AI Overview

HUAWEI

Foreword

- Mankind is welcoming the fourth industrial revolution represented by intelligent technology. New technologies such as AI, IoT, 5G and bioengineering are integrated into all aspects of human society; driving changes in global macro trends, such as sustainable social development and economic growth. New kinetic energy, smart city upgrading, industrial digital transformation, consumer experience, etc.
- As the world's leading provider of ICT (information and communications) infrastructure and smart terminals, Huawei actively participates in the transformation of artificial intelligence and proposes Huawei's full-stack full-scenario AI strategy. This chapter will mainly introduce AI Overview, Technical Fields and Application Fields of AI, Huawei's AI Development Strategy, AI Disputes, Future Prospects of AI.

HUAWEI

Objectives

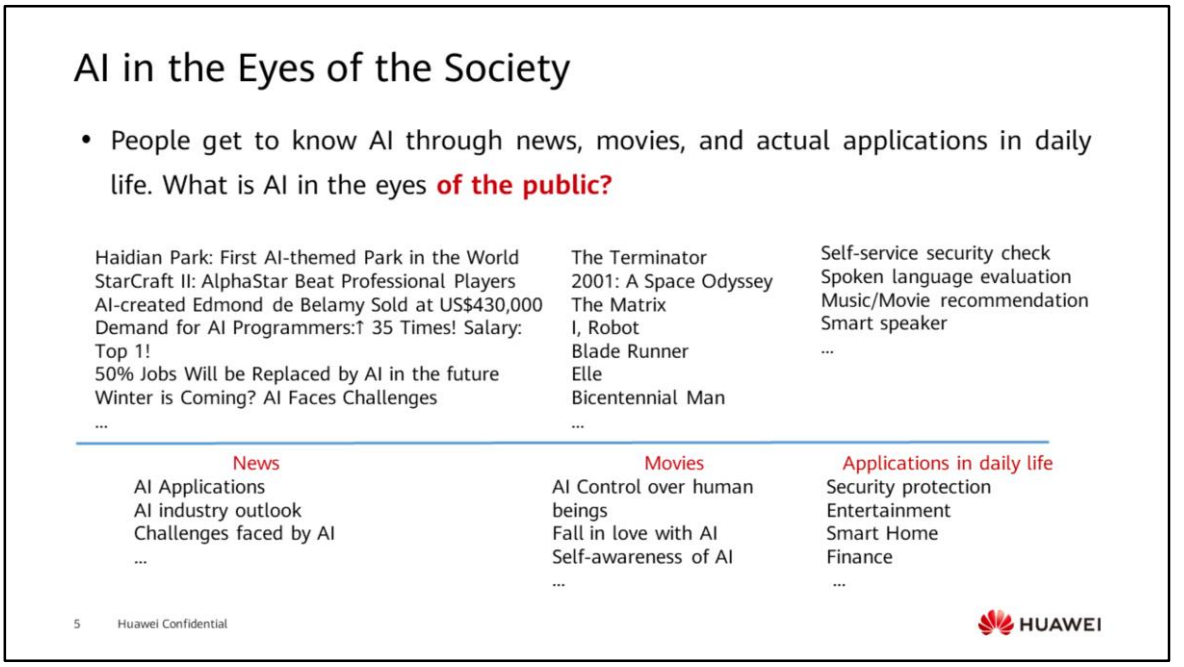
Upon completion of this course, you will be able to:

- Understand basic concepts of AI.
- Understand AI technologies and their development history.
- Understand the application technologies and application fields of AI.
- Know Huawei's AI development strategy.
- Know the development trends of AI.

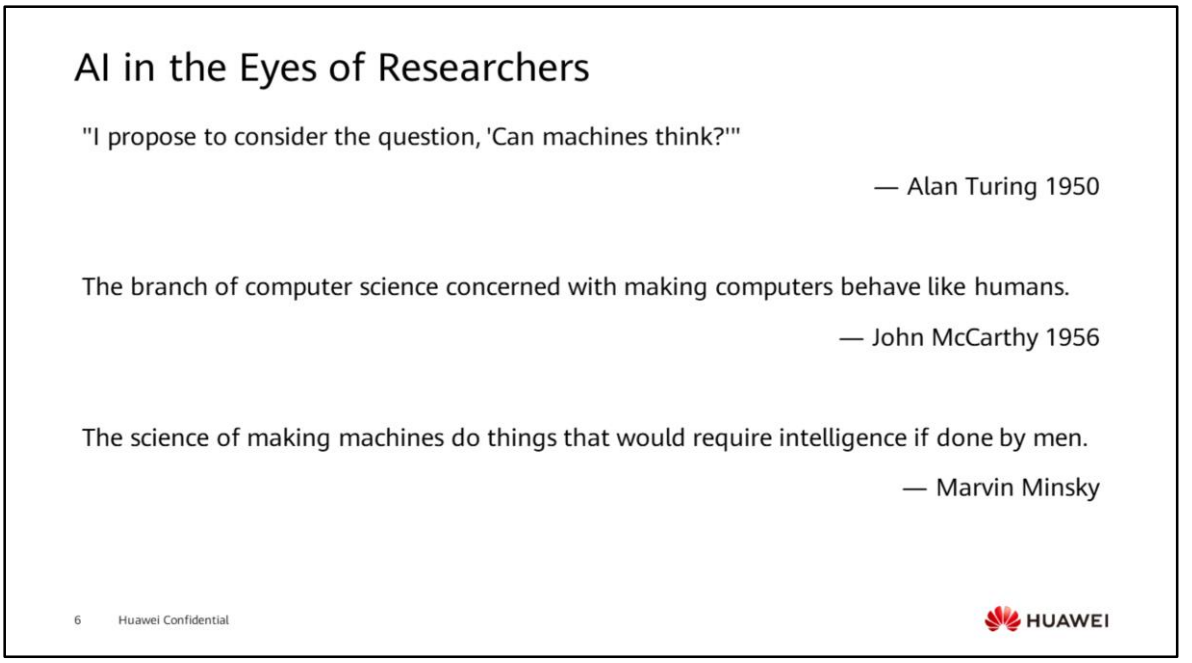
HUAWEI

Contents

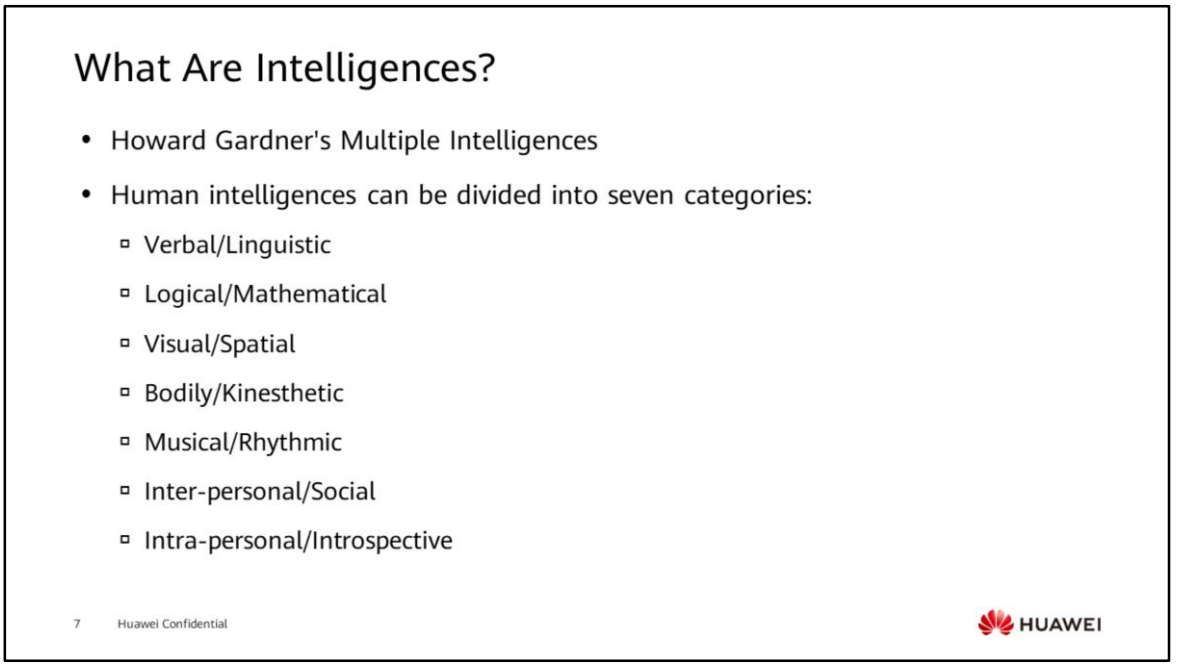
HUAWEI



- News: Exaggerated titles were used to report AI. Movies: Virtual AI was built with rich imagination. Applications in life: AI makes life more convenient while brings privacy concerns.



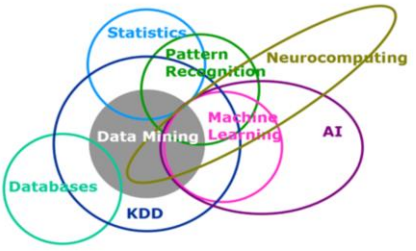
- "The branch of computer science concerned with making computers behave like humans." — A popular definition of artificial intelligence, and an earlier definition in this field proposed by John McCarthy| at the Dartmouth Conference in 1956. However, it seems that this definition ignores the possibility of strong AI. According to another definition, artificial intelligence is the intelligence (weak AI) demonstrated by artificial machines.
- Alan Turing discussed the question of "Can machines think?" in his seminal paper Computing Machinery and Intelligence.

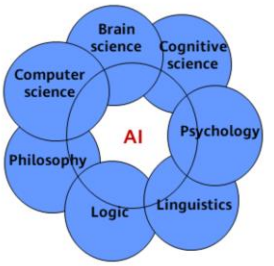


- 1. Language intelligence
- It refers to the ability to express thoughts and understand others by using oral languages or in words, and to master speech, semantics, and grammar flexibly, with the ability to think in words, express in words, and appreciate the deep meaning of languages. Ideal professions for people mastering language intelligence include politicians, hosts, lawyers, speakers, editors, writers, journalists, and teachers.
- 2. Logical-mathematical intelligence
- It refers to the ability to calculate, measure, infer, conclude, and classify, and to carry out complex mathematical operations. This intelligence includes sensitivity to logical ways and relationships, statements and propositions, functions, and other related abstract concepts. Ideal professions for people mastering logical mathematical intelligence include scientists, accountants, statisticians, engineers, and computer software developers.

What Is AI?

- Artificial Intelligence (AI) is a new technical science that studies and develops theories, methods, techniques, and application systems for simulating and extending human intelligence. In 1956, the concept of AI was first proposed by John McCarthy, who defined the subject as "science and engineering of making intelligent machines, especially intelligent computer program". AI is concerned with making machines work in an intelligent way, similar to the way that the human mind works. At present, AI has become an interdisciplinary course that involves various fields.



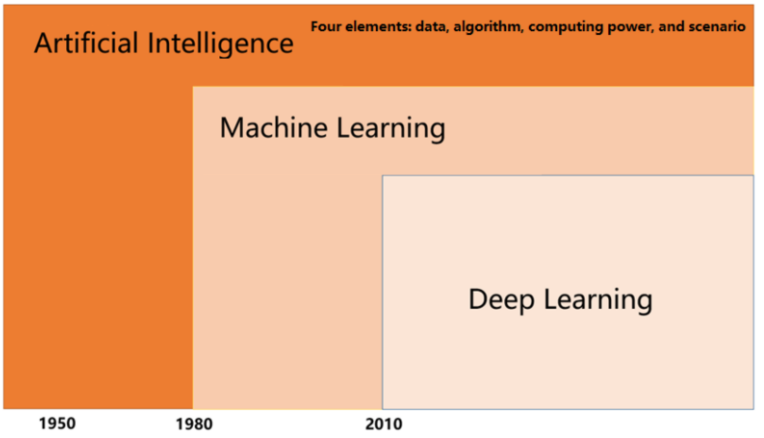


Identification of concepts related to AI and machine learning
AI Development Report 2020

HUAWEI

- Machine learning can be understood from multiple aspects. Tom Mitchell, a global machine learning scientist, provided a widely quoted definition: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E." These definitions are simple and abstract. However, as we deepen our understanding of machine learning, we will find that the connotation and extension of machine learning are changing over time. Because a variety of fields and applications are involved and machine learning develops rapidly, it is not easy to define machine learning simply and clearly.
- In general knowledge, processing systems and algorithms of machine learning (ML) are an identification mode that performs prediction by finding a hidden mode in data. ML is an important subfield of AI, which also intersects with data mining (DM) and knowledge discovery in database (KDD), for better understanding and distinguishing of artificial intelligence, machine learning, data mining, pattern recognition, statistics, and neural computing.

Relationship of AI, Machine Learning, and Deep Learning



HUAWEI

Relationship of AI, Machine Learning and Deep Learning

- AI: A new technical science that focuses on the research and development of theories, methods, techniques, and application systems for simulating and extending human intelligence.
- Machine learning: A core research field of AI. It focuses on the study of how computers can obtain new knowledge or skills by simulating or performing learning behavior of human beings, and reorganize existing knowledge architecture to improve its performance. It is one of the core research fields of AI.
- Deep learning: A new field of machine learning. The concept of deep learning originates from the research on artificial neural networks. The multi-layer perceptron (MLP) is a type a deep learning architecture. Deep learning aims to simulate the human brain to interpret data such as images, sounds, and texts.

HUAWEI

Three Major Schools of Thought: Symbolism

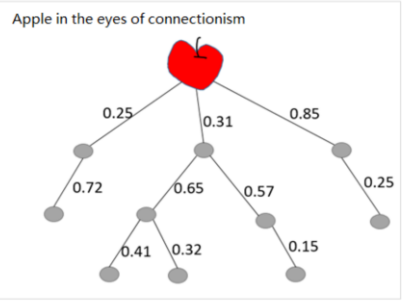
- Basic thoughts
  - The cognitive process of human beings is the process of inference and operation of various symbols.
  - A human being is a physical symbol system, and so is a computer. Computers, therefore, can be used to simulate intelligent behavior of human beings.
  - The core of AI lies in knowledge representation, knowledge inference, and knowledge application. Knowledge and concepts can be represented with symbols. Cognition is the process of symbol processing while inference refers to the process of solving problems by using heuristic knowledge and search.
- Representative of symbolism: inference, including symbolic inference and machine inference

HUAWEI

- Symbolism (logicism, psychology, computer science): Symbols represent knowledge, and inference produces cognitive results.
- In symbolism, the concept of apple can be represented by the apple node or a group of nodes that represent its attributes. The focus of symbolism is theoretical logic inference. Connectionism, being weak in logic, is result-oriented.

Three Major Schools of Thought: Connectionism

- Basic thoughts
  - The basis of thinking is neurons rather than the process of symbol processing.
  - Human brains vary from computers. A computer working mode based on connectionism is proposed to replace the computer working mode based on symbolic operation.
- Representative of connectionism: neural networks and deep learning



HUAWEI

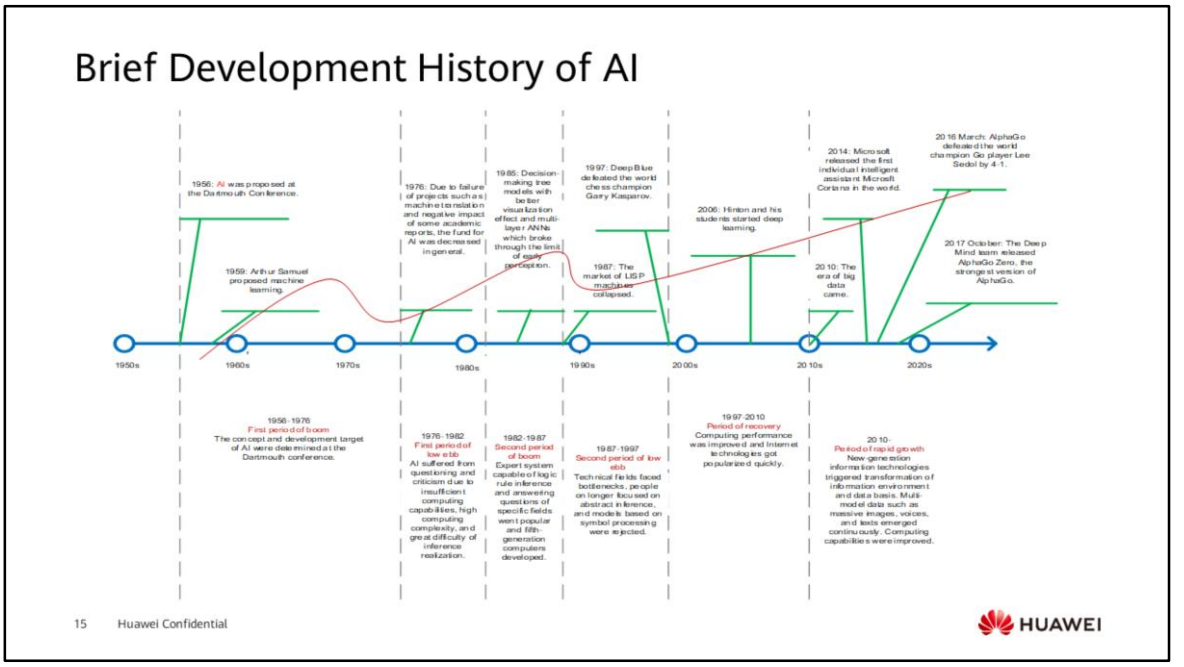
- Connectionism is derived from bionics, especially studies of the human brain model. In connectionism, a concept is represented by a set of numbers, vectors, matrices, or tensors in a specific activation mode of the entire network. Each node, without specific meaning, plays its role in the representation of the concept. For example, in symbolism, the concept of a cat may be represented by a "cat node" or a set of nodes representing the cat's attributes, such as "two eyes", "four legs", and "fluffy". However, in connectionism, each node does not represent a particular concept, and it is impossible to find a cat node or an eye neuron.

Three Major Schools of Thought: Behaviorism

- Basic thoughts:
  - Intelligence depends on perception and action. The perception-action mode of intelligent behavior is proposed.
  - Intelligence requires no knowledge, representation, or inference. AI can evolve like human intelligence. Intelligent behavior can only be demonstrated in the real world through the constant interaction with the surrounding environment.
- Representative of behaviorism: behavior control, adaptation, and evolutionary computing

HUAWEI

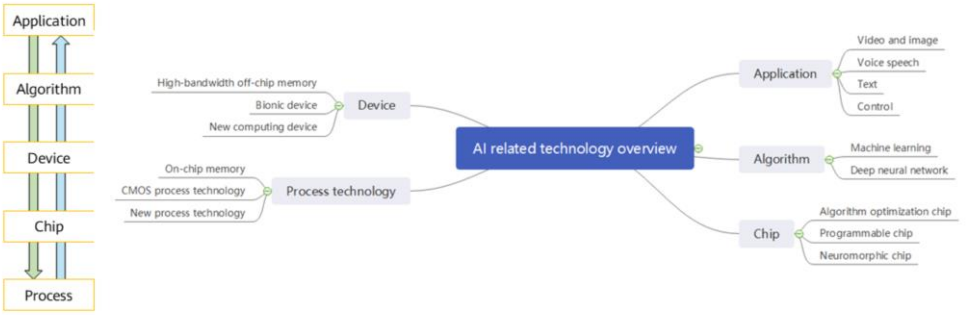
- Behaviorism concerns more about application practices and how to learn from the environment continuously to make corrections.



- In August 1956, a group of scientists gathered at Dartmouth College in the US to discuss an otherworldly topic: how to make computers simulate learning behavior and other intelligent actions of human beings. Many well-known scientists attended the Dartmouth Conference, including John McCarthy (founder of LISP), Marvin Minsky (expert in AI and cognition), and Claude Shannon (founder of the information theory), Allen Newell (computer scientist), and Herbert Simon (Nobel laureate in economics).
- The meeting lasted for two months. Although no consensus was reached, a name was given to the discussed object: artificial intelligence. The year of 1956, therefore, is the birth year of AI.

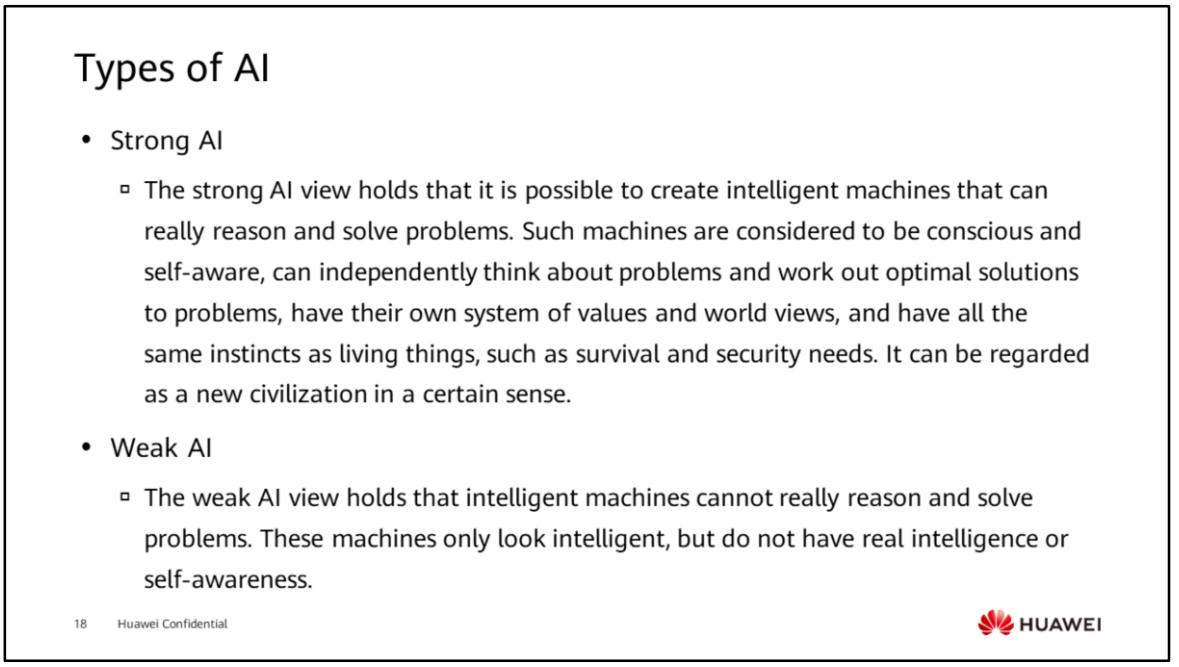
Overview of AI Technologies

- AI technologies are multi-layered, covering the application, algorithm mechanism, toolchain, device, chip, process, and material layers.



HUAWEI

- On one hand, the rapid development of applications and algorithms, especially deep learning and convolutional neural networks, raises performance optimization requirements for AI chips by two to three orders of magnitude, which has triggered the upsurge of AI chip R&D in recent years. On the other hand, the rapid development of new materials, processes, and components, such as 3D stacked memory and process evolution, also makes significant improvements in performance and power consumption of AI chips possible. This driving force comes from breakthroughs in basic research. In general, the above driving forces empower rapid advancement of AI chip technologies in recent years.
- Application layer:
- Text: text analysis, language translation, man-machine dialog, reading comprehension, recommendation system...
- Control: autonomous driving, drone, robot, industrial automation...



- A key counter of AI is to achieve a superhuman level in challenging fields through self-learning without any prior knowledge.
- Strong AI can compete with humans in all aspects. Therefore, it aims to enable robots to implement human-like capabilities in all aspects rather than a specific field. Strong AI can think, make plans, solve problems, perform abstract thinking, understand complex concepts, quickly learn, and learn from experience. Currently, it is believed that if we can simulate the human brain and copy all its neurons and synapses on the same scale, strong AI will naturally occur.
- Now we are in the weak AI phase. The emergence of weak AI alleviates human intellectual labor, similar to advanced bionics. Both AlphaGo and robots that can write press releases and novels fall in the weak AI phase because they are better than humans only in some ways. The roles of data and computing power are self-evident in the era of weak AI, and promote the commercialization of AI. In the era of strong AI, these two factors are still critical. At the same time, the research on quantum computing by technology giants like Google and IBM also provides powerful support for humans to enter the era of strong AI.

Classification of Intelligent Robots

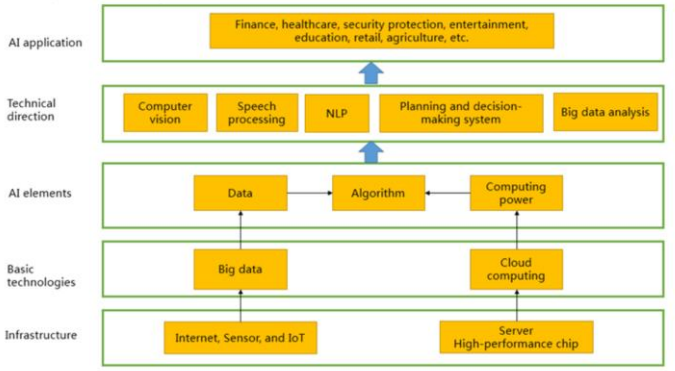
- Currently, there is no unified definition of AI research. Intelligent robots are generally classified into the following four types:
  - "Thinking like human beings": weak AI, such as Watson and AlphaGo
  - "Acting like human beings": weak AI, such as humanoid robot, iRobot, and Atlas of Boston Dynamics
  - "Thinking rationally": strong AI (Currently, no intelligent robots of this type have been created due to the bottleneck in brain science.)
  - "Acting rationally": strong AI

HUAWEI

- Generally, AI can be divided into four categories: "thinking like a human", "acting like a human", "thinking rationally", and "acting rationally". The "acting" here should be understood in a broad sense as taking actions or making decisions, rather than physical movements. Mainstream research focuses on weak AI, and it is generally believed that this research field has achieved considerable achievements.
- Boston Dynamics 10 Years challenge on Robots：https://www.youtube.com/watch?v=WjE1_XEUoGE

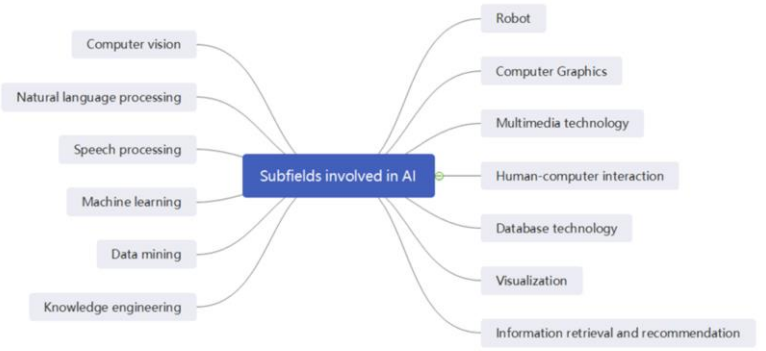
AI Industry Ecosystem

- The four elements of AI are data, algorithm, computing power, and scenario. To meet requirements of these four elements, we need to combine AI with cloud computing, big data, and IoT to build an intelligent society.



HUAWEI

Sub-fields of AI



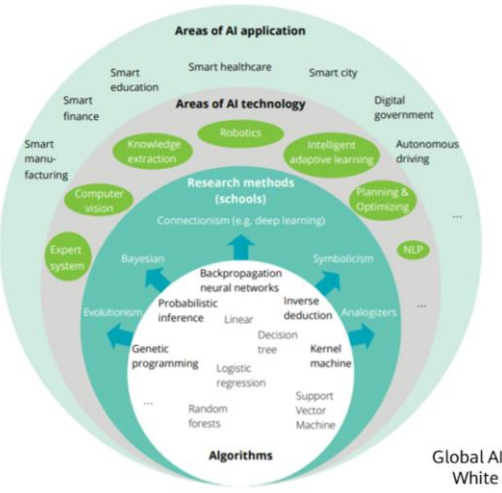
AI Development Report 2020

HUAWEI

Contents

HUAWEI

Technical Fields and Application Fields of AI

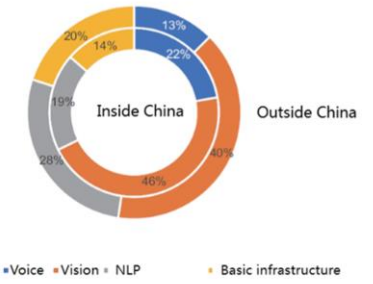


Global AI Development White Paper 2020

HUAWEI

Distribution of AI Application Technologies in Enterprises Inside and Outside China

- At present, application directions of AI technologies mainly include:
  - **Computer vision**: a science of how to make computers "see"
  - **Speech processing**: a general term for various processing technologies used to research the voicing process, statistical features of speech signals, speech recognition, machine-based speech synthesis, and speech perception
  - **Natural language processing (NLP)**: a subject that use computer technologies to understand and use natural language



Distribution of AI application technologies in enterprises inside and outside China

China AI Development Report 2018

HUAWEI

Voice Processing Application Scenario (1)

- The main topics of voice processing research include voice recognition, voice synthesis, voice wakeup, voiceprint recognition, and audio-based incident detection. Among them, the most mature technology is voice recognition. As for near field recognition in a quite indoor environment, the recognition accuracy can reach 96%.
- Application scenarios:

Question Answering Bot (QABot)

Voice navigation

HUAWEI

Voice Processing Application Scenario (2)

Intelligent education

Real-time conference records

- Other applications:
  - Spoken language evaluation
  - Diagnostic robot
  - Voiceprint recognition
  - Smart sound box
  - ...

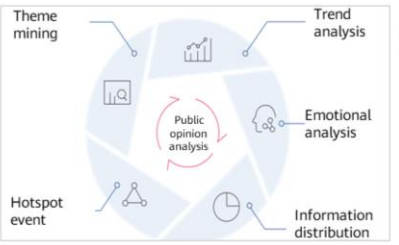
HUAWEI

- Telephone follow-up: converts telephone follow-up content into human voices to communicate with customers, improving the overall convenience of calling customers.
- Intelligent education: converts textbooks into life-like voices to simulate classroom teaching, helping teachers find new, innovative ways to better the education of their students.
- Conference records: quickly identifies audio files of conference records and converts them into text for easy recording.
- Real-time conference records: converts the audio files of a video conference or a conference call into text in real time and allows users to check, modify, and retrieve the conference content, improving the conferencing efficiency.

NLP Application Scenario (1)

- The main topics of NLP research include machine translation, text mining, and sentiment analysis. NLP imposes high requirements on technologies but confronts low technology maturity. Due to high complexity of semantics, it is hard to reach the human understanding level using parallel computing based on big data and parallel computing only.
- In future, NLP will achieve more growth: understanding of shallow semantics → automatic extraction of features and understanding of deep semantics; single-purpose intelligence (ML) → hybrid intelligence (ML, DL, and RL)
- Application scenarios:

Public opinion analysis



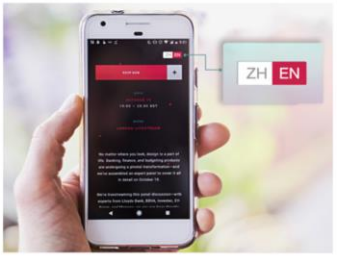
Evaluation analysis

HUAWEI

- Public opinion analysis: uses algorithm models such as keyword extraction, text clustering, and theme mining to mine emergencies and public opinion orientation, discover topics and trends, and analyze public opinions. It analyzes emotions, hot topics, trends, and prorogation channels from multiple dimensions to master public opinion trends in a timely manner.
- Evaluation analysis: uses approaches such as emotion analysis and opinion extraction to extract emotional tendency and key opinion information from a large number of user comments.

