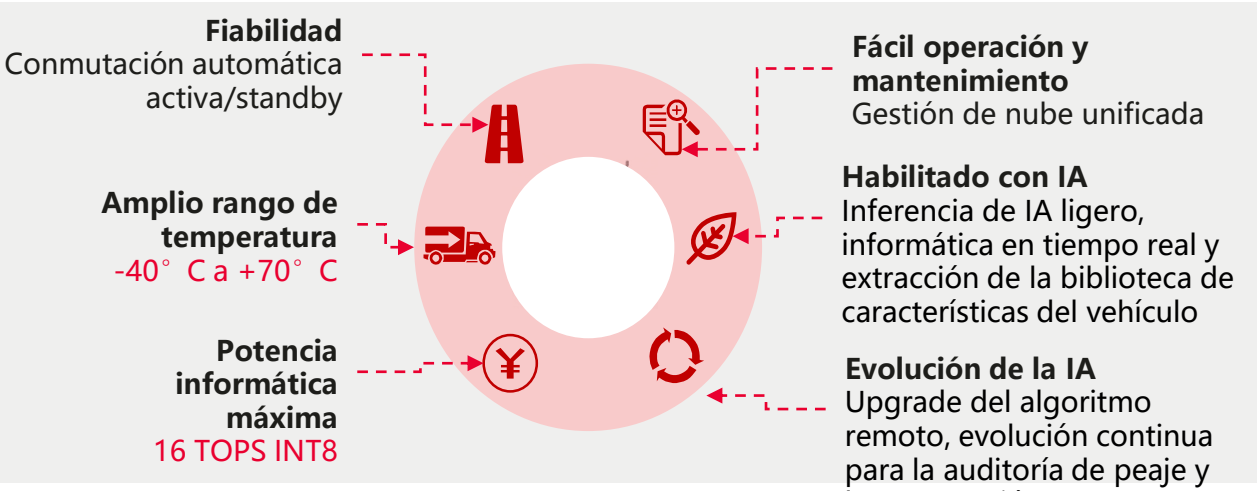
Omisión cero	Alta eficiencia de producción	Sinergias de la nube	Reducción de los costos laborales

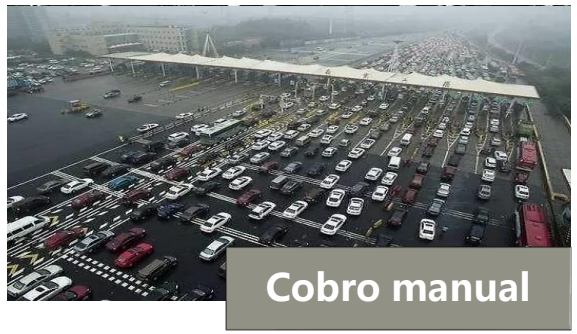
Chip de batería defectuoso	*6*
Duración de la detección	*1,36s*
Exactitud	*100%*

HUAWEI

Transporte: AI acelera las autopistas con 5x de eficiencia







- Bajo rendimiento
- Tiempo en fila largo

- Eficiencia de paso 5X ↑
- Ahorro de energía y reducción de emisiones

- Control de seguridad proactivo
- Gestión de la cooperación vial
- Conducción autónoma de vehículos

HUAWEI

Supercomputación: Atlas ayuda a PCL a construir CloudBrain Fase II

Laboratorio Peng Cheng (PCL)
Plataforma básica innovadora para las misiones nacionales

Capa de aplicación
- Conducción automatizada
- Cerebro de la ciudad
- Atención sanitaria inteligente
- Reconocimiento de voz
- Procesamiento natural de Idioma

Capa de plataforma básica
Infraestructura crítica para la IA

Capa de recursos físicos
- Fase I de Peng Cheng CloudBrain

- Fase II de Peng Cheng CloudBrain

Peng Cheng CloudBrain Fase II construyó principalmente por Atlas 900, **el clúster de entrenamiento más rápido del mundo**

Potencia de computación máxima

Potencia de computación de IA de nivel E

Red de clústeres principales

La comunicación HCCL soporta la creación de redes en plano de parámetros no bloqueantes de 100 TB/s

Eficiencia energética máxima

Cluster de IA PUE < 1.1

Atraer a más desarrolladores basados en la comunidad de desarrolladores Ascend

+

Salón técnico anual

- Realizado en más de 10 ciudades
- Más de 20 entrenadores superiores
- Decenas de discursos

Concurso de desarrolladores

- **Más de 1.500 equipos**
- Premio anual de más de 1 millón de RMB
- Igualdad de oportunidades para las empresas y las universidades

Soporte al desarrollador

- Vouchers publicos de cloud
- Tickets de curso de certificación gratis
- Kits de desarrolladores de Atlas gratis

- **Plataforma de habilitación centrada en el desarrollador**
- **http://www.ascend.huawei.com/home**

Resumen

- Este capítulo describe los productos de la plataforma de computación Atlas de Huawei y le ayuda a comprender los principios de trabajo de los chips Ascend de Huawei. Se centra en las arquitecturas de hardware y software de los chips Ascend y los escenarios de aplicación de la plataforma de computación Atlas AI.

Quiz

1. ¿Cuáles son las principales aplicaciones de Ascend 310? ( )
   A. Inferencia de modelo
   B. Entrenamiento de modelo
   C. Construcción de modelos

Recomendaciones

- Comunidad Ascend:
  - https://ascend.huawei.com/

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

HUAWEI

Plataforma de IA abierta de Huawei para dispositivos inteligentes

HUAWEI

Prefacio

- Huawei HiAI es una plataforma abierta de inteligencia artificial (IA) para dispositivos inteligentes, que adopta una arquitectura "chip-device-cloud", abriendo chips, aplicaciones y capacidades de servicio para un ecosistema totalmente inteligente. Esto ayuda a los desarrolladores a ofrecer una mejor experiencia de aplicaciones inteligentes para los usuarios al aprovechar plenamente las potentes capacidades de procesamiento de IA de Huawei.

Objetivos

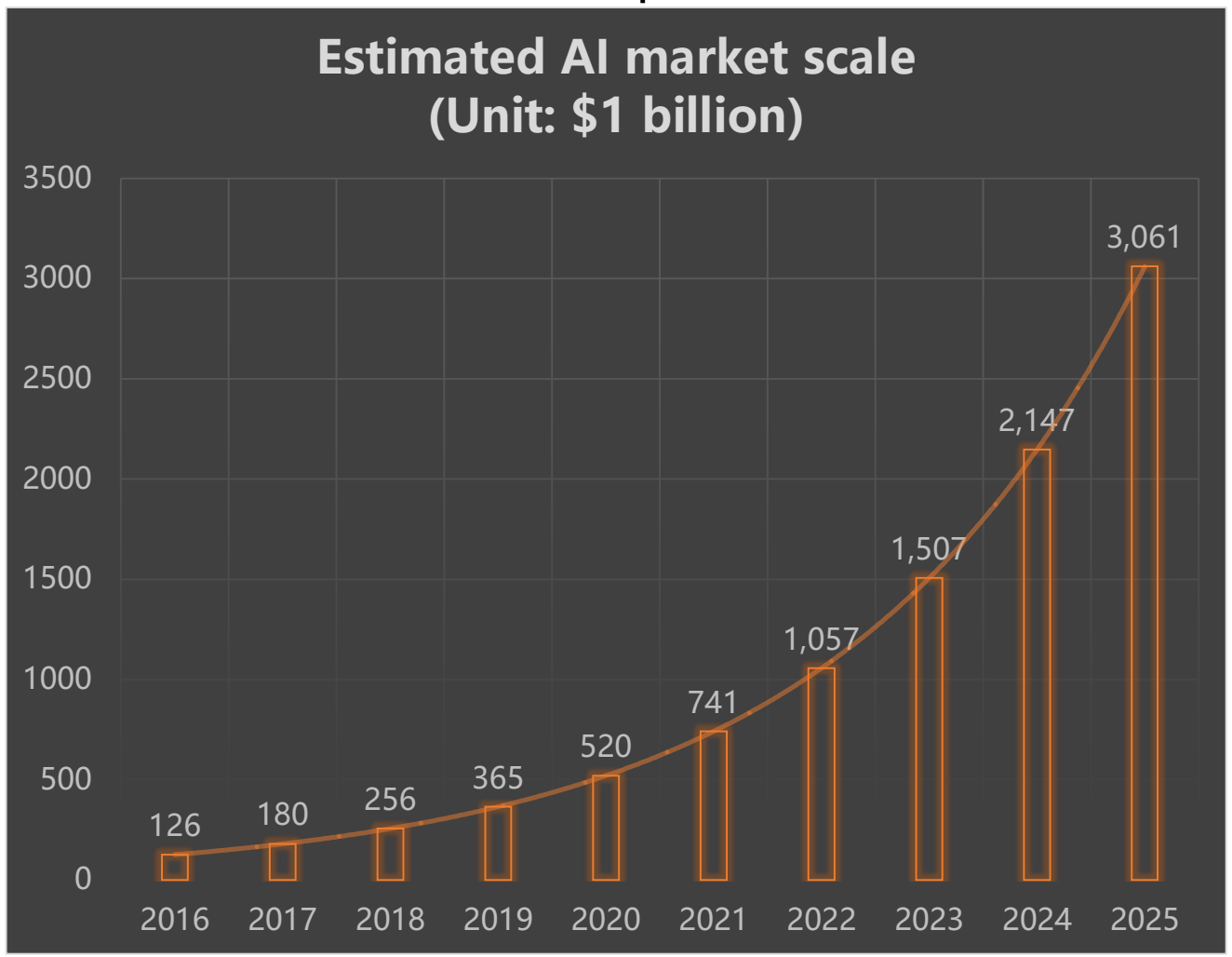
Después de este curso, podrá:

- Dominar el uso de la plataforma HiAI de Huawei.
- Comprender las poderosas funciones de la plataforma HiAI de Huawei.

Contenido

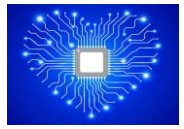
1. **Ecosistema industrial de IA**
2. Plataforma HiAI de Huawei
3. Desarrollo de aplicaciones basadas en la plataforma HiAI de Huawei

Grandes oportunidades: Se prevee ubicuidad de la IA en un mercado de $3 trillones



Industrias participantes: automóvil, finanzas, bienes de consumo y venta al por menor, atención médica, educación, manufactura, comunicaciones, energía, turismo, cultura y entretenimiento, transporte, logística, bienes raíces y protección del medio ambiente

Descubrimiento de potencia de computación



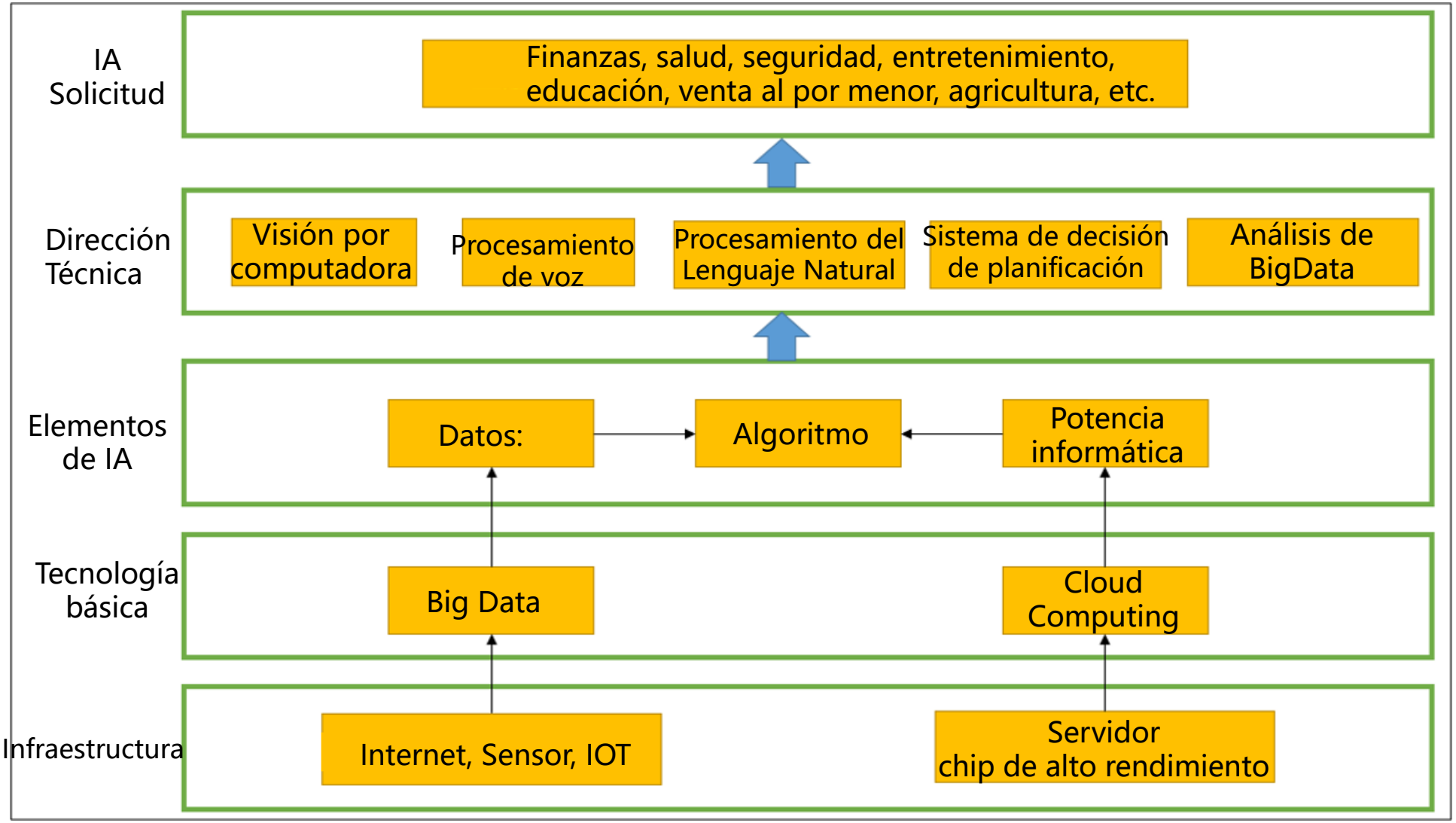
Avances en Algoritmos

Avances de Data

Fuente de datos: Forrester, Transparency Market Research, Asociación China de Inteligencia Artificial y Roland Berger

HUAWEI

Arquitectura de la plataforma de aplicaciones de IA



Desafíos en el desarrollo y aplicación de capacidades de IA

Umbrales altos

Requerimientos demandantes

6 meses: LM y LD
2 meses: Estadísticas
4 meses: Álgebra lineal
3 meses: Cálculo

15 meses

Baja eficiencia

Largos ciclos de entrenamiento

Recogida y limpieza de datos
Entrenamiento y optimización de modelos
Mejora personalizada de la experiencia

3 - 8 meses

Diversas solicitudes

Experiencia fragmentada

Se deberá repetir todo el ciclo de entrenamiento para cada escenario.
Ausencia de heredabilidad o transferibilidad