NLP Application Scenario (2)

Machine translation

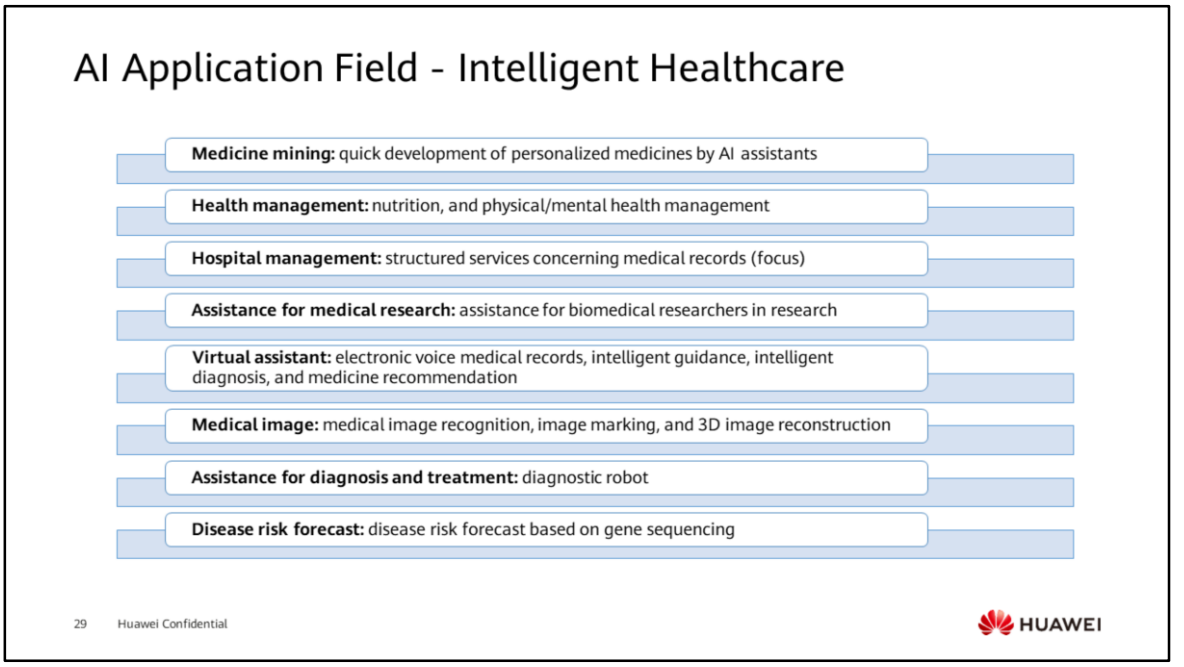


Text classification

- Other applications:
  - Knowledge graph
  - Intelligent copywriting
  - Video subtitle
  - ...

HUAWEI

- Translation center: builds a machine translation system that meets specific requirements and efficiently and accurately translates emails, papers, and news.
- Text classification: classifies a large number of files based on the preset categories. Typical application scenarios include category labeling, content retrieval, and personalized content recommendation.
- Intelligent copywriting: automatically generates new copywriting by learning the writing mode and structure of existing documents and using machine learning algorithms.
- Video subtitles: uses machine translation and speech recognition to implement real-time translation and provide bilingual subtitles, helping people quickly understand each other.
- Automobile knowledge graph drives technological transformation: The knowledge graph can be well applied in the automobile field with various attributes. It is able to provide consumers with a comprehensive shopping guide map by integrating the automotive information of different models and brands. In the pre-sales scenario, knowledge graph helps answer consumer questions about the product price, maintenance, configurations, and comparisons of price and performance. In addition, it also recommends customers specified models with outstanding features, such as technology and security. By building a knowledge graph for the automotive field, we establish a multi-round dialog system. This system analyzes the entities and relations in the user discourse, query in the knowledge graph, and select the optimal dialog strategy.



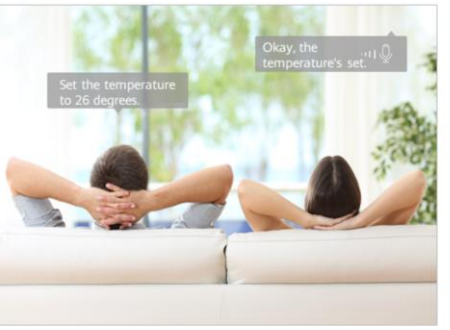
- With AI technologies, we can enable AI to "learn" professional medical knowledge, "remember" numerous historical medical cases, and identify medical images with computer vision technologies to provide reliable and efficient assistance for doctors.

AI Application Field - Smart Home

- Based on IoT technologies, a smart home ecosystem is formed with hardware, software, and cloud platforms, providing users personalized life services and making home life more convenient, comfortable, and safe.

Control smart home products with voice processing such as air conditioning temperature adjustment, curtain switch control, and voice control on the lighting system.

Develop user profiles and recommend content to users with the help of machine learning and deep learning technologies and based on historical records of smart speakers and smart TVs.



HUAWEI

AI Application Field - Retail

- AI will bring revolutionary changes to the retail industry. A typical symptom is unmanned supermarkets. For example, Amazon Go, unmanned supermarket of Amazon, uses sensors, cameras, computer vision, and deep learning algorithms to completely cancel the checkout process, allowing customers to pick up goods and "just walk out".
- One of the biggest challenges for unmanned supermarket is how to charge the right fees to the right customers. So far, Amazon Go is the only successful business case and even this case involves many controlled factors. For example, only Prime members can enter Amazon Go. Other enterprises, to follow the example of Amazon, have to build their membership system first.

HUAWEI

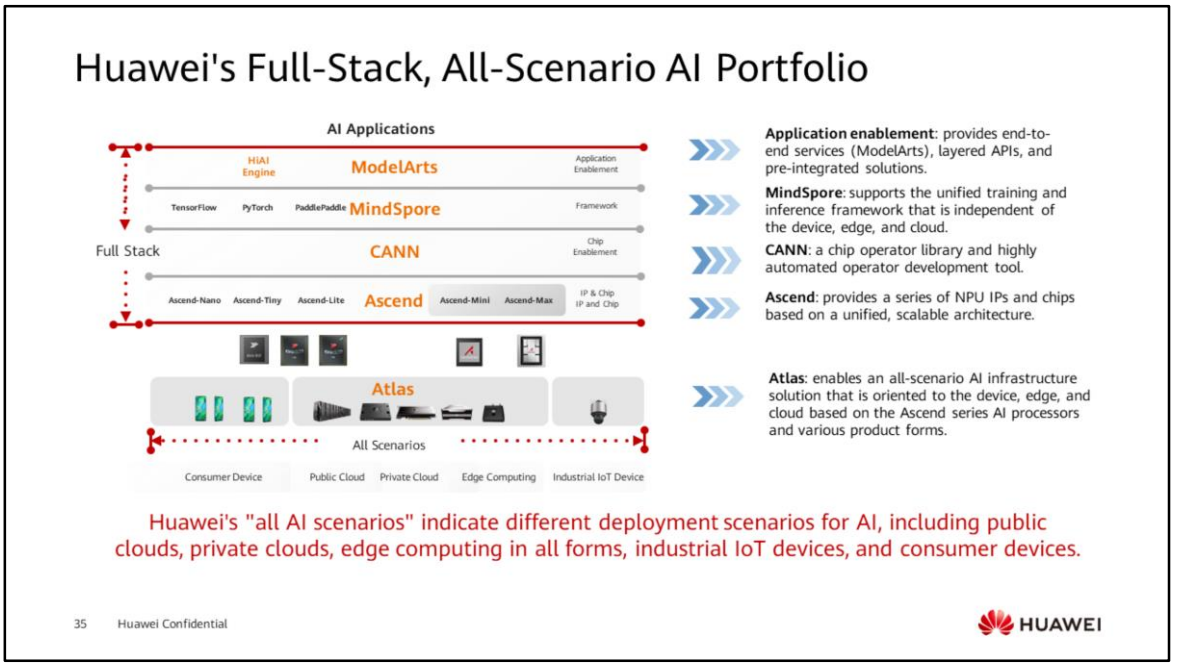
AI Application Field - Autonomous Driving

- The Society of Automotive Engineers (SAE) in the U.S. defines 6 levels of driving automation ranging from 0 (fully manual) to 5 (fully autonomous). L0 indicates that the driving of a vehicle completely depends on the driver's operation. The system above L3 can implement the driver's hand-off operation in specific cases, L5 depends on the system when vehicles are driving in all scenarios.
- Currently, only some commercial passenger vehicle models, such as Audi A8, Tesla, and Cadillac, support L2 and L3 Advanced driver-assistance systems (ADAS). It is estimated that by 2020, more L3 vehicle models will emerge with the further improvement of sensors and vehicle-mounted processors. L4 and L5 autonomous driving is expected to be first implemented on **commercial vehicles in closed campuses**. A wider range of passenger vehicles require advanced autonomous driving, which requires further improvement of technologies, policies, and infrastructure. It is estimated that L4 and L5 autonomous driving will be supported by common roads in 2025–2030.

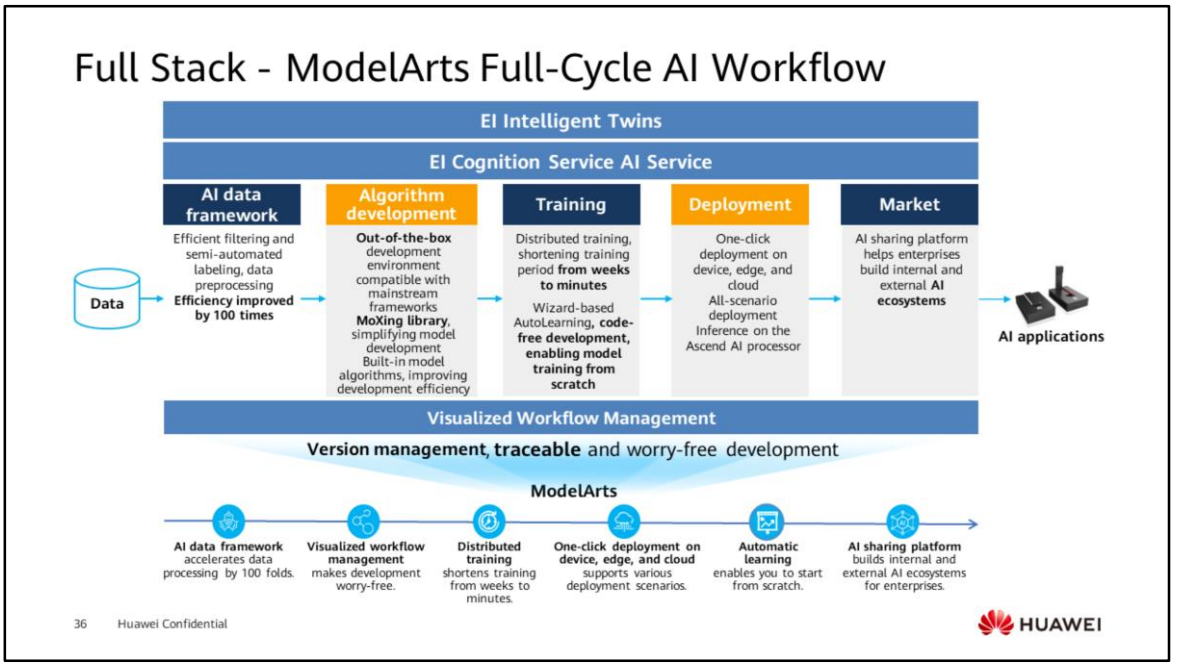
- Video material: Why is it so difficult to implement autonomous driving? PaperClip X NIO https://www.youtube.com/watch?v=MYPZbNEI77o
- Understand the autonomous driving in 5 minutes! |Science Big Bang 2-EP.25: https://www.youtube.com/watch?v=09tRb1G5afU
- Can not tell the difference between autonomous driving and unmanned driving? After watching this audio, you will be able to share the up-to-date car technologies with friends: https://www.youtube.com/watch?v=EaeV526oqWE
- Currently, the electronic control system uses the distributed ECU architecture. The information, entertainment, vehicle body, vehicle movement, and power assembly systems and their subdivision functions are separately controlled by different ECUs. Some high-end models have more than 100 ECUs. In the future, as vehicles enter the autonomous driving era of L3 or higher, more vehicle-mounted sensors will be used, and the amount of data generated by the sensors will increase sharply. The distributed electronic system cannot meet requirements for efficient convergence processing on a large amount of diversified sensor data and vehicle control decision making based on all sensor data. To meet the preceding requirements, the electronic control system needs to be centralized towards the domain control unit (DCU) and multi-domain control unit (MDC). In the future, the electronic control system will further be centralized and platform-based and support software and hardware decoupling. A unified supercomputing platform will be used to process, converge, and make decisions on sensor data to achieve high-level autonomous driving.

Contents

HUAWEI



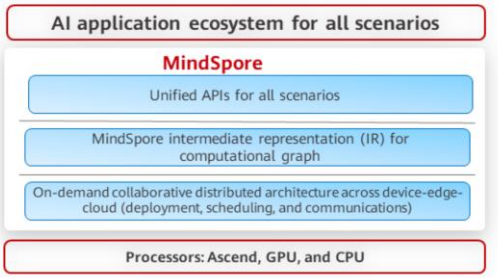
- Huawei announced that it will open source the server OS on December 31, 2020, the standalone GaussDB OLTP database in June 2020, and the MindSpore all-scenario AI computing framework in the first quarter of 2020.
- "Full-stack" refers to its technical function. Huawei's full-stack portfolio includes chips, chip enablement, a training and inference framework, and application enablement.
- By "all scenarios", Huawei means different deployment scenarios for AI, including public clouds, private clouds, edge computing in all forms, industrial IoT devices, and consumer devices.
- As the cornerstone of Huawei full-stack AI solution, Atlas provides modules, cards, and servers based on the Ascend AI processor to meet customers' computing requirements in all scenarios.



- ModelArts is a one-stop development platform for AI developers. With data preprocessing, semi-automated data labeling, distributed training, automated model building, and model deployment on the device, edge, and cloud, ModelArts helps AI developers build models quickly and manage the lifecycle of AI development.
- Automatic learning: can automate model design, parameter adjustment, and model training, compression, and deployment with the labeled data. The process is code-free and requires no model development experience.
- Device-edge-cloud: indicates devices, Huawei intelligent edge devices, and HUAWEI CLOUD, respectively.
- Online inference: a web service that synchronously provides the inference result for each inference request.
- Batch inference: a job that processes batch data for inference.

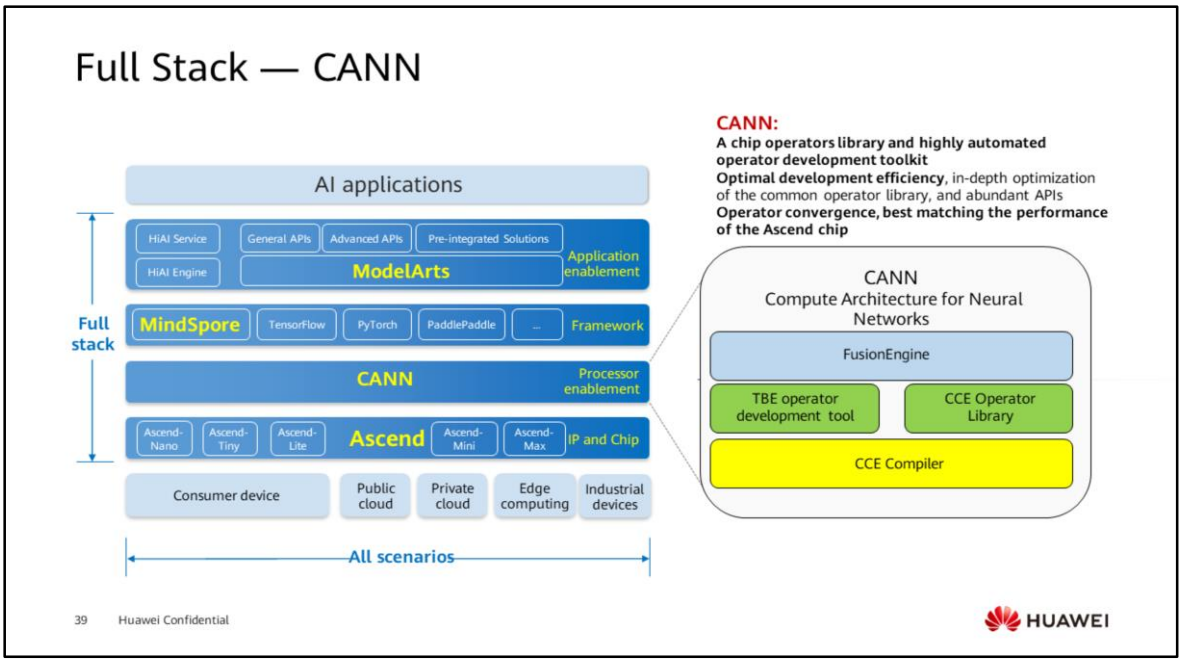
Full Stack — MindSpore (Huawei AI Computing Framework)

- MindSpore provides automatic parallel capabilities. With MindSpore, senior algorithm engineers and data scientists who focus on data modeling and problem solving can run algorithms on dozens or even thousands of AI computing nodes with only a few lines of description.
- The MindSpore framework supports both large-scale and small-scale deployment, adapting to independent deployment in all scenarios. In addition to the Ascend AI processors, MindSpore also supports other processors such as GPUs and CPUs.



HUAWEI

- In the intelligent era, AI applications in device-edge-cloud scenarios are booming. However, AI still faces huge challenges. Technical barriers, high development cost, and long deployment period hinder the development of the AI developer ecosystem in the entire industry. The all-scenario AI computing framework MindSpore is developed based on the principles of friendly development, efficient operation, and flexible deployment.
- In terms of deep learning framework, Huawei MindSpore is the strongest challenger to TensorFlow (Google), MXnet (Amazon), PyTorch (Facebook), and CNTK (Microsoft), which are listed as the four major players.
- MindSpore will be open-source on March 30, 2020. It competes with frameworks such as TensorFlow (Google), PyTorch (Facebook), PaddlePaddle (Baidu), and Caffe.



- CANN is a chip enabling layer developed by Huawei for deep neural networks and Ascend chips. It consists of four functional modules:

(1) FusionEngine: FusionEngine is an operator-level fusion engine. It fuses operators, reduces the memory transfer between operators, and improves the performance by 50%.

(2) CCE operator library: The optimized general operator library provided by Huawei can meet the requirements of most mainstream vision and NLP neural networks. (It is estimated that the CCE operator library will open APIs in 2020 Q1.)

Inevitably, customers and partners will have custom operator requirements for timeliness, privacy, and research. In this case, the third functional module is used.

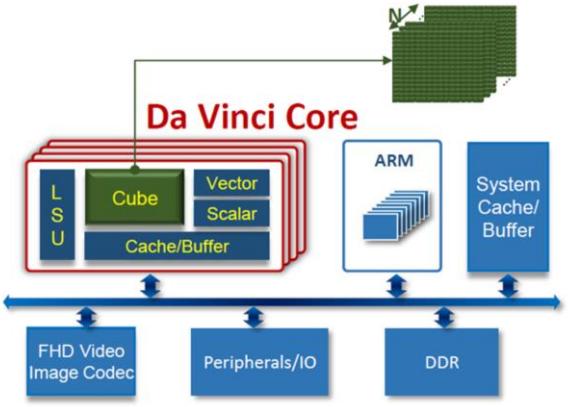
(3) Tensor Boost Engine (TBE): TBE is an efficient and high-performance custom operator development tool. It abstracts hardware resources as APIs, enabling customers to quickly construct required operators. (This functional module is expected to be available in 2020 Q4.)

(4) The last module is the bottom-layer compiler that optimizes performance and supports Ascend processors in all scenarios.

Note: TE-DSL is short for Tensor Engine-Description language.

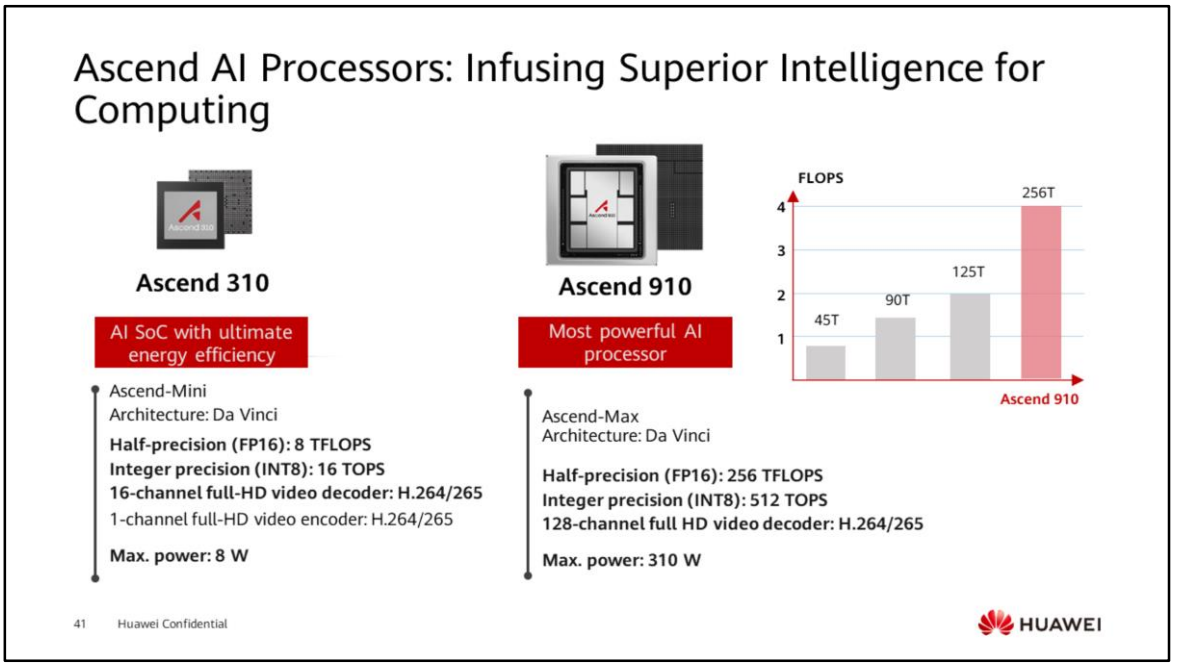
Full Stack — Ascend 310 AI Processor and Da Vinci Core

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

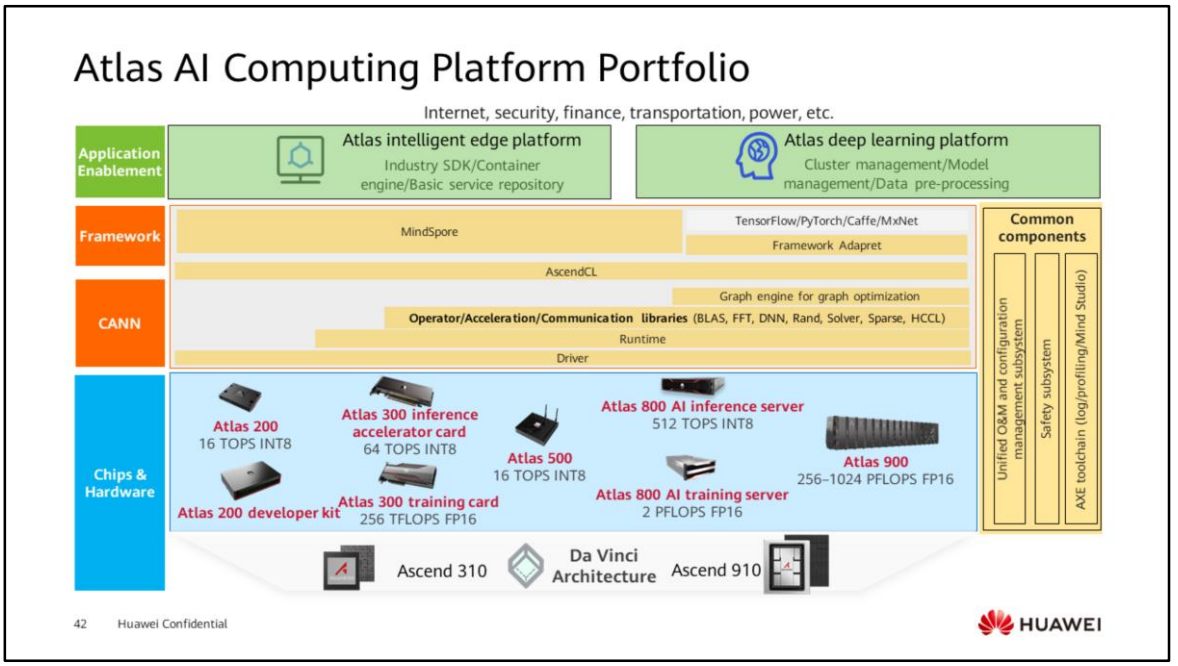


Note: This is typical configuration, high performance and low power sku can be offered based on your requirement.

HUAWEI



- Demands for AI are soaring worldwide. However, with the market being dominated by only a few vendors, AI processors are sold at a very high price. The delivery cycle is long and the local service support is weak. Therefore, the AI requirements of many industries cannot be effectively met.
- At HUAWEI CONNECT held in October 2018, Huawei unveiled its Ascend 310 processor for AI inference and Ascend 910 processor for AI training. Built upon the unique Da Vinci 3D Cube architecture, Huawei's Ascend AI processors boast high computing power, energy efficiency, and scalability.
- Ascend 310, an AI SoC with ultimate performance per watt, is designed for edge inference. It provides up to 16 TOPS of computing power, with a power consumption of only 8 watts. This makes it a perfect choice for edge computing.
- The Ascend 910 AI processor delivers the industry's highest computing density on a single AI chip. It applies to AI training and delivers 512 TOPS of computing power, with a maximum power consumption of 310 watts.
- (In the chart, Google TPU v2, Google TPU v3, NVIDIA Tesla V100, and Huawei Ascend 910 are shown from left to right.)



- Powered by the Ascend series AI processors, the Huawei Atlas AI computing platform supports rich form factors, including modules, cards, edge stations, servers, and clusters. Atlas enables AI solutions for all scenarios across the device, edge, and cloud. As an important part of Huawei's full-stack AI solution, Atlas launches the training platform this year following the inference platform unveiled last year, providing the industry with a complete AI solution. Huawei will also enhance all-scenario deployment, and drive full collaboration across the device, edge, and cloud, enabling every phase of the AI industry chain.

Huawei Atlas Computational Reasoning Platform

HUAWEI

HUAWEI CLOUD AI and HUAWEI Mobile Phones Help RFCx Protect the Rainforest

HUAWEI

Contents

HUAWEI

Seeing = Believing?

- With the development of computer vision technologies, reliability of images and videos is decreasing. Fake images can be produced with technologies such as PS and generative adversarial networks (GAN), making it hard to identify whether images are true or not.
- Example:
  - A suspect provided fake evidence by forging an image in which the suspect is in a place where he has never been to or with someone he has never seen using PS technologies.
  - In advertisements for diet pills, people's appearances before and after weight loss can be changed with PS technologies to exaggerate the effect of the pills.
  - Lyrebird, a tool for simulating voice of human beings based on recording samples of minutes, may be used by criminals.
  - Household images released on rent and hotel booking platforms may be generated through GAN.

HUAWEI

AI Development = Rising Unemployment?

- Looking back, human beings have always been seeking ways to improve efficiency, that is, obtain more with less resources. We used sharp stones to hunt and collect food more efficiently. We used steam engines to reduce the need for horses. Every step in achieving automation will change our life and work. In the era of AI, what jobs will be replaced by AI?
- The answer is repetitive jobs that involve little creativity and social interaction.

Jobs Most Likely to Be Replaced by AI	Jobs Most Unlikely to Be Replaced by AI
Courier	Writer
Taxi driver	Management personnel
Soldier	Software engineers
Accounting	HR manager
Telesales personnel	Designer
Customer service	Activity planner
...	...

HUAWEI

Problems to Be Solved

- Are AI-created works protected by copyright laws?
- Who gives authority to robots?
- What rights shall be authorized to robots?
- ...

HUAWEI

-

Contents

HUAWEI

Development Trends of AI Technologies

- Framework: easier-to-use development framework
- Algorithm: algorithm models with better performance and smaller size
- Computing power: comprehensive development of device-edge-cloud computing
- Data: more comprehensive basic data service industry and more secure data sharing
- Scenario: continuous breakthroughs in industry applications

HUAWEI

Easier-to-Use Development Framework

- Various AI development frameworks are evolving towards ease-of-use and omnipotent, continuously lowering the threshold for AI development.

HUAWEI

Tensorflow 2.0

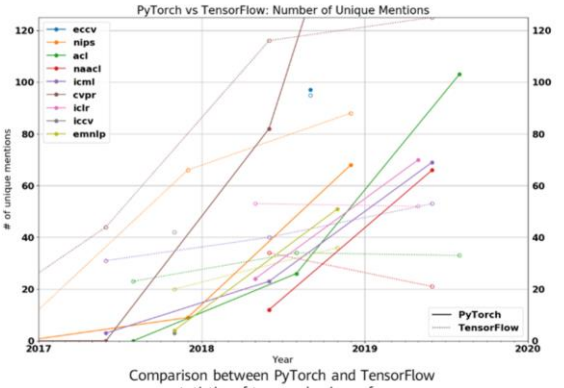
- TensorFlow 2.0 has been officially released. It integrates Keras as its high-level API, greatly improving usability.

HUAWEI

- Source: : https://medium.com/@himanshurawlani/getting-started-with-tensorflow-2-0-faf5428febae

Pytorch vs Tensorflow

- PyTorch is widely recognized by academia for its ease of use.



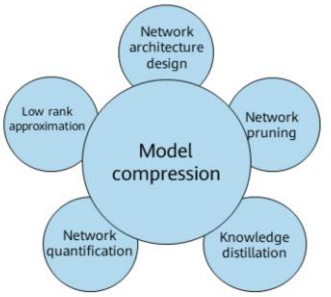
Comparison between PyTorch and TensorFlow usage statistics of top academic conferences

HUAWEI

- Source: https://thegradient.pub/state-of-ml-frameworks-2020-pytorch-dominates-research-tensorflow-dominates-industry/
- The comparison between the solid line point (PyTorch) and the dotted line (TensorFlow) in the picture shows that PyTorch is more widely used than TensorFlow in academia.
- In the industry, TensorFlow supports industrial deployment better and still has a larger market share in enterprises than PyTorch, which, however, shows a trend of continuous catch-up.

Smaller Deep Learning Models

- A model with better performance usually has a larger quantity of parameters, and a large model has lower running efficiency in industrial applications. More and more model compression technologies are proposed to further compress the model size while ensuring the model performance, meeting the requirements of industrial applications.
  - Low rank approximation
  - Network pruning
  - Network quantification
  - Knowledge distillation
  - Compact network design



HUAWEI

- Low-rank approximation
- If the weight matrix of the original network is considered as a full-rank matrix, can multiple low-rank matrices be used to approximate the original matrix to simplify the network? The answer is yes. The original dense full-rank matrix may be represented as a combination of several low-rank matrices, and a low-rank matrix may be broken down as a product of small-scale matrices.
- Network pruning
- The aim of network pruning is to remove the relatively unimportant weights from the weight matrix, and then fine-tune the network.
- Major procedure:
  Measure the importance of neurons.
  Remove some unimportant neurons.
  Fine-tune the network.
  Return to the first step to perform the next round of pruning.
- Compact network design
- MobileNet series, compact network model design based on depthwise separable convolution

Computing Power with Comprehensive Device-Edge-Cloud Development

- The scale of AI chips applied to the cloud, edge devices, and mobile devices keeps increasing, further meeting the computing power demand of AI.

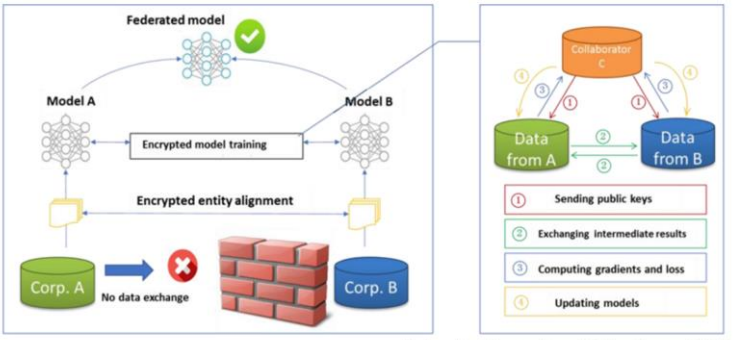
China AI Chip Industry Development White Paper 2020
Market Scale and Growth Prediction of AI Chips in China from 2020 to 2021

HUAWEI

- The above figure comes from the 2020 China AI Chip Development White Paper.