N veces la carga de trabajo

Iteración lenta

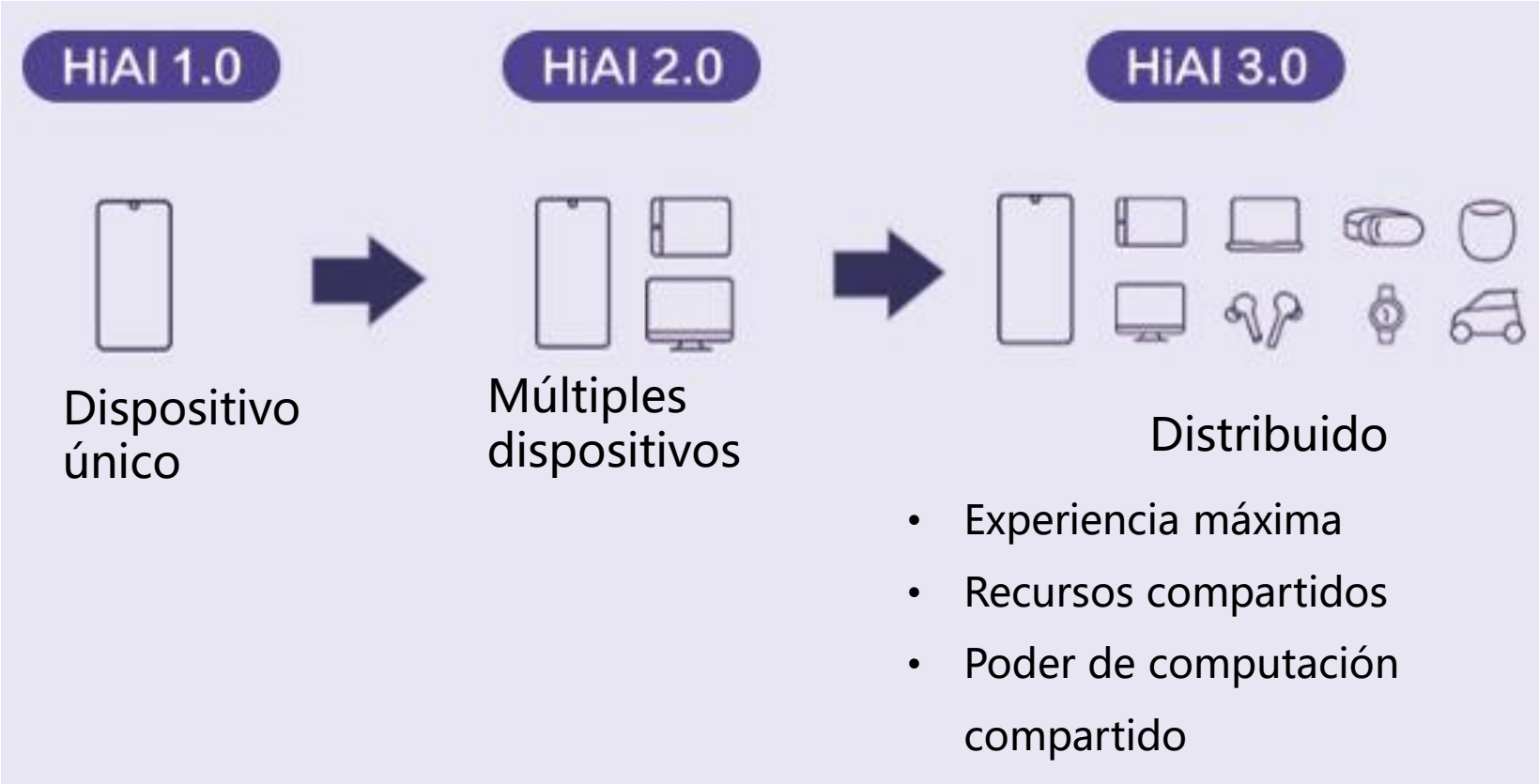
Mejora limitada de la capacidad

Dificultades en la actualización del modelo
Dificultades para obtener datos válidos

Contenido

1. Ecosistema industrial de IA
2. **Plataforma HiAI de Huawei**
3. Desarrollo de aplicaciones basadas en la plataforma HiAI de Huawei

HiAI 3.0: Habilitar la máxima experiencia en AI en todos los escenarios

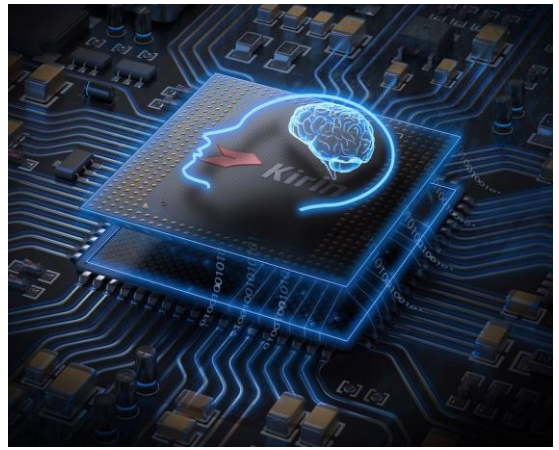


- Más de 4000 socios
- Más de 96 millones de usuarios activos diarios
- Más de 600 mil millones de llamadas mensuales

Huawei HiAI 3.0: Habilitando la Distribución por IA en todos los Escenarios

Cloud. 1000+ Servicios automatizados.

Dispositivo. 40+ interfaces de programación de aplicaciones (APIs)



Chip 300+ operadores

Servicio HiAI de Huawei

Apertura de la capacidad de servicio para beneficios mutuos
Servicios Push basados en los requerimientos del usuario de forma activa.

Motor HiAI de Huawei

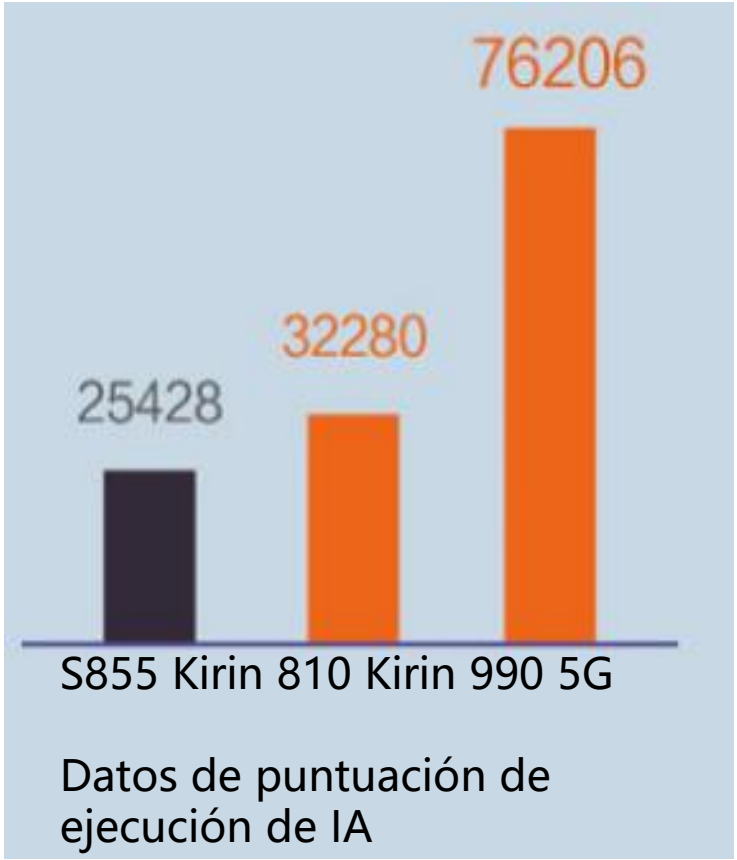
Apertura de la capacidad de IA para la simplicidad
Integre varias capacidades de IA en aplicaciones de forma sencilla, haciendo que las aplicaciones sean más inteligentes y potentes.

Fundación Huawei HiAI

Apertura de la capacidad del chip para lograr una alta eficiencia
Convertir y migrar rápidamente los modelos existentes para obtener un desempeño óptimo basado en la programación heterogénea y la aceleración de la unidad de proceso de red. NPU

Fundación de HiAI

- Las API de la Fundación de HiAI constituyen una biblioteca informática de IA de una plataforma de computación móvil, lo que permite a los desarrolladores compilar de forma eficiente aplicaciones de IA que se pueden ejecutar en dispositivos móviles.
  - Aprovechando el alto rendimiento y la alta precisión de los chips Kirin, un mejor rendimiento de IA del dispositivo será proporcionado por una potencia de computación más potente.
  - Apoyar el mayor número de operadores (300 +) en la industria y más marcos (frameworks), mejorando considerablemente la flexibilidad y la compatibilidad.
  - Los chips Honghu, Kirin y cámaras con IA permiten la capacidad de IA para más dispositivos.



Motor HiAI – HiAI Engine

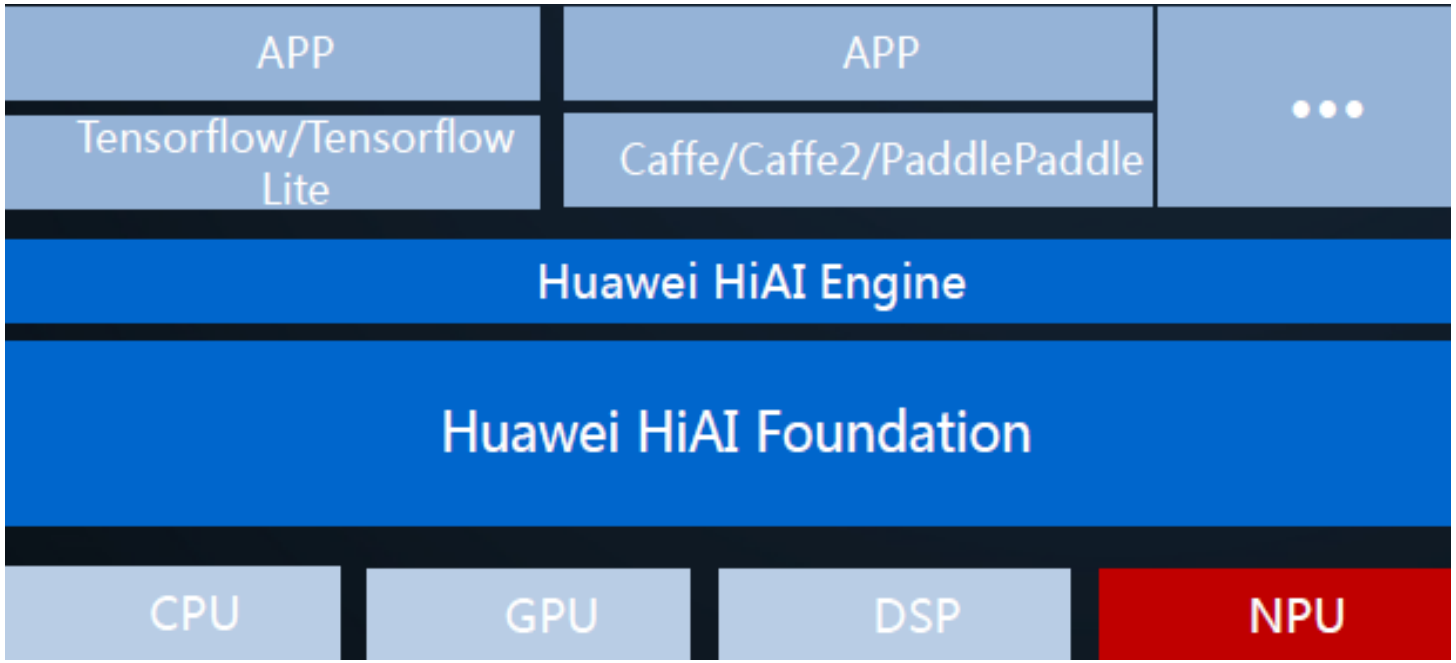
- HiAI Engine abre las capacidades de las aplicaciones e integra múltiples capacidades de IA en las aplicaciones, haciendo las aplicaciones más inteligentes y más potentes.
  - Proporciona reconocimiento de escritura a mano y capacidades de reconocimiento de gestos dinámico, con 40+ API subyacentes.
  - La visión informática y el reconocimiento de voz se desarrollarán hacia un modo distribuido, ayudando a los desarrolladores a ofrecer una experiencia de vida inteligente en todos los escenarios.

La IA distribuida ayuda al deporte y la salud, y a la seguridad en la conducción

Servicio HiAI

- El servicio permite a los desarrolladores reutilizar servicios en múltiples dispositivos, como teléfonos móviles, tabletas y pantallas grandes, utilizando un solo acceso al servicio, implementando de forma eficiente la distribución.

Arquitectura de la plataforma de computación móvil HiAI



Soporta diversos marcos front-end principales.

Proporciona varias API de servicios de capa superior para garantizar la ejecución eficiente en los dispositivos móviles.

Permite una programación flexible de recursos heterogéneos, satisfaciendo la demanda de los desarrolladores para acelerar la informática de modelo de red neuronal y el cómputo del operador.

Tool chain

Documentos completos

Diferentes tipos de API

Códigos de origen que permiten un inicio rápido

HUAWEI

¿Cómo pueden beneficiarse las aplicaciones de Huawei HiAI?

Tiempo real

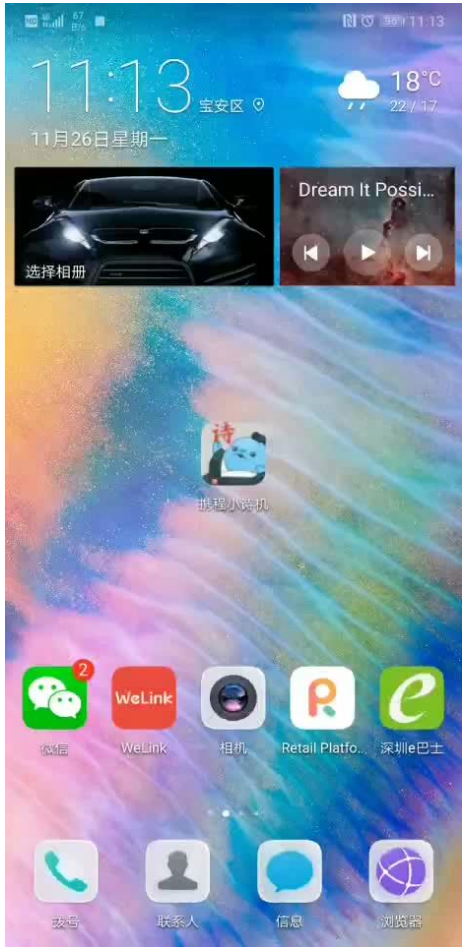
Listo para usar

Estabilidad

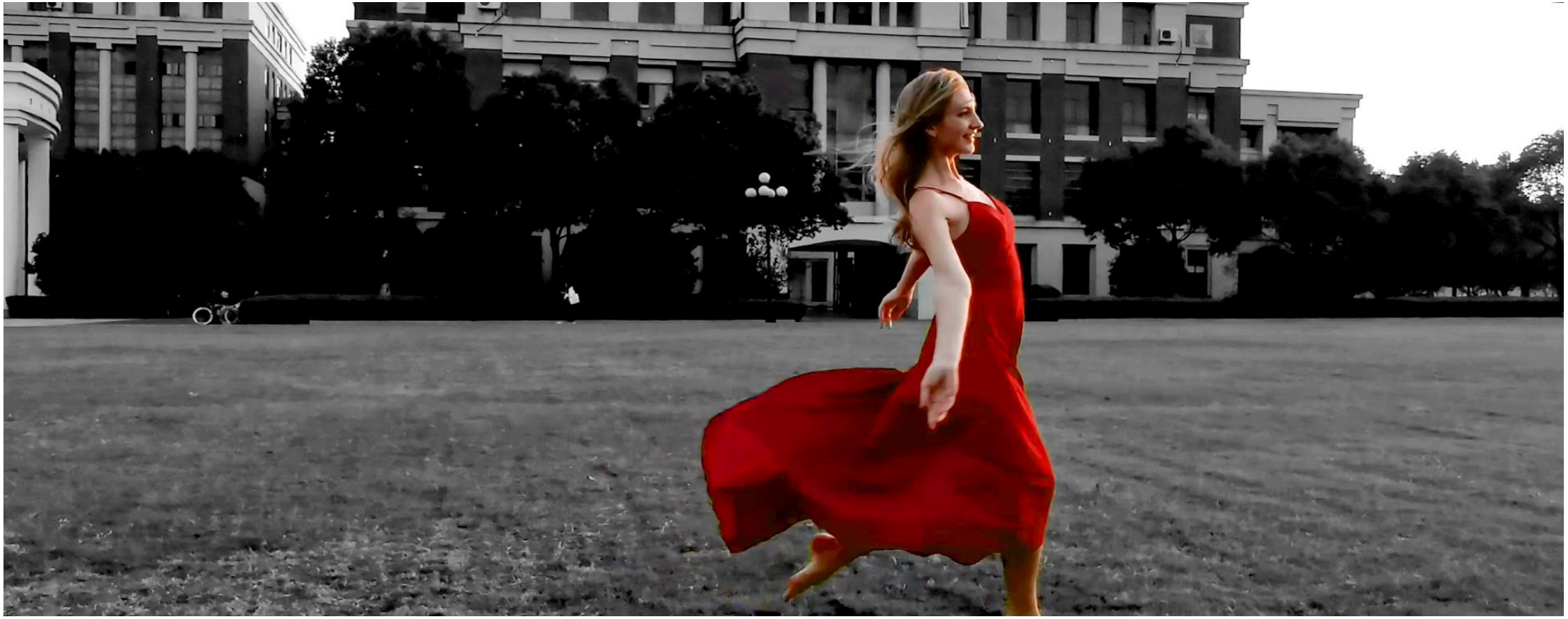
Seguridad

Menor coste

Huawei HiAI + Ctrip ayuda a tomar fotos poéticas



Segmentación de imágenes de IA de la NPU en tiempo real



Interfaz de imagen humana para la conmutación flexible en varios escenarios

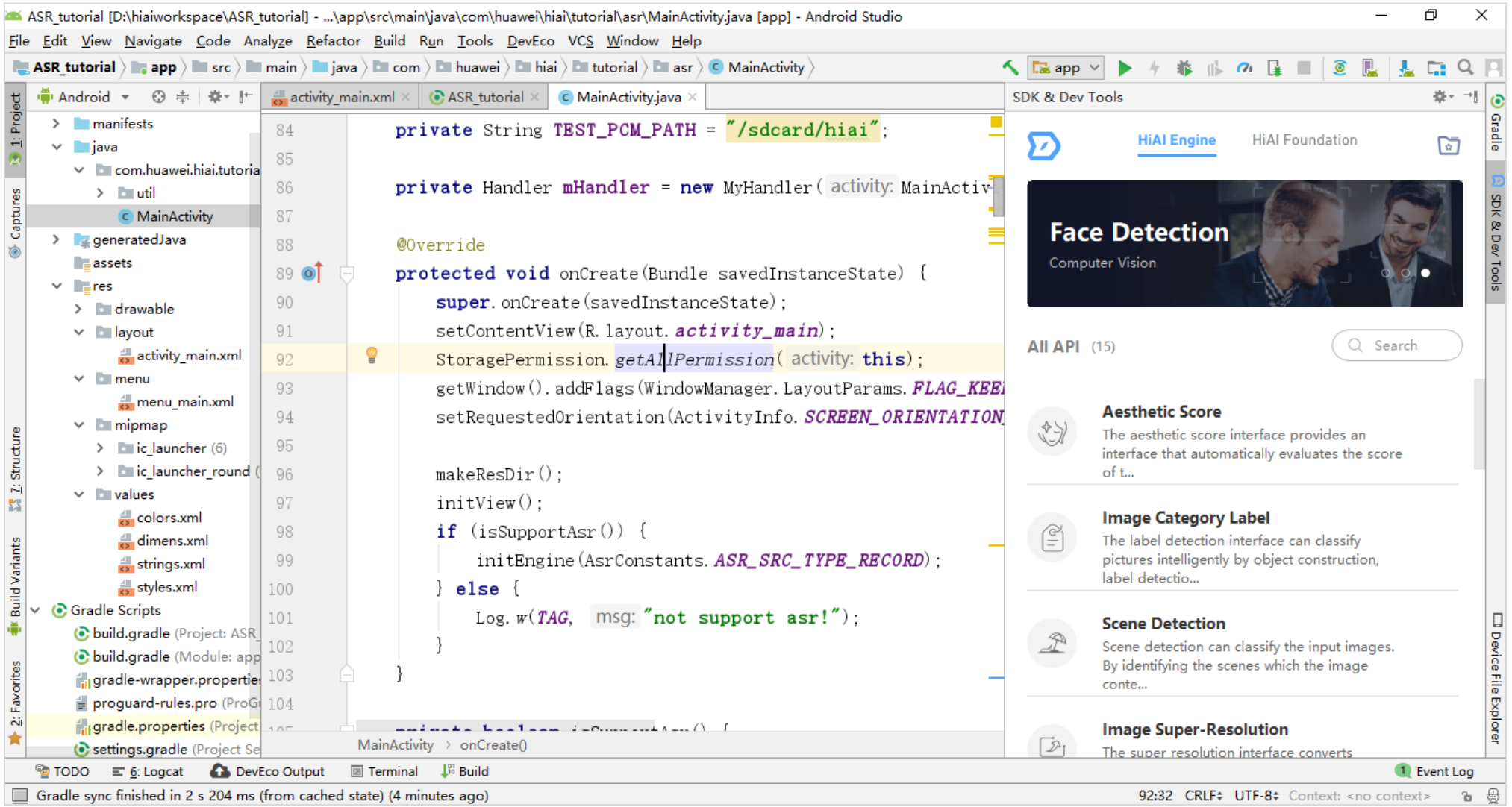
- http:// www.bhaisajya.net http:// www.bhaisajya.net http:// www.bhaisajya.net

Proveedor de capacidades de IA, aceleración del desarrollo de aplicaciones

Video corto y streaming en vivo	Redes sociales	Realidad aumentada (RA)	Toma de fotos y retoques	Compras	Traducción y procesamiento de textos
• Reconocimiento facial • Reconocimiento de gestos • Segmentación de retrato • Reconocimiento de postura • Estilo de vídeo • Control de voz • Profundidad inteligente de control de campo • Reconocimiento de escenas de imagen	• Categorización de fotos • Reconocimiento de imagen • Superresolución de imagen (RS) • Reconocimiento de datos sensibles	• Conciencia contextual • Control de voz • Estimación de profundidad • Estimación de la luz	• Embellecimiento • Mejora de imagen • Puntuación estética • Generación de álbumes • Fotografía por voz • Fotografía por gesto	• Escaneo de códigos QRName • Prestación directa de servicios y recomendación • Reconocimiento de tarjeta de identificación • Reconocimiento de tarjeta bancaria • Identificación del producto	• Traducir haciendo una foto • Reconocimiento óptico de caracteres • división de palabras • Reconocimiento de entidad nombrada • Reconocimiento de emociones de texto • Respuesta de texto inteligente • Texto e imagen
Visión de ordenador (VC), reconocimiento automático de voz (RAS)	VC, comprensión del lenguaje natural (ULN)	CA, VC	VC	VC	ULN, VC, ASR

Proveedor de capacidades de IA, aceleración del desarrollo de aplicaciones

Herramientas integrales para desarrolladores



HUAWEI

Contenido

1. Ecosistema industrial de IA
2. Plataforma HiAI de Huawei
3. **Desarrollo de aplicaciones basadas en la plataforma HiAI de Huawei**

Huawei HiAI ayuda a personas sordas y mudas

Experiencia móvil de próxima generación con Huawei HiAI

Rápida **Simple** **Beneficios mutuos**

HUAWEI

Conectar a los desarrolladores y estimular la innovación para lograr un ecosistema beneficioso para todos (win-win)

Conexión offline para una comunicación a profundidad

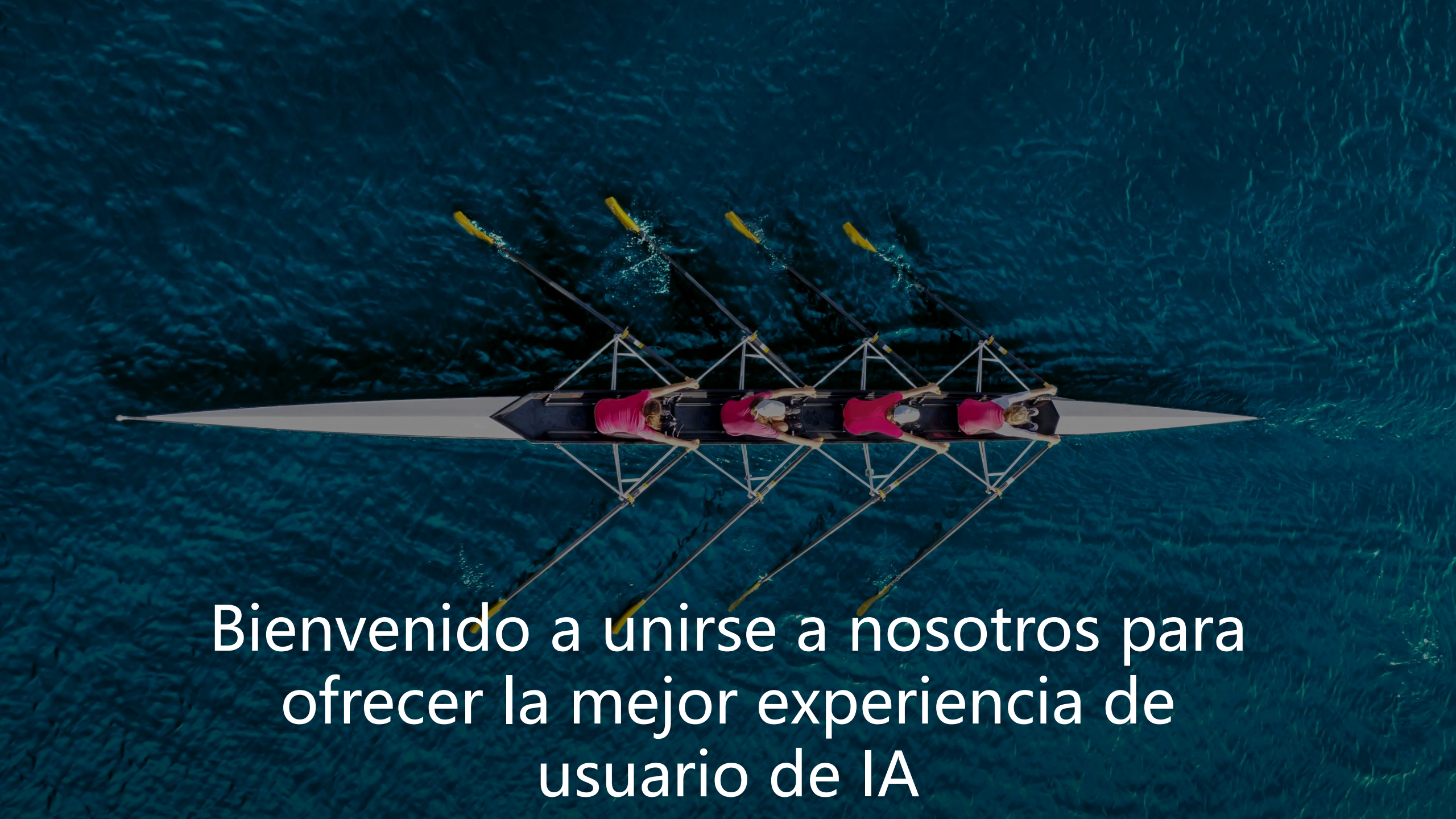
- Salones en las ciudades
- Cursos abiertos de HiAI
- Simposios técnicos

Inversión de 1,000 millones de USD, para estimular las innovaciones en todos los escenarios

- Apertura e innovación de las capacidades de los dispositivos
- Innovación en servicios digitales en todos los escenarios
- Co-construcción del ecosistema de servicios en nube

Competencia de innovación para el desarrollo continuo

- Concurso de innovación de aplicaciones de IA
- Concurso de creatividad para aplicaciones futuras
- Concurso de innovación de aplicaciones de realidad aumentada - AR


Bienvenido a unirse a nosotros para
ofrecer la mejor experiencia de
usuario de IA

Resumen

- Creemos que la IA puede mejorar la vida al traer una comodidad sin precedentes tanto para el desarrollo (back-end) como para los dispositivos. Sin embargo, esto requiere escenarios de aplicación reales que permitan a más empresas y desarrolladores desempeñar un papel en la mejora sustancial de la experiencia del usuario. Huawei está dispuesto a trabajar con los partners para promover conjuntamente la transformación inteligente de las industrias con más desarrolladores y empresas basados en la plataforma HiAI3.0.

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

HUAWEI

Plataforma de aplicaciones de inteligencia empresarial de HUAWEI CLOUD

HUAWEI

Objetivos

Al finalizar este curso, podrás:

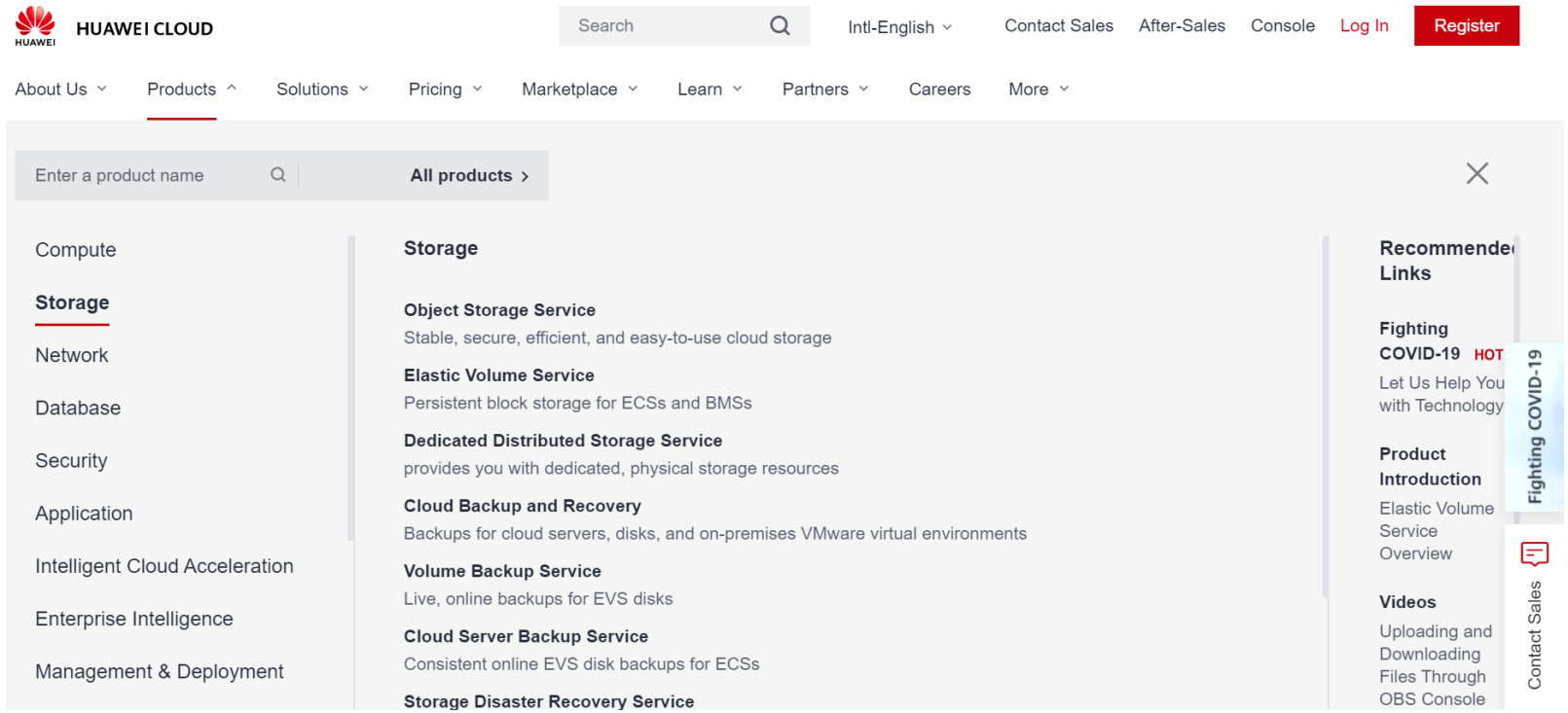
- Conocer el ecosistema y los servicios de inteligencia empresarial (IE) de HUAWEI CLOUD.
- Conocer la plataforma ModelArts de Huawei y cómo realizar operaciones en la plataforma.