More Secure Data Sharing

- Federated learning uses different data sources to train models, further breaking data bottlenecks while ensuring data privacy and security.



Federated Learning White Paper V1.0

HUAWEI

- The above figure comes from the Federated Learning White Paper V1.0 released by WeBank in September 2018.
- Federated learning is essentially a distributed machine learning technology or machine learning framework.
- Federated learning aims to implement joint modeling and improve the AI model effect while ensuring data privacy security and legal compliance.
- Federated learning was first proposed by Google in 2016 and originally used to solve the problem of updating models locally for Android mobile phone users.

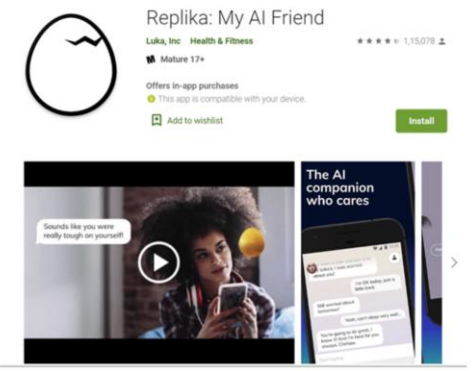
Continuous Breakthroughs in Application Scenarios

- With the continuous exploration of AI in various verticals, the application scenarios of AI will be continuously broken through.
  - Mitigating psychological problems
  - Automatic vehicle insurance and loss assessment
  - Office automation
  - ...

HUAWEI

Mitigating Psychological Problems

- AI chat robots help alleviate mental health problems such as autism by combining psychological knowledge.



HUAWEI

- The figure comes from https://medium.com/syncedreview/2020-in-review-10-brilliant-ai-apps-60d690976ccb
- AI Chat robots learn the human-like conversations from a large number of conversation corpora to help mitigate psychological problems such as autism in a warm manner.

Automatic Vehicle Insurance and Loss Assessment

- AI technologies help insurance companies optimize vehicle insurance claims and complete vehicle insurance loss assessment using deep learning algorithms such as image recognition.

Vehicle Damage Assessment

HUAWEI

- The figure comes from http://www.sohu.com/a/230901312_260616
- Video link: https://v.qq.com/x/page/w0648w88thu.html
- Ant Financial officially announced that it will fully open up its app "Dingsunbao", which uses AI technologies to help insurance companies optimize vehicle insurance claim procedures. This will not only improve the loss assessment efficiency, reduce the operation cost, and solve the problem of insufficient manpower in remote areas or peak hours, but also enhance user experience and reduce fraud risks.
- Yin Ming, president of Ant Financial's Insurance Business Unit, demonstrated how to use Dingsunbao: First, upload the panoramic photo of the vehicle, photo of the damaged part, and the details of the damaged part as instructed by Dingsunbao.
- Then, confirm the uploaded photo. Dingsunbao will use the algorithm model on the cloud server to determine the loss, provide accurate loss assessment conclusion within several seconds, provide information on nearby repair factories, and predict the premium of the next year.
- "Although Dingsunbao looks simple, it requires numerous efforts. We have overcome many difficulties. " Yin Ming said that the image-based damage assessment process includes core technologies such as automatic photo distribution, component identification, de-reflection, and image angle correction.

Office Automation

- AI is automating management, but the different nature and format of data makes it a challenging task. While each industry and application has its own unique challenges, different industries are gradually adopting machine learning-based workflow solutions.

HUAWEI

- WeLink introduction video: https://bbs-video.huaweicloud.com/video/media/20201015/20201015105917_56819/HuaWeiWeLinkxuanchuanshipin.mp4

Summary

- This chapter introduces the definition and development history of AI, describes the technical fields and application fields of AI, briefly introduces Huawei's AI development strategy, and finally discusses the disputes and the development trends of AI.

HUAWEI

Quiz

1. (Multiple-answer question) Which of the following are AI application fields?
   A. Smart household
   B. Smart healthcare
   C. Smart city
   D. Smart education
2. (True or False) By "all AI scenarios", Huawei means different deployment scenarios for AI, including public clouds, private clouds, edge computing in all forms, industrial IoT devices, and consumer devices.
   A. True
   B. False

HUAWEI

- Answers: 1. ABCD 2. A

More Information

Online learning website

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

Huawei Knowledge Base

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

HUAWEI

Thank you.

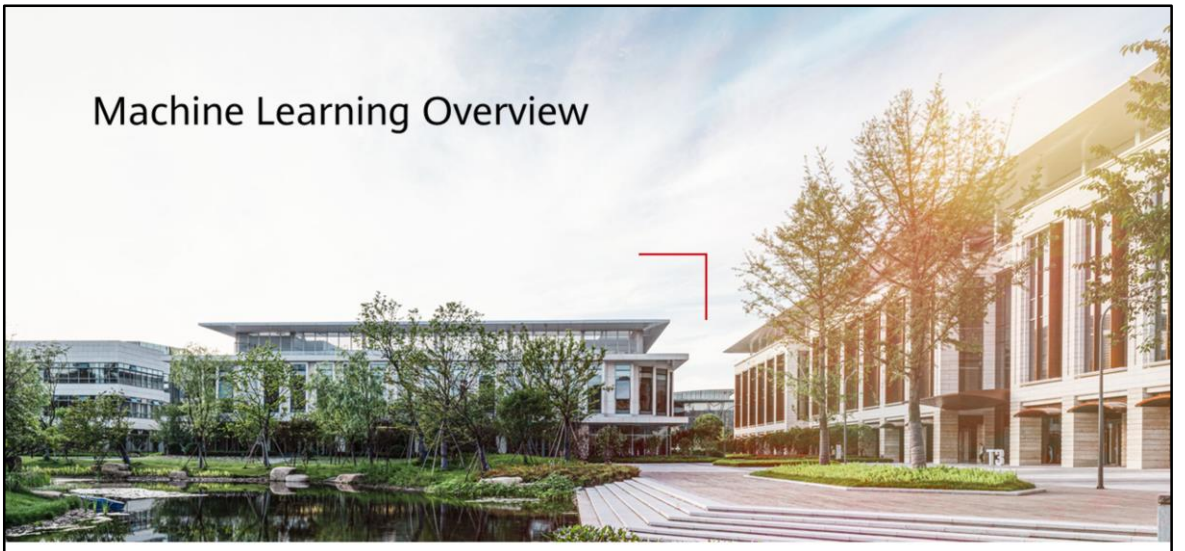
把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

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

Copyright©2020 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.

Machine Learning Overview



HUAWEI

Foreword

- Machine learning is a core research field of AI, and it is also a necessary knowledge for deep learning. Therefore, this chapter mainly introduces the main concepts of machine learning, the classification of machine learning, the overall process of machine learning, and the common algorithms of machine learning.

HUAWEI

Objectives

Upon completion of this course, you will be able to:

- Master the learning algorithm definition and machine learning process.
- Know common machine learning algorithms.
- Understand concepts such as hyperparameters, gradient descent, and cross validation.

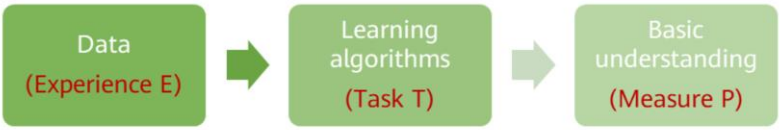
HUAWEI

Contents

HUAWEI

Machine Learning Algorithms (1)

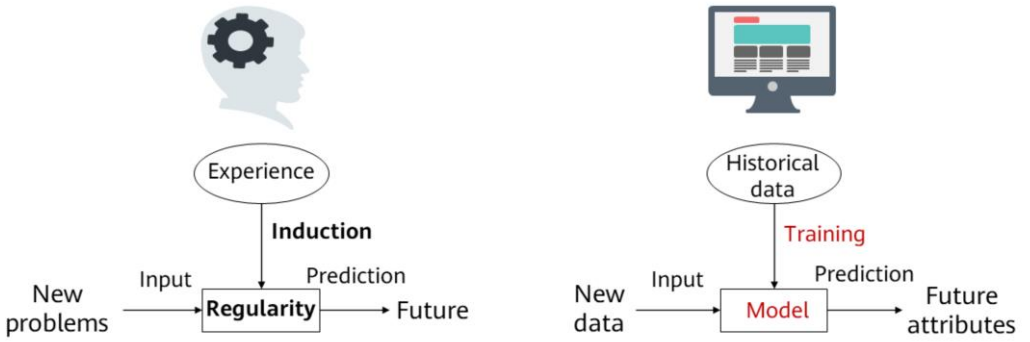
- Machine learning (including deep learning) is a study of learning algorithms. A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E.



HUAWEI

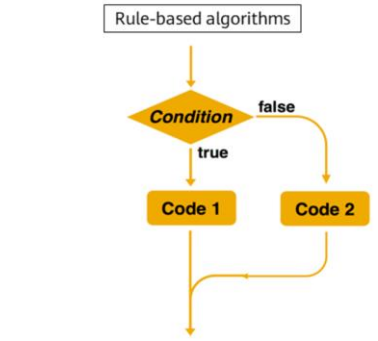
- Deep learning is a specific branch of machine learning. To understand deep learning well, one must have a solid understanding of the basic principles of machine learning.
  - Task T: how the machine learning system should process a sample. A sample is a collection of features that have been quantitatively measured from some objects or events that we want the machine learning system to process, such as classification, regression, and machine translation.
  - Performance measure P: evaluates the abilities of a machine learning algorithm, such as accuracy and error rate.
  - Experience E: Most machine learning algorithms can be understood as being allowed to experience an entire dataset. Some machine learning algorithms do not just experience a fixed dataset. For example, reinforcement learning algorithms interact with an environment, so there is a feedback loop between the learning system and its training process. Machine learning algorithms can be broadly categorized as unsupervised or supervised by what kind of experience they are allowed to have during the learning process.
- To learn the game of Go:
  - Experience E_1: playing with itself — Unsupervised and indirect learning
  - Experience E_2: inquiring humans when playing with itself — Semi-supervised learning
  - Experience E_3: historical human games — Supervised and direct learning
- Handwritten text recognition: Task T: recognizes handwritten texts. Performance measure P: classification accuracy. Experience E: classified example library (supervised and direct learning).
- Robots' desire to advance: Look for new games and practice their skills through tiny changes in the same situation, enriching their training examples.

Machine Learning Algorithms (2)

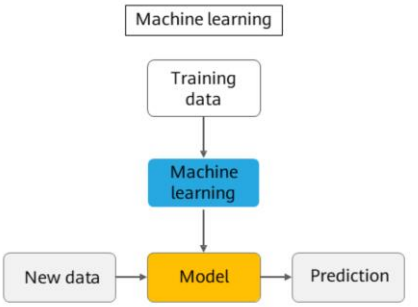


HUAWEI

Differences Between Machine Learning Algorithms and Traditional Rule-Based Algorithms



- Explicit programming is used to solve problems.
- Rules can be manually specified.



- Samples are used for training.
- The decision-making rules are complex or difficult to describe.
- Rules are automatically learned by machines.

HUAWEI

Application Scenarios of Machine Learning (1)

- The solution to a problem is complex, or the problem may involve a large amount of data without a clear data distribution function.
- Machine learning can be used in the following scenarios:

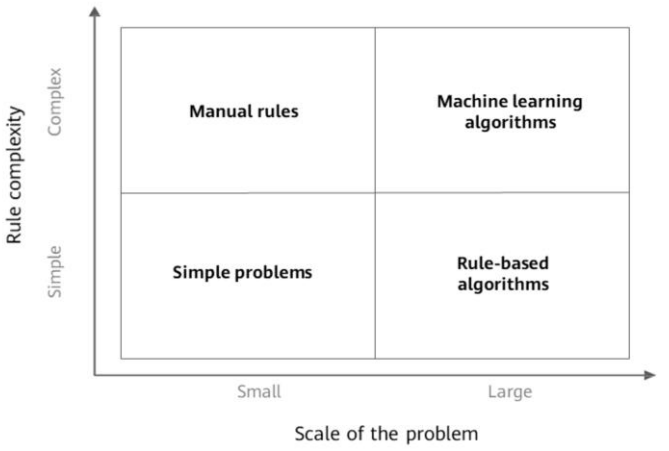
Rules are complex or cannot be described, such as voice recognition.

Task rules change over time. For example, in the part-of-speech tagging task, new words or meanings are generated at any time.

Data distribution changes over time, requiring constant readaptation of programs, such as predicting the trend of commodity sales.

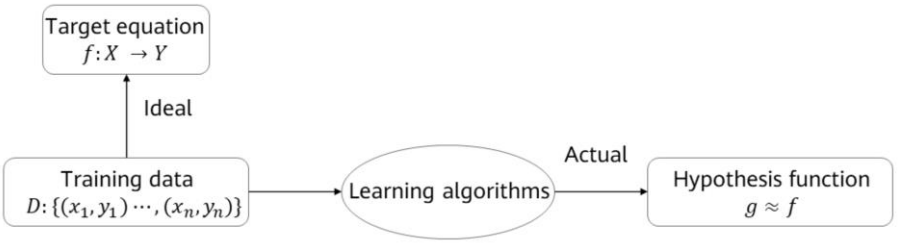
HUAWEI

Application Scenarios of Machine Learning (2)



HUAWEI

Rational Understanding of Machine Learning Algorithms



- Target function *f* is unknown. Learning algorithms cannot obtain a perfect function *f*.
- Assume that hypothesis function *g* **approximates** function *f*, but may be different from function *f*.

HUAWEI

Main Problems Solved by Machine Learning

- Machine learning can deal with many types of tasks. The following describes the most typical and common types of tasks.
  - Classification: A computer program needs to specify which of the k categories some input belongs to. To accomplish this task, learning algorithms usually output a function $f: R^n \rightarrow (1,2,\ldots,k)$. For example, the image classification algorithm in computer vision is developed to handle classification tasks.
  - Regression: For this type of task, a computer program predicts the output for the given input. Learning algorithms typically output a function $f: R^n \rightarrow R$. An example of this task type is to predict the claim amount of an insured person (to set the insurance premium) or predict the security price.
  - Clustering: A large amount of data from an unlabeled dataset is divided into multiple categories according to internal similarity of the data. Data in the same category is more similar than that in different categories. This feature can be used in scenarios such as image retrieval and user profile management.
- Classification and regression are two main types of prediction, accounting from 80% to 90%. The output of classification is discrete category values, and the output of regression is continuous numbers.

HUAWEI

Contents

HUAWEI

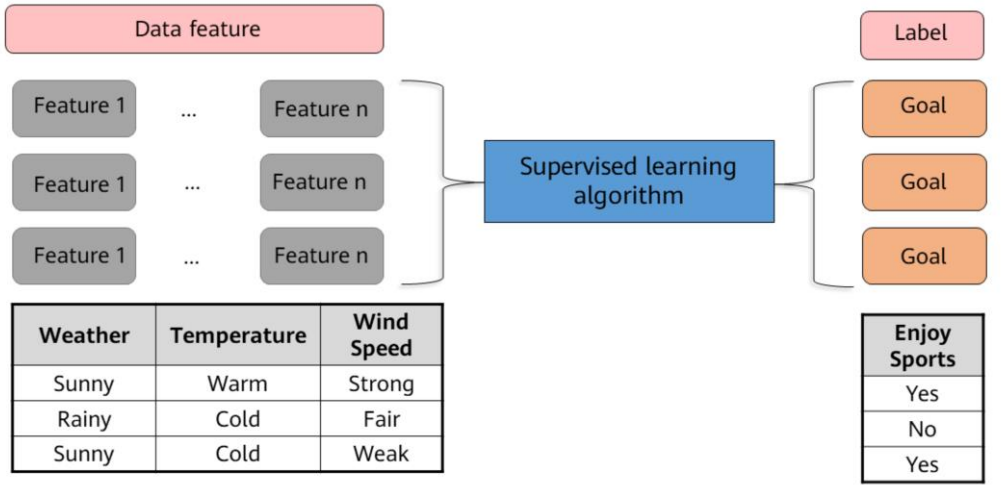
Machine Learning Classification

- **Supervised learning**: Obtain an optimal model with required performance through training and learning based on the samples of known categories. Then, use the model to map all inputs to outputs and check the output for the purpose of classifying unknown data.
- **Unsupervised learning**: For unlabeled samples, the learning algorithms directly model the input datasets. Clustering is a common form of unsupervised learning. We only need to put highly similar samples together, calculate the similarity between new samples and existing ones, and classify them by similarity.
- **Semi-supervised learning**: In one task, a machine learning model that automatically uses a large amount of unlabeled data to assist learning directly of a small amount of labeled data.
- **Reinforcement learning**: It is an area of machine learning concerned with how agents ought to take actions in an environment to maximize some notion of cumulative reward. The difference between reinforcement learning and supervised learning is the teacher signal. The reinforcement signal provided by the environment in reinforcement learning is used to evaluate the action (scalar signal) rather than telling the learning system how to perform correct actions.

HUAWEI

- Supervised learning: We give a computer a bunch of choice questions (training samples) and provide standard answers. The computer tries to adjust its model parameters to make their predictions close to standard answers. In this way, the computer learns how to deal with this type of problem. Then the computer can help us reply to choice questions whose answers are not given (test samples).
- Unsupervised learning: We give a computer a bunch of choice questions (training samples), but do not provide standard answers. The computer tries to analyze the relationships between these questions and classify them. It does not know the answers to these questions, but it thinks that the answers to the questions in the same category should be the same.

Supervised Learning



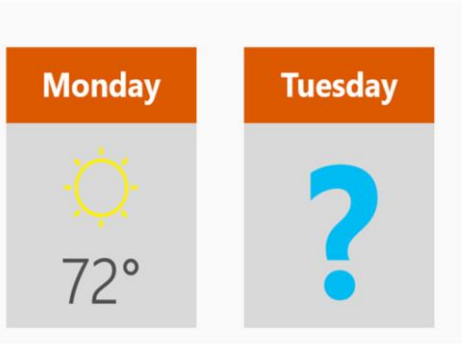
Weather	Temperature	Wind Speed
Sunny	Warm	Strong
Rainy	Cold	Fair
Sunny	Cold	Weak

Enjoy Sports
Yes
No
Yes

HUAWEI

Supervised Learning - Regression Questions

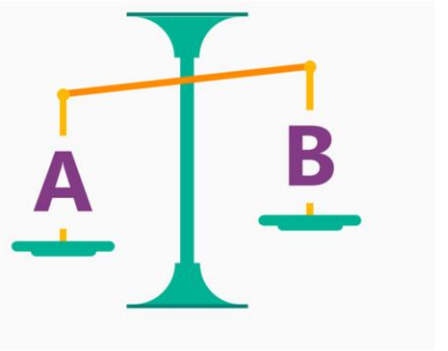
- Regression: reflects the features of attribute values of samples in a sample dataset. The dependency between attribute values is discovered by expressing the relationship of sample mapping through functions.
  - How much will I benefit from the stock next week?
  - What's the temperature on Tuesday?



HUAWEI

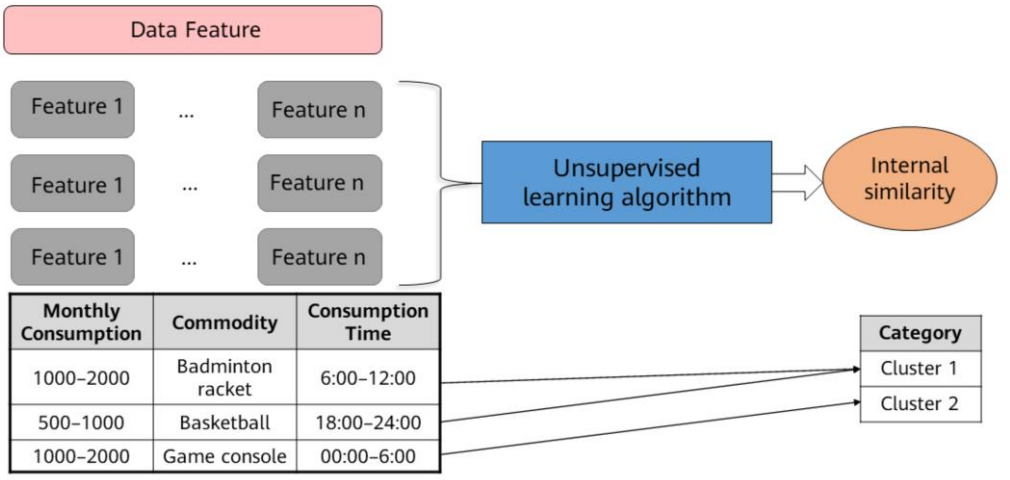
Supervised Learning - Classification Questions

- Classification: maps samples in a sample dataset to a specified category by using a classification model.
  - Will there be a traffic jam on XX road during the morning rush hour tomorrow?
  - Which method is more attractive to customers: 5 yuan voucher or 25% off?



HUAWEI

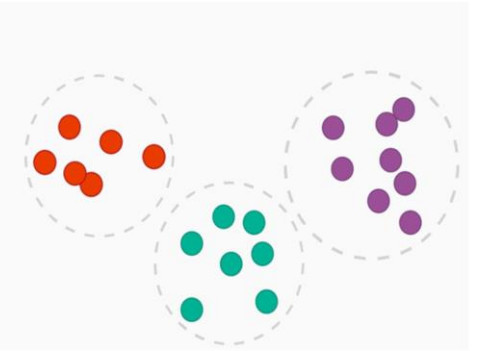
Unsupervised Learning



HUAWEI

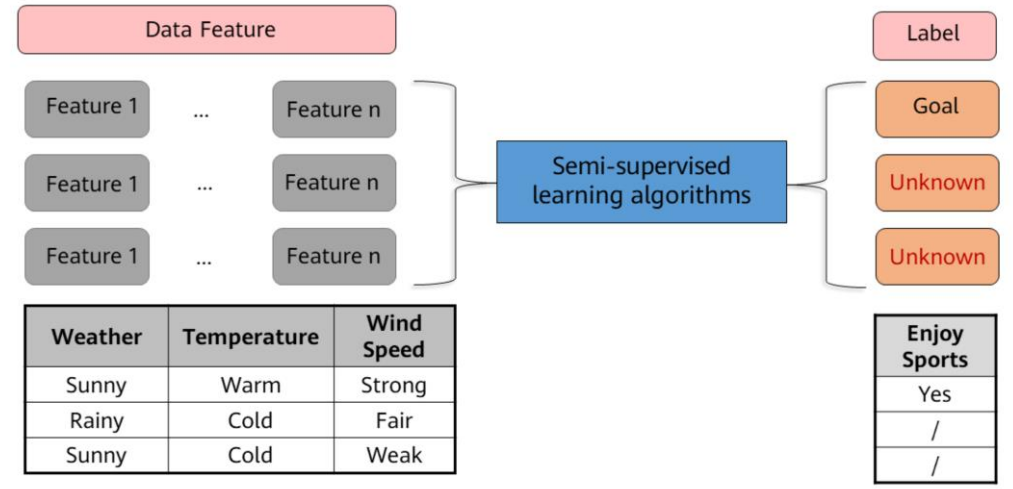
Unsupervised Learning - Clustering Questions

- Clustering: classifies samples in a sample dataset into several categories based on the clustering model. The similarity of samples belonging to the same category is high.
  - Which audiences like to watch movies of the same subject?
  - Which of these components are damaged in a similar way?



HUAWEI

Semi-Supervised Learning



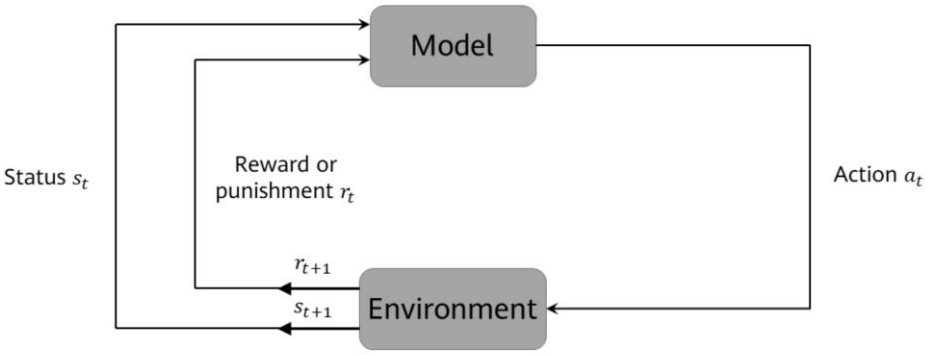
Weather	Temperature	Wind Speed
Sunny	Warm	Strong
Rainy	Cold	Fair
Sunny	Cold	Weak

Enjoy Sports
Yes
/
/

HUAWEI

Reinforcement Learning

- The model perceives the environment, takes actions, and makes adjustments and choices based on the status and award or punishment.

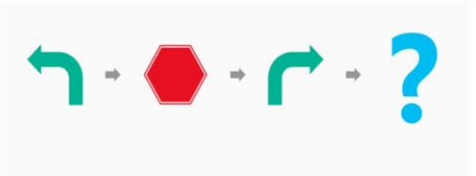


HUAWEI

- Reinforcement learning uses a series of actions to maximize the reward function to learn models.
- Both good and bad behaviors can help reinforcement learning in model learning.
- For example, autonomous vehicles learn by continuously interacting with the environment.

Reinforcement Learning - Best Behavior

- Reinforcement learning: always looks for best behaviors. Reinforcement learning is targeted at machines or robots.
  - Autopilot: Should it brake or accelerate when the yellow light starts to flash?
  - Cleaning robot: Should it keep working or go back for charging?

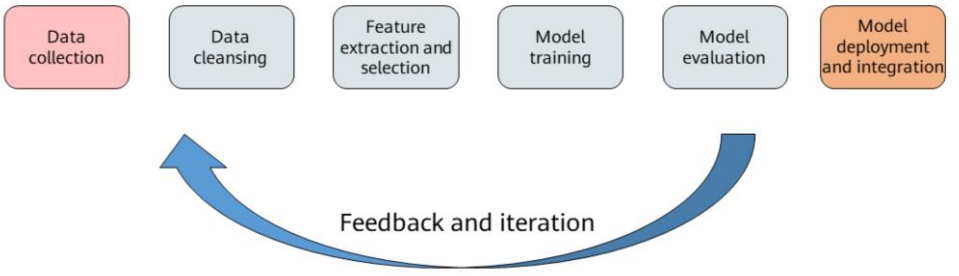


HUAWEI

Contents

HUAWEI

Machine Learning Process



HUAWEI

Basic Machine Learning Concept — Dataset

- Dataset: a collection of data used in machine learning tasks. Each data record is called a sample. Events or attributes that reflect the performance or nature of a sample in a particular aspect are called features.
- Training set: a dataset used in the training process, where each sample is referred to as a training sample. The process of creating a model from data is called learning (training).
- Test set: Testing refers to the process of using the model obtained after learning for prediction. The dataset used is called a test set, and each sample is called a test sample.

HUAWEI

Checking Data Overview

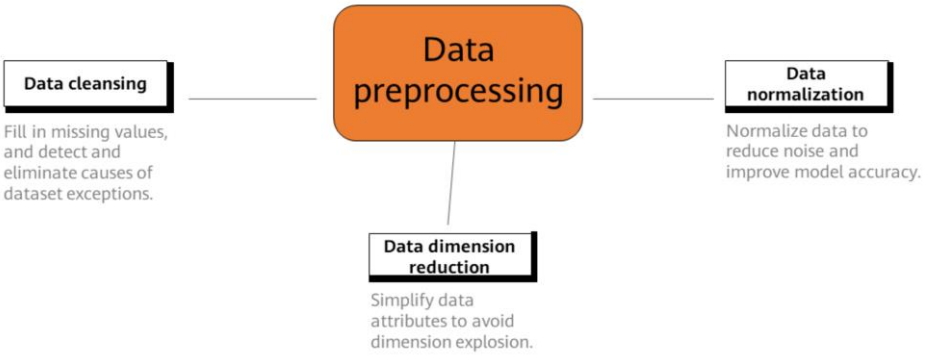
- Typical dataset form

		Feature 1	Feature 2	Feature 3	Label
	No.	Area	School Districts	Direction	House Price
Training set	1	100	8	South	1000
	2	120	9	Southwest	1300
	3	60	6	North	700
	4	80	9	Southeast	1100
Test set	5	95	3	South	850

HUAWEI

Importance of Data Processing

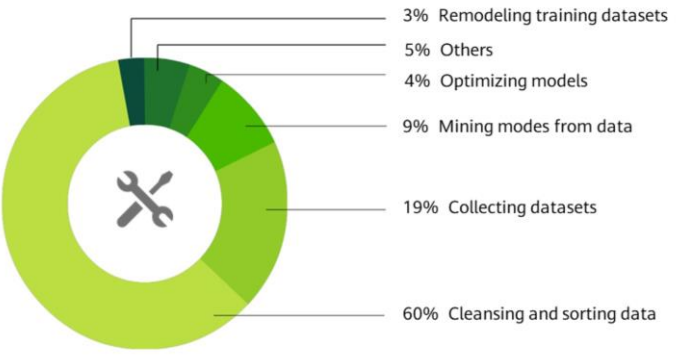
- Data is crucial to models. It is the ceiling of model capabilities. Without good data, there is no good model.



HUAWEI

Workload of Data Cleansing

- Statistics on data scientists' work in machine learning



CrowdFlower Data Science Report 2016

HUAWEI

Data Cleansing

- Most machine learning models process features, which are usually numeric representations of input variables that can be used in the model.
- In most cases, the collected data can be used by algorithms only after being preprocessed. The preprocessing operations include the following:
  - Data filtering
  - Processing of lost data
  - Processing of possible exceptions, errors, or abnormal values
  - Combination of data from multiple data sources
  - Data consolidation

HUAWEI

Dirty Data (1)

- Generally, real data may have some quality problems.
  - Incompleteness: contains missing values or the data that lacks attributes
  - Noise: contains incorrect records or exceptions.
  - Inconsistency: contains inconsistent records.

HUAWEI

Dirty Data (2)

#	Id	Name	Birthday	Gender	IsTeacher	#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	08/08/1948	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

Missing value

Invalid value

Value that should be in another column

Misspelling

Invalid duplicate item

Incorrect format

Attribute dependency

HUAWEI

Data Conversion

- After being preprocessed, the data needs to be converted into a representation form suitable for the machine learning model. Common data conversion forms include the following:
  - With respect to classification, category data is encoded into a corresponding numerical representation.
  - Value data is converted to category data to reduce the value of variables (for age segmentation).
  - Other data
    - In the text, the word is converted into a word vector through word embedding (generally using the word2vec model, BERT model, etc).
    - Process image data (color space, grayscale, geometric change, Haar feature, and image enhancement)
  - Feature engineering
    - Normalize features to ensure the same value ranges for input variables of the same model.
    - Feature expansion: Combine or convert existing variables to generate new features, such as the average.

HUAWEI

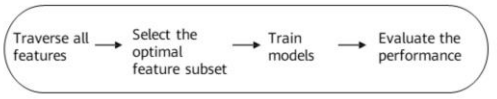
Necessity of Feature Selection

- Generally, a dataset has many features, some of which may be redundant or irrelevant to the value to be predicted.
- Feature selection is necessary in the following aspects:

HUAWEI

Feature Selection Methods - Filter

- Filter methods are independent of the model during feature selection.



Procedure of a filter method

By evaluating the correlation between each feature and the target attribute, these methods use a statistical measure to assign a value to each feature. Features are then sorted by score, which is helpful for preserving or eliminating specific features.

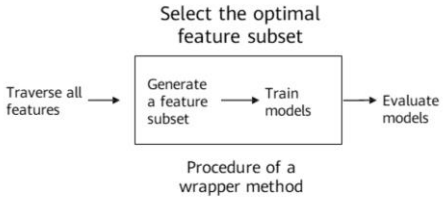
Common methods
- Pearson correlation coefficient
- Chi-square coefficient
- Mutual information

Limitations
- The filter method tends to select redundant variables as the relationship between features is not considered.