Contenido

1. **Visión panorámica de HUAWEI CLOUD EI**
2. ModelArts
3. Soluciones HUAWEI CLOUD EI

Servicios HUAWEI CLOUD EI

- HUAWEI CLOUD EI es una fuerza motriz para la transformación inteligente de las empresas. HUAWEI CLOUD EI, que depende de la IA y de las tecnologías de datos grandes, proporciona una plataforma abierta, confiable e inteligente a través de servicios en la nube (en modo, como la nube pública o la nube dedicada) que permite a los sistemas de aplicaciones empresariales comprender y analizar ze las imágenes, vídeos, idiomas y textos para satisfacer los requisitos de diferentes escenarios, de modo que cada vez más empresas puedan utilizar los servicios de IA y de Big Data de forma conveniente, acelerando el desarrollo empresarial y contribuyendo al progreso de la sociedad.



HUAWEI CLOUD EI

Historia de desarrollo Huawei CLOUD EI

- Tener CMP y compromisos para proyectos principales1. Hadoop core/HBase: 7

2. Spark + CarbonData: 8

- CarbonData: proyectos de alto nivel de Apache

- CID: La posición de mercado de los servicios de macrodatos de Huawei ocupa el primer lugar en la región de China.
- Derechos de patente: 190 +

- 59 servicios en la nube y 159 funciones
- Intelligent Twins en varios dominios
- Dedicado a la IA inclusiva

Servicios en nube de IE

Servicios Cloud IE

Servicios en nube de IE

Servicios Cloud IE

Plataforma de macrodatos de clase empresarial (FusionInsight)

Autogestión segura y confiable

Soluciones de Big Data (industria de las telecomunicaciones)

Orientado al rendimiento y basado en equipos

Investigación de Big Data

Optimización del núcleo Hadoop y contribuciones de la comunidad

Inteligencia empresarial (BI) tradicionall

(industria de las telecomunicaciones)

Tecnología de análisis y de la LTE

Investigación de IA

Práctica de IA: centrarse en los requerimientos internos de Huawei para soportar actualizaciones inteligentes

2002 **2007** **2011** **2013** **2015** **2017** **2019**

HUAWEI

Historia de desarrollo Huawei CLOUD EI

- Tener CMP y compromisos para proyectos principales1. Hadoop core/HBase: 7

2. Spark + CarbonData: 8

- CarbonData: proyectos de alto nivel de Apache

- CID: La posición de mercado de los servicios de macrodatos de Huawei ocupa el primer lugar en la región de China.
- Derechos de patente: 190 +

- 59 servicios en la nube y 159 funciones
- Intelligent Twins en varios dominios
- Dedicado a la IA inclusiva

Servicios en nube de IE

Servicios Cloud IE

Servicios en nube de IE

Servicios Cloud IE

Plataforma de macrodatos de clase empresarial (FusionInsight)

Autogestión segura y confiable

Soluciones de Big Data (industria de las telecomunicaciones)

Orientado al rendimiento y basado en equipos

Investigación de Big Data

Optimización del núcleo Hadoop y contribuciones de la comunidad

Inteligencia empresarial (BI) tradicionall

(industria de las telecomunicaciones)

Tecnología de análisis y de la LTE

Investigación de IA

Práctica de IA: centrarse en los requerimientos internos de Huawei para soportar actualizaciones inteligentes

2002 2007 2011 2013 2015 2017 2019

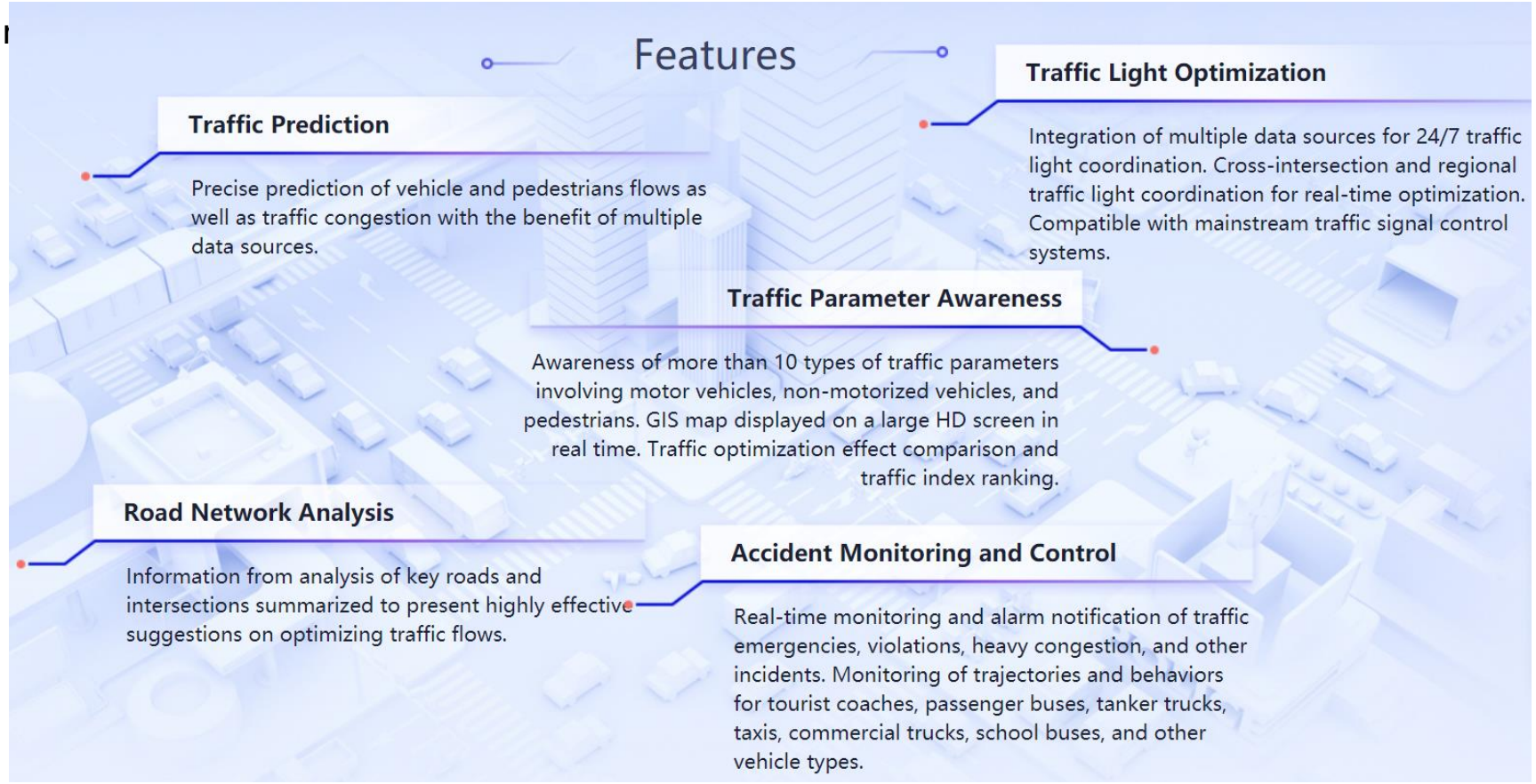
HUAWEI

EI Intelligent Twins

- Los Intelligent Twins de la IE integran las tecnologías de la IA en escenarios de aplicación de diversas industrias y aprovechan plenamente las ventajas de las tecnologías de la IA para mejorar la eficiencia y la experiencia.

Traffic Intelligent Twins (TrafficGo)

- Los Intelligent Twins de Tráfico (Traffic Intelligent Twins) habilitan el monitoreo de condiciones de tráfico 24/7 y todo el área, la detección de incidentes de tráfico, la programación de señales de tráfico en tiempo real, la visualización de situaciones de tráfico en pantalla grande y la gestión de vehículos clave, lo que proporciona una experiencia de viaje eficiente, i

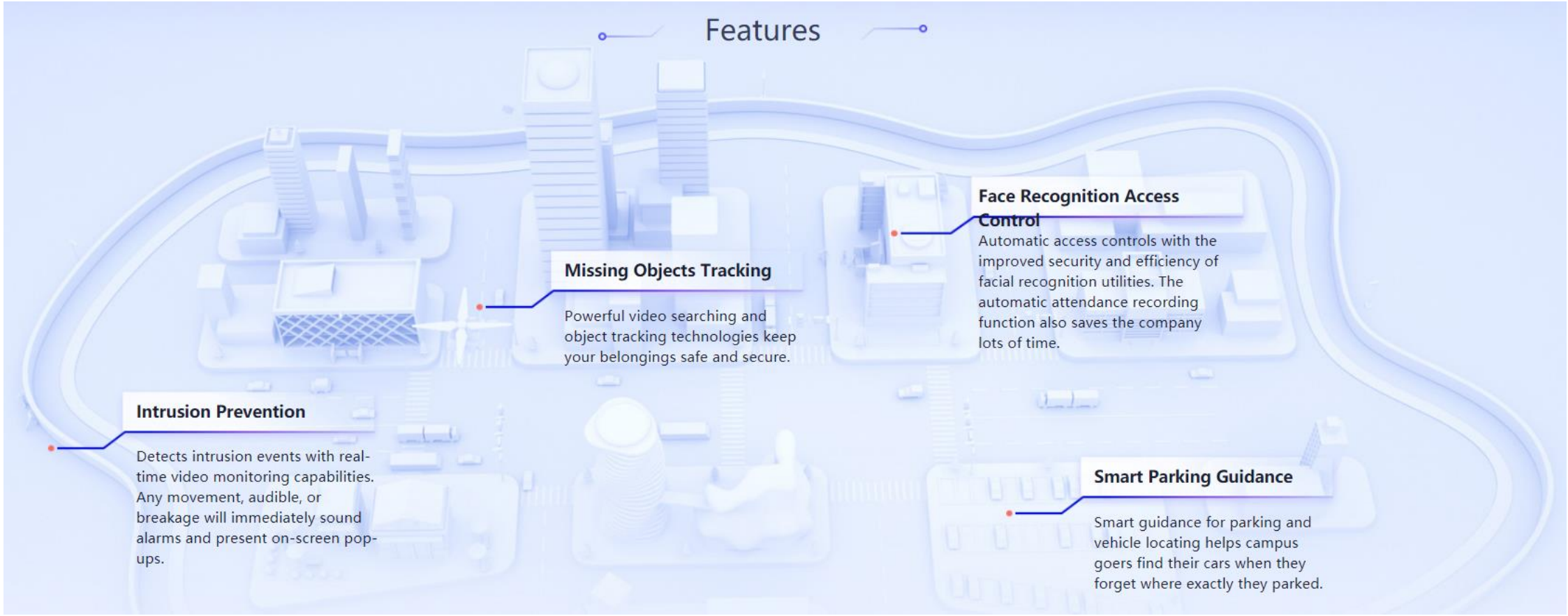


Intelligent Twins industriales

- Los Gemelos Industriales Inteligentes utilizan tecnologías de Big Data e IA para proporcionar una serie completa de servicios que abarcan el diseño, la producción, la logística, las ventas y el servicio. Ayuda a las empresas a obtener una posición de liderazgo.

Intelligent Twins del campus

- Los Campus Intelligent Twins gestionan y supervisan los campus industriales, residenciales y comerciales. Adopta tecnologías de IA como el análisis de vídeo y la minería de datos para hacer nuestro trabajo y vida más conveniente y eficiente.



Productos y servicios de IE

ModelArts

ModelArts is a one-stop development platform that helps AI developers build models and manage the AI development lifecycle with data preprocessing, semi-automated data labeling, and distributed training.

Graph Engine Service

Graph Engine Service (GES) facilitates querying and analysis of graph-structure data based on various relationships. It is specifically suited for scenarios requiring analysis of rich relationship data.

Data Lake Insight

Data Lake Insight (DLI) is a Serverless big data compute and analysis service that is fully compatible with Apache Spark and Apache Flink ecosystems and supports batch streaming.

Video Ingestion Service

Video Ingestion Service (VIS) ingests massive volumes of video data in real time. Its superb data collection, real-time transmission, and video retention capabilities allow easy intelligent video analysis.

Data Warehouse Service

Data Warehouse Service (DWS) is a fast, easy-to-use, and reliable enterprise-class converged data warehouse service that can extend queries and analysis to your data lake with the help of DWS Express.

Cloud Stream Service

Cloud Stream Service (CS) is designed to process streaming data in real time. Computing clusters are fully managed by CS, allowing you to run StreamSQL or custom jobs without learning any programming skills.

MapReduce Service

MapReduce Service (MRS) provides enterprise-level big data clusters on the cloud. Tenants can fully control clusters and easily run big data components such as Hadoop, Spark, HBase, Kafka, and Storm.

Question Answering Bot

Question Answering Bot (QABot) helps enterprises quickly build, publish, and manage intelligent Q&A bots.

Image Recognition

Image Recognition uses deep learning technologies to accurately identify objects, scenes, and concepts in images using a pool of visual content tags.