HUAWEI

Feature Selection Methods - Wrapper

- Wrapper methods use a prediction model to score feature subsets.



Procedure of a wrapper method

Wrapper methods consider feature selection as a search issue for which different combinations are evaluated and compared. A predictive model is used to evaluate a combination of features and assign a score based on model accuracy.

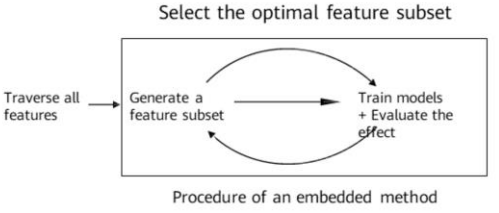
Common methods
- Recursive feature elimination (RFE)

Limitations
- Wrapper methods train a new model for each subset, resulting in **a huge number of computations**.
- A feature set with the best performance is usually provided for a specific type of model.

HUAWEI

Feature Selection Methods - Embedded

- Embedded methods consider feature selection as a part of model construction.



Procedure of an embedded method

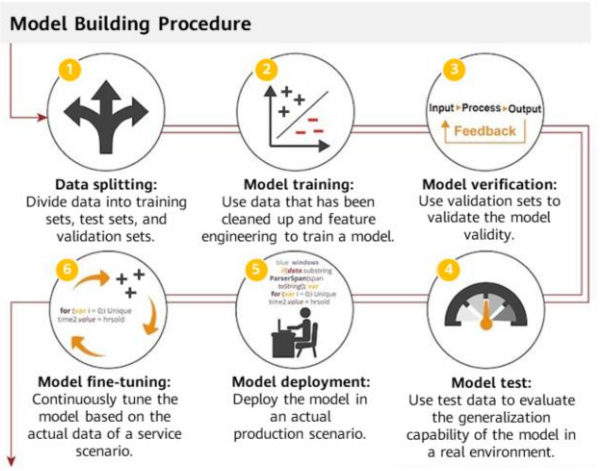
The most common type of embedded feature selection method is the **regularization method**. Regularization methods are also called penalization methods that introduce additional constraints into the optimization of a predictive algorithm that bias the model toward lower complexity and reduce the number of features.

Common methods
- Lasso regression
- Ridge regression

HUAWEI

- The penalty attribute of the model is highlighted, which is used to eliminate unimportant features.

Overall Procedure of Building a Model

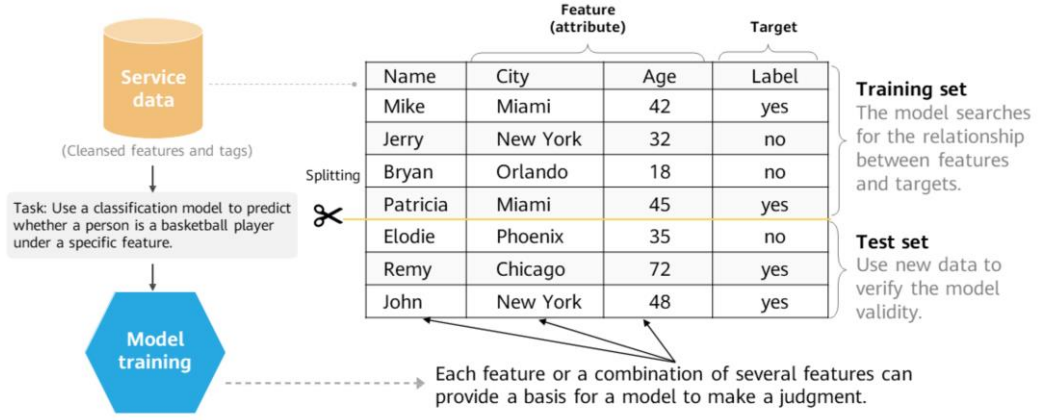


HUAWEI

- After data cleansing and feature extraction, we need to start building the model. The general procedure for building a model is shown above (supervised learning).

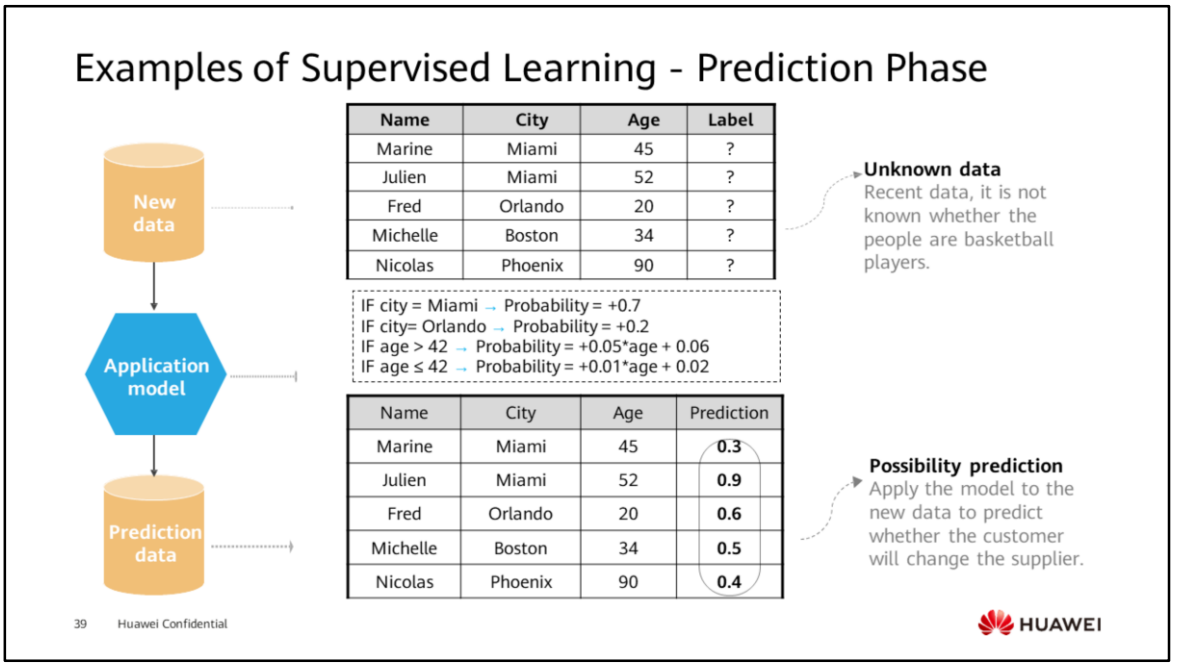
Examples of Supervised Learning - Learning Phase

- Use the classification model to predict whether a person is a basketball player.



HUAWEI

- Model example



- Model example
- The model here can be considered similar to the decision tree model.

What Is a Good Model?

- **Generalization capability**
  Can it accurately predict the actual service data?
- **Interpretability**
  Is the prediction result easy to interpret?
- **Prediction speed**
  How long does it take to predict each piece of data?
- **Practicability**
  Is the prediction rate still acceptable when the service volume increases with a huge data volume?

HUAWEI

- Which factors are used to determine a model?
- The last three are engineering factors. The generalization capability is the most important factor.

Model Validity (1)

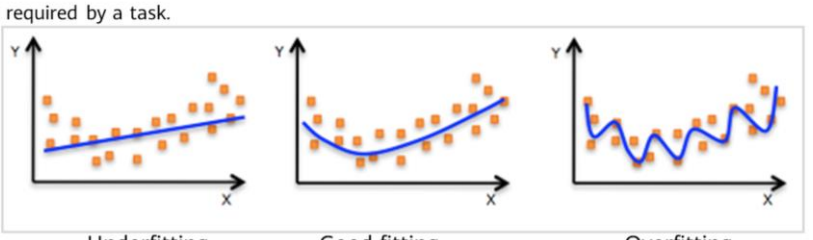
- Generalization capability: The goal of machine learning is that the model obtained after learning should perform well on new samples, not just on samples used for training. The capability of applying a model to new samples is called generalization or robustness.
- Error: difference between the sample result predicted by the model obtained after learning and the actual sample result.
  - Training error: error that you get when you run the model on the training data.
  - Generalization error: error that you get when you run the model on new samples. Obviously, we prefer a model with a smaller generalization error.
- Underfitting: occurs when the model or the algorithm does not fit the data well enough.
- Overfitting: occurs when the training error of the model obtained after learning is small but the generalization error is large (poor generalization capability).

HUAWEI

- Once the form of a problem's hypothesis is given, all possible functions constitute a space, which is hypothesis space. The problem of machine learning is searching for a suitable fitting function in a hypothesis space.
- Overfitting: It occurs frequently in complex mathematical models. To prevent overfitting, we can simplify mathematical models, end training before overfitting, or use dropout/weight decay methods.
- Underfitting: It occurs if the mathematical model is too simple or the training time is too short. To solve the underfitting problem, use a more complex model or extend the training time.

Model Validity (2)

- Model capacity: model's capability of fitting functions, which is also called model complexity.
  - When the capacity suits the task complexity and the amount of training data provided, the algorithm effect is usually optimal.
  - Models with insufficient capacity cannot solve complex tasks and underfitting may occur.
  - A high-capacity model can solve complex tasks, but overfitting may occur if the capacity is higher than that required by a task.



Underfitting
Not all features are learned.

Good fitting

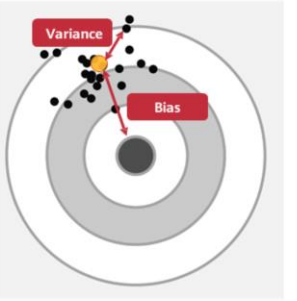
Overfitting
Noises are learned.

HUAWEI

- The effective capacity is restricted by algorithms, parameters, and regularization.

Overfitting Cause — Error

- Total error of final prediction = Bias2 + Variance + Irreducible error
- Generally, the prediction error can be divided into two types:
  - Error caused by "bias"
  - Error caused by "variance"
- Variance:
  - Offset of the prediction result from the average value
  - Error caused by the model's sensitivity to small fluctuations in the training set
- Bias:
  - Difference between the expected (or average) prediction value and the correct value we are trying to predict.

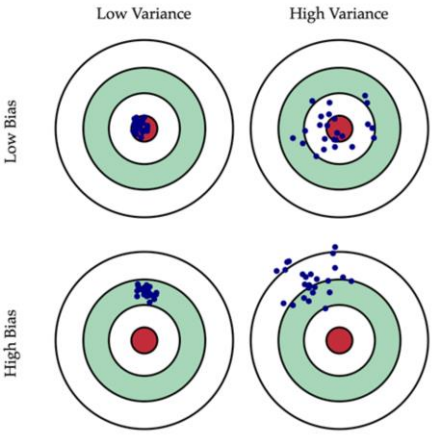


HUAWEI

- About the unresolvable error
  - Theoretically, if there is infinite amount of data and a perfect model, the error can be eliminated.
  - Actually, all models are imperfect, and our data volume is limited.

Variance and Bias

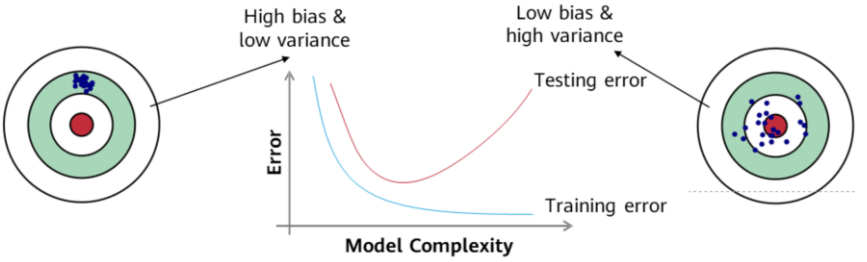
- Combinations of variance and bias are as follows:
  - Low bias & low variance –> Good model
  - Low bias & high variance
  - High bias & low variance
  - High bias & high variance –> Poor model
- Ideally, we want a model that can accurately capture the rules in the training data and summarize the invisible data (new data). However, it is usually impossible for the model to complete both tasks at the same time.



HUAWEI

Model Complexity and Error

- As the model complexity increases, the training error decreases.
- As the model complexity increases, the test error decreases to a certain point and then increases in the reverse direction, forming a convex curve.



HUAWEI

Machine Learning Performance Evaluation - Regression

- The closer the Mean Absolute Error (MAE) is to 0, the better the model can fit the training data.

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

- Mean Square Error (MSE)

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

- The value range of R^2 is (–∞, 1]. A larger value indicates that the model can better fit the training data. TSS indicates the difference between samples. RSS indicates the difference between the predicted value and sample value.

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

HUAWEI

Machine Learning Performance Evaluation - Classification (1)

- Terms and definitions:
  - P: positive, indicating the number of real positive cases in the data.
  - N: negative, indicating the number of real negative cases in the data.
  - TP : true positive, indicating the number of positive cases that are correctly classified by the classifier.
  - TN: true negative, indicating the number of negative cases that are correctly classified by the classifier.
  - FP: false positive, indicating the number of positive cases that are incorrectly classified by the classifier.
  - FN: false negative, indicating the number of negative cases that are incorrectly classified by the classifier.

Actual amount \ Estimated amount	yes	no	Total
yes	TP	FN	P
no	FP	TN	N
Total	P'	N'	$P+N$

Confusion matrix

- Confusion matrix: at least an $m \times m$ table. $CM_{i,j}$ of the first m rows and m columns indicates the number of cases that actually belong to class i but are classified into class j by the classifier.
  - Ideally, for a high accuracy classifier, most prediction values should be located in the diagonal from $CM_{1,1}$ to $CM_{m,m}$ of the table while values outside the diagonal are 0 or close to 0. That is, FP and FP are close to 0.

HUAWEI

Machine Learning Performance Evaluation - Classification (2)

Measurement	Ratio
Accuracy and recognition rate	$\frac{TP+TN}{P+N}$
Error rate and misclassification rate	$\frac{FP+FN}{P+N}$
Sensitivity, true positive rate, and recall	$\frac{TP}{P}$
Specificity and true negative rate	$\frac{TN}{N}$
Precision	$\frac{TP}{TP+FP}$
F_1, harmonic mean of the recall rate and precision	$\frac{2 \times precision \times recall}{precision + recall}$
F_β, where β is a non-negative real number	$\frac{(1+\beta^2) \times precision \times recall}{\beta^2 \times precision + recall}$

HUAWEI

Example of Machine Learning Performance Evaluation

- We have trained a machine learning model to identify whether the object in an image is a cat. Now we use 200 pictures to verify the model performance. Among the 200 images, objects in 170 images are cats, while others are not. The identification result of the model is that objects in 160 images are cats, while others are not.

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

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

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

Actual amount \ Estimated amount	yes	no	Total
yes	140	30	170
no	20	10	30
Total	160	40	200

HUAWEI

Contents

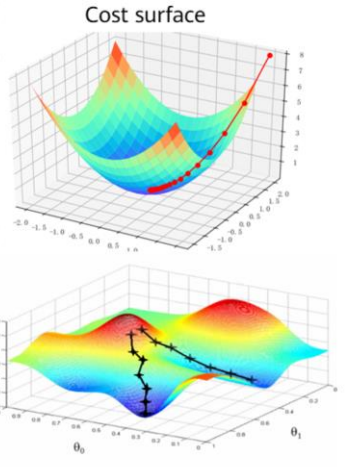
HUAWEI

Machine Learning Training Method - Gradient Descent (1)

- The gradient descent method uses the negative gradient direction of the current position as the search direction, which is the steepest direction. The formula is as follows:

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

- In the formula, η indicates the learning rate and i indicates the data record number i. The weight parameter w indicates the change in each iteration.
- Convergence: The value of the objective functior changes very little, or the maximum number o iterations is reached.



HUAWEI

Machine Learning Training Method - Gradient Descent (2)

- Batch Gradient Descent (BGD) uses the samples (m in total) in all datasets to update the weight parameter based on the gradient value at the current point.

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

- Stochastic Gradient Descent (SGD) randomly selects a sample in a dataset to update the weight parameter based on the gradient value at the current point.

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

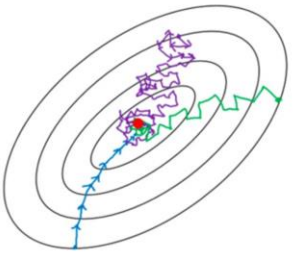
- Mini-Batch Gradient Descent (MBGD) combines the features of BGD and SGD and selects the gradients of n samples in a dataset to update the weight parameter.

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

HUAWEI

Machine Learning Training Method - Gradient Descent (3)

- Comparison of three gradient descent methods
  - In the SGD, samples selected for each training are stochastic. Such instability causes the loss function to be unstable or even causes reverse displacement when the loss function decreases to the lowest point.
  - BGD has the highest stability but consumes too many computing resources. MBGD is a method that balances SGD and BGD.



BGD
Uses **all** training samples for training each time.

SGD
Uses **one** training sample for training each time.

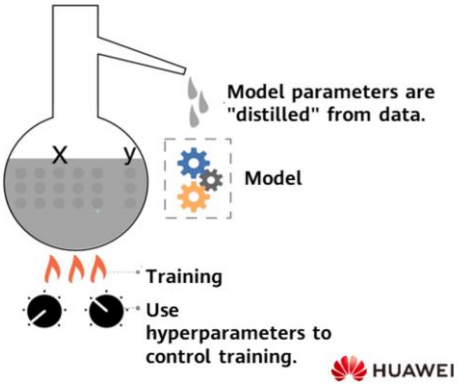
MBGD
Uses a certain number of training samples for training each time.

HUAWEI

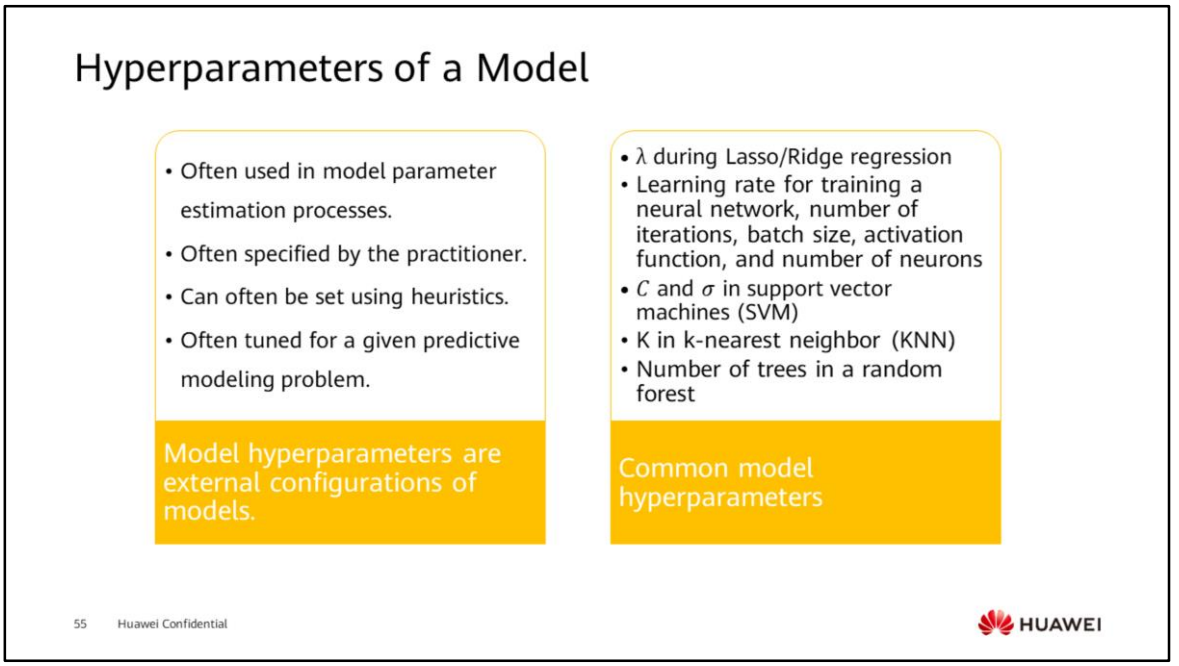
- Reverse displacement is caused by noise data from datasets.

Parameters and Hyperparameters in Models

- The model contains not only parameters but also hyperparameters. The purpose is to enable the model to learn the optimal parameters.
  - Parameters are automatically learned by models.
  - Hyperparameters are manually set.

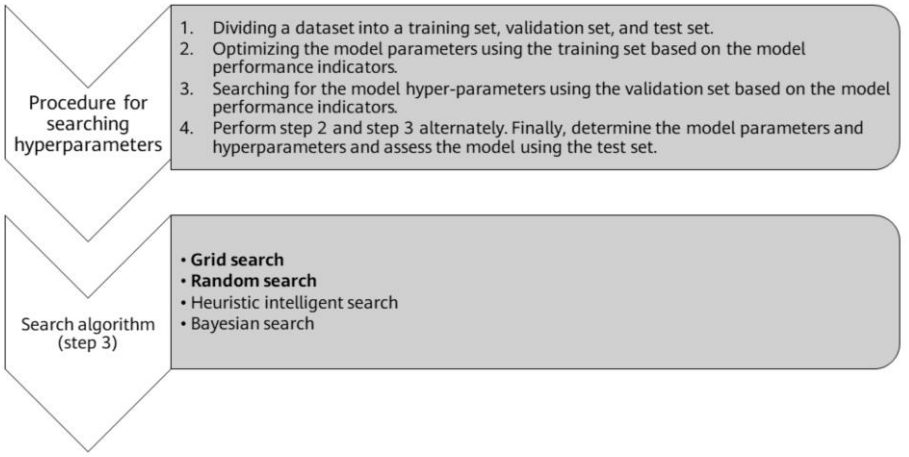


HUAWEI



- Parameters are a part of a model that is learned from historical training data and key to machine learning algorithms. They have the following features:
  - They are required by a model to make predictions.
  - Their values define the model functions.
  - They are obtained by data estimation or data learning.
  - They are often not set manually by the practitioner.
  - They are often saved as a part of the learned model.
- Examples:
  - Weights in an artificial neural network
  - Support vectors in a support vector machine
  - Coefficients in linear regression or logistic regression

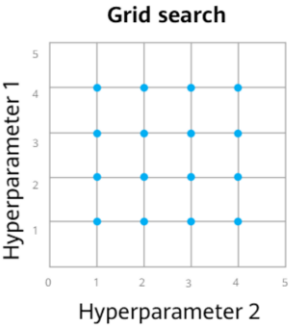
Hyperparameter Search Procedure and Method



HUAWEI

Hyperparameter Searching Method - Grid Search

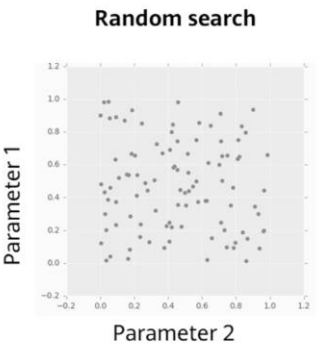
- Grid search attempts to **exhaustively search** all possible hyperparameter combinations to form a hyperparameter value grid.
- In practice, the range of hyperparameter values to search is specified manually.
- Grid search is an expensive and time-consuming method.
  - This method works well when the number of hyperparameters is relatively small. Therefore, it is applicable to generally machine learning algorithms but inapplicable to neural networks
    (see the deep learning part).



HUAWEI

Hyperparameter Searching Method - Random Search

- When the hyperparameter search space is large, **random search** is better than grid search.
- In random search, each setting is sampled from the distribution of possible parameter values, in an attempt to find the best subset of hyperparameters.
- Note:
  - Search is performed within a coarse range, which then will be narrowed based on where the best result appears.
  - Some hyperparameters are more important than others, and the search deviation will be affected during random search.



HUAWEI

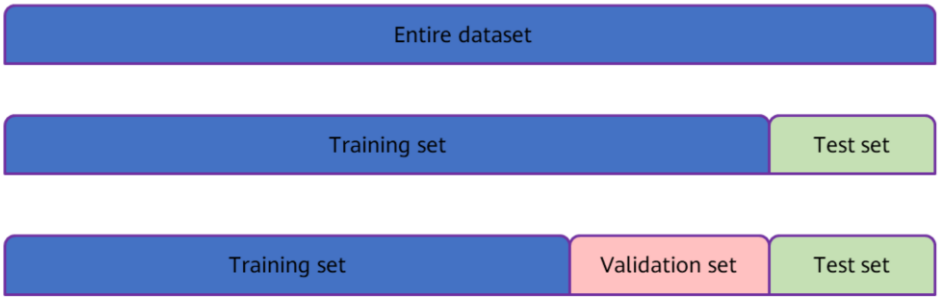
Cross Validation (1)

- **Cross validation**: It is a statistical analysis method used to validate the performance of a classifier. The basic idea is to divide the original dataset into two parts: training set and validation set. Train the classifier using the training set and test the model using the validation set to check the classifier performance.
- **k-fold cross validation ($K-CV$):**
  - Divide the raw data into k groups (generally, evenly divided).
  - Use each subset as a validation set, and use the other $k-1$ subsets as the training set. A total of k models can be obtained.
  - Use the mean classification accuracy of the final validation sets of k models as the performance indicator of the $K-CV$ classifier.

HUAWEI

- Dividing a dataset into a fixed training set and a fixed test set can be problematic if the error of the test set is small. A small test set implies statistical uncertainty around the estimated mean test error, making it difficult to claim that algorithm A works better than algorithm B on the given task. When the dataset has hundreds of thousands of samples or more, this is not a serious problem. When the dataset is too small, alternative procedures enable one to use all the samples in the estimation of the mean test error, at the price of increased computational workload.
- K-CV: Generally, the value of k is greater than or equal to 2. In practice, the value is greater than or equal to 3. The value 2 is used only when the original dataset is small. $K-CV$ can effectively avoid over-learning and under-learning, and the final result is also persuasive.

Cross Validation (2)



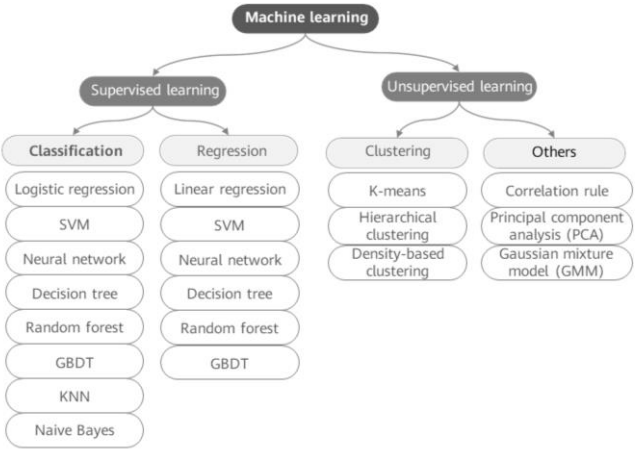
- Note: The K value in K-fold cross validation is also a hyperparameter.

HUAWEI

Contents

HUAWEI

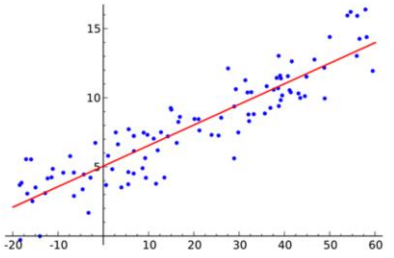
Machine Learning Algorithm Overview



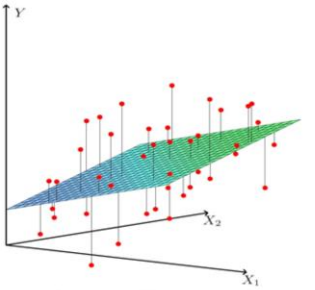
HUAWEI

Linear Regression (1)

- Linear regression: a statistical analysis method to determine the quantitative relationships between two or more variables through regression analysis in mathematical statistics.
- Linear regression is a type of supervised learning.



Unary linear regression



Multi-dimensional linear regression

HUAWEI

- Unary linear regression analysis only involves one independent variable and one dependent variable and their relationship can be approximately represented by a straight line. Multiple linear regression analysis involves two or more independent variables and the relationship between independent variables and dependent variables is linear. The learning result is not necessarily a straight line. It is a straight line only when the variable x is one-dimensional, and it is a hyperplane when the variable is high-dimensional. For example, the price of an apartment is determined by a variety of factors such as the area, layout, and location. Prediction of the apartment price based on these factors can be abstracted into a linear regression problem.
- The above figures show polynomial linear regression.

Linear Regression (2)

- The model function of linear regression is as follows, where w indicates the weight parameter, b indicates the bias, and x indicates the sample attribute.

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

- The relationship between the value predicted by the model and actual value is as follows, where y indicates the actual value, and ε indicates the error.

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

- The error ε is influenced by many factors independently. According to the central limit theorem, the error ε follows normal distribution. According to the normal distribution function and maximum likelihood estimation, the loss function of linear regression is as follows:

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

- To make the predicted value close to the actual value, we need to minimize the loss value. We can use the gradient descent method to calculate the weight parameter w when the loss function reaches the minimum, and then complete model building.

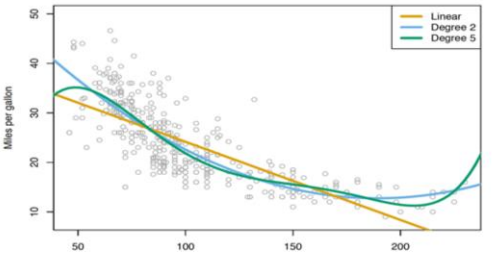
HUAWEI

Linear Regression Extension - Polynomial Regression

- Polynomial regression is an extension of linear regression. Generally, the complexity of a dataset exceeds the possibility of fitting by a straight line. That is, obvious underfitting occurs if the original linear regression model is used. The solution is to use polynomial regression.

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

- where, the nth power is a polynomial regression dimension (degree).
- Polynomial regression belongs to linear regression as the relationship between its weight parameters w is still linear while its nonlinearity is reflected in the feature dimension.



Comparison between linear regression and polynomial regression

HUAWEI

Linear Regression and Overfitting Prevention

- Regularization terms can be used to reduce overfitting. The value of w cannot be too large or too small in the sample space. You can add a square sum loss on the target function.

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

- Regularization terms (norm): The regularization term here is called L2-norm. Linear regression that uses this loss function is also called Ridge regression.

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

- Linear regression with absolute loss is called Lasso regression.

HUAWEI

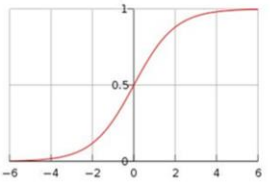
Logistic Regression (1)

- Logistic regression: The logistic regression model is used to solve classification problems. The model is defined as follows:

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

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

where w indicates the weight, b indicates the bias, and $wx + b$ is regarded as the linear function of x. Compare the preceding two probability values. The class with a higher probability value is the class of x.



HUAWEI

Logistic Regression (2)

- Both the logistic regression model and linear regression model are generalized linear models. Logistic regression introduces nonlinear factors (the sigmoid function) based on linear regression and sets thresholds, so it can deal with binary classification problems.
- According to the model function of logistic regression, the loss function of logistic regression can be estimated as follows by using the maximum likelihood estimation:

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

- where w indicates the weight parameter, m indicates the number of samples, x indicates the sample, and y indicates the real value. The values of all the weight parameters w can also be obtained through the gradient descent algorithm.

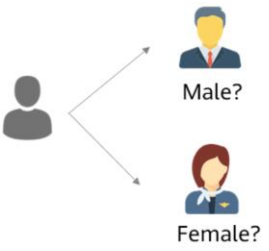
HUAWEI

- After regularization terms are added, logistic regression can also prevent overfitting.

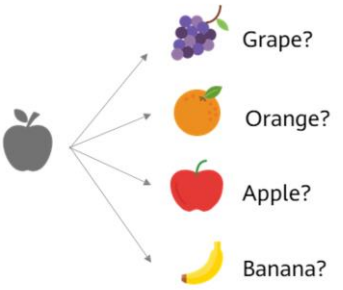
Logistic Regression Extension - Softmax Function (1)

- Logistic regression applies only to binary classification problems. For multi-class classification problems, use the Softmax function.

Binary classification problem



Multi-class classification problem



HUAWEI

Logistic Regression Extension - Softmax Function (2)

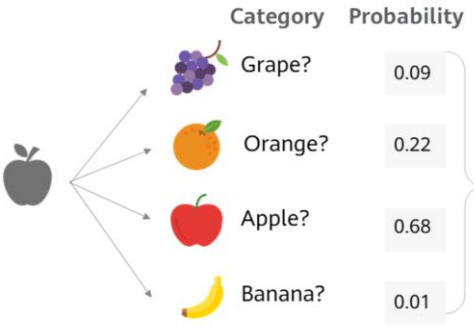
- Softmax regression is a generalization of logistic regression that we can use for K-class classification.
- The Softmax function is used to map a K-dimensional vector of arbitrary real values to another K-dimensional vector of real values, where each vector element is in the interval (0, 1).
- The regression probability function of Softmax is as follows:

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

HUAWEI

Logistic Regression Extension - Softmax Function (3)

- Softmax assigns a probability to each class in a multi-class problem. These probabilities must add up to 1.
  - Softmax may produce a form belonging to a particular class. Example:

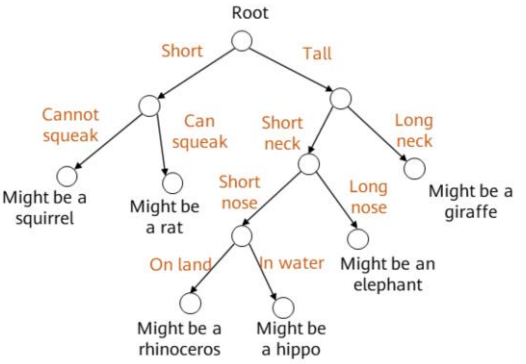


- **Sum of all probabilities:**
  - 0.09 + 0.22 + 0.68 + 0.01 =1
- Most probably, this picture is an apple.

HUAWEI

Decision Tree

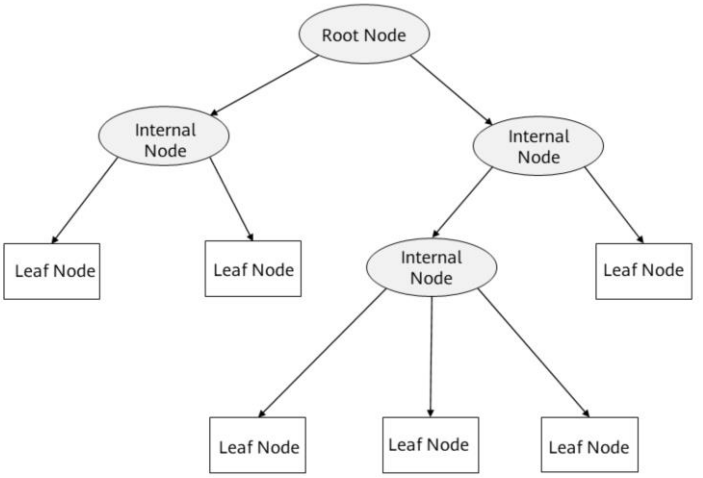
- A decision tree is a tree structure (a binary tree or a non-binary tree). Each non-leaf node represents a test on a feature attribute. Each branch represents the output of a feature attribute in a certain value range, and each leaf node stores a category. To use the decision tree, start from the root node, test the feature attributes of the items to be classified, select the output branches, and use the category stored on the leaf node as the final result.



HUAWEI

- How to construct a decision tree is very important. We should determine the topological structure of feature attributes by selecting attributes based on quantitative values. The key step is to split attributes. That is, different branches are constructed based on the differences of a feature attribute on a node.
- The decision tree learning algorithm is used to generate decision trees. Common learning algorithms include ID3, C4.5, and CART.

Decision Tree Structure



HUAWEI

- All nodes except the root node are called leaf nodes.

Key Points of Decision Tree Construction

- To create a decision tree, we need to select attributes and determine the tree structure between feature attributes. The key step of constructing a decision tree is to divide data of all feature attributes, compare the result sets in terms of 'purity', and select the attribute with the highest 'purity' as the data point for dataset division.
- The metrics to quantify the 'purity' include the information entropy and GINI Index. The formula is as follows:

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

- where p_k indicates the probability that the sample belongs to class k (there are K classes in total). A greater difference between purity before segmentation and that after segmentation indicates a better decision tree.
- Common decision tree algorithms include ID3, C4.5, and CART.

HUAWEI

Decision Tree Construction Process

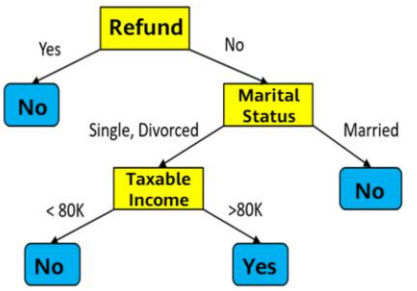
- **Feature selection:** Select a feature from the features of the training data as the split standard of the current node. (Different standards generate different decision tree algorithms.)
- **Decision tree generation:** Generate internal node upside down based on the selected features and stop until the dataset can no longer be split.
- **Pruning:** The decision tree may easily become overfitting unless necessary pruning (including pre-pruning and post-pruning) is performed to reduce the tree size and optimize its node structure.

HUAWEI

Decision Tree Example

- The following figure shows a classification when a decision tree is used. The classification result is impacted by three attributes: Refund, Marital Status, and Taxable Income.

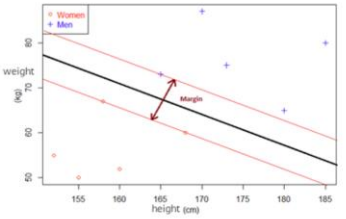
Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125,000	No
2	No	Married	100,000	No
3	No	Single	70,000	No
4	Yes	Married	120,000	No
5	No	Divorced	95,000	Yes
6	No	Married	60,000	No
7	Yes	Divorced	220,000	No
8	No	Single	85,000	Yes
9	No	Married	75,000	No
10	No	Single	90,000	Yes

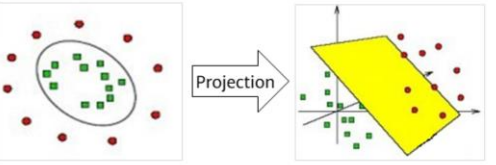


HUAWEI

SVM

- SVM is a binary classification model whose basic model is a linear classifier defined in the eigenspace with the largest interval. SVMs also include kernel tricks that make them nonlinear classifiers. The SVM learning algorithm is the optimal solution to convex quadratic programming.





Complex segmentation in low-dimensional space

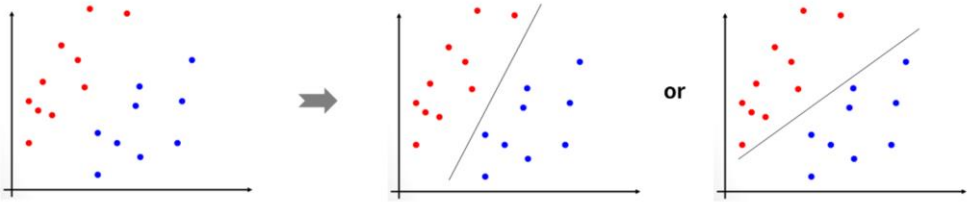
Easy segmentation in high-dimensional space

HUAWEI

- The main ideas of SVM include two points:
  - In the case of linear inseparability, non-linear mapping algorithms are used to convert the linearly inseparable samples of low-dimensional input space into high-dimensional eigenspace. In this way, samples are linearly separable. Then the linear algorithm can be used to analyze the non-linear features of samples.
  - Based on the structural risk minimization principle, an optimal hyperplane is constructed in the eigenspace, so that the learner is optimized globally, and the expectation of the whole sample space satisfies an upper boundary with a certain probability.

Linear SVM (1)

- How do we split the red and blue datasets by a straight line?



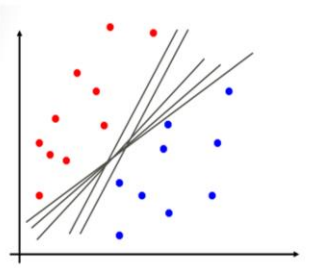
With binary classification
Two-dimensional dataset

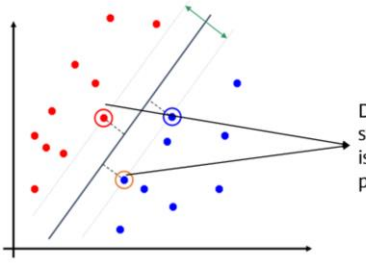
Both the left and right methods can be used to divide datasets. Which of them is correct?

HUAWEI

Linear SVM (2)

- Straight lines are used to divide data into different classes. Actually, we can use multiple straight lines to divide data. The core idea of the SVM is to find a straight line and keep the point close to the straight line as **far** as possible from the straight line. This can enable strong generalization capability of the model. These points are called **support vectors**.
- In two-dimensional space, we use straight lines for segmentation. In high-dimensional space, we use **hyperplanes** for segmentation.

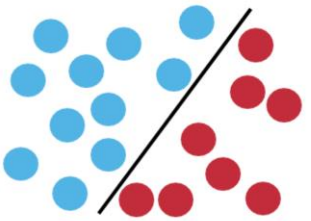




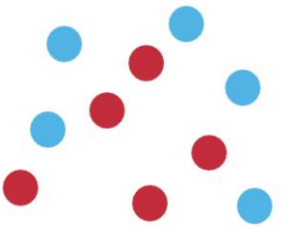
HUAWEI

Nonlinear SVM (1)

- How do we classify a nonlinear separable dataset?



Linear SVM can function well for linear separable datasets.

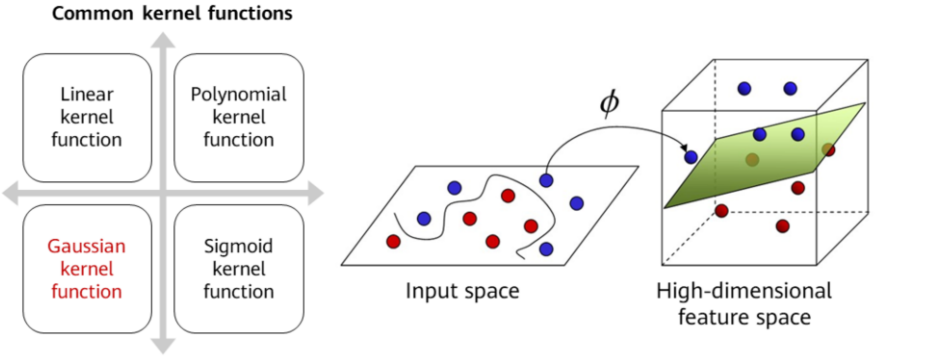


Nonlinear datasets cannot be split with straight lines.

HUAWEI

Nonlinear SVM (2)

- Kernel functions are used to construct nonlinear SVMs.
- Kernel functions allow algorithms to fit the largest hyperplane in a transformed high-dimensional feature space.

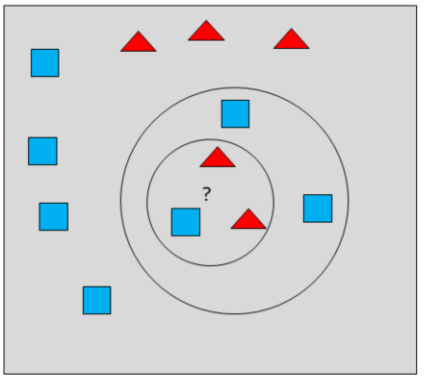


HUAWEI

- Gaussian kernel functions are most frequently used.

KNN Algorithm (1)

- The KNN classification algorithm is a theoretically mature method and one of the simplest machine learning algorithms. According to this method, if the majority of k samples most similar to one sample (nearest neighbors in the eigenspace) belong to a specific category, this sample also belongs to this category.

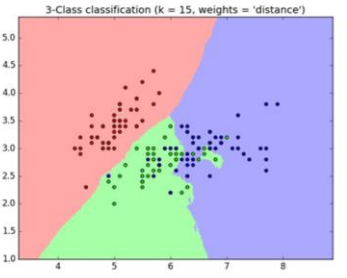


The target category of point ? varies with the number of the most adjacent nodes.

HUAWEI

KNN Algorithm (2)

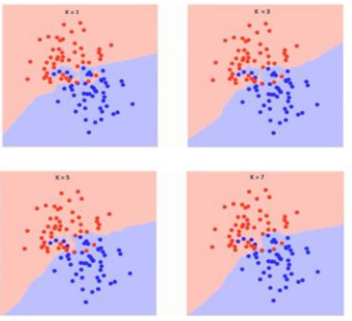
- As the prediction result is determined based on the number and weights of neighbors in the training set, the KNN algorithm has a simple logic.
- KNN is a non-parametric method which is usually used in datasets with irregular decision boundaries.
  - The KNN algorithm generally adopts the majority voting method for classification prediction and the average value method for regression prediction.
- KNN requires a huge number of computations.



HUAWEI

KNN Algorithm (3)

- Generally, a larger k value reduces the impact of noise on classification, but obfuscates the boundary between classes.
  - A larger k value means a higher probability of underfitting because the segmentation is too rough. A smaller k value means a higher probability of overfitting because the segmentation is too refined.



- The boundary becomes smoother as the value of k increases.
- As the k value increases to infinity, all data points will eventually become all blue or all red.

HUAWEI

Naive Bayes (1)

- Naive Bayes algorithm: a simple multi-class classification algorithm based on the Bayes theorem. It assumes that features are independent of each other. For a given sample feature X, the probability that a sample belongs to a category H is:

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

  - $X_1, \ldots, X_n$ are data features, which are usually described by measurement values of m attribute sets.
    - For example, the color feature may have three attributes: red, yellow, and blue.
  - C_k indicates that the data belongs to a specific category C
  - $P(C_k|X_1, \ldots, X_n)$ is a posterior probability, or a posterior probability of under condition C_k.
  - $P(C_k)$ is a prior probability that is independent of $X_1, \ldots, X_n$
  - $P(X_1, \ldots, X_n)$ is the priori probability of X.

HUAWEI

- Class conditional independence: The Bayes classifier assumes that the effect of an attribute value on a given class is independent of the values of other attributes. This assumption is made to simplify the calculation and becomes "naive" in this sense.
- The Bayes classifier, featuring high accuracy and speed, can be applied to large databases.

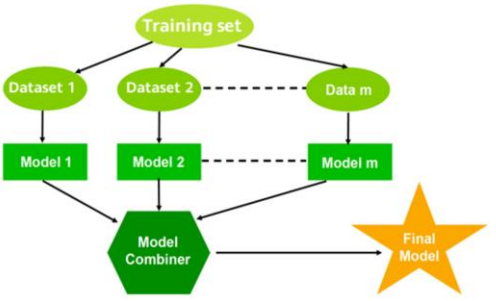
Naive Bayes (2)

- Independent assumption of features.
  - For example, if a fruit is red, round, and about 10 cm (3.94 in.) in diameter, it can be considered an apple.
  - A Naive Bayes classifier considers that each feature independently contributes to the probability that the fruit is an apple, regardless of any possible correlation between the color, roundness, and diameter.

HUAWEI

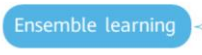
Ensemble Learning

- Ensemble learning is a machine learning paradigm in which multiple learners are trained and combined to solve the same problem. When multiple learners are used, the integrated generalization capability can be much stronger than that of a single learner.
- If you ask a complex question to thousands of people at random and then summarize their answers, the summarized answer is better than an expert's answer in most cases. This is the wisdom of the masses.



HUAWEI

Classification of Ensemble Learning



Bagging (Random Forest)

- Independently builds several basic learners and then averages their predictions.
- On average, a composite learner is usually better than a single-base learner because of a smaller variance.

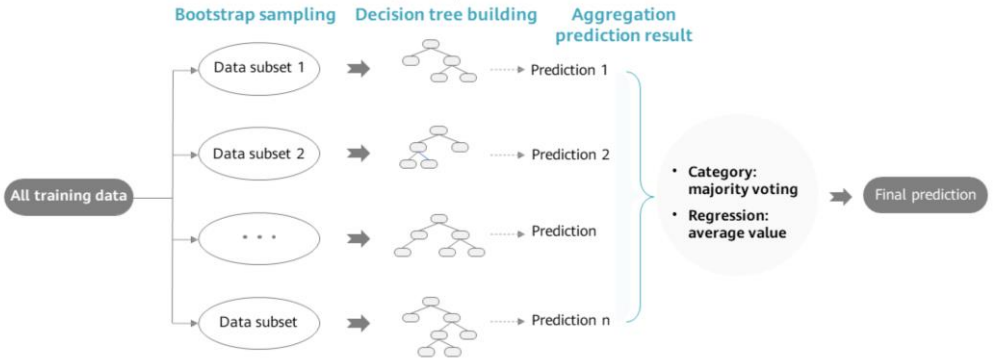
Boosting (Adaboost, GBDT, and XGboost)

Constructs basic learners in sequence to gradually reduce the bias of a composite learner. The composite learner can fit data well, which may also cause overfitting.

HUAWEI

Ensemble Methods in Machine Learning (1)

- Random forest = Bagging + CART decision tree
- Random forests build multiple decision trees and merge them together to make predictions more accurate and stable.
  - Random forests can be used for classification and regression problems.

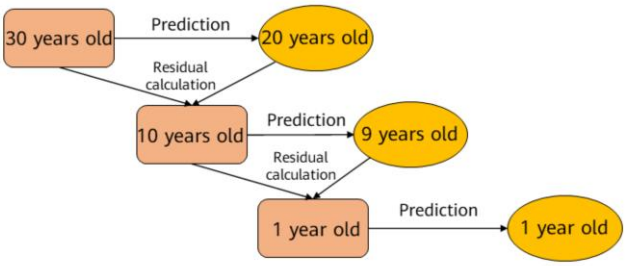


HUAWEI

- The Bootstrap bagging method is based on repeatable random sampling.

Ensemble Methods in Machine Learning (2)

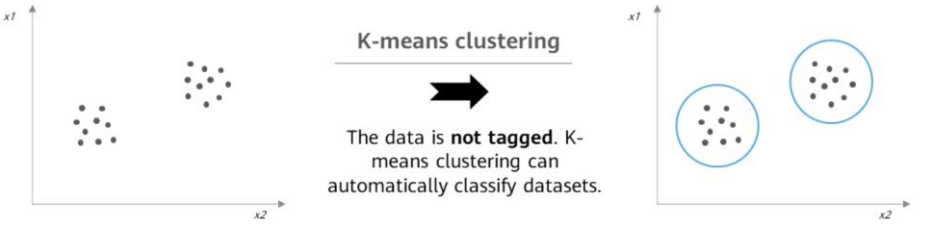
- GBDT is a type of boosting algorithm.
- For an aggregative mode, the sum of the results of all the basic learners equals the predicted value. In essence, the residual of the error function to the predicted value is fit by the next basic learner. (The residual is the error between the predicted value and the actual value.)
- During model training, GBDT requires that the sample loss for model prediction be as small as possible.



HUAWEI

Unsupervised Learning - K-means

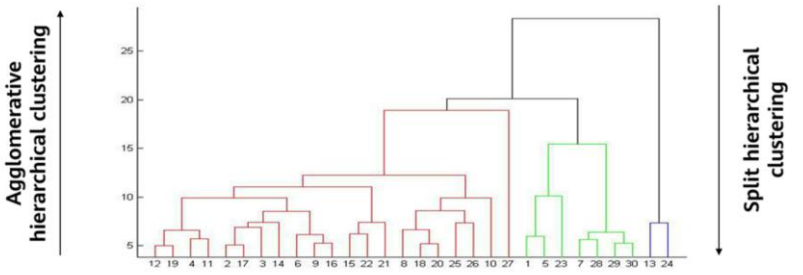
- K-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster.
- For the k-means algorithm, specify the final number of clusters (k). Then, divide n data objects into k clusters. The clusters obtained meet the following conditions: (1) Objects in the same cluster are highly similar. (2) The similarity of objects in different clusters is small.



HUAWEI

Unsupervised Learning - Hierarchical Clustering

- Hierarchical clustering divides a dataset at different layers and forms a tree-like clustering structure. The dataset division may use a "bottom-up" aggregation policy, or a "top-down" splitting policy. The hierarchy of clustering is represented in a tree graph. The root is the unique cluster of all samples, and the leaves are the cluster of only a sample.



HUAWEI

Contents

HUAWEI

Comprehensive Case

- Assume that there is a dataset containing the house areas and prices of 21,613 housing units sold in a city. Based on this data, we can predict the prices of other houses in the city.

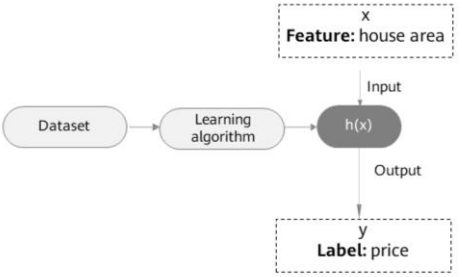
House Area	Price
1,180	221,900
2,570	538,000
770	180,000
1,960	604,000
1,680	510,000
5,420	1,225,000
1,715	257,500
1,060	291,850
1,160	468,000
1,430	310,000
1,370	400,000
1,810	530,000
...	...

Dataset

HUAWEI

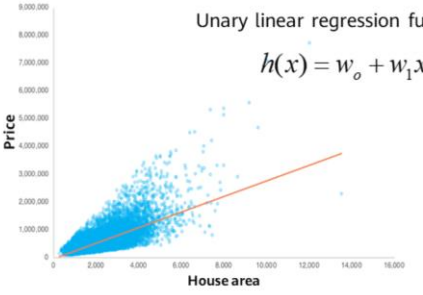
Problem Analysis

- This case contains a large amount of data, including input x (house area), and output y (price), which is a continuous value. We can use **regression** of **supervised learning**. Draw a scatter chart based on the data and use **linear regression**.
- Our goal is to build a model function h(x) that infinitely approximates the function that expresses true distribution of the dataset.
- Then, use the model to predict unknown price data.



Unary linear regression function

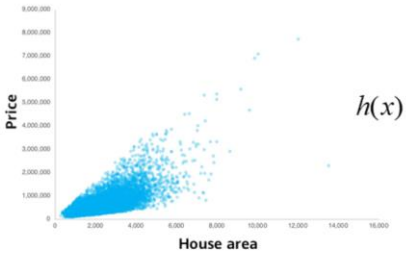
$$h(x) = w_o + w_1 x$$



HUAWEI

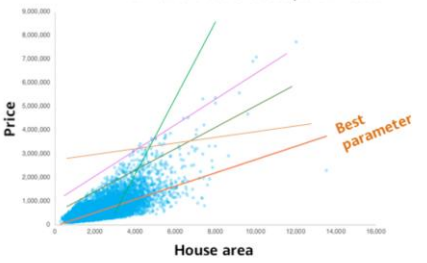
Goal of Linear Regression

- Linear regression aims to find a straight line that best fits the dataset.
- Linear regression is a parameter-based model. Here, we need learning parameters w_0 and w_1. When these two parameters are found, the best model appears.



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

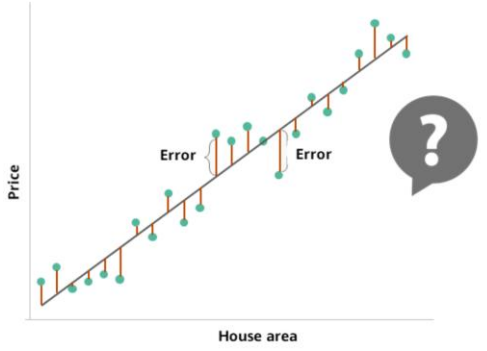
Which line is the best parameter?



HUAWEI

Loss Function of Linear Regression

- To find the optimal parameter, construct a loss function and find the parameter values when the loss function becomes the minimum.



Loss function of linear regression: $J(w)=\frac{1}{2m}\sum\left(h(x)-y\right)^2$

Goal:

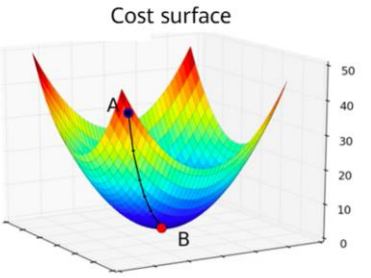
$$\arg\min_{w} J(w)=\frac{1}{2m}\sum\left(h(x)-y\right)^2$$

- where, m indicates the number of samples,
- $h(x)$ indicates the predicted value, and y indicates the actual value.

HUAWEI

Gradient Descent Method

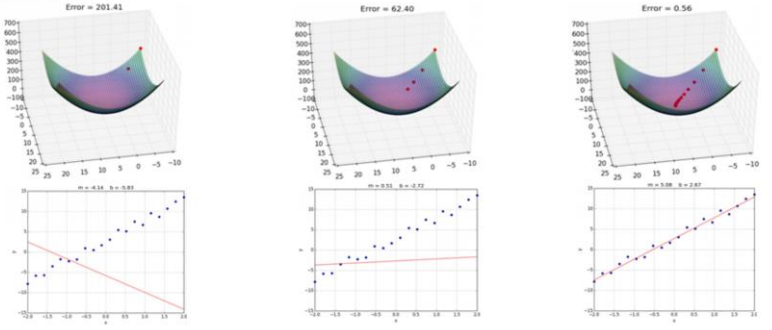
- The gradient descent algorithm finds the minimum value of a function through iteration.
- It aims to randomize an initial point on the loss function, and then find the global minimum value of the loss function based on the negative gradient direction. Such parameter value is the optimal parameter value.
  - **Point A:** the position of w_0 and w_1 after random initialization. w_0 and w_1 are the required **parameters.**
  - **A-B connection line:** a path formed based on descents in a negative gradient direction. Upon each descent, values w_0 and w_1 **change**, and the regression line also changes.
  - **Point B:** global minimum value of the loss function. Final values of w_0 and w_1 are also found.



HUAWEI

Iteration Example

- The following is an example of a gradient descent iteration. We can see that as red points on the loss function surface gradually approach a lowest point, fitting of the linear regression red line with data becomes better and better. At this time, we can get the best parameters.



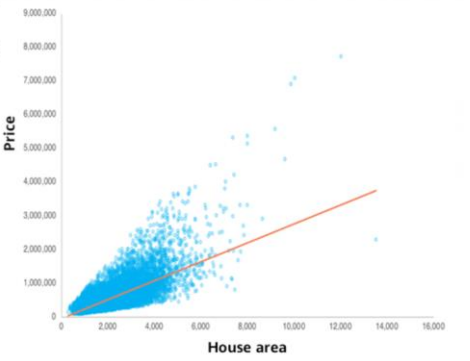
HUAWEI

Model Debugging and Application

- After the model is trained, test it with the test set to ensure the generalization capability.
- If overfitting occurs, use Lasso regression or Ridge regression with regularization terms and tune the hyperparameters.
- If underfitting occurs, use a more complex regression model, such as GBDT.
- Note:
  - For real data, pay attention to the functions of data cleansing and feature engineering.

The final model result is as follows:

$$h(x) = 280.62x - 43581$$



HUAWEI

Summary

- First, this course describes the definition and classification of machine learning, as well as problems machine learning solves. Then, it introduces key knowledge points of machine learning, including the overall procedure (data collection, data cleansing, feature extraction, model training, model training and evaluation, and model deployment), common algorithms (linear regression, logistic regression, decision tree, SVM, naive Bayes, KNN, ensemble learning, K-means, etc.), gradient descent algorithm, parameters and hyper-parameters.
- Finally, a complete machine learning process is presented by a case of using linear regression to predict house prices.

HUAWEI

Quiz

1. (True or false) Gradient descent iteration is the only method of machine learning algorithms. ( )
   A. True
   B. False
2. (Single-answer question) Which of the following algorithms is not supervised learning ? ( )
   A. Linear regression
   B. Decision tree
   C. KNN
   D. K-means

HUAWEI

- Answers: 1. B 2. D

Recommendations

- Online learning website
  - https://e.huawei.com/en/talent/#/
- Huawei Knowledge Base
  - https://support.huawei.com/enterprise/en/knowledge?lang=en

HUAWEI

Thank you.

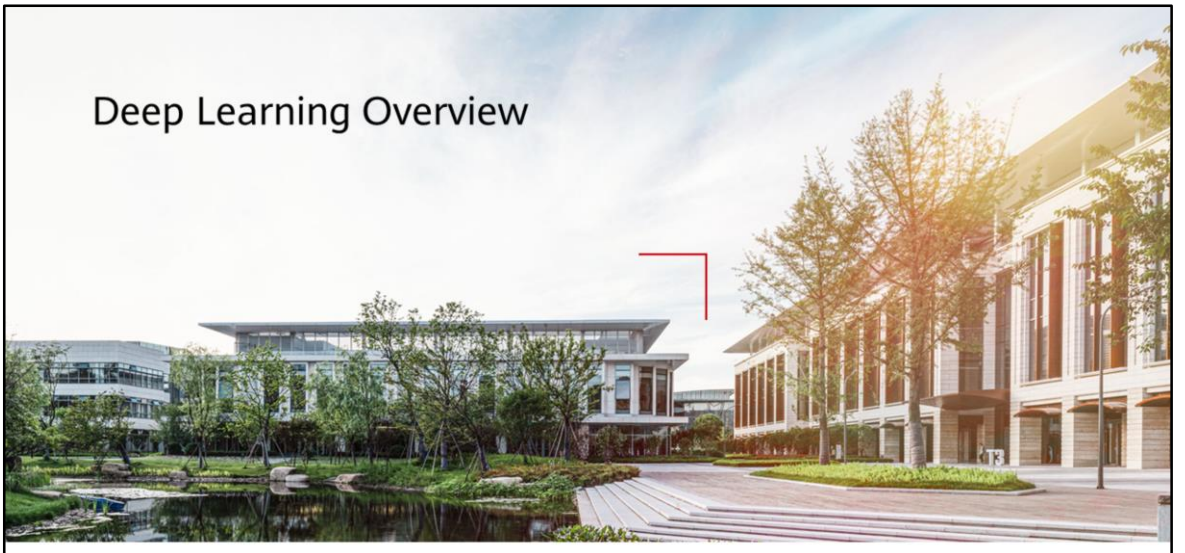
把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

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

Copyright©2020 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.

Deep Learning Overview



HUAWEI

Foreword

- The chapter describes the basic knowledge of deep learning, including the development history of deep learning, components and types of deep learning neural networks, and common problems in deep learning projects.

HUAWEI

Objectives

On completion of this course, you will be able to:

- Describe the definition and development of neural networks.
- Learn about the important components of deep learning neural networks.
- Understand training and optimization of neural networks.
- Describe common problems in deep learning.