Plataforma esencial de IE

ModelArts

Plataforma de desarrollo integral de IA

HiLens de Huawei

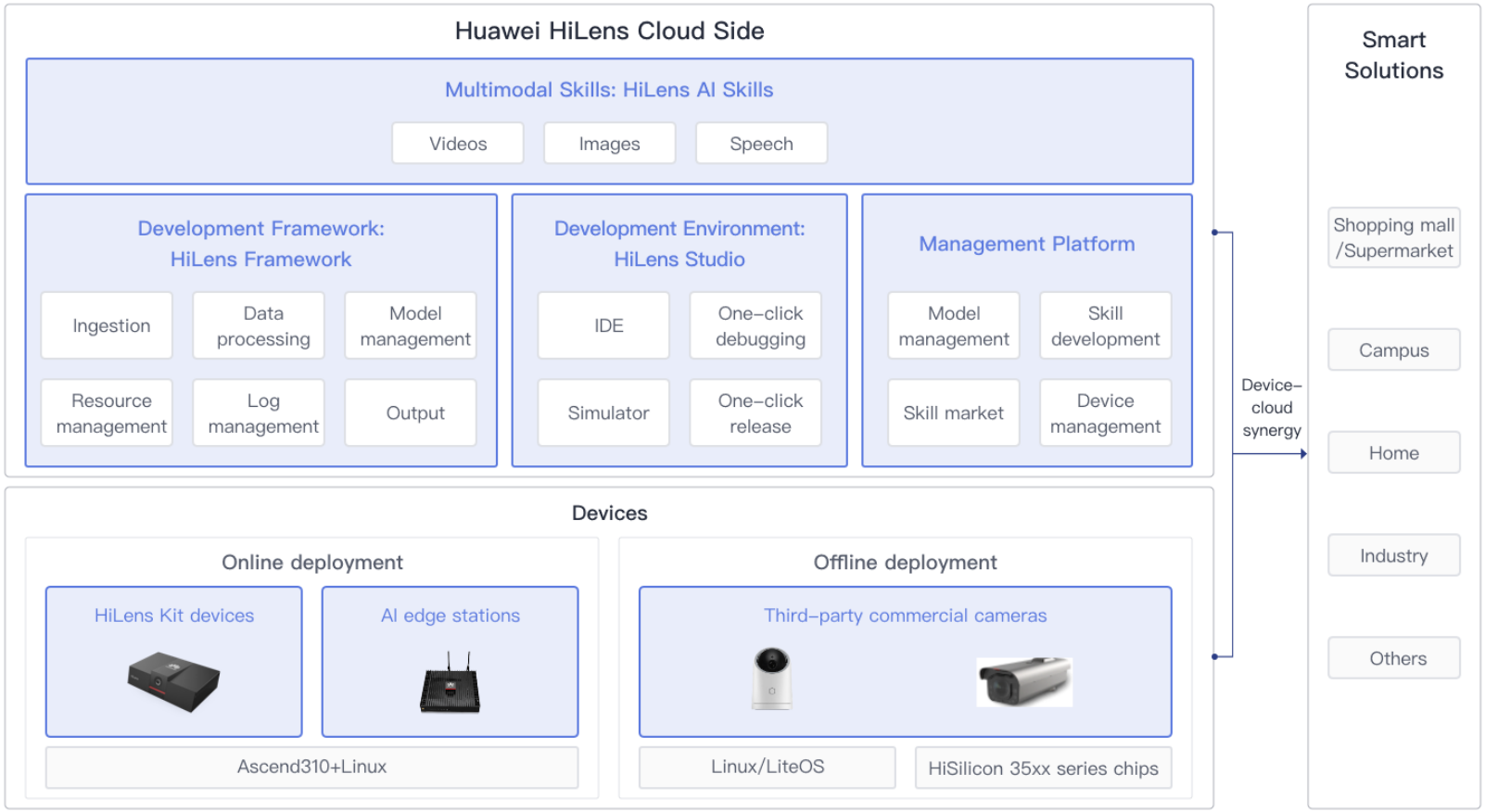
Plataforma de desarrollo de IA multimodal que permite la sinergia entre dispositivos y cloud

Graph Engine Service (GES)

Primer motor gráfico nativo distribuido autoconstruido comercial con derechos de propiedad intelectual independientes en China

HiLens de Huawei

- El sistema HiLens de Huawei incluye dispositivos de computación y una plataforma de desarrollo basada en la nube, y ofrece un marco de desarrollo, un entorno de desarrollo y una plataforma de gestión para ayudar a los usuarios a desarrollar aplicaciones de IA multimodales y entregarlas a los dispositivos, para implementar soluciones inteligentes en múltiples escenarios.



SES

GES facilita la consulta y el análisis de los datos de la estructura de gráficos basados en diversas relaciones. Usa el motor de gráficos de alto rendimiento EYWA como su núcleo, y se le otorgan muchos derechos de propiedad intelectual independientes. GES desempeña un papel importante en escenarios tales como aplicaciones sociales, aplicaciones de análisis de relaciones empresariales, distribución logística, planificación de rutas de autobús, gráfico de conocimientos empresariales y control de riesgos.

- Relaciones sociales
- Registros de transacciones
- Registros de llamadas
- propagación de información
- Exploración de registros
- Redes de tráfico
- Redes de comunicaciones
- **Los datos asociados masivos y complejos son datos gráficos en la naturaleza.**

GES

- Datos diversificados independientes de las estructuras
- Capacidad de propagación y asociación de datos
- Cambios dinámicos de datos y análisis interactivo en tiempo real sin formación
- Resultados visualizados e interpretables

Individual Analisis

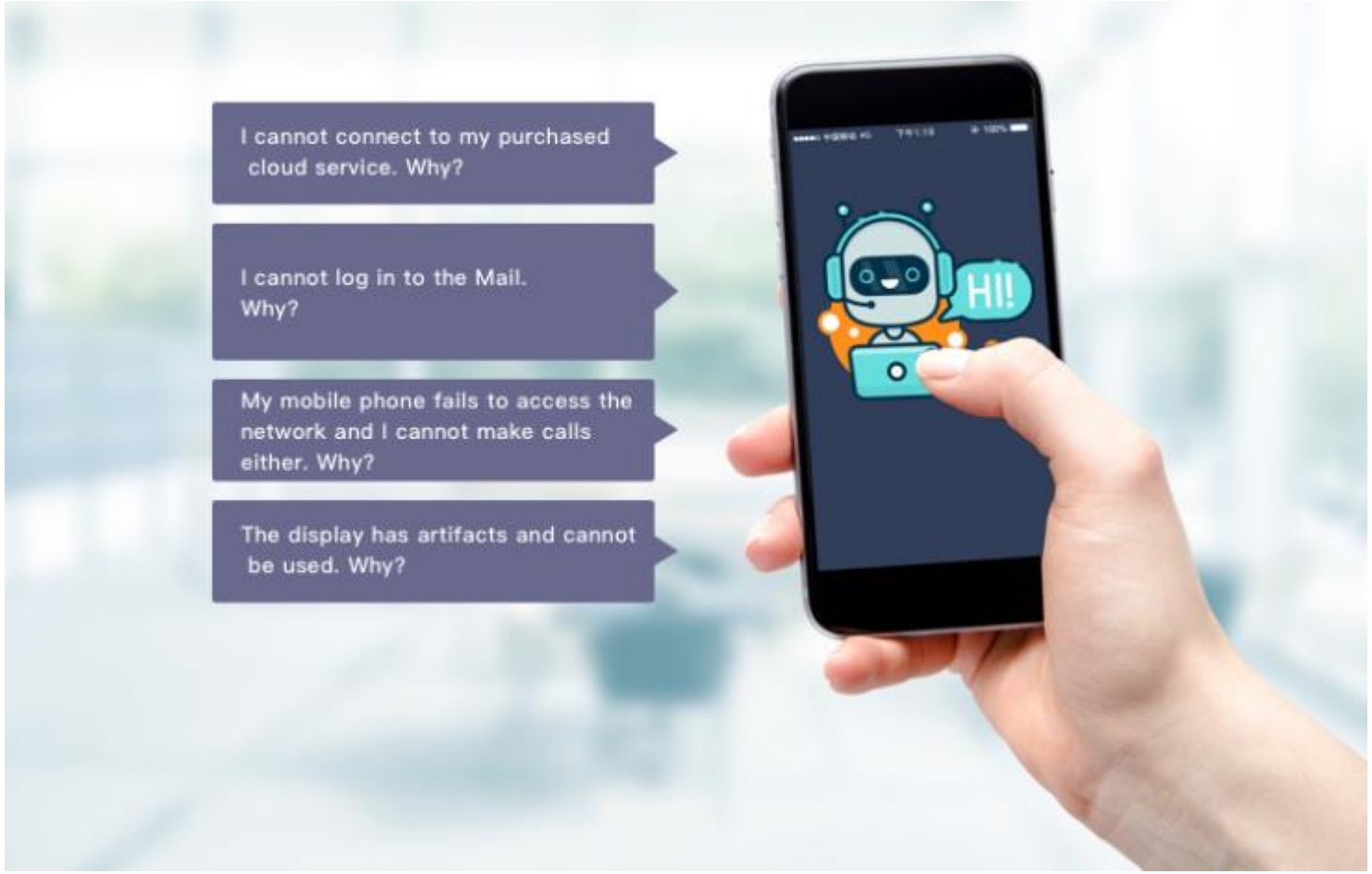
Agrupar Analisis

Vínculo Analisis

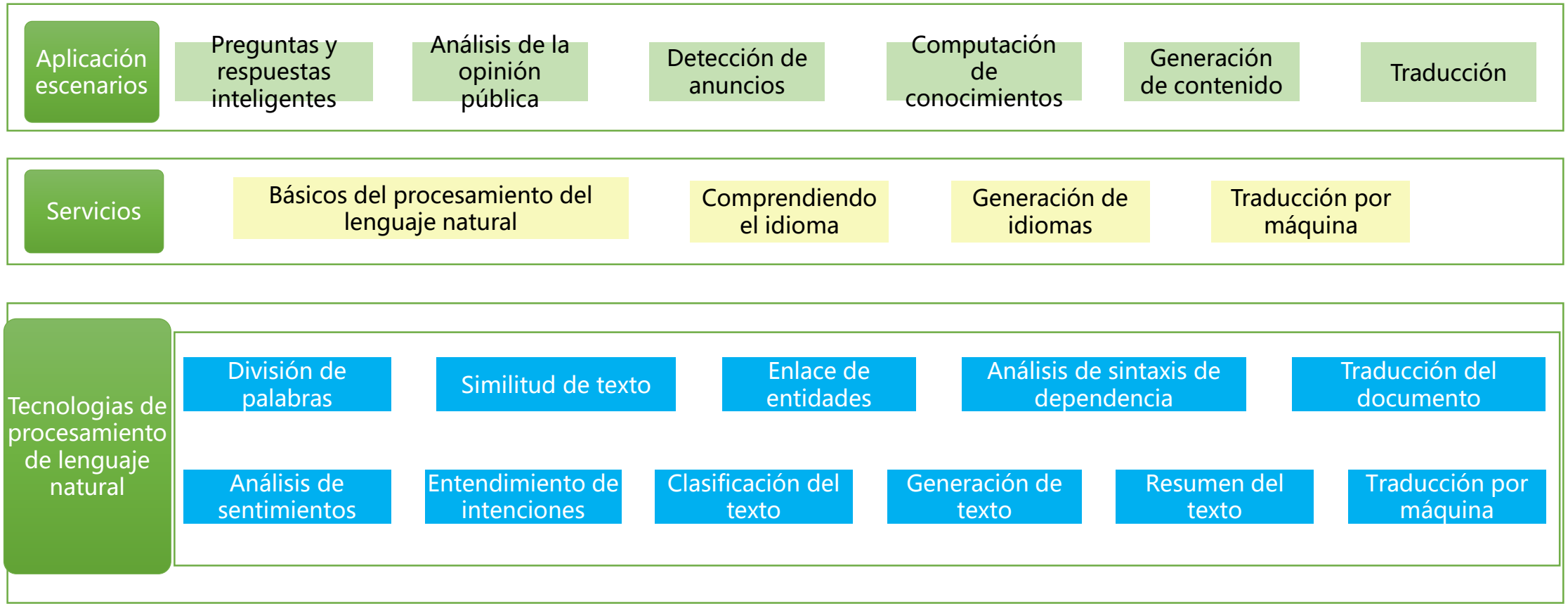
HUAWEI

Servicio de Bot Conversacional (CBS)

- robot de Preguntas-Respuestas(QABot)
- robot de conversación orientado a tareas (TaskBot)
- Análisis del habla (CBS-SA)
- Personalización del CBS



Procesamiento de lenguaje natural



Interacción de voz

Reconocimiento de palabras y enunciados cortos

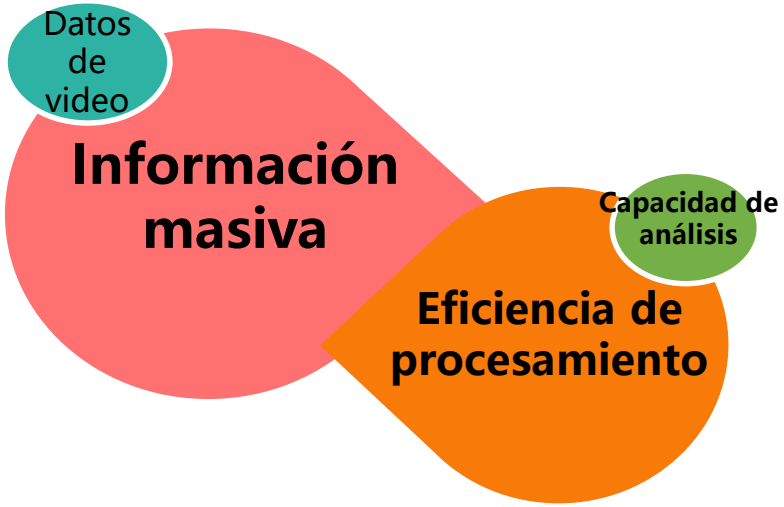
Reconocimiento de voz en tiempo real

Reconocimiento de grabación de audio

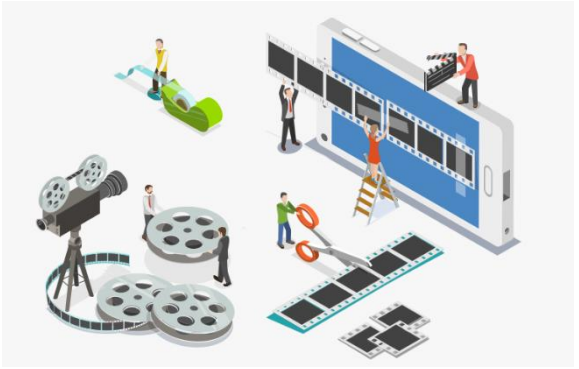
Audiobooks

HUAWEI

Análisis de vídeo



Análisis de contenido de video

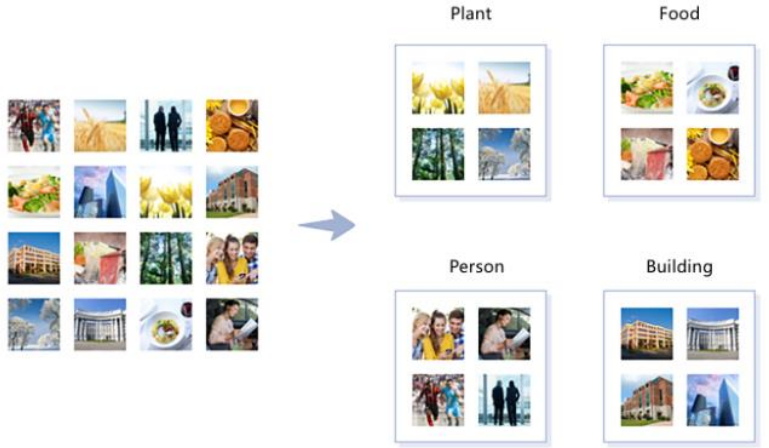


Proporcione las capacidades de cubierta, división y resumen basadas en el análisis de vídeo general.

Edición de vídeo

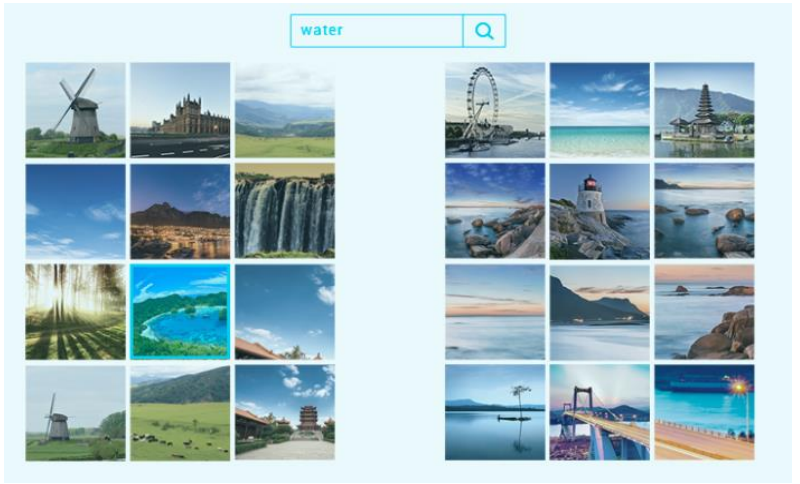
Reconocimiento de imagen

Análisis de escenarios



Álbum inteligente

Detección de objetos



Recuperación de imágenes

HUAWEI

Moderación del contenido

La moderación de contenido adopta tecnologías de vanguardia de detección de imágenes, texto y vídeo que detectan con precisión anuncios, material pornográfico o relacionado con el terrorismo e información política sensible, reduciendo los riesgos de incumplimiento en su negocio.

Moderación (imagen y texto)

Sexy.

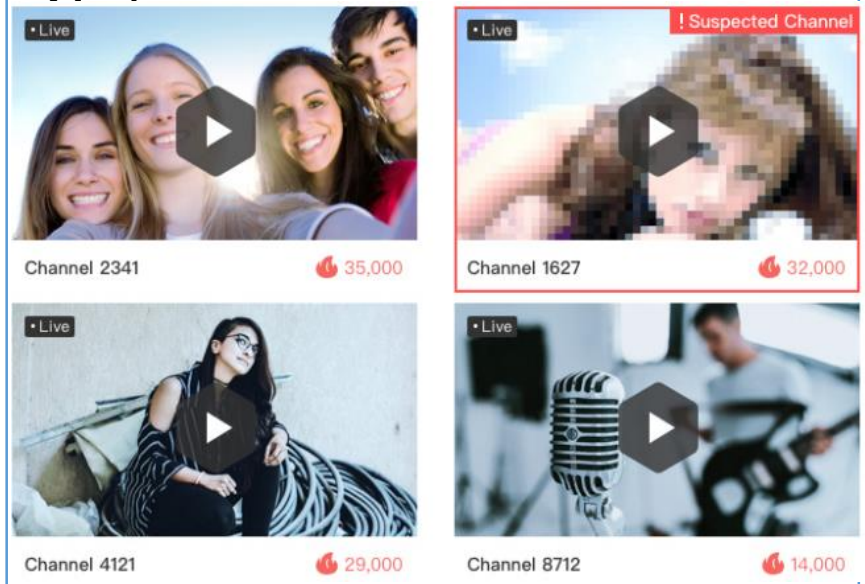
Pornografico

Identificación de contenido obsceno

Identificación del terrorismo

Detección de información política

Moderación (contenido de

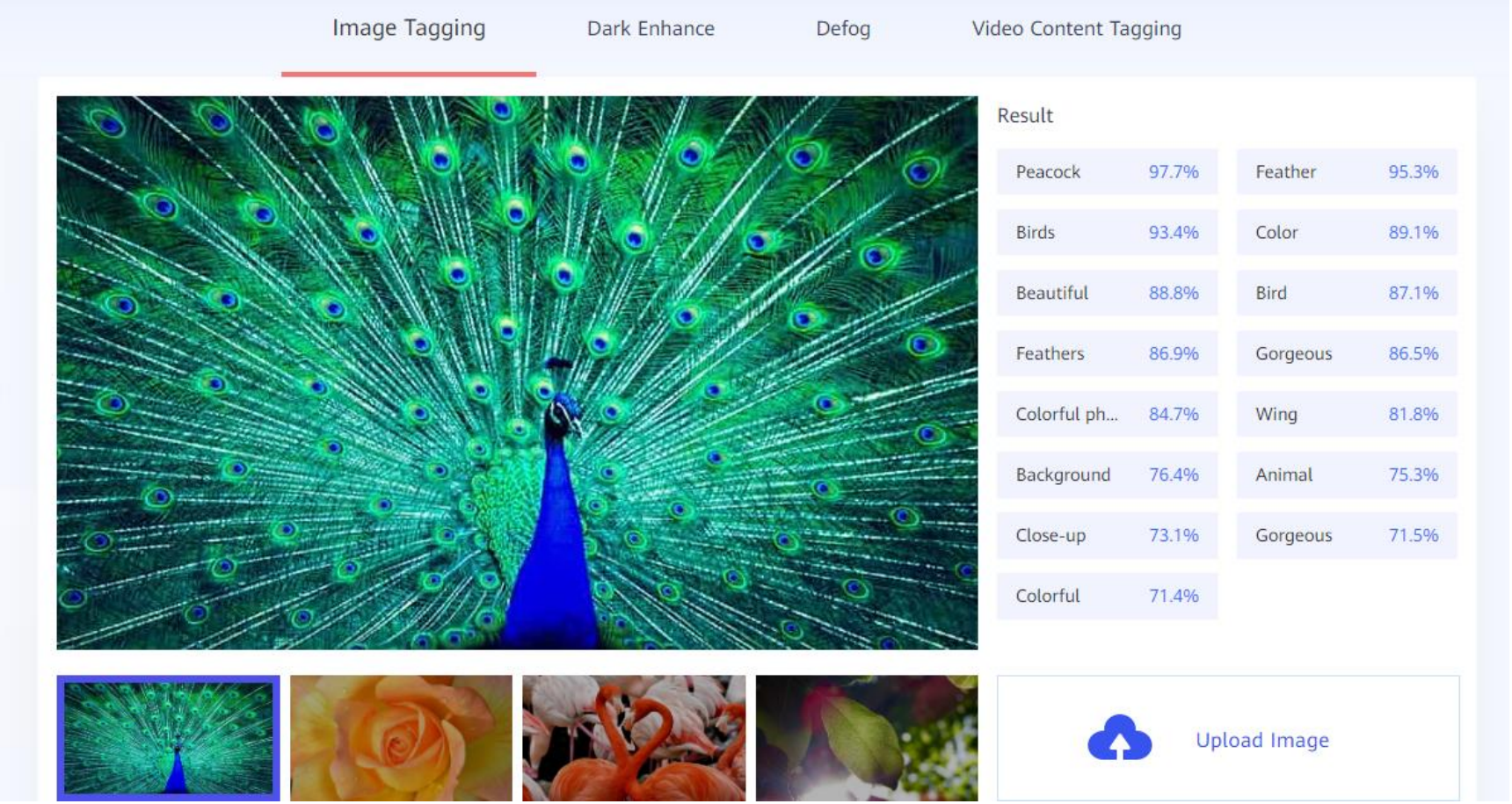


Determine si un vídeo tiene riesgos de incumplimiento y proporcione información de incumplimiento de varias dimensiones, como imagen, sonido y subtítulos.

HUAWEI

Centro de experiencia EI

- El centro de experiencia de EI es una ventana de experiencia de IA construida por Huawei, dedicada a reducir el umbral para el uso de IA y hacer la IA ubicua.



Contenido

1. Aspectos generales de Huawei Cloud EI
2. **ModelArts**
3. Soluciones de Huawei Cloud EI

ModelArts

- ModelArts es una plataforma de desarrollo integral para desarrolladores de IA. Con preprocesamiento de datos, etiquetado de datos semiautomático, formación distribuida a gran escala, modelado automático e implementación de modelos bajo demanda en dispositivos, bordes y nubes, ModelArts ayuda a los desarrolladores de IA a construir modelos rápidamente y gestionar el ciclo de vida del desarrollo de IA.

Funciones de ModelArts

Data Management

ModelArts manages data preparation, such as collection, filtering, and labeling, and dataset versions, especially for deep learning datasets.

Rapid and Simplified Model Training

Huawei's MoXing deep learning framework enables high-performance distributed training. To accelerate model development, it uses automatic hyperparameter tuning and standalone and distributed training.

Model Deployment

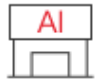
ModelArts deploys models in various production environments such as devices, the edge, and the cloud, and supports online and batch inference jobs.

ExeML

ModelArts supports code-free modeling and auto learning with image classification, object detection, and predictive analytics.

Visualized Workflow

ModelArts works with Graph Engine Service (GES) to manage and visualize the lifecycle of AI development workflows, implementing data and model lineage.



AI Marketplace

ModelArts supports common models and datasets, and internal or public sharing of enterprise models in the marketplace.

Aplicaciones de ModelArts

Ciclo de vida de desarrollo de la IA

Datos:

Preparación de datos

- Tres escenarios (imagen, voz y texto)
- Siete escenas de etiquetado

Edificio Modelo

- Desarrollo inmediato y en línea
- Informática potente y desarrollo acelerado

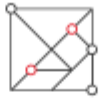
Despliegue de modelos

- Alto rendimiento y baja latencia
- Inferencia por lotes
- Combinado con HiLens y fácilmente desplegable en dispositivos

Factores destacados de ModelArts

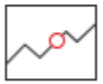
One-Stop Platform

The out-of-the-box and full-lifecycle AI development platform provides one-stop data processing, model development, training, management, and deployment.



Easy to Use

Various built-in open source models and automatic hyperparameter tuning help you start model training from scratch. Models can be deployed on the device, edge, and cloud with just one click.



Excellent Performance

The Huawei-developed MoXing framework delivers high-performance algorithm development and training. GPU utilization is optimized for online inference. Huawei Ascend chips significantly accelerate inference.

High Flexibility

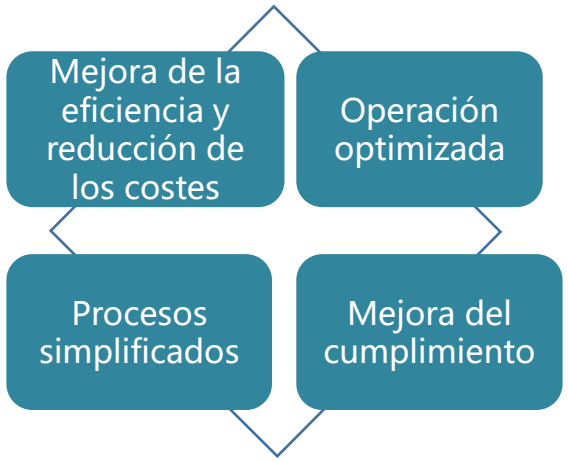
ModelArts supports multiple mainstream open source frameworks, such as TensorFlow and Apache Spark MLlib, mainstream GPUs, and the Huawei-developed Ascend AI chips. Exclusive use of resources and custom images ensure flexible development experience.

Contenido

1. Panorámica de HUAWEI CLOUD EI
2. ModelAarts
3. **Soluciones HUAWEI CLOUD EI**

Caso: OCR implementa la automatización de proceso completo para el reembolso a través de facturas.

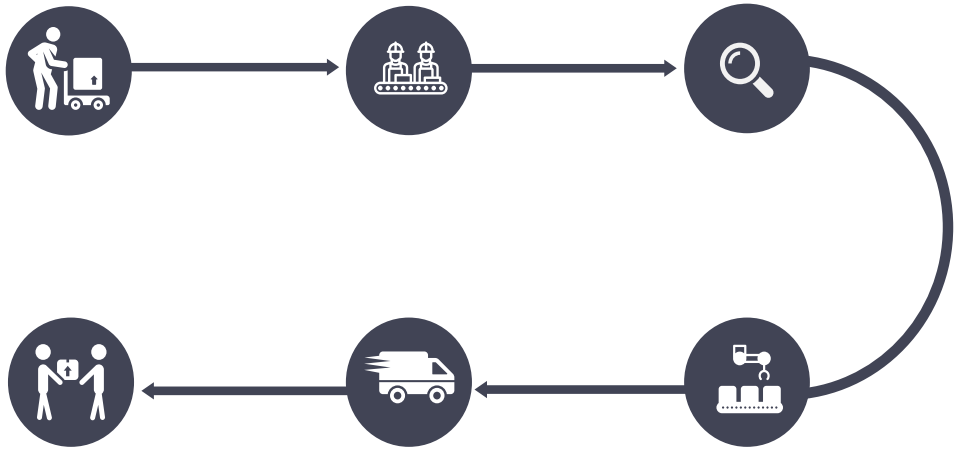
- **Múltiples modos de acceso:** conexión automática a los escáneres para obtener imágenes en lotes; captura de imágenes mediante escáneres de documentos de alta velocidad y teléfonos móviles
- **Implementación flexible:** modos de implementación múltiples, como nube pública, HCS y equipos, y API estándar unificadas
- **Soporte de diversas facturas:** facturas regulares/especiales/electrónicas/ETC/IVA (IVA), y facturas de taxi, tren, itinerario de vuelo, cuota y peaje
- **Una imagen para múltiples facturas:** clasificación automática e identificación de múltiples tipos de facturas
- **Comparación visualizada:** devuelve la información de ubicación de caracteres de OCR y la convierte en un archivo Excel para la recopilación y análisis de estadísticas



Caso: Logística Inteligente con OCR

➢ **Tarjeta de ID OCR**
- Fotografía, reconocimiento y verificación de tarjetas de identidad con aplicaciones móviles

➢ **Captura de pantalla OCR**
- Una vez que una plataforma de comercio electrónico recibe la dirección del comprador y las capturas de chat, OCR reconoce y extrae la información automáticamente.