HUAWEI

Contents

HUAWEI

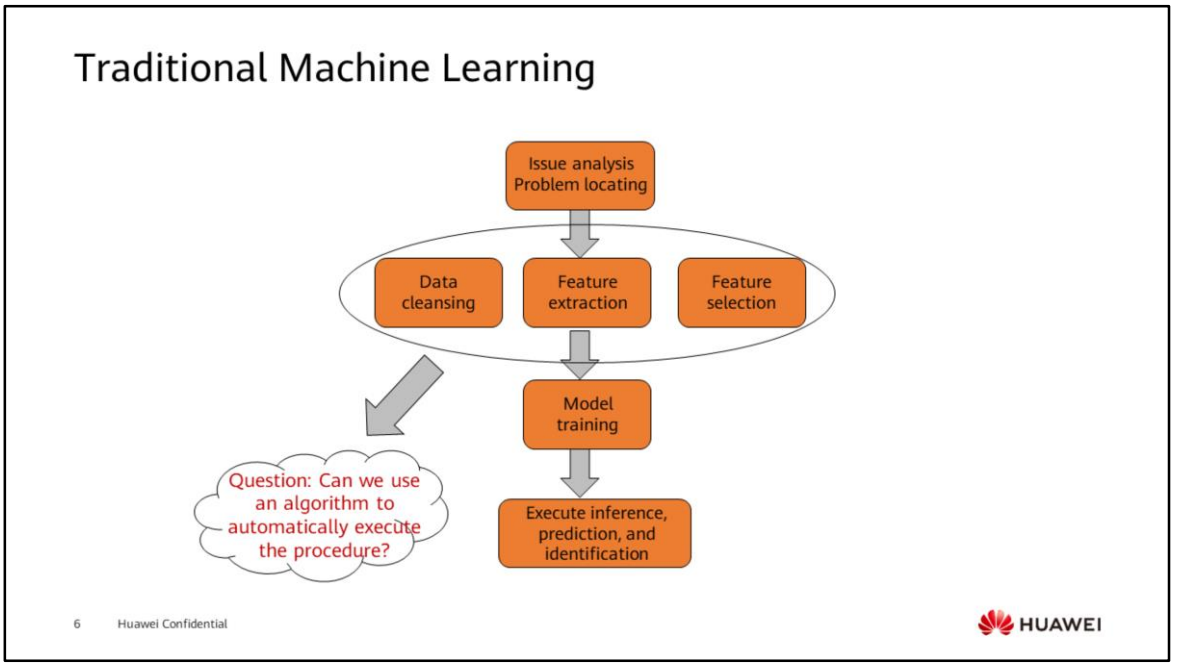
Traditional Machine Learning and Deep Learning

- As a model based on unsupervised feature learning and feature hierarchy learning, deep learning has great advantages in fields such as computer vision, speech recognition, and natural language processing.

Traditional Machine Learning	Deep Learning
Low hardware requirements on the computer: Given the limited computing amount, the computer does not need a GPU for parallel computing generally.	Higher hardware requirements on the computer: To execute matrix operations on massive data, the computer needs a GPU to perform parallel computing.
Applicable to training under a small data amount and whose performance cannot be improved continuously as the data amount increases.	The performance can be high when high-dimensional weight parameters and massive training data are provided.
Level-by-level problem breakdown	E2E learning
Manual feature selection	Algorithm-based automatic feature extraction
Easy-to-explain features	Hard-to-explain features

HUAWEI

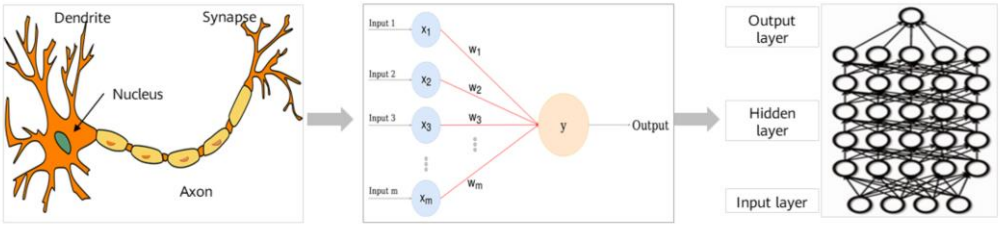
- As a model based on unsupervised feature learning and feature hierarchy learning, deep learning has great advantages in fields such as computer vision, speech recognition, and natural language processing.
- The two kinds of learning are compared from five aspects.



- Issue analysis and task determining
- Data cleansing
- Signature extraction
- Feature selection
- Model training
- Inference, prediction, and identification
- Generally, features in machine learning are selected manually. More features indicate more information and higher identification accuracy.
- However, more features indicate higher calculation complexity and larger search space. Training data appears to be sparse in the overall feature vectors, affecting similarity determination, that is, dimension exploration.
- In addition, features that have nothing to do with classification may interfere with the learning effect.
- Conclusion: More features do not mean better learning effect. Proper feature selection is the key to identification success. The number of required features can be determined by the learning problem.

Deep Learning

- Generally, the deep learning architecture is a deep neural network. "Deep" in "deep learning" refers to the number of layers of the neural network.



Human neural network | Perceptron | Deep neural network

HUAWEI

- What is deep learning?
- Generally, the deep learning architecture is a deep neural network. "Deep" in "deep learning" refers to the number of layers of the neural network.
- The network is built by simulating the human neural network.
- In the design and application of artificial neural networks, the following factors need to be considered: neuron functions, connection modes among neurons, and network learning (training).

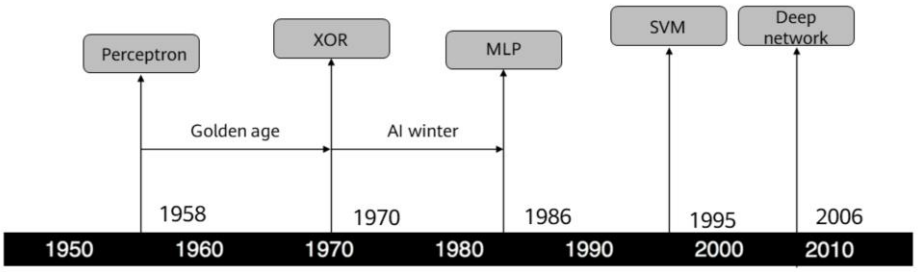
Neural Network

- Currently, the definition of the neural network has not been determined yet. Hecht Nielsen, a neural network researcher in the U.S., defines a neural network as a computer system composed of simple and highly interconnected processing elements, which process information by dynamic response to external inputs.
- A neural network can be simply expressed as an information processing system designed to imitate the human brain structure and functions based on its source, features, and explanations.
- Artificial neural network (neural network): Formed by artificial neurons connected to each other, the neural network extracts and simplifies the human brain's microstructure and functions. It is an important approach to simulate human intelligence and reflect several basic features of human brain functions, such as concurrent information processing, learning, association, model classification, and memory.

HUAWEI

- Let's see what a neural network is.

Development History of Neural Networks

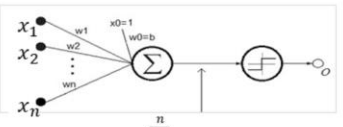


HUAWEI

- Development history of neural networks (milestones of deep learning):
- Budding neural network (1958–1969)
  - In 1958, Frank Rosenblatt invented the perceptron algorithm.
  - In 1969, Marvin Minsky, pioneer of artificial intelligence in the US, questioned that the perceptron could handle only linear classification problems and failed to classify even the simplest XOR problems. The research on the perceptron was doomed to failure.
- Developing neural network (1986–1998)
  - The second-generation neural network: In 1986, G. E. Hinton, a deep learning expert, developed a BP algorithm suitable for multilayer perceptron (MLP) and used Sigmoid for non-linear mapping, which solved the problem of non-linear classification and learning.
  - Universal approximation theorem: In 1989, Robert Hecht-Nielsen proved that a continuous function f in any closed interval could be approximated by a BP network containing a hidden layer.
- Emerging neural network (since 2006)
  - Year 2006 is the first year of deep learning. In 2006, Hinton proposed a solution to the vanishing gradient problem in the deep neural network training: weight initialization for unsupervised pre-training+weight adjustment for supervised training.
  - In 2012, CNN neural network of Hinton's team overwhelmed other methods and won the first prize in the top image recognition competition ImageNet, setting off an upsurge of deep learning.
  - In 2016, the deep learning AI program AlphaGo developed by Google beat the Go world champion Lee Sedol who is a player of 9 dan rank, further promoting the popularity of deep learning.

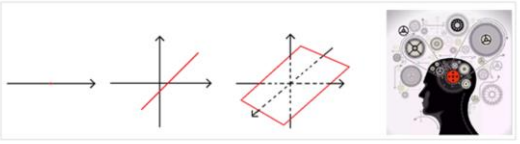
Single-Layer Perceptron

- Input vector: $X = [x_0, x_1, \ldots, x_n]^T$
- Weight: $W = [\omega_0, \omega_1, \ldots, \omega_n]^T$, in which ω_0 is the offset.
- Activation function: $O = sign(net) = \begin{cases} 1, net > 0, \\ -1, otherwise. \end{cases}$



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

- The preceding perceptron is equivalent to a classifier. It uses the high-dimensional X vector as the input and performs binary classification on input samples in the high-dimensional space. When $\boldsymbol{W^T X} > 0$, $O = 1$. In this case, the samples are classified into a type. Otherwise, $O = -1$. In this case, the samples are classified into the other type. The boundary of these two types is $\boldsymbol{W^T X} = 0$, which is a high-dimensional hyperplane.



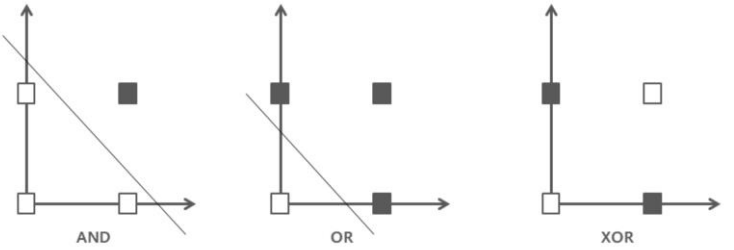
Classification point	Classification line	Classification plane	Classification hyperplane
$Ax + B = 0$	$Ax + By + C = 0$	$Ax + By + Cz + D = 0$	$W^T X + b = 0$

HUAWEI

- First, let's take a look at the simplest neural network — single-layer perceptron.
- In 1958, Frank Rosenblatt invented the perceptron algorithm. Such an algorithm coexisted with machine learning for more than a decade.
- In 1969, Marvin Minsky, pioneer of artificial intelligence in the US, questioned that the perceptron could handle only linear classification problems and failed to classify even the simplest XOR problems. The research on the perceptron was doomed to failure.

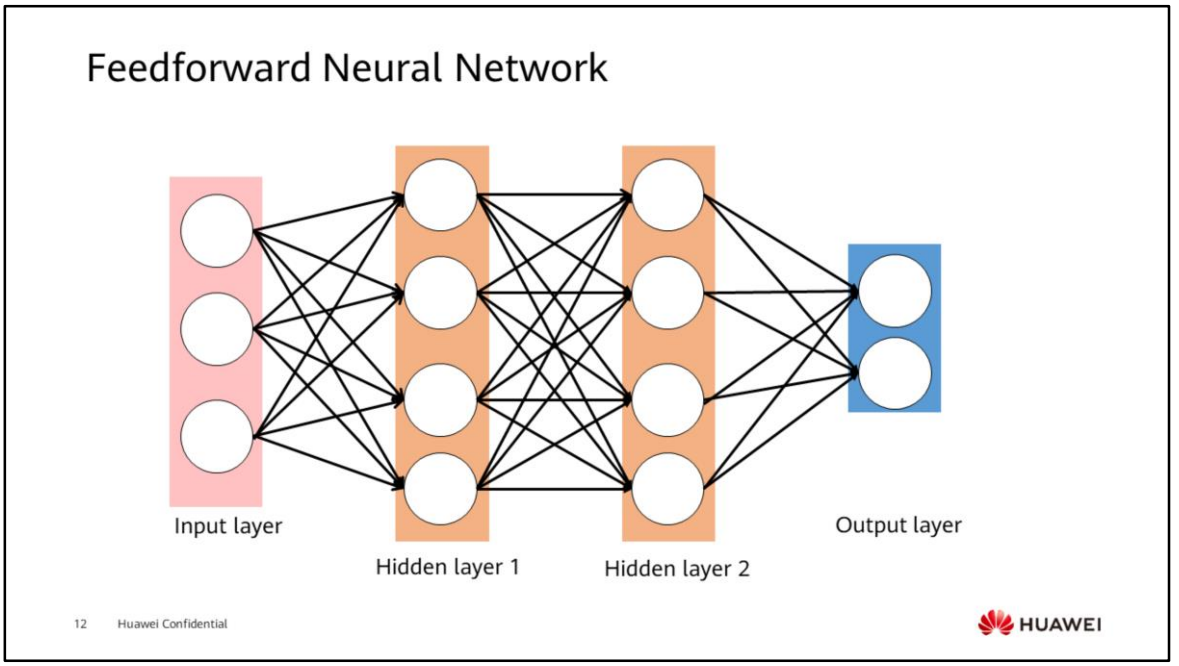
XOR Problem

- In 1969, Minsky, an American mathematician and AI pioneer, proved that a perceptron is essentially a linear model that can only deal with linear classification problems, but cannot process non-linear data.



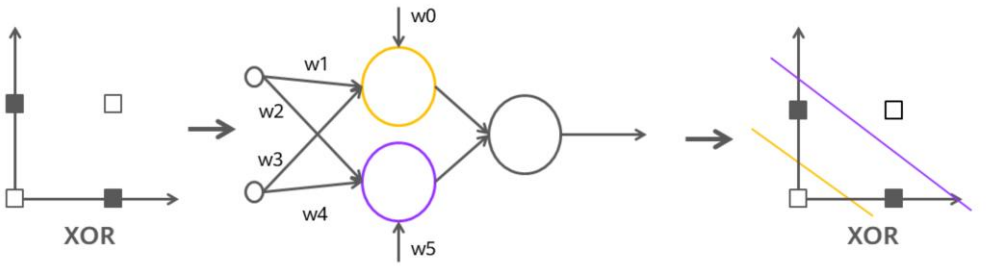
HUAWEI

- Restriction of the single-layer perceptron: XOR problem
- Reference: https://blog.csdn.net/qq_18515405/article/details/42123697
- Under this context, the multilayer perceptron was invented.



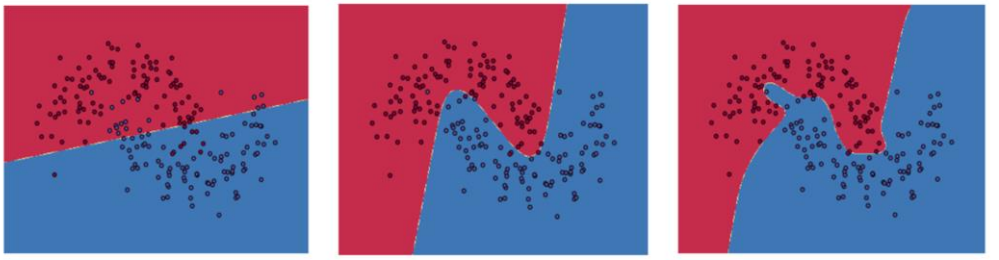
- The multilayer perceptron is a feedforward neural network.
- It is the simplest neural network with neurons arranged in layers. It is more widely used and develops faster than other artificial neural networks.
- Input nodes do not have the calculation capability and are only used to represent element values of the input vector.
- The neuron having the computing capability at each layer is referred to as the computing unit. Each neuron is connected only to the neurons of the previous layer.
- A unidirectional multi-layer structure is used to receive the output of the previous layer and send the output to the next layer. Each layer includes several neurons, and the neurons at the same layer are not connected to each other. Only unidirectional inter-layer information transmission is supported.

Solution of XOR



HUAWEI

Impacts of Hidden Layers on A Neural Network



0 hidden layers　　3 hidden layers　　20 hidden layers

HUAWEI

- More hidden layers indicate the stronger identification capability of the neural network.

Contents

HUAWEI

Gradient Descent and Loss Function

- The gradient of the multivariate function $o = f(x) = f(x_0, x_1, \dots, x_n)$ at $X' = [x_0', x_1', \dots, x_n']^T$ is shown as follows:

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

  The direction of the gradient vector is the fastest growing direction of the function. As a result, the direction of the negative gradient vector $-\nabla f$ is the fastest descent direction of the function.

- During the training of the deep learning network, target classification errors must be parameterized. **A loss function (error function)** is used, which reflects the error between the target output and actual output of the perceptron. For a single training sample x, the most common error function is the **Quadratic cost function**.

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

  In the preceding function, d is one neuron in the output layer, D is all the neurons in the output layer, t_d is the target output, and o_d is the actual output.

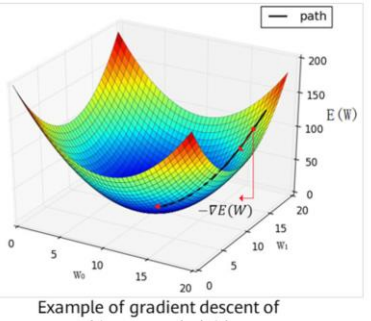
- The gradient descent method enables the loss function to search along the negative gradient direction and update the parameters iteratively, finally minimizing the loss function.

HUAWEI

- This loss function has the following features:
  - Uses the weight vector as the independent variable.
  - Uses the sum of squares of the difference between the target output t_d and actual output o_d of each training sample as the main body.
  - There is a coefficient 1/2.
- Once the training sample is given, the input and target output are constants. The actual output varies with W. The independent variable of the error function is W.
- Usage of the coefficient 1/2 which is difficult to understand is as follows: When you take a derivative of E with respect to the independent variable, the coefficient 1/2 is multiplied by the index 2 and the number 1 is obtained. The specific calculation will be described in the following text.

Extrema of the Loss Function

- Purpose: The loss function $E(W)$ is defined on the weight space. The objective is to search for the weight vector W that can minimize $E(W)$.
- Limitation: No effective method can solve the extremum in mathematics on the complex high-dimensional surface of $E(W) = \frac{1}{2}\sum_{d \in D}(t_d - o_d)^2$.



Example of gradient descent of binary paraboloid

HUAWEI

- Approach: The core approach of the gradient descent method is as follows: The negative gradient direction is the fastest descent direction of the function. Therefore, the minimum point of $E(W)$ is expected to exist along the −() direction.

Common Loss Functions in Deep Learning

- Quadratic cost function:

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

- Cross entropy error function:

$$E(W) = -\sum_{d \in D} t_d \ln o_d$$

- The cross entropy error function depicts the distance between two probability distributions, which is a widely used loss function for classification problems.
- Generally, the mean square error function is used to solve the regression problem, while the cross entropy error function is used to solve the classification problem.

HUAWEI

Batch Gradient Descent Algorithm (BGD)

- In the training sample set X, each sample is recorded as $< \mathrm{x}, t >$, in which X is the input vector, t the target output, o the actual output, and η the learning rate.
  - Initializes each w_i to a random value with a smaller absolute value.
  - Before the end condition is met:
    - Initializes each Δw_i to zero.
    - For each iteration:
      - Input all the x to this unit and calculate the output o_x.
      - For each w_i in this unit: Δw_i+= -$\eta \frac{1}{n} \sum_{x \in X} \frac{\partial C(t_x, o_x)}{\partial w_i}$.
    - For each w_i in this unit: w_i+= Δw_i.
- The gradient descent algorithm of this version is not commonly used because:
  - The convergence process is very slow as all training samples need to be calculated every time the weight is updated.

HUAWEI

Stochastic Gradient Descent Algorithm (SGD)

- To address the BGD algorithm defect, a common variant called Incremental Gradient Descent algorithm is used, which is also called the Stochastic Gradient Descent (SGD) algorithm. One implementation is called Online Learning, which updates the gradient based on each sample:

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

- ONLINE-GRADIENT-DESCENT
  - Initializes each w_i to a random value with a smaller absolute value.
  - Before the end condition is met:
  - Generates a random <x, t> from X and does the following calculation:
    - Input X to this unit and calculate the output o_x.
    - For each w_i in this unit: w_i += $-\eta \frac{\partial C(t_x, o_x)}{\partial w_i}$.

HUAWEI

- This gradient descent algorithm goes to another extreme, that is, updating the weight based on each sample. Most training samples contain noises. As a result, when the extrema is approximated to, the gradient direction is oriented up and down near the extrema but difficult to converge to the extrema.

Mini-Batch Gradient Descent Algorithm (MBGD)

- To address the defects of the previous two gradient descent algorithms, the Mini-batch Gradient Descent Algorithm (MBGD) was proposed and has been most widely used. A small number of Batch Size (BS) samples are used at a time to calculate Δw_i, and then the weight is updated accordingly.
- Batch-gradient-descent
  - Initializes each w_i to a random value with a smaller absolute value.
  - Before the end condition is met:
    - Initializes each Δw_i to zero.
    - For each $< x, t >$ in the BS samples in the next batch in B:
      - Input x to this unit and calculate the output o_x.
      - For each w_i in this unit: $\Delta w_i \mathrel{+}= -\eta \frac{1}{n} \sum_{x \in B} \frac{\partial C(t_x, o_x)}{\partial w_i}$
    - For each w_i in this unit: $w_i \mathrel{+}= \Delta w_i$
    - For the last batch, the training samples are mixed up in a random order.

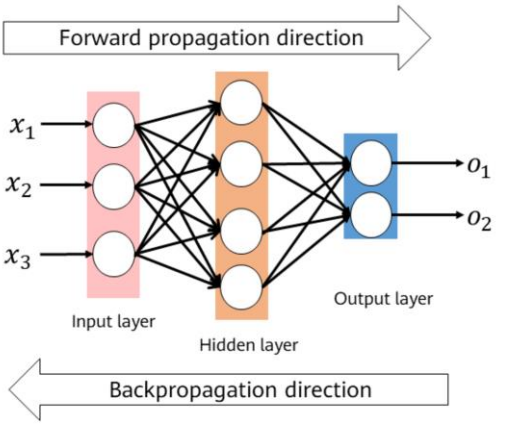
HUAWEI

- This gradient descent algorithm considers both the efficiency and gradient stability. It is easy to overshoot the local minimum and is the most commonly used gradient descent algorithm in actual work. The value of BS varies with specific problems. Generally, the value is 128.

Backpropagation Algorithm (1)

- Signals are propagated in forward direction, and errors are propagated in backward direction.
- In the training sample set D, each sample is recorded as <X, t>, in which X is the input vector, t the target output, o the actual output, and w the weight coefficient.
- Loss function:

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



HUAWEI

Backpropagation Algorithm (2)

- According to the following formulas, errors in the input, hidden, and output layers are accumulated to generate the error in the loss function.
- wc is the weight coefficient between the hidden layer and the output layer, while wb is the weight coefficient between the input layer and the hidden layer. f is the activation function, D is the output layer set, and C and B are the hidden layer set and input layer set respectively. Assume that the loss function is a quadratic cost function:
  - Output layer error: $E = \frac{1}{2}\sum_{(d \in D)}(t_d - o_d)^2$
  - Expanded hidden layer error: $E = \frac{1}{2}\sum_{(d \in D)}\left[t_d - f(net_d)\right]^2 = \frac{1}{2}\sum_{(d \in D)}\left[t_d - f(\sum_{(c \in C)} w_c y_c)\right]^2$
  - Expanded input layer error: $E = \frac{1}{2}\sum_{(d \in D)}\left[t_d - f\left(\sum_{(c \in C)} w_c f(net_c)\right)\right]^2 = \frac{1}{2}\sum_{(d \in D)}\left[t_d - f\left(\sum_{(c \in C)} w_c f\left(\sum_{b \in B} w_b x_b\right)\right)\right]^2$

HUAWEI

Backpropagation Algorithm (3)

- To minimize error E, the gradient descent iterative calculation can be used to solve W_c and W_b, that is, calculating W_c and W_b to minimize error E.
- Formula:

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

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

- If there are multiple hidden layers, chain rules are used to take a derivative for each layer to obtain the optimized parameters by iteration.

HUAWEI

Backpropagation Algorithm (4)

- For a neural network with any number of layers, the arranged formula for training is as follows:

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

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

- The BP algorithm is used to train the network as follows:
  - Takes out the next training sample <X, T>, inputs X to the network, and obtains the actual output o.
  - Calculates output layer δ according to the output layer error formula (1).
  - Calculates δ of each hidden layer from output to input by iteration according to the hidden layer error propagation formula (2).
  - According to the δ of each layer, the weight values of all the layer are updated.

HUAWEI

Contents

HUAWEI

Activation Function

- Activation functions are important for the neural network model to learn and understand complex non-linear functions. They allow introduction of non-linear features to the network.
- Without activation functions, output signals are only simple linear functions. The complexity of linear functions is limited, and the capability of learning complex function mappings from data is low.

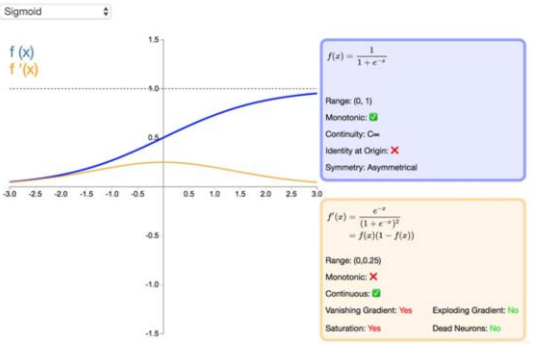
Activation Function

$$output = f(w_1x_1 + w_2x_2 + w_3x_3\text{K}) = f(W^t \bullet X)$$

HUAWEI

Sigmoid

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

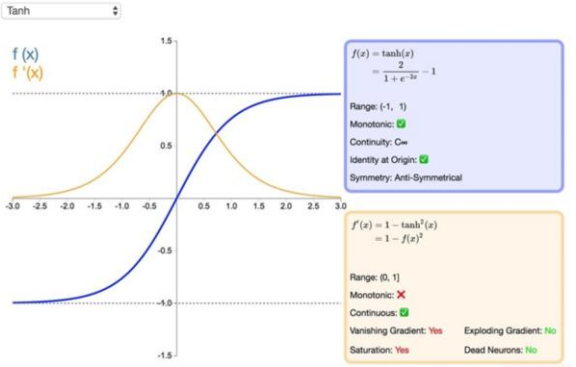


HUAWEI

- The sigmoid function is monotonic, continuous, and easy to derive. The output is bounded, and the network is easy to converge. However, we see that the derivative of the sigmoid function is close to 0 at the position away from the central point. When the network is very deep, more and more backpropagation gradients fall into the saturation area so that the gradient module becomes smaller. Generally, if the sigmoid network has five or fewer layers, the gradient is degraded to 0, which is difficult to train. This phenomenon is a vanishing gradient. In addition, the output of the sigmoid is not zero-centered.

Tanh

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

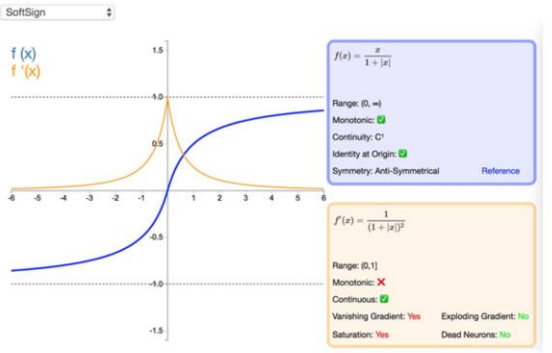


HUAWEI

- Tanh function and sigmoid function have similar shortcomings. The derivative of the tanh function is nearly 0 at its extremes. However, because the tanh function is symmetric with respect to the origin, the average of the outputs is closer to 0 than that of the sigmoid function. Therefore, SGD can reduce the required number of iterations because it is closer to the natural gradient descent.

Softsign

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

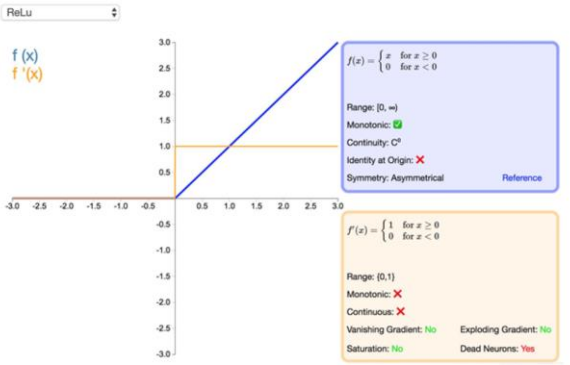


HUAWEI

- This function saturates more slowly than the tanh function.
- When the sigmoid, tanh, and softsign functions are used to train a deep neural network, the vanishing gradient problem is inevitable. The derivative of the functions approaches 0 at its extremes. When the network is very deep, more and more backpropagation gradients fall into the saturation area so that the gradient module becomes smaller and finally close to 0, and the weight cannot be updated.
- Generally, if the neural network has more than five layers, the gradient is degraded to 0, which is difficult to train.

Rectified Linear Unit (ReLU)

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

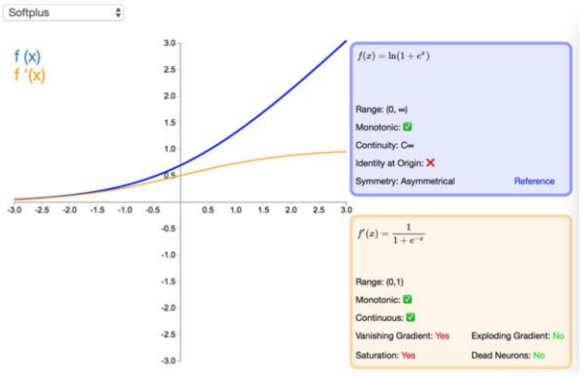


HUAWEI

- Advantages:
  - Compared with sigmoid and tanh, ReLU supports fast convergence in SGD.
  - Compared with the sigmoid and tanh functions involving exponentiation, the ReLU can be implemented more easily.
  - The vanishing gradient problem can be effectively alleviated.
  - The ReLU has a good performance during unsupervised pre-training.
- Disadvantages:
  - There is no upper bound, so that the training is relatively easy to diverge.
  - The ReLU is not differentiable at x = 0 and a derivative is forcibly defined at this point.
  - The surface defined at the zero point is not smooth enough in some regression problems.
- Reduces the computation workload
  - When functions such as sigmoid are used, the activation function involves exponent operation, which requires a large amount of computation. When the error gradient is calculated through backpropagation, the derivation involves division and the computation workload is heavy. However, the ReLU activation function can reduce much of the computation workload.
- Effectively mitigates the vanishing gradient problem.
  - The ReLU gradient is unsaturated. When the sigmoid function is close to the saturation area (far from the function center), the transformation is too slow and the derivative is close to 0. Therefore, in the backpropagation process, the ReLU function mitigates the vanishing gradient problem, and parameters of the first several layers of the neural network can be quickly updated.

Softplus

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



HUAWEI

- Compared with ReLU, this function has more complex computation. However, it has a continuous derivative and defines a smooth curved surface.

Softmax

- Softmax function:

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

- The Softmax function is used to map a K-dimensional vector of arbitrary real values to another K-dimensional vector of real values, where each vector element is in the interval (0, 1). All the elements add up to 1.
- The Softmax function is often used as the output layer of a multiclass classification task.

HUAWEI

Contents

HUAWEI

Normalizer

- Regularization is an important and effective technology to reduce generalization errors in machine learning. It is especially useful for deep learning models that tend to be over-fit due to a large number of parameters. Therefore, researchers have proposed many effective technologies to prevent over-fitting, including:
  - Adding constraints to parameters, such as L_1 and L_2 norms
  - Expanding the training set, such as adding noise and transforming data
  - Dropout
  - Early stopping

HUAWEI

Penalty Parameters

- Many regularization methods restrict the learning capability of models by adding a penalty parameter $\Omega(\theta)$ to the objective function J. Assume that the target function after regularization is $\tilde{J}$.

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

- Where $\alpha\epsilon[0, \infty)$ is a hyperparameter that weights the relative contribution of the norm penalty term Ω and the standard objective function $J(X; \theta)$. If α is set to 0, no regularization is performed. The penalty in regularization increases with α.

HUAWEI

- Generally, in deep learning, we add constraints only to affine parameter w but not to biases. The main reason is that biases usually require less data for more accurate fitting. Adding constraints often leads to underfitting.

L_1 Regularization

- Add L_1 norm constraint to model parameters, that is,

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

- If a gradient method is used to resolve the value, the parameter gradient is

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

HUAWEI

- In special cases, for secondary optimization and assuming that the corresponding Hessian matrix is a diagonal matrix, the parameter recursive formula is $w_i = sign(w_i^*)\max(|w_i^*| - \frac{\alpha}{\lambda_i}, 0)$. According to this formula, the parameter value is reduced to 0 if $|w_i^*| < \frac{\alpha}{\lambda_i}$. This implementation is different from L_2 regularization. Compared with L_2 regularization, L_2 regularization does not directly reduce the parameter value to 0 but a value close to 0.

L_2 Regularization

- Add norm penalty term L_2 to prevent overfitting.