➢ **OCR de conocimiento de correo electrónico**
- Extracción automática: número de conocimiento de embarque y nombre, número de teléfono y dirección del receptor/expediente

➢ **OCR de la carta de porte**
- Detección de textos y sellos

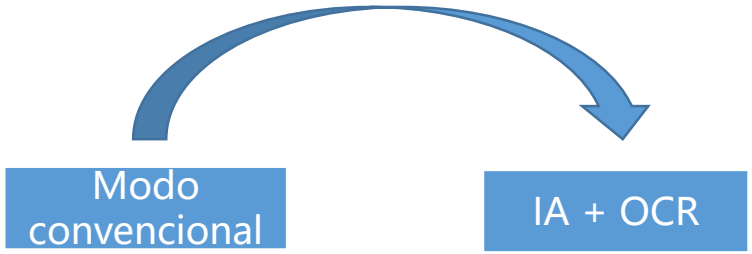
➢ **OCR de recepción**
- Reconocimiento de información de la factura

Eficiencia

Precisión

Coste

Privacidad



Servicio 24/7, identificación de una sola guía de circulación en sólo 2 segundos

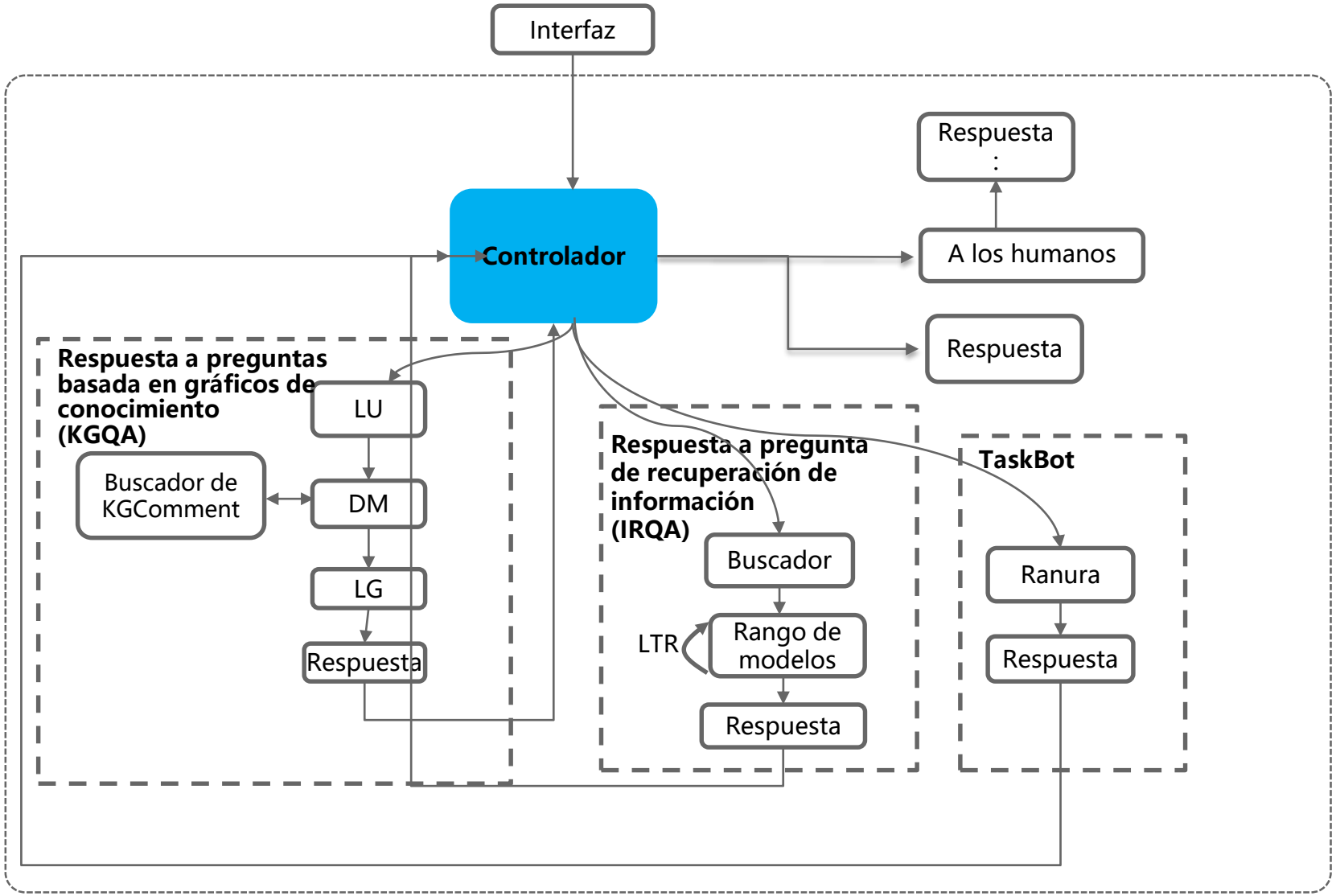
Hasta un 98 % de precisión, reduciendo la repetición innecesaria de disparos y eliminando interferencias externas

Proceso automatizado optimizado, reduciendo la intervención manual y los costes

Identificación automática sin intervención manual, garantizando la seguridad de la privacidad

HUAWEI

CBS



Integración inteligente de múltiples robots para servicios más completos

Los robots con sus respectivas ventajas están integrados y pueden aprender automáticamente el conocimiento y optimizar los servicios para recomendar respuestas óptimas a los clientes.

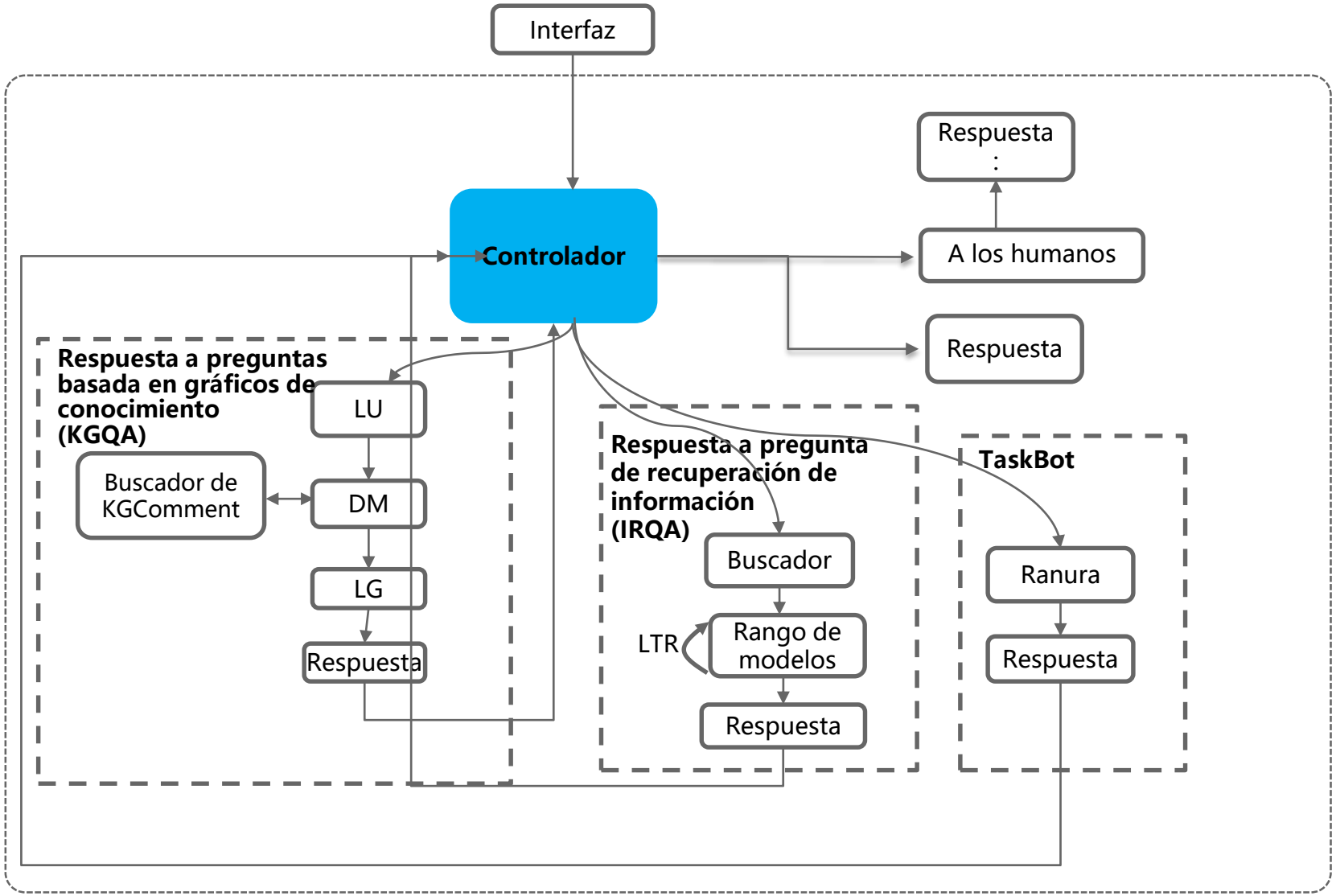
Guía inteligente a través de múltiples rondas de interacciones para comprenderle mejor

Se realizan múltiples rondas de interacciones naturales para identificar con precisión las intenciones de los usuarios y comprender los significados ocultos de los usuarios.

Gráficos de conocimiento para obtener más inteligencia

Modelos lingüísticos de dominio común + gráficos de conocimiento de dominio
Actualizaciones dinámicas del contenido del gráfico
Más inteligencia de robots basados en gráficos

CBS



Integración inteligente de múltiples robots para servicios más completos

Los robots con sus respectivas ventajas están integrados y pueden aprender automáticamente el conocimiento y optimizar los servicios para recomendar respuestas óptimas a los clientes.

Guía inteligente a través de múltiples rondas de interacciones para comprenderle mejor

Se realizan múltiples rondas de interacciones naturales para identificar con precisión las intenciones de los usuarios y comprender los significados ocultos de los usuarios.

Gráficos de conocimiento para obtener más inteligencia

Modelos lingüísticos de dominio común + gráficos de conocimiento de dominio
Actualizaciones dinámicas del contenido del gráfico
Más inteligencia de robots basados en gráficos

Caso: Botón Conversacional con Conocimiento de Vehículos

Preguntas sobre las recomendaciones de los vehículos

Recomendaciones

- ✓ Por favor, recomienda un coche a un precio de alrededor de 30,0000 CNY.
- ✓ Por favor, recomienda un vehículo comercial.
- ✓ Recomienda modelos de vehículos con un par superior a 310 Nm.

Cuestiones relativas a la evaluación de los vehículos

Consultoría abierta

- ✓ ¿Qué tal Mercedes-Benz E?
- ✓ ¿Cómo está la seguridad del S90?
- ✓ ¿Cómo es el interior de Audi A6L?

Preguntas sobre la comparación de modelos de vehículos

Comparación

- ✓ Comparación entre Mercedes-Benz E300L Sedan y BMW 530Li xDrive
- ✓ ¿Cuál de las series Audi A6L y BMW 5 es mejor?

Consulta de preguntas sobre modelos de vehículos especificados

Rendimiento

- ✓ ¿Cuál es la base de ruedas de 530 Li?
- ✓ ¿Cuál es el consumo de combustible de Mercedes-Benz E300L?
- ✓ ¿Se pueden calentar los asientos de Lexus ES?

Preguntas posteriores a la venta

Después de la venta

- ✓ ¿Qué puedo hacer si el volante no se mueve?
- ✓ ¿Por qué el indicador sigue parpadeando? (Figura)
- ✓ ¿Cuánto tiempo se pagará el seguro?

Preguntas sobre la reserva de conducción de pruebas

Preventa

- ✓ ¿Está disponible la conducción de pruebas para el S90?
- ✓ ¿Qué tienda 4S tiene inventario?
- ✓ ¿Puedo probar Audi A4L el próximo miércoles?

Respuestas precisas basadas en gráficos de conocimiento del vehículo

¿Qué tal el sedan deportivo Mercedes-Benz A200?

La Mercedes-Benz A200 deportiva... está equipado con... (detalles)

¿Cómo está el desempeño de seguridad?

El frenado activo se proporciona en la configuración estándar... (Introducir las ventajas de seguridad.) La solución única xxx se proporciona además...

Múltiples rondas de interacciones y entradas multimodales

¿Cómo está el interior?

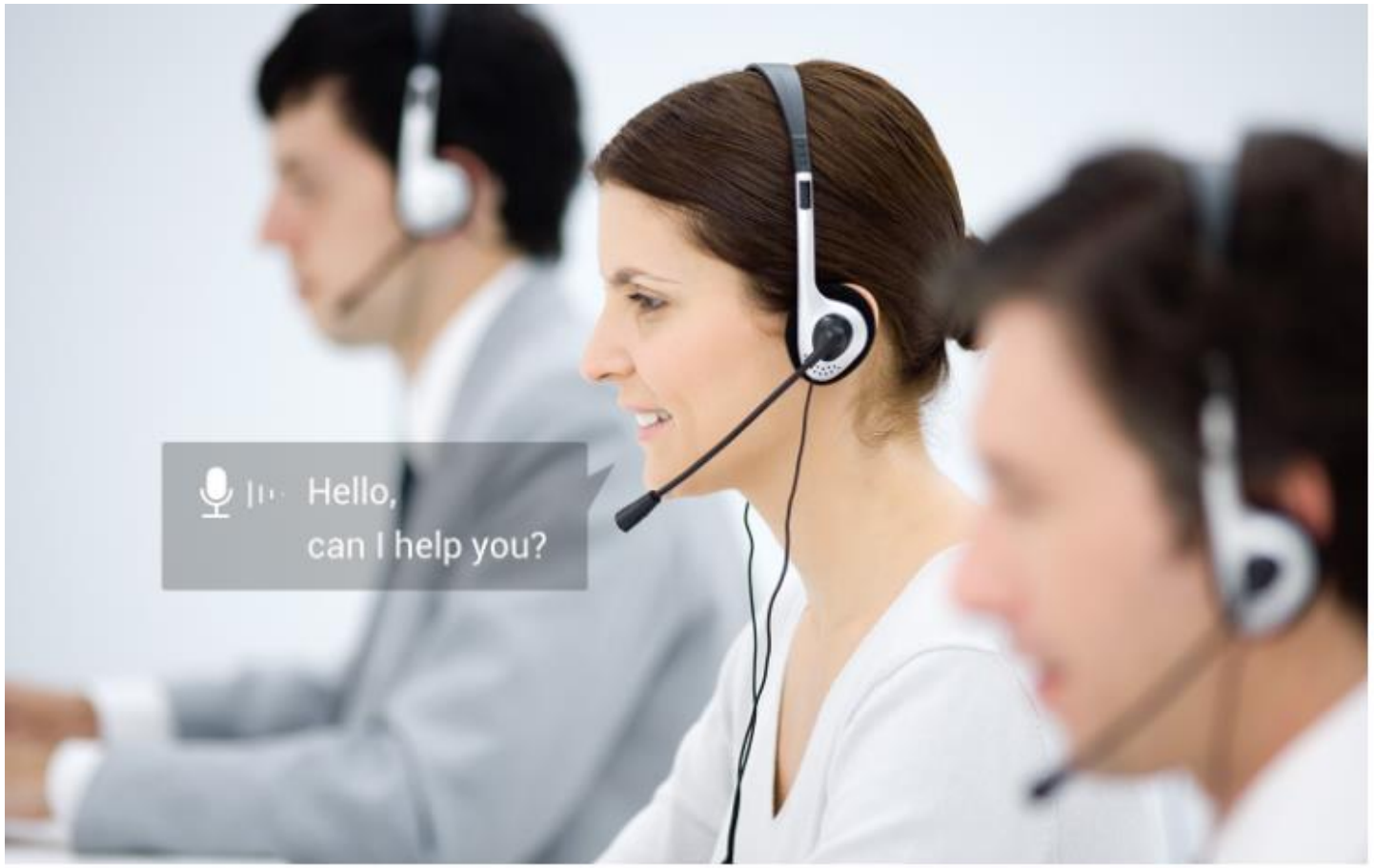
... xxx de cuero superior... (descripción del interior)

Orientación proactiva y respuestas preferenciales

¿Cuál es la diferencia entre S90 T5 y BMW 530Li?

El precio de S90-T5 es CNY410.800, y la de la BMW 530Li es CNY519.900.
BMW 530Li tiene 9 configuraciones estándar como puntos de venta, mientras que S90 T5 tiene 10. El costo de BMW 530Li aumenta si se necesita agregar una configuración adicional para BMW 530Li para igualar S90 T5.