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

- A parameter optimization method can be inferred using an optimization technology (such as a gradient method):

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

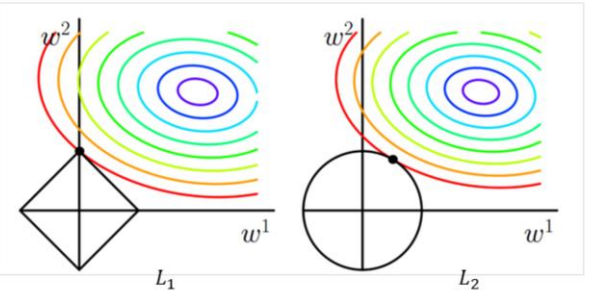
- where ε is the learning rate. Compared with a common gradient optimization formula, this formula multiplies the parameter by a reduction factor.

HUAWEI

- If J is a secondary optimization formula, the model parameter may be further represented as $\widetilde{w}_i = \frac{\lambda_i}{\lambda_i + \alpha} w_i$, that is, adding a control factor to the original parameter, where λ is an eigenvalue of the parameter Hessian matrix. Therefore:
  - When $\lambda_i \gg \propto$, the penalty factor has a small effect.
  - When $\lambda_i \ll \propto$, the corresponding parameter value is reduced to 0.

L_1 v.s. L_2

- The major differences between L_2 and L_1:
  - According to the preceding analysis, L_1 can generate a more sparse model than L_2. When the value of parameter w is small, L_1 regularization can directly reduce the parameter value to 0, which can be used for feature selection.
  - From the perspective of probability, many norm constraints are equivalent to adding prior probability distribution to parameters. In L_2 regularization, the parameter value complies with the Gaussian distribution rule. In L_1 regularization, the parameter value complies with the Laplace distribution rule.



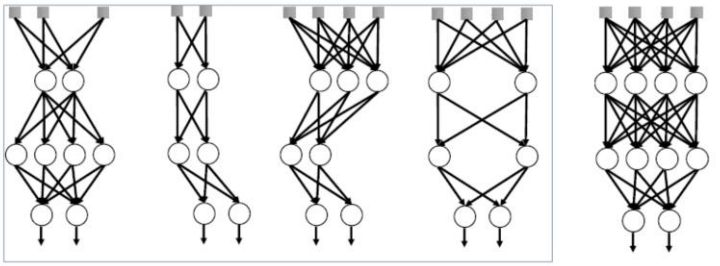
Dataset Expansion

- The most effective way to prevent over-fitting is to add a training set. A larger training set has a smaller over-fitting probability. Dataset expansion is a time-saving method, but it varies in different fields.
  - A common method in the object recognition field is to rotate or scale images. (The prerequisite to image transformation is that the type of the image cannot be changed through transformation. For example, for handwriting digit recognition, categories 6 and 9 can be easily changed after rotation).
  - Random noise is added to the input data in speech recognition.
  - A common practice of natural language processing (NLP) is replacing words with their synonyms.
  - Noise injection can add noise to the input or to the hidden layer or output layer. For example, for Softmax classification, noise can be added using the label smoothing technology. If noise is added to categories 0 and 1, the corresponding probabilities are changed to $\frac{\varepsilon}{k}$ and $1-\frac{k-1}{k}\varepsilon$ respectively.

HUAWEI

Dropout

- Dropout is a common and simple regularization method, which has been widely used since 2014. Simply put, Dropout randomly discards some inputs during the training process. In this case, the parameters corresponding to the discarded inputs are not updated. As an integration method, Dropout combines all sub-network results and obtains sub-networks by randomly dropping inputs. See the figures below:



Dropout in training

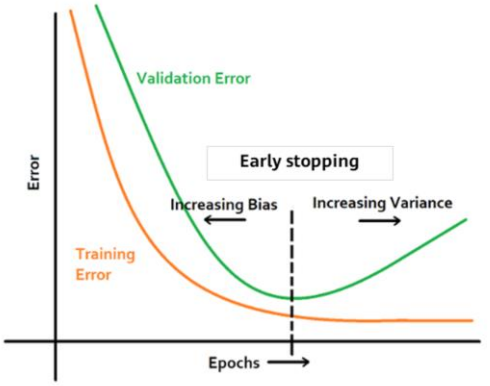
Testing

HUAWEI

- The sampling probability of each entry is 0.8 for the input and 0.5 for the hidden layers.
- Advantages:
  - Compared with weight decay and norm constraints, this strategy is more effective.
  - It is computationally cheap and simple and can be used in other non-deep-learning models.
  - However, it is less effective when the training data is insufficient.
  - Stochasticity is not necessary or sufficient to achieve the regularizing effect of dropout. Invariant shielding parameters can be constructed to obtain good solutions.
- In addition to the preceding methods, we can also use semi-supervised learning, multi-task learning, early stopping, parameter sharing, ensemble methods, and adversarial training.

Early Stopping

- A test on data of the validation set can be inserted during the training. When the data loss of the verification set increases, perform early stopping.



HUAWEI

Contents

HUAWEI

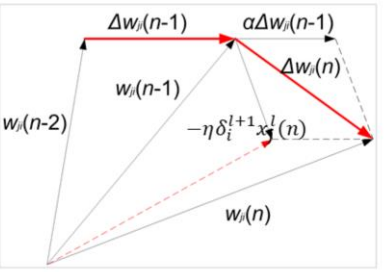
Optimizer

- There are various optimized versions of gradient descent algorithms. In object-oriented language implementation, different gradient descent algorithms are often encapsulated into objects called optimizers.
- Purposes of the algorithm optimization include but are not limited to:
  - Accelerating algorithm convergence.
  - Preventing or jumping out of local extreme values.
  - Simplifying manual parameter setting, especially the learning rate (LR).
- Common optimizers: common GD optimizer, momentum optimizer, Nesterov, AdaGrad, AdaDelta, RMSProp, Adam, AdaMax, and Nadam.

HUAWEI

Momentum Optimizer

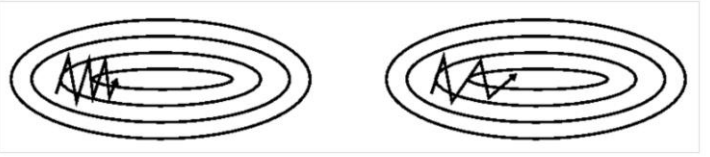
- A most basic improvement is to add momentum terms for Δw_{ji}. Assume that the weight correction of the n-th iteration is $\Delta w_{ji}(n)$. The weight correction rule is:
- $\Delta w_{ji}^{l}(n) = -\eta \delta_i^{l+1} x_j^{l}(n) + \alpha \Delta w_{ji}^{l}(n-1)$
- where α is a constant ($0 \le \alpha < 1$) called Momentum Coefficient and $\alpha \Delta w_{ji}(n-1)$ is a momentum term.
- Imagine a small ball rolls down from a random point on the error surface. The introduction of the momentum term is equivalent to giving the small ball inertia.



HUAWEI

Advantages and Disadvantages of Momentum Optimizer

- Advantages:
  - Enhances the stability of the gradient correction direction and reduces mutations.
  - In areas where the gradient direction is stable, the ball rolls faster and faster (there is a speed upper limit because $\alpha < 1$), which helps the ball quickly overshoot the flat area and accelerates convergence.
  - A small ball with inertia is more likely to roll over some narrow local extrema.
- Disadvantages:
  - The learning rate η and momentum α need to be manually set, which often requires more experiments to determine the appropriate value.



HUAWEI

AdaGrad Optimizer (1)

- The common feature of the random gradient descent algorithm (SGD), small-batch gradient descent algorithm (MBGD), and momentum optimizer is that each parameter is updated with the same LR.
- According to the approach of AdaGrad, different learning rates need to be set for different parameters.

$$g_t = \frac{\partial C(t,o)}{\partial w_t}$$ Gradient calculation

$$r_t = r_{t-1} + g_t^2$$ Square gradient accumulation

$$\Delta w_t = -\frac{\eta}{\varepsilon + \sqrt{r_t}} g_t$$ Computing update

$$w_{t+1} = w_t + \Delta w_t$$ Application update

- g_t indicates the t-th gradient, r is a gradient accumulation variable, and the initial value of r is 0, which **increases continuously**. η indicates the global LR, which needs to be set manually. ε is a small constant, and is set to about 10^{-7} for numerical stability.

HUAWEI

AdaGrad Optimizer (2)

- The AdaGrad optimization algorithm shows that the r continues increasing while the overall learning rate keeps decreasing as the algorithm iterates. This is because we hope LR to decrease as the number of updates increases. In the initial learning phase, we are far away from the optimal solution to the loss function. As the number of updates increases, we are closer to the optimal solution, and therefore LR can decrease.
- Pros:
  - The learning rate is automatically updated. As the number of updates increases, the learning rate decreases.
- Cons:
  - The denominator keeps accumulating so that the learning rate will eventually become very small, and the algorithm will become ineffective.

HUAWEI

RMSProp Optimizer

- The RMSProp optimizer is an improved AdaGrad optimizer. It introduces an attenuation coefficient to ensure a certain attenuation ratio for r in each round.
- The RMSProp optimizer solves the problem that the AdaGrad optimizer ends the optimization process too early. It is suitable for non-stable target handling and has good effects on the RNN.

$$g_t = \frac{\partial C(t,o)}{\partial w_t}$$ Gradient calculation

$$r_t = \beta r_{t-1} + (1-\beta) g_t^2$$ Square gradient accumulation

$$\Delta w_t = -\frac{\eta}{\varepsilon + \sqrt{r_t}} g_t$$ Computing update

$$w_{t+1} = w_t + \Delta w_t$$ Application update

- g_t indicates the t-th gradient, r is a gradient accumulation variable, and the initial value of r is 0, which **may not increase and needs to be adjusted using a parameter**. β is the attenuation factor, η indicates the global LR, which needs to be set manually. ε is a small constant, and is set to about 10^{-7} for numerical stability.

HUAWEI

- m indicates the number of samples used.

Adam Optimizer (1)

- Adaptive Moment Estimation (Adam): Developed based on AdaGrad and AdaDelta, Adam maintains two additional variables m_t and v_t for each variable to be trained:

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

- Where t represents the t-th iteration and g_t is the calculated gradient. m_t and v_t are moving averages of the gradient and square gradient. From the statistical perspective, m_t and v_t are estimates of the first moment (the average value) and the second moment (the uncentered variance) of the gradients respectively, which also explains why the method is so named.

HUAWEI

- Adam attempts to calculate adaptive learning rates for each parameter. This is very useful in complex network structures because different parts of the network have different sensitivity to weight adjustment. A very sensitive part usually requires a smaller learning rate. It is difficult or complex to manually identify the sensitive part and set a learning rate. It may be the best optimizer at present.

Adam Optimizer (2)

- If m_t and v_t are initialized using the zero vector, m_t and v_t are close to 0 during the initial iterations, especially when β_1 and β_2 are close to 1. To solve this problem, we use $\hat{m}_t$ and $\hat{v}_t$:

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

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

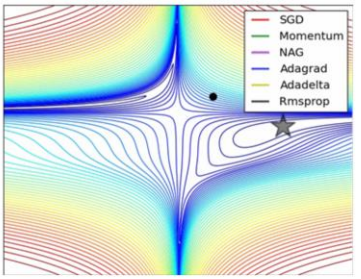
- The weight update rule of Adam is as follows:

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

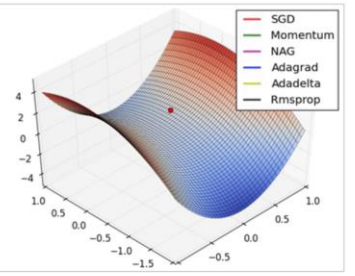
- Although the rule involves manual setting of η, β_1, and β_2, the setting is much simpler. According to experiments, the default settings are $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, and $\eta = 0.001$. In practice, Adam will converge quickly. When convergence saturation is reached, xx can be reduced. After several times of reduction, a satisfying local extremum will be obtained. Other parameters do not need to be adjusted.

HUAWEI

Optimizer Performance Comparison



Comparison of optimization algorithms in contour maps of loss functions



Comparison of optimization algorithms at the saddle point

HUAWEI

Contents

HUAWEI

Convolutional Neural Network

- A convolutional neural network (CNN) is a feedforward neural network. Its artificial neurons may respond to **surrounding units within the coverage range**. CNN excels at image processing. It includes a **convolutional layer**, a **pooling layer**, and a **fully connected layer**.
- In the 1960s, Hubel and Wiesel studied cats' cortex neurons used for local sensitivity and direction selection and found that their unique network structure could simplify feedback neural networks. They then proposed the CNN.
- Now, CNN has become one of the research hotspots in many scientific fields, especially in the pattern classification field. The network is widely used because it can avoid complex pre-processing of images and directly input original images.

HUAWEI

- A filter matrix is a set of fixed weight and can be seen as a constant filter (kernel). The convolution (adding each element, weighted by the kernel) is performed between an image (data from different data windows) and a kernel. This type of network is called CNN.
- Local receptive field: It is generally considered that human perception of the outside world is from local to global. Spatial correlations among local pixels of an image are closer than those among pixels that are far away. Therefore, each neuron does not need to know the global image. It only needs to know the local image and then the local information is combined at a higher level to generate global information. The idea of local network connection is also inspired by the biological visual system structure. The neurons in the visual cortex receive local information (respond to stimuli of certain regions).
- Parameter sharing: One or more filters can be used to scan the input images. The parameters of the filter are weights. At the layers scanned by the same filter, each filter uses the same parameters to perform weighted calculation. Weight sharing means that the parameter values of each filter does not change when the filter scans the entire image. For example, if we have three feature filters and each filter scans the entire image. During the scanning process, the parameter values of the filters do not change. In other words, all elements of the image share the same weights.

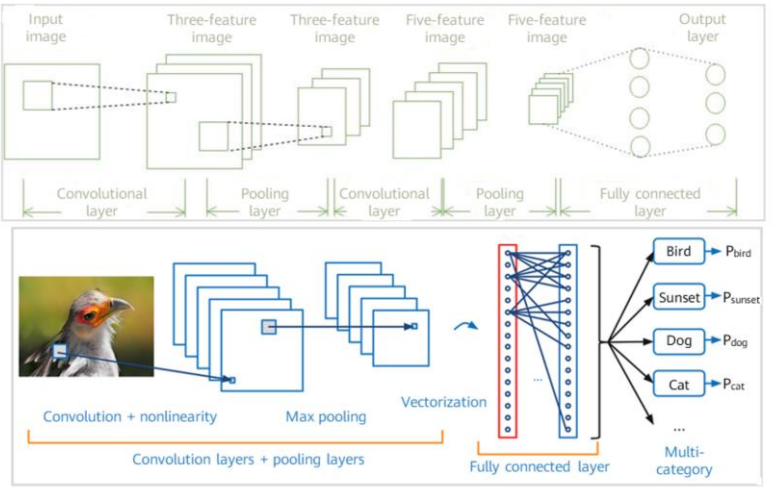
Main Concepts of CNN

- **Local receptive field:** It is generally considered that human perception of the outside world is from local to global. **Spatial correlations among local pixels of an image are closer than those among distant pixels**. Therefore, each neuron does not need to know the global image. It only needs to know the local image. The local information is combined at a higher level to generate global information.
- **Parameter sharing:** One or more filters/kernels may be used to scan input images. Parameters carried by the filters are weights. In a layer scanned by filters, each filter uses the same parameters during weighted computation. Weight sharing means that when each filter scans an entire image, parameters of the filter are fixed.

HUAWEI

- Local receptive field: The idea of local network connection is also inspired by the biological visual system structure. The neurons in the visual cortex receive local information (respond to stimuli of certain regions).
- Parameter sharing: For example, if we have three feature convolution kernels and each kernel scans the entire image, the parameter values of the convolution kernels do not change during the scanning process. In other words, all elements of the image share the same weights. This means that the features learned from a part of the image can also be applied to other parts of the image or other images, which is called position invariance.

Architecture of Convolutional Neural Network



HUAWEI

- Input layer: inputs data.
- Convolutional layer: composed of several convolutional units. The parameters of each convolutional unit are obtained by optimizing the backpropagation algorithm. The purpose of convolution calculation is to extract different input features. The first convolutional layer may extract only some low-level features such as edges, lines, and angles. A multi-layer network can extract more complex features based on the low-level features.
- Rectified linear units layer (ReLU layer): uses ReLU f(x) = max(0, x) as the activation function.
- Pooling layer: partitions features obtained from the convolutional layer into some areas and outputs the maximum or minimum value, generating new features with a smaller spatial size.
- Fully connected layer: integrates all local features into global features to calculate the final scores for each type.
- Output layer: outputs the final result.

Single-Filter Calculation (1)

- Description of convolution calculation

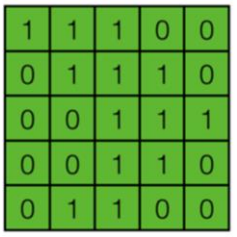
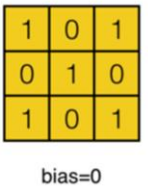
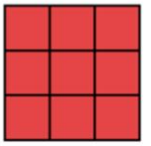


image 5*5



bias=0

filter 3*3

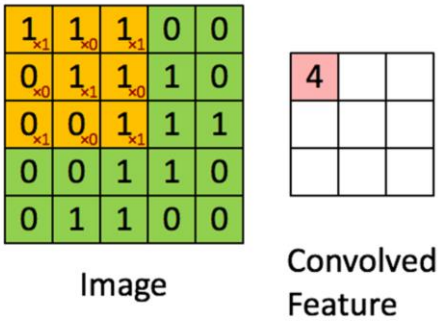


feature map 3*3

HUAWEI

Single-Filter Calculation (2)

- Demonstration of the convolution calculation

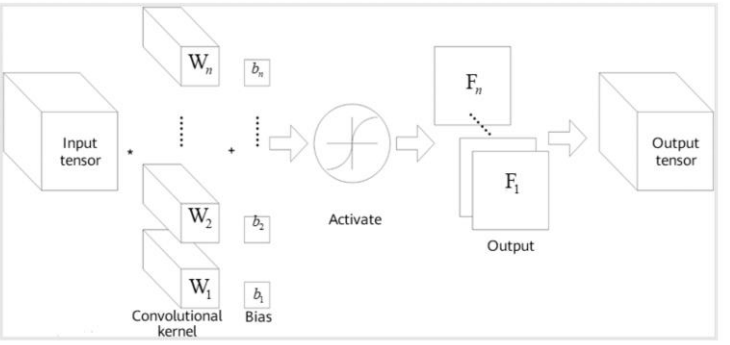


Han Bingtao, 2017, Convolutional Neural Network

HUAWEI

Convolutional Layer

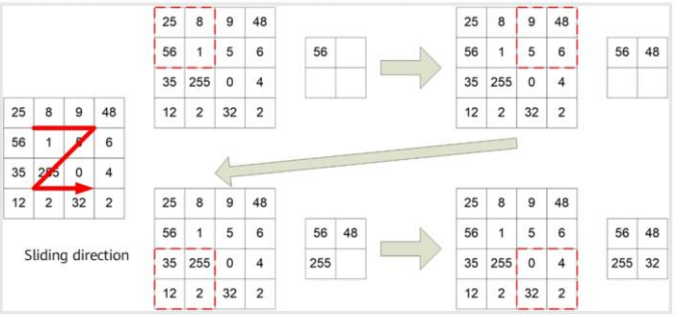
- The basic architecture of a CNN is multi-channel convolution consisting of multiple single convolutions. The output of the previous layer (or the original image of the first layer) is used as the input of the current layer. It is then convolved with the filter in the layer and serves as the output of this layer. The convolution kernel of each layer is the weight to be learned. Similar to FCN, after the convolution is complete, the result should be biased and activated through activation functions before being input to the next layer.



HUAWEI

Pooling Layer

- Pooling combines nearby units to reduce the size of the input on the next layer, reducing dimensions. Common pooling includes max pooling and average pooling. When max pooling is used, the maximum value in a small square area is selected as the representative of this area, while the mean value is selected as the representative when average pooling is used. The side of this small area is the pool window size. The following figure shows the max pooling operation whose pooling window size is 2.



HUAWEI

- The actual classification networks are feedforward networks that are formed by interconnected convolutional and pooling layers. The pooling layer has the following functions:
  - Invariance: Max pooling ensures invariance within a certain range, because the maximum value of a region is the last output value regardless of where the value is.
  - Reducing the input size for the next layer: Pooling effectively reduces the size of the input data for the next layer, the number of parameters, and computation workload.
  - Obtaining fixed-length data: By properly setting the pooling window size and stride, we can obtain fixed-length outputs from variable-length inputs.
  - Increasing the scale: The features of the previous layer can be extracted from a larger scale.
  - Preventing overfitting: Pooling simplifies the network and reduces the fitting precision. Therefore, it can prevent overfitting (pay attention to the possible underfitting).

Fully Connected Layer

- The fully connected layer is essentially a classifier. The features extracted on the convolutional layer and pooling layer are straightened and placed at the fully connected layer to output and classify results.
- Generally, the Softmax function is used as the activation function of the final fully connected output layer to combine all local features into global features and calculate the score of each type.

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

HUAWEI

Recurrent Neural Network

- The recurrent neural network (RNN) is a neural network that captures dynamic information in sequential data through periodical connections of hidden layer nodes. It can classify sequential data.
- Unlike other forward neural networks, the RNN can keep a context state and even store, learn, and express related information in context windows of any length. Different from traditional neural networks, it is not limited to the space boundary, but also supports time sequences. In other words, there is a side between the hidden layer of the current moment and the hidden layer of the next moment.
- The RNN is widely used in scenarios related to sequences, such as videos consisting of image frames, audio consisting of clips, and sentences consisting of words.

HUAWEI

Recurrent Neural Network Architecture (1)

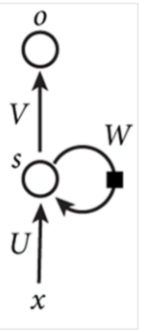
- X_t is the input of the input sequence at time t.
- S_t is the memory unit of the sequence at time t and caches previous information.

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

- O_t is the output of the hidden layer of the sequence at time t.

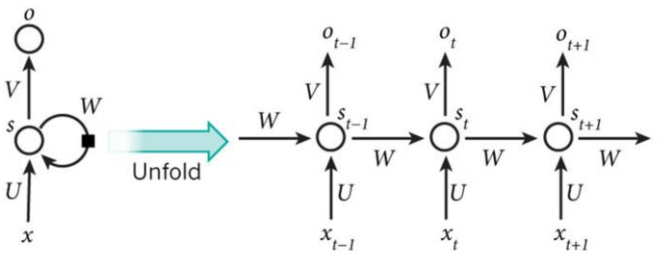
  $$O_t = tanh(VS_t)$$

- O_t after through multiple hidden layers, it can get the final output of the sequence at time t.



HUAWEI

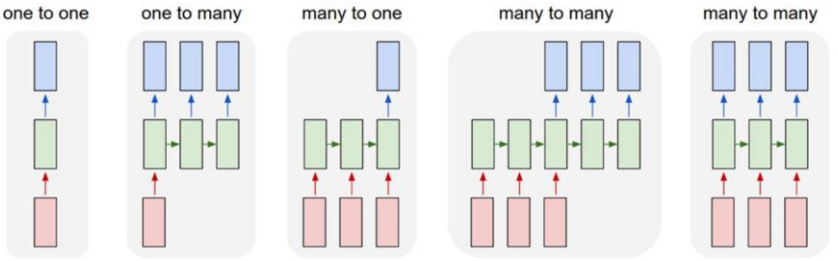
Recurrent Neural Network Architecture (2)



LeCun, Bengio, and G. Hinton, 2015, A Recurrent Neural Network and the Unfolding in Time of the Computation Involved in Its Forward Computation

HUAWEI

Types of Recurrent Neural Networks



Andrej Karpathy, 2015, The Unreasonable Effectiveness of Recurrent Neural Networks

HUAWEI

Backpropagation Through Time (BPTT)

- BPTT:
  - Traditional backpropagation is the extension on the time sequence.
  - There are two sources of errors in the sequence at time of memory unit: first is from the hidden layer output error at t time sequence; the second is the error from the memory cell at the next time sequence t + 1.
  - The longer the time sequence, the more likely the loss of the last time sequence to the gradient of w in the first time sequence causes the vanishing gradient or exploding gradient problem.
  - The total gradient of weight w is the accumulation of the gradient of the weight at all time sequence.
- Three steps of BPTT:
  - Computing the output value of each neuron through forward propagation.
  - Computing the error value of each neuron through backpropagation δ_j.
  - Computing the gradient of each weight.
- Updating weights using the SGD algorithm.

HUAWEI

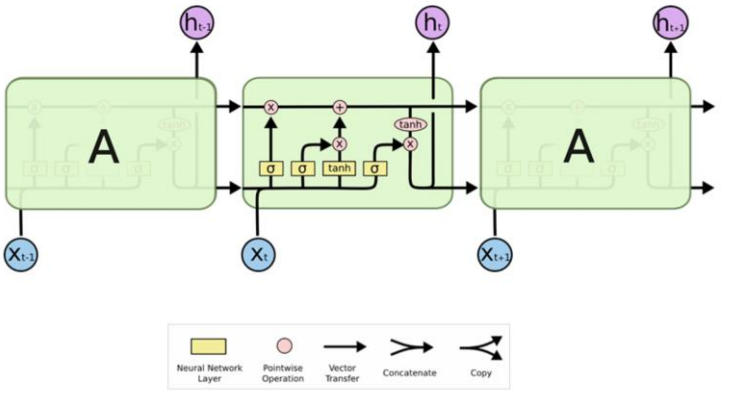
- z_t is the value before the memory cell enters the activation function, $\delta_t = \frac{\partial C_t}{\partial z_t} + w^T \times f_t'(z_t) \times \delta_{t+1}$

Recurrent Neural Network Problem

- $S_t = \sigma(UX_t + WS_{t-1})$ is extended on the time sequence.
- $S_t = \sigma\left(UX_t + W\left(\sigma\left(UX_{t-1} + W\left(\sigma\left(UX_{t-2} + W(\ldots)\right)\right)\right)\right)\right)$
- Despite that the standard RNN structure solves the problem of information memory, the information attenuates during long-term memory.
- Information needs to be saved long time in many tasks. For example, a hint at the beginning of a speculative fiction may not be answered until the end.
- The RNN may not be able to save information for long due to the limited memory unit capacity.
- We expect that memory units can remember key information.

HUAWEI

Long Short-term Memory Network

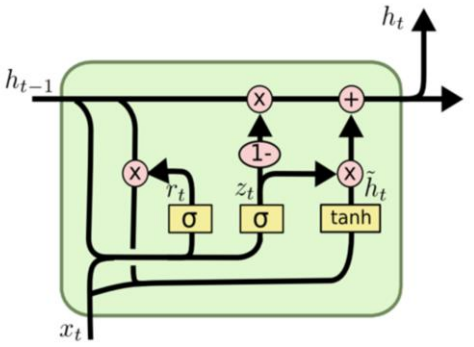


Colah, 2015, Understanding LSTMs Networks

HUAWEI

- Long short-term memory (LSTM) applies to the scenario with a large gap between the relevant information and the point where it is needed. It can connect previous information for long periods of time to the present task.

Gated Recurrent Unit (GRU)



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

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

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

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

HUAWEI

- As a variant of LSTM, GRU combines the Forget Gate and the Input Gate into a single Update Gate. It mixes the cell state and hidden state and also incorporates other changes. As a popular variant, its final model is simpler than the standard LSTM model. [] indicates concatenation, * indicates element product, and· indicates matrix multiplication.

Generative Adversarial Network (GAN)

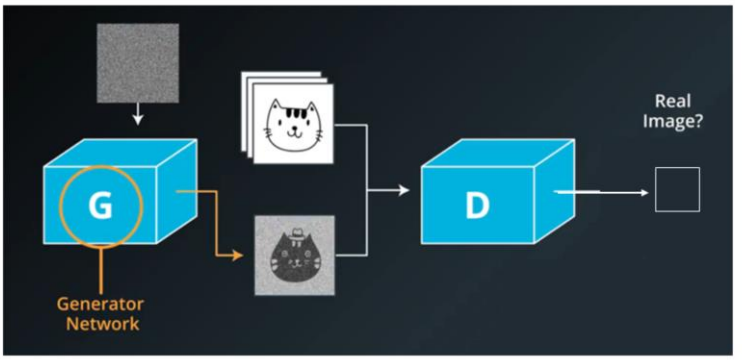
- Generative Adversarial Network is a framework that trains generator G and discriminator D through the adversarial process. Through the adversarial process, the discriminator can tell whether the sample from the generator is fake or real. GAN adopts a mature BP algorithm.
- (1) Generator G: The input is noise z, which complies with manually selected prior probability distribution, such as even distribution and Gaussian distribution. The generator adopts the network structure of the multilayer perceptron (MLP), uses maximum likelihood estimation (MLE) parameters to represent the derivable mapping G(z), and maps the input space to the sample space.
- (2) Discriminator D: The input is the real sample x and the fake sample G(z), which are tagged as real and fake respectively. The network of the discriminator can use the MLP carrying parameters. The output is the probability D(G(z)) that determines whether the sample is a real or fake sample.
- GAN can be applied to scenarios such as image generation, text generation, speech enhancement, image super-resolution.

HUAWEI

- GAN: Traditional deep generative models require Markov chains or maximum likelihood estimation (MLE), which brings a lot of probability issues difficult to compute.

GAN Architecture

- Generator/Discriminator



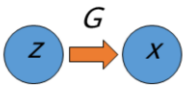
HUAWEI

- There are two kinds of models in a GAN: generative model and discriminative model. The discriminative model is used to predict the classification based on input variables. The generative model is used to randomly generate observable-data values, typically using some given hidden parameters. Example:
- Generative model
  - Generates a new cat image that does not exist in the dataset with random data given.
- Discriminative model
  - Discriminates the animal (whether it is a cat or not) in a given image.

Generative Model and Discriminative Model

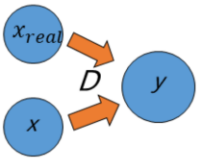
- Generative network
  - Generates sample data
    - Input: Gaussian white noise vector z
    - Output: sample data vector x

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



- Discriminator network
  - Determines whether sample data is real
    - Input: real sample data x_{real} and generated sample data $x = G(z)$
    - Output: probability that determines whether the sample is real

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



HUAWEI

Training Rules of GAN

- Optimization objective:
  - Value function

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

  - In the early training stage, when the outcome of G is very poor, D determines that the generated sample is fake with high confidence, because the sample is obviously different from training data. In this case, log(1-D(G(z))) is saturated (where the gradient is 0, and iteration cannot be performed). Therefore, we choose to train G only by minimizing [-log(D(G(z))].

HUAWEI

- When the value function is used during training, the GAN is called minimax GAN. When [-log(D(G(z))] is used during training, the GAN is called non-saturating GAN.