Caso: Preguntas y Respuestas Inteligentes de las Empresas en un determinado Distrito



Agent Assistant

Improves productivity and customer satisfaction with real-time analysis and helps on improving the interaction between agents in the call center and customers. During a call, the bot automatically extracts keywords, coils problems, searches for and displays the best answers to matching semantics, and provides real-time support for agents.

Advantages

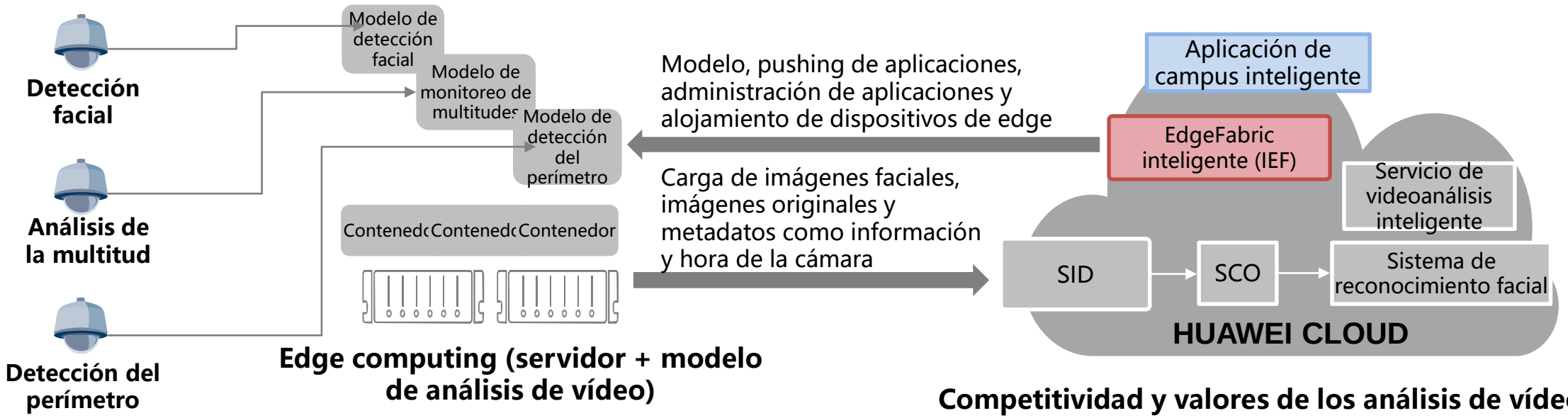
- **Real-Time Support**

  Offers real-time support for human agents by attempting to understand customer questions and matching the answers with high relevancy.

- **Improved Efficiency & Satisfaction**

  Enables human agents to answer customers' questions at a faster speed.

Caso: Campus inteligente



Edge computing (servidor + modelo de análisis de vídeo)

IPC de alta definición comunes del lado del dispositivo:

- Captura de rostros
- Análisis de vídeo en el borde

Campus de vigilancia

Lado del borde:

- Se recomienda utilizar servidores GPU.
- El FEI presiona los algoritmos de detección facial, monitoreo de multitudes y detección de perímetro para su despliegue en nodos de borde.
- La FEI gestiona el ciclo de vida de la aplicación (con los algoritmos optimizados iterativamente).
- La FIE gestiona de forma centralizada los contenedores y las aplicaciones de borde.

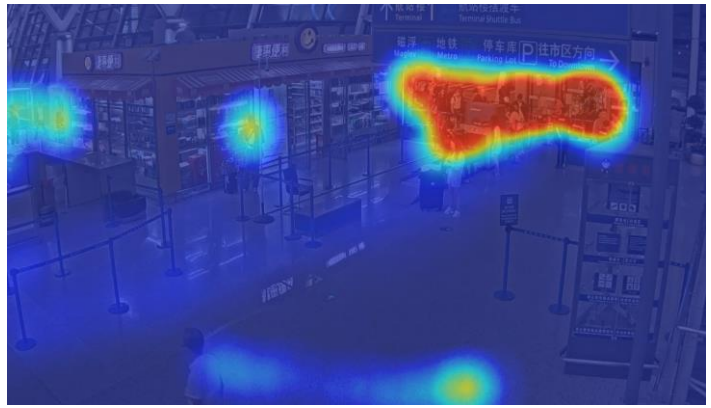
Competitividad y valores de los análisis de vídeo de vanguardia

1. Valores de servicio: Analice de forma inteligente los videos de vigilancia para detectar eventos de seguridad anormales, como intrusiones y reuniones de gran cantidad de personas en tiempo real, lo que reduce los costos de mano de obra.
2. Sinergia Edge-Cloud: Lleve a cabo la gestión del ciclo de vida completo y la actualización sin problemas de las aplicaciones de vanguardia.
3. Capacitación sobre modelos en la nube: Implementar capacitación automática utilizando algoritmos que tengan buena escalabilidad y fácil actualización.
4. Alta compatibilidad: reutilizar los IPC existentes en los campus como cámaras inteligentes mediante la sinergia edge-cloud.

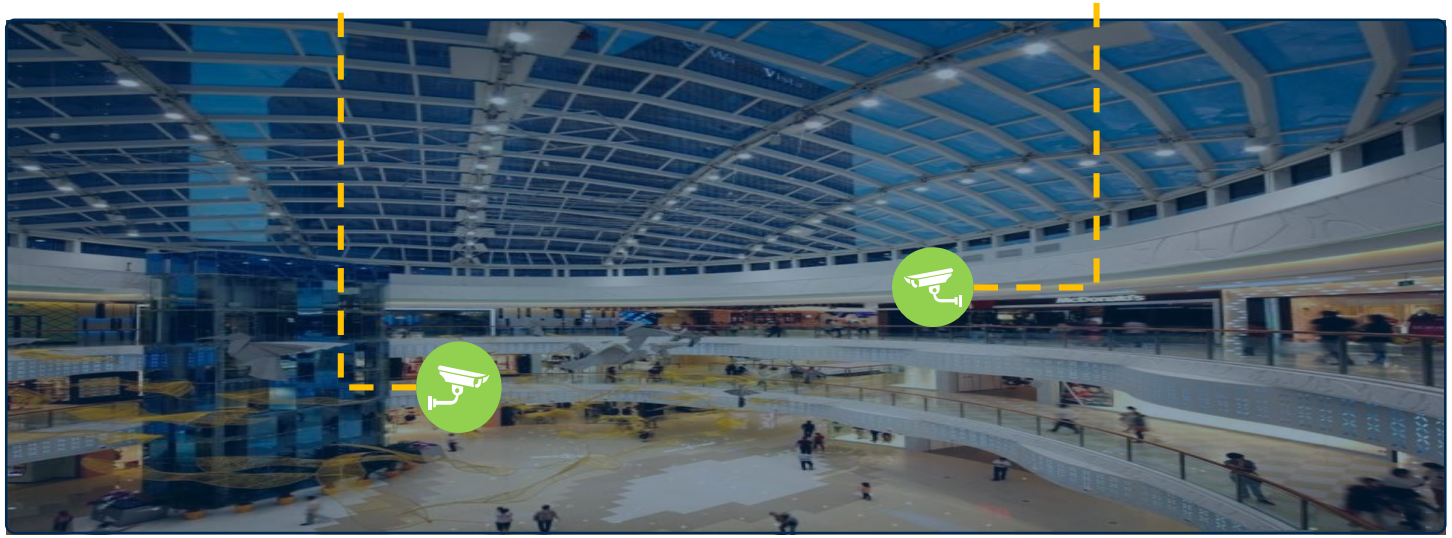
HUAWEI

Caso: Estadísticas de multitudes y mapa de calor

Estadísticas de multitudes de regiones

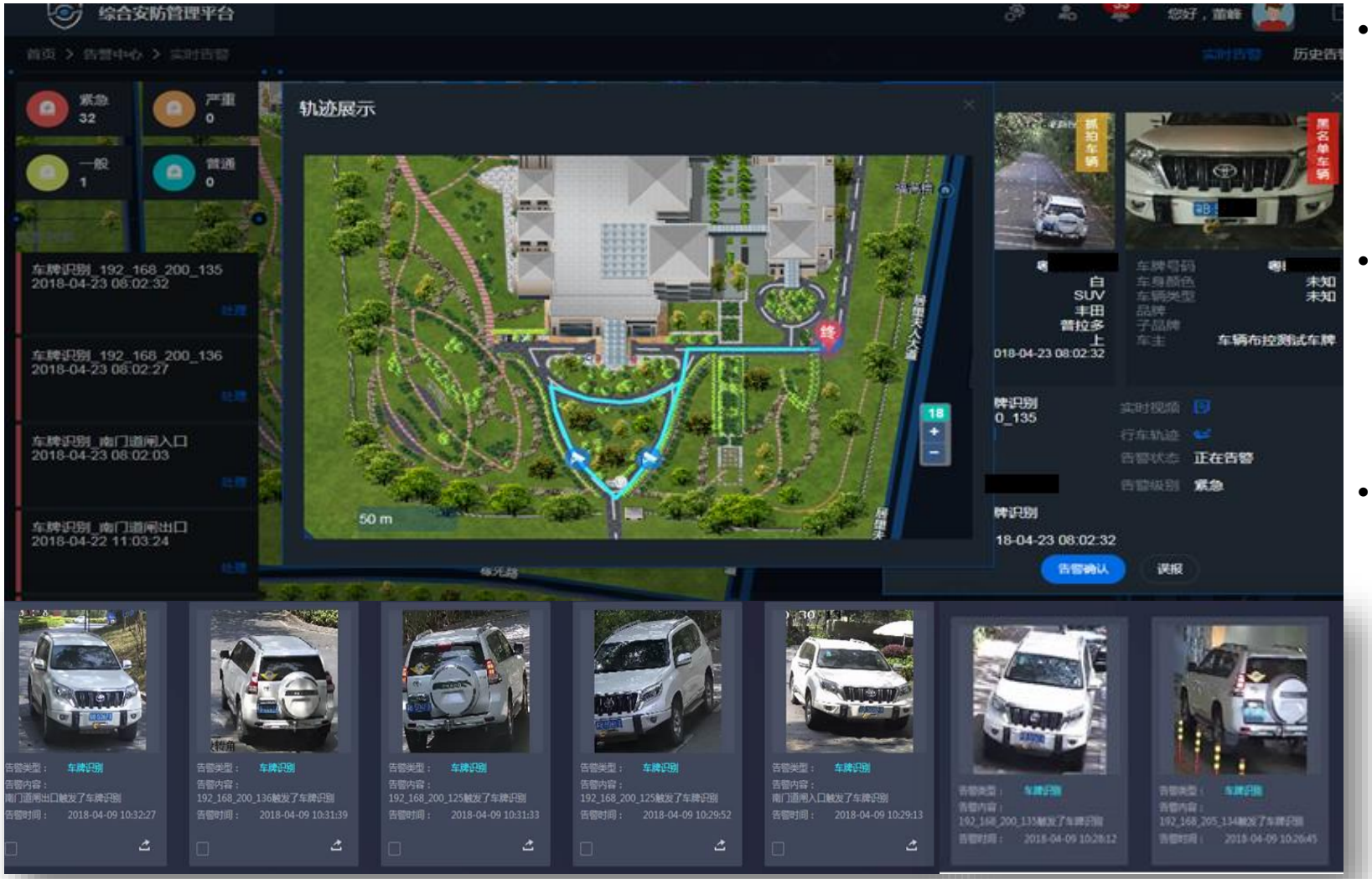


Mapa de calor de la región



- Funciones:
  - Conteo de multitudes en una imagen.
  - Recopilación de estadísticas de popularidad de una imagen.
  - Soporte de ajustes de tiempo personalizados.
  - Activar intervalos configurables para el envío de resultados estadísticos.
- Escenarios:
  - Estadísticas de tráfico de clientes
  - Estadísticas de visitantes
  - Identificación de la popularidad del distrito empresarial
- Ventajas:
  - Fuerte rendimiento anti-interferencia: contado de multitudes en escenarios complejos, como bloqueo de la cara y bloqueo parcial del cuerpo
  - Alta escalabilidad: envío simultáneo de estadísticas de la región de cruce peatonal y estadísticas de mapas de calor
  - Facilidad de uso: compatible con cualquier cámara de vigilancia de 1080p

Caso: Reconocimiento del vehículo



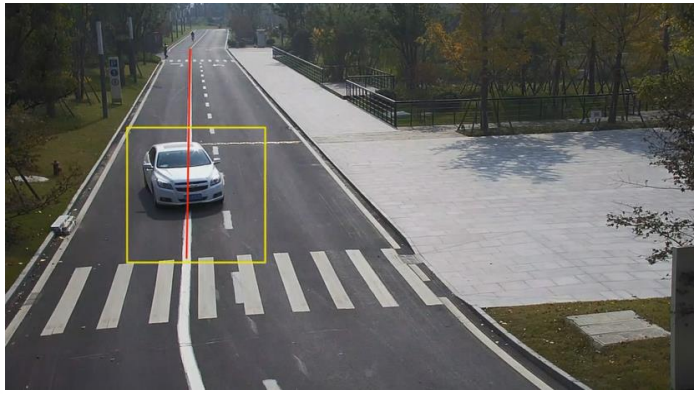
- **Funciones:**
  - Detección del modelo de vehículo
  - Reconocimiento del color del vehículo
  - Reconocimiento de placas de licencia (LPR)
- Escenarios:
  - Gestión de vehículos del campus
  - Gestión de vehículos de estacionamiento
  - Seguimiento del vehículo
- **Ventajas:**
  - Posibilidades completas: reconocimiento de modelos de vehículos, estilos, colores y matrículas en diversos escenarios, como ePolice y puntos de control
  - Facilidad de uso: Compatible con cualquier cámara de vigilancia de 1080p

Caso: Detección de intrusión

Detección de pasos de cables de trampa para el personal

Detección de intrusiones de área

Detección de escalada



Detección de cruce de cable de trampa para vehículos

Funciones:

- Extraer objetos en movimiento del campo de visión de una cámara y generar una alarma cuando un objeto cruza un área especificada.
- Ajuste del número mínimo de personas en un área de alarma.
- Configuración de la hora de activación de alarmas.
- Configuración del período de detección de algoritmos.

Escenarios:

- Identificación del acceso no autorizado a zonas clave
- Identificación del acceso no autorizado a zonas peligrosas
- Detección de escalada

Ventajas:

- ✓ **Alta flexibilidad: configuración del tamaño y tipo de un objeto de alarma**
- ✓ **Baja tasa de notificación incorrecta: personas inv #/inv # alarma de intrusión basada en vehículos, sin interferencia de otros objetos**
- ✓ **Fácil de usar: compatible con cualquier cámara de vigilancia de 1080 p**

Contenido

- Este capítulo describe los siguientes contenidos:
  - El ecosistema Huawei Cloud EI, y ayuda a comprender los servicios de Huawei Cloud EI
  - Servicios de ModelArts en combinación con experimentos, ayudándole a entender los servicios de ModelArts de forma más eficiente
  - Casos de IE

Quiz

1. ¿A cuál de los siguientes escenarios se puede aplicar IE? ()
   A. Gobierno inteligente
   B. Ciudad inteligente
   C. Fabricación inteligente
   D. Finanzas inteligentes

Más información

Sitio web de Huawei Talent

https://e.huawei.com/en/talent/#/home

WeChat cuentas publicas:

IME

Laboratorio abierto de Huawei Device

Desarrolladores de Huawei

Smart-E

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home, and organization for a fully connected, intelligent world.

HUAWEI