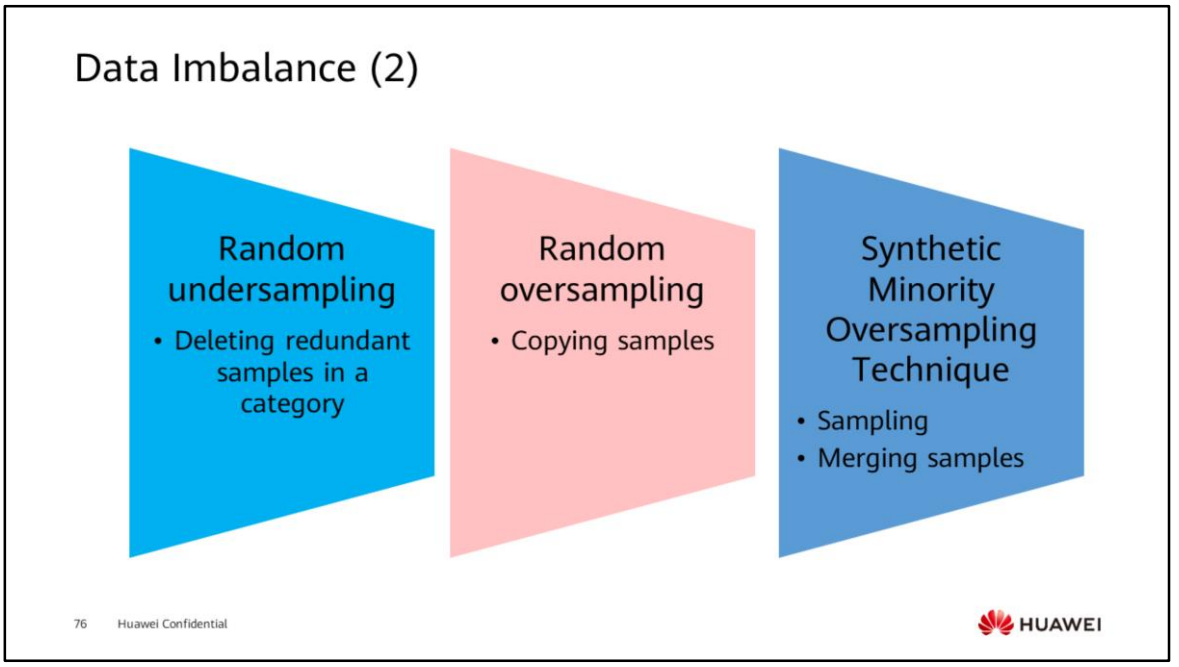
Contents

HUAWEI

Data Imbalance (1)

- Problem description: In the dataset consisting of various task categories, the number of samples varies greatly from one category to another. One or more categories in the predicted categories contain very few samples.
- For example, in an image recognition experiment, more than 2,000 categories among a total of 4251 training images contain just one image each. Some of the others have 2-5 images.
- Impacts:
  - Due to the unbalanced number of samples, we cannot get the optimal real-time result because model/algorithm never examines categories with very few samples adequately.
  - Since few observation objects may not be representative for a class, we may fail to obtain adequate samples for verification and test.

HUAWEI



- Random undersampling: Classes with sufficient observations are randomly deleted to enable significant relative ratio between the two classes. Although this method is easy to use, the deleted data packet may contain important information about the prediction class.
    - Advantage
        - It increases the running time. In addition, when the training dataset is large, a storage problem may be resolved by reducing samples.
    - Disadvantage
        - It discards potentially valuable information that is important for building a rule classifier.
        - Samples selected by random undersampling may have a bias. It does not accurately represent the majority. Therefore, the result for an actual test dataset is not accurate.
- Random oversampling: For unbalanced classes, we randomly increase the number of observations by copying existing samples. Ideally, we have sufficient samples, but oversampling may lead to overfitting training data.
    - Advantage
        - It does not cause information loss.
        - It outperforms undersampling.
    - Disadvantage
        - It increases the likelihood of overfitting because of the replication of

minority class event.

Vanishing Gradient and Exploding Gradient Problem (1)

- Vanishing gradient: As network layers increase, the derivative value of backpropagation decreases, which causes a vanishing gradient problem.
- Exploding gradient: As network layers increase, the derivative value of backpropagation increases, which causes an exploding gradient problem.
- Cause: $y_i = \sigma_{(z_i)} = \sigma(w_i x_i + b_i)$ Where σ is sigmoid function.

- Backpropagation can be deduced as follows:

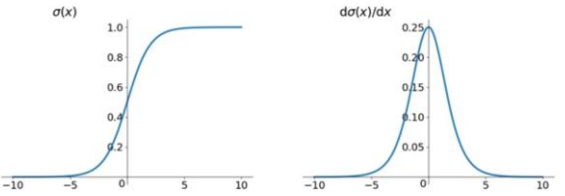
$$\frac{\partial C}{\partial b_1} = \frac{\partial C}{\partial y_4}\frac{\partial y_4}{\partial z_4}\frac{\partial z_4}{\partial x_4}\frac{\partial x_4}{\partial z_3}\frac{\partial z_3}{\partial x_3}\frac{\partial x_3}{\partial z_2}\frac{\partial z_2}{\partial x_2}\frac{\partial x_2}{\partial z_1}\frac{\partial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

HUAWEI

- (It is assumed that each layer has only one neuron and for each layer, the activation function is a sigmoid function.)

Vanishing Gradient and Exploding Gradient Problem (2)

- The maximum value of $\sigma'(x)$ is $\frac{1}{4}$:



- However, the network weight $|w|$ is usually smaller than 1. Therefore, $|\sigma'(z)w| \leq \frac{1}{4}$. According to the chain rule, as layers increase, the derivation result $\frac{\partial C}{\partial b_1}$ decreases, resulting in the vanishing gradient problem.
- When the network weight $|w|$ is large, resulting in $|\sigma'(z)w| > 1$, the exploding gradient problem occurs.
- Solution: For example, gradient clipping is used to alleviate the exploding gradient problem, ReLU activation function and LSTM are used to alleviate the vanishing gradient problem.

HUAWEI

- This problem occurs when the value of w is large. However, this problem seldom occurs when the sigmoid activation function is used because $\sigma'(z)$ is a
- Gradient clipping is proposed for the exploding gradient problem. A gradient clipping threshold is set. When being updated, if the gradient exceeds the threshold, the gradient is forcibly limited within the range. This prevents the exploding gradient problem.
- Another approach for resolving the exploding gradient problem is weight regularization, among which l1 regularization and l2 regularization are commonly used. APIs in each deep learning framework may use regularization. For example, in TensorFlow, if regularization parameters have been set during network setup, you can directly calculate the regularization loss by calling the following code.
- ReLU: Its idea is simple. If the derivative of the activation function is 1, the vanishing gradient or exploding gradient problem will not occur. That's why the ReLU is developed.
- For LSTM, the vanishing gradient problem does not easily occur thanks to the complex gates inside the LSTM. As shown in the following figure, LSTM can "remember" the "residual memory" of the previous training based on the internal gates when it is being updated. Therefore, it is often used in generative text. Currently, there is also CNN-based LSTM. Try it if you are interested.

Overfitting

- Problem description: The model performs well in the training set, but badly in the test set.
- Root cause: There are too many feature dimensions, model assumptions, and parameters, too much noise, but very few training data. As a result, the fitting function perfectly predicts the training set, while the prediction result of the test set of new data is poor. Training data is over-fitted without considering generalization capabilities.
- Solution: For example, data augmentation, regularization, early stopping, and dropout

HUAWEI

- Overfitting causes: insufficient data and complex model
- More data: Obtain more data from the data source to achieve data augmentation.
  Proper model: Reduce the number of network layers and neurons can limit the fitting capability of the network.
  Dropout;
  Regularization: The weight increase is restricted during training.
- Limit the training time. Pass the evaluation test.
- Increase the noise: Input + Weight (Gaussian initialization)
- Data data cleansing/pruning: Rectify or delete incorrect labels.
- Multiple models: Bagging uses diverse models to fit training sets of different parts. Boosting uses only simple neural networks.
- L1 and L2 regularization: Penalty terms are introduced to solve the problem of excessive parameters in the model based on the Occam's Razor principle. In deep learning, L1 tends to generate a small number of features, and other features are all 0 to increase network sparsity. L2 selects more features, which are close to 0 to prevent overfitting. The neural network requires neurons at each layer to extract meaningful features as much as possible. Therefore, L2 regularization is applied more widely.

Summary

- This chapter describes the definition and development of neural networks, perceptrons and their training rules, common types of neural networks (CNN, RNN, and GAN), and the Common Problems of neural networks in AI engineering and solutions.

HUAWEI

Quiz

1. (True or false) Compared with the recurrent neural network, the convolutional neural network is more suitable for image recognition. ( )
   A. True
   B. False
2. (True or false) GAN is a deep learning model, which is one of the most promising methods for unsupervised learning of complex distribution in recent years. ( )
   A. True
   B. False

HUAWEI

- Answers: 3. A 4. A

Quiz

3. (Single-choice) There are many types of deep learning neural networks. Which of the following is not a deep learning neural network? ( )
   A. CNN
   B. RNN
   C. LSTM
   D. Logistic
4. (Multi-choice) There are many important "components" in the convolutional neural network architecture. Which of the following are the convolutional neural network "components"? ( )
   A. Activation function
   B. Convolutional kernel
   C. Pooling
   D. Fully connected layer

HUAWEI

- Answers: 1. D 2. ABCD

Recommendations

- Online learning website
  - https://e.huawei.com/cn/talent/#/home
- Huawei Knowledge Base
  - https://support.huawei.com/enterprise/servicecenter?lang=zh

HUAWEI

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.

Copyright©2020 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.

Mainstream Development Frameworks in the Industry

HUAWEI

Foreword

- This chapter describes:
  - Definition of deep learning framework and its advantages, and two mainstream deep learning frameworks PyTorch and TensorFlow
  - Basic operations and common modules of TensorFlow 2.x (by focusing on code)
  - MNIST handwritten digit recognition experiment performed based on TensorFlow for deeply understanding and getting familiar with a deep learning modeling process

HUAWEI

Objectives

On completion of this course, you will be able to:

- Describe a deep learning framework.
- Know mainstream deep learning frameworks.
- Know the features of PyTorch.
- Know the features of TensorFlow.
- Differentiate between TensorFlow 1.x and 2.x.
- Master the basic syntax and common modules of TensorFlow 2.x.
- Master the process of an MNIST handwritten digit recognition experiment.

HUAWEI

Contents

HUAWEI

Deep Learning Framework

- A deep learning framework is an interface, library or a tool which allows us to build deep learning models more easily and quickly, without getting into the details of underlying algorithms. A deep learning framework can be regarded as a set of building blocks. Each component in the building blocks is a model or algorithm. Therefore, developers can use components to assemble models that meet requirements, and do not need to start from scratch.
- The emergence of deep learning frameworks lowers the requirements for developers. Developers no longer need to compile code starting from complex neural networks and back-propagation algorithms. Instead, they can use existing models to configure parameters as required, where the model parameters are automatically trained. Moreover, they can add self-defined network layers to the existing models, or select required classifiers and optimization algorithms directly by invoking existing code.

HUAWEI

Contents

HUAWEI

PyTorch

- PyTorch is a Python-based machine learning computing framework developed by Facebook. It is developed based on Torch, a scientific computing framework supported by a large number of machine learning algorithms. Torch is a tensor operation library similar to NumPy, featured by high flexibility, but is less popular because it uses the programming language Lua. This is why PyTorch is developed.
- In addition to Facebook, institutes such as Twitter, GMU also use PyTorch.

Image source: http://PyTorch123.com/FirstSection/PyTorchIntro/

HUAWEI

Features of PyTorch

- **Python first**: PyTorch does not simply bind Python to a C++ framework. PyTorch directly supports Python access at a fine grain. Developers can use PyTorch as easily as using NumPy or SciPy. This not only lowers the threshold for understanding Python, but also ensures that the code is basically consistent with the native Python implementation.
- **Dynamic neural network**: Many mainstream frameworks such as TensorFlow 1.x do not support this feature. To run TensorFlow 1.x, developers must create static computational graphs in advance, and run the **feed** and **run** commands to repeatedly execute the created graphs. In contrast, PyTorch with this feature is free from such complexity, and PyTorch programs can dynamically build/adjust computational graphs during execution.
- **Easy to debug**: PyTorch can generate dynamic graphs during execution. Developers can stop an interpreter in a debugger and view output of a specific node.
- PyTorch provides tensors that support CPUs and GPUs, greatly accelerating computing.

HUAWEI

Contents

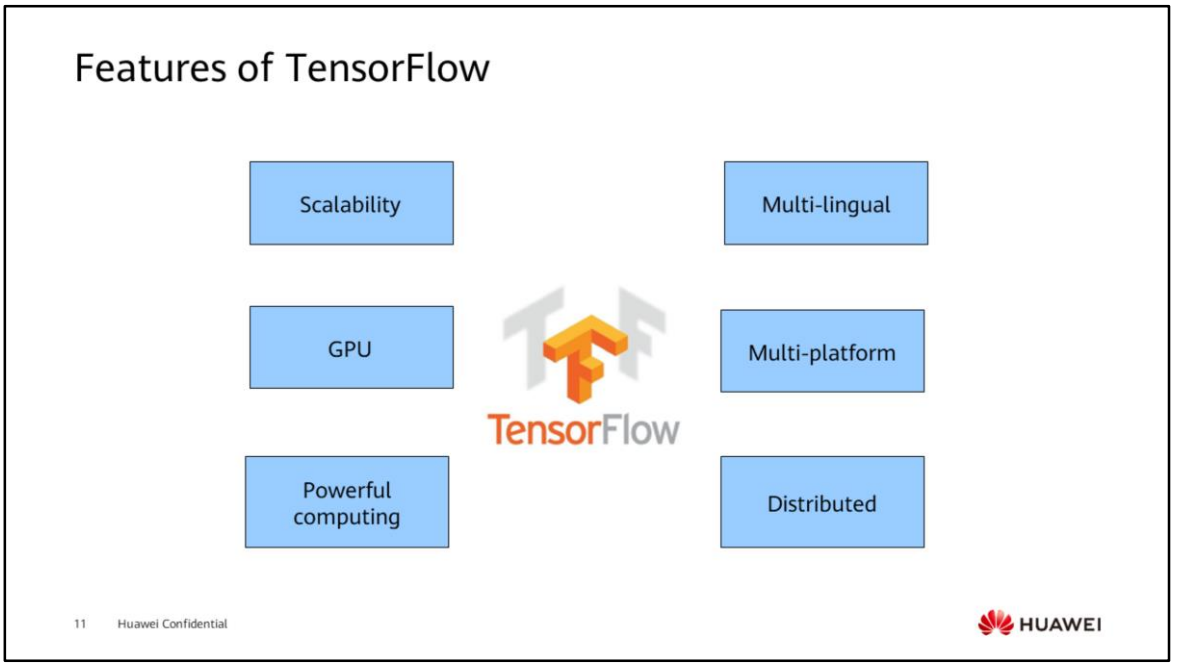
HUAWEI

TensorFlow

- TensorFlow is Google's second-generation open-source software library for digital computing. The TensorFlow computing framework supports various deep learning algorithms and multiple computing platforms, ensuring high system stability.

Image source: https://www.TensorFlow.org/

HUAWEI



- Multi-platform: All platforms that support the Python development environment support TensorFlow. However, TensorFlow depends on other software such as the NVIDIA CUDA Toolkit and cuDNN to access a supported GPU.
- GPU: TensorFlow supports certain NVIDIA GPUs, which are compatible with NVIDIA CUDA Toolkit versions that meet specific performance standards.
- Distributed: TensorFlow supports distributed computing, allowing graphics to be computed on different processes. These processes may be located on different servers.
- Multi-lingual: TensorFlow primarily uses Python for programming. It also supports C++, Java, and Go application programming interfaces (APIs), but cannot guarantee stability when these languages are used. Similarly, TensorFlow cannot ensure stability when third parties use TensorFlow based on C#, Haskell, Julia, Rust, Ruby, Scala, and R (even PHP). Google recently released a mobile-optimized TensorFlow Lite library for running TensorFlow applications on Android.
- Powerful computing: TensorFlow can achieve optimal performance on Google TPUs, but still strives to deliver high performance on various platforms, which include not only servers and desktops, but also embedded systems and mobile devices.
- Scalability: One of the main advantages of TensorFlow lies in its modular, scalable, and flexible design. Developers can easily port models among the CPU, GPU, and TPU with a few code changes. Python developers can develop their own models by using native and low-level APIs (or core APIs) of TensorFlow, or develop built-in models by using advanced API libraries of TensorFlow. TensorFlow has many built-in and distributed libraries. TensorFlow can be overlaid with an advanced deep learning framework (such as Keras) to serve as an advanced API.

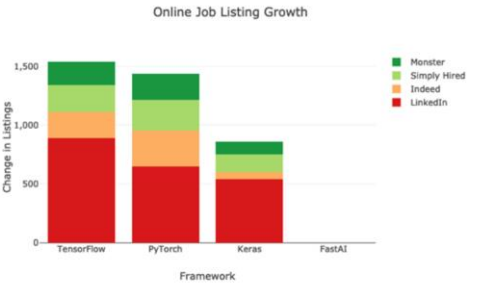
TensorFlow - Distributed

- TensorFlow can run on different computers:
  - From smartphones to computer clusters, to generate desired training models.
- Currently, supported native distributed deep learning frameworks include only TensorFlow, CNTK, Deeplearning4J, and MXNet.
- When a single GPU is used, most deep learning frameworks rely on cuDNN, and therefore support almost the same training speed, provided that the hardware computing capabilities or allocated memories slightly differ. However, for large-scale deep learning, massive data makes it difficult for the single GPU to complete training in a limited time. To handle such cases, TensorFlow enables distributed training.

HUAWEI

Why TensorFlow?

- TensorFlow is considered as one of the best libraries for neural networks, and can reduce difficulty in deep learning development. In addition, as it is open-source, it can be conveniently maintained and updated, thus the efficiency of development can be improved.
- Keras, ranking third in the number of stars on GitHub, is packaged into an advanced API of TensorFlow 2.0, which makes TensorFlow 2.x mor flexible, and easier to debug.



Demand on the recruitment market

HUAWEI

TensorFlow 2.x vs. TensorFlow 1.x

- Disadvantages of TensorFlow 1.0:
  - After a tensor is created in TensorFlow 1.0, the result cannot be returned directly. To obtain the result, the session mechanism needs to be created, which includes the concept of graph, and code cannot run without session.run. This style is more like the hardware programming language VHDL.
  - Compared with some simple frameworks such as PyTorch, TensorFlow 1.0 adds the session and graph concepts, which are inconvenient for users.
  - It is complex to debug TensorFlow 1.0, and its APIs are disordered, making it difficult for beginners. Learners will come across many difficulties in using TensorFlow 1.0 even after gaining the basic knowledge. As a result, many researchers have turned to PyTorch.

HUAWEI

TensorFlow 2.x vs. TensorFlow 1.x

- Features of TensorFlow 2.x:
  - Advanced API Keras:
    - Easy to use: The graph and session mechanisms are removed. What you see is what you get, just like Python and PyTorch.
  - Major improvements:
    - The core function of TensorFlow 2.x is the dynamic graph mechanism called eager execution. It allows users to compile and debug models like normal programs, making TensorFlow easier to learn and use.
    - Multiple platforms and languages are supported, and compatibility between components can be improved via standardization on exchange formats and alignment of APIs.
    - Deprecated APIs are deleted and duplicate APIs are reduced to avoid confusion.
    - Compatibility and continuity: TensorFlow 2.x provides a module enabling compatibility with TensorFlow 1.x.
    - The tf.contrib module is removed. Maintained modules are moved to separate repositories. Unused and unmaintained modules are removed.

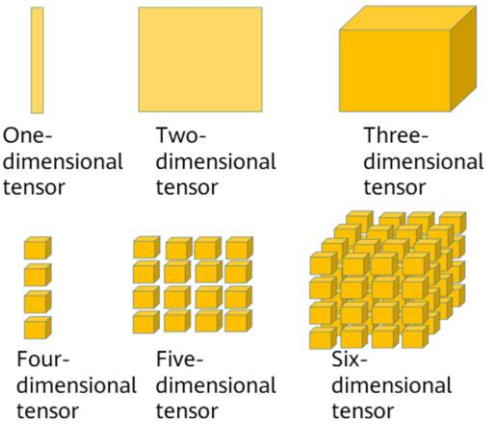
HUAWEI

Contents

HUAWEI

Tensors

- Tensors are the most basic data structures in TensorFlow. All data is encapsulated in tensors.
- Tensor: a multidimensional array
  - A scalar is a rank-0 tensor. A vector is a rank-1 tensor. A matrix is a rank-2 tensor.
- In TensorFlow, tensors are classified into:
  - Constant tensors
  - Variable tensors



HUAWEI

- Jupyter; tensor introduction chapter

Basic Operations of TensorFlow 2.x

- The following describes common APIs in TensorFlow by focusing on code. The main content is as follows:
  - Methods for creating constants and variables
  - Tensor slicing and indexing
  - Dimension changes of tensors
  - Arithmetic operations on tensors
  - Tensor concatenation and splitting
  - Tensor sorting

HUAWEI

- The following describes common APIs in TensorFlow by focusing on code.

Eager Execution Mode of TensorFlow 2.x

- Static graph: TensorFlow 1.x using static graphs (graph mode) separates computation definition and execution by using computational graphs. This is a declarative programming model. In graph mode, developers need to build a computational graph, start a session, and then input data to obtain an execution result.
- Static graphs are advantageous in distributed training, performance optimization, and deployment, but inconvenient for debugging. Executing a static graph is similar to invoking a compiled C language program, and internal debugging cannot be performed in this case. Therefore, eager execution based on dynamic computational graphs emerges.
- Eager execution is a command-based programming method, which is the same as native Python. A result is returned immediately after an operation is performed.

HUAWEI

AutoGraph

- Eager execution is enabled in TensorFlow 2.x by default. Eager execution is intuitive and flexible for users (easier and faster to run a one-time operation), but may compromise performance and deployability.
- To achieve optimal performance and make a model deployable anywhere, you can run @tf.function to add a decorator to build a graph from a program, making Python code more efficient.
- tf.function can build a TensorFlow operation in the function into a graph. In this way, this function can be executed in graph mode. Such practice can be considered as encapsulating the function as a TensorFlow operation of a graph.

HUAWEI

Contents

HUAWEI

Common Modules of TensorFlow 2.x (1)

- tf: Functions in the tf module are used to perform common arithmetic operations, such as tf.abs (calculating an absolute value), tf.add (adding elements one by one), and tf.concat (concatenating tensors). Most operations in this module can be performed by NumPy.
- tf.errors: error type module of TensorFlow
- tf.data: implements operations on datasets.
  - Input pipes created by tf.data are used to read training data. In addition, data can be easily input from memories (such as NumPy).
- tf.distributions: implements various statistical distributions.
  - The functions in this module are used to implement various statistical distributions, such as Bernoulli distribution, uniform distribution, and Gaussian distribution.

HUAWEI

Common Modules of TensorFlow 2.x (2)

- tf.io.gfile: implements operations on files.
  - Functions in this module can be used to perform file I/O operations, copy files, and rename files.
- tf.image: implements operations on images.
  - Functions in this module include image processing functions. This module is similar to OpenCV, and provides functions related to image luminance, saturation, phase inversion, cropping, resizing, image format conversion (RGB to HSV, YUV, YIQ, or gray), rotation, and sobel edge detection. This module is equivalent to a small image processing package of OpenCV.
- tf.keras: a Python API for invoking Keras tools.
  - This is a large module that enables various network operations.

HUAWEI

Keras Interface

- TensorFlow 2.x recommends Keras for network building. Common neural networks are included in **Keras.layers**.
- Keras is a high-level API used to build and train deep learning models. It can be used for rapid prototype design, advanced research, and production. It has the following three advantages:
  - Easy to use
    Keras provides simple and consistent GUIs optimized for common cases. It provides practical and clear feedback on user errors.
  - Modular and composable
    You can build Keras models by connecting configurable building blocks together, with little restriction.
  - Easy to extend
    You can customize building blocks to express new research ideas, create layers and loss functions, and develop advanced models.

HUAWEI

Common Keras Methods and Interfaces

- The following describes common methods and interfaces of tf.keras by focusing on code. The main content is as follows:
  - Dataset processing: datasets and preprocessing
  - Neural network model creation: Sequential, Model, Layers...
  - Network compilation: compile, Losses, Metrics, and Optimizers
  - Network training and evaluation: fit, fit_generator, and evaluate

HUAWEI

Contents

HUAWEI

TensorFlow Environment Setup in Windows 10

- Environment setup in Windows 10:
  - Operating system: Windows 10
  - pip software built in Anaconda 3 (adapting to Python 3)
  - TensorFlow installation:
    - Open Anaconda Prompt and run the pip command to install TensorFlow.
    - Run pip install TensorFlow in the command line interface.

```
(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)
```


HUAWEI

TensorFlow Environment Setup in Ubuntu/Linux

- The simplest way for installing TensorFlow in Linux is to run the pip command.

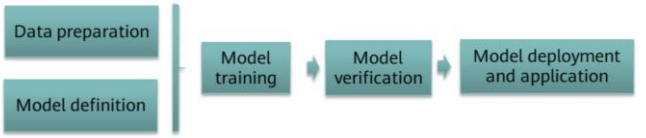
- pip command: pip install TensorFlow==2.1.0

HUAWEI

- If the installation is slow, replace the existing mirror with a Tsinghua mirror, and run the following command on the terminal:
  - pip install pip –U
  - pip config set global.index-url https://pypi.tuna.tsinghua.edu.cn/simple

TensorFlow Development Process

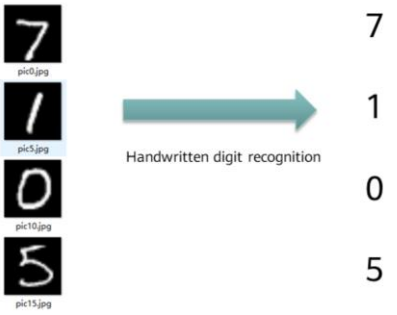
- Data preparation
  - Data exploration
  - Data processing
- Network construction
  - Defining a network structure.
  - Defining loss functions, selecting optimizers, and defining model evaluation indicators.
- Model training and verification
- Model saving
- Model restoration and invoking



HUAWEI

Project Description

- Handwritten digit recognition is a common image recognition task where computers recognize text in handwriting images. Different from printed fonts, handwriting of different people has different sizes and styles, making it difficult for computers to recognize handwriting. This project applies deep learning and TensorFlow tools to train and build models based on the MNIST handwriting dataset.



HUAWEI

Data Preparation

- MNIST datasets
  - Download the MNIST datasets from http://yann.lecun.com/exdb/mnist/.
  - The MNIST datasets consist of a training set and a test set.
    - Training set: 60,000 handwriting images and corresponding labels
    - Test set: 10,000 handwriting images and corresponding labels

Examples

Corresponding labels

[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]

HUAWEI

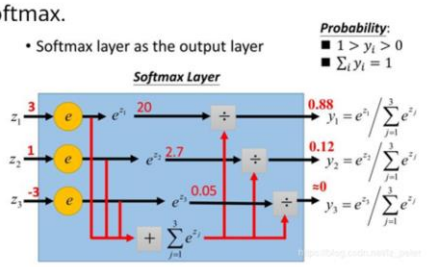
Network Structure Definition (1)

- Softmax regression model

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

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

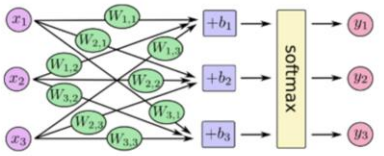
- The softmax function is also called normalized exponential function. It is a derivative of the binary classification function sigmoid in terms of multi-class classification. The following figure shows the calculation method of softmax.



HUAWEI

Network Structure Definition (2)

- The process of model establishment is the core process of network structure definition.
- The network operation process defines how model output is calculated based on input.



- Matrix multiplication and vector addition are used to express the calculation process of softmax.

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

HUAWEI

Network Structure Definition (3)

- TensorFlow-based softmax regression model

```
## 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)
```


HUAWEI

Network Compilation

- Model compilation involves the following two parts:
  - Loss function selection
- In machine learning/deep learning, an indicator needs to be defined to indicate whether a model is proper. This indicator is called cost or loss, and is minimized as far as possible. In this project, the cross entropy loss function is used.
  - Gradient descent method
- A loss function is constructed for an original model needs to be optimized by using an optimization algorithm, to find optimal parameters and further minimize a value of the loss function. Among optimization algorithms for solving machine learning parameters, the gradient descent-based optimization algorithm (Gradient Descent) is usually used.

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


HUAWEI

Model Training

- Training process:
  - All training data is trained through batch iteration or full iteration. In the experiment, all data is trained five times.
  - In TensorFlow, model.fit is used for training, where epoch indicates the number of training iterations.

```
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
```


HUAWEI

Model Evaluation

- You can test the model using the test set, compare predicted results with actual ones, and find correctly predicted labels, to calculate the accuracy of the test set.

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

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

Loss value

Accuracy

HUAWEI

Quiz

1. In TensorFlow 2.x, eager execution is enabled by default. ( )
   A. True
   B. False
2. Which of the following statements about tf.keras.Model and tf.keras.Sequential is incorrect when the tf.keras interface is used to build a network model? ( )
   A. tf.keras.Model supports network models with multiple inputs, while tf.keras.Sequential does not.
   B. tf.keras.Model supports network models with multiple outputs, while tf.keras.Sequential does not.
   C. tf.keras.Model is recommended for model building when a sharing layer exists on the network.
   D. tf.keras.Sequential is recommended for model building when a sharing layer exists on the network.

HUAWEI

- Answers:
  - A
  - D

Summary

- This chapter describes the following content by focusing on code: Features of common deep learning frameworks, including PyTorch and TensorFlow Basic syntax and common modules of TensorFlow 2.x Development procedure of TensorFlow.

HUAWEI

More Information

Official TensorFlow website: https://tensorflow.google.cn

Official PyTorch website: https://PyTorch.org/

HUAWEI

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.

Copyright©2020 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.

Huawei AI Development Framework — MindSpore

HUAWEI

Foreword

- This chapter describes Huawei AI development framework — MindSpore. It introduces MindSpore architecture, design ideas, and then MindSpore features through development challenges of AI computing frameworks. At last, it describes the MindSpore development and application to help you further understand this development framework.

HUAWEI

Objectives

On completion of this course, you will be able to:

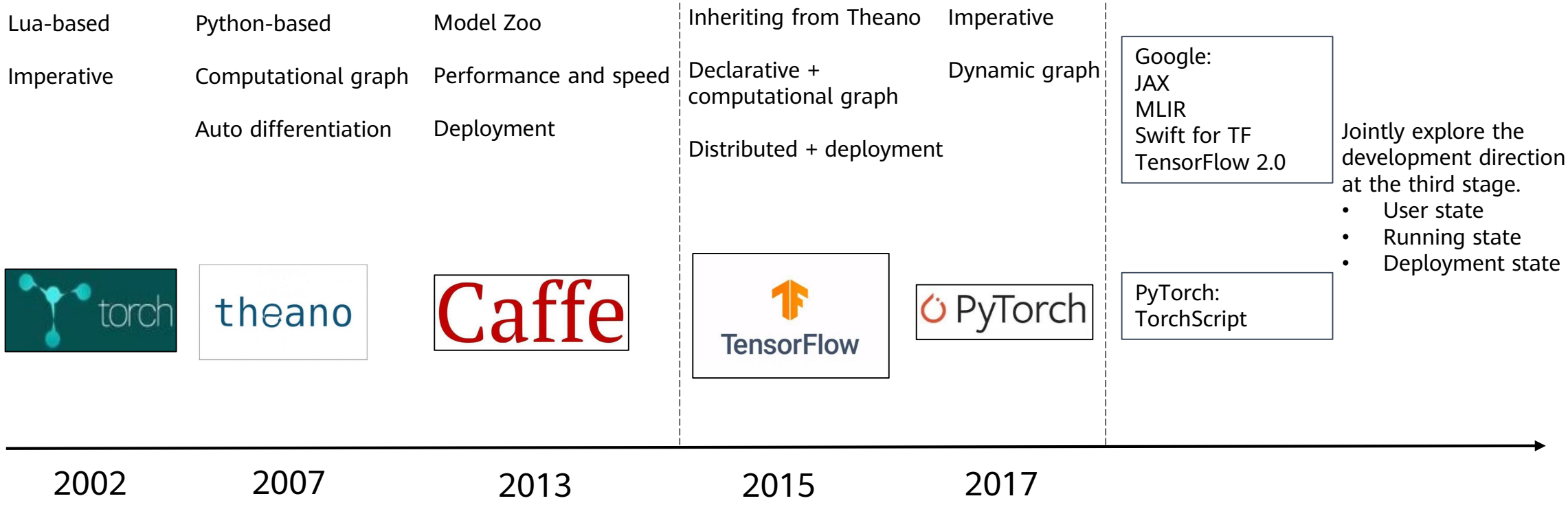
- Describe MindSpore.
- Understand the MindSpore framework.
- Understand MindSpore design ideas.
- Understand MindSpore features.
- Understand MindSpore environment setup process and development cases.

HUAWEI

Contents

HUAWEI

AI Framework Development History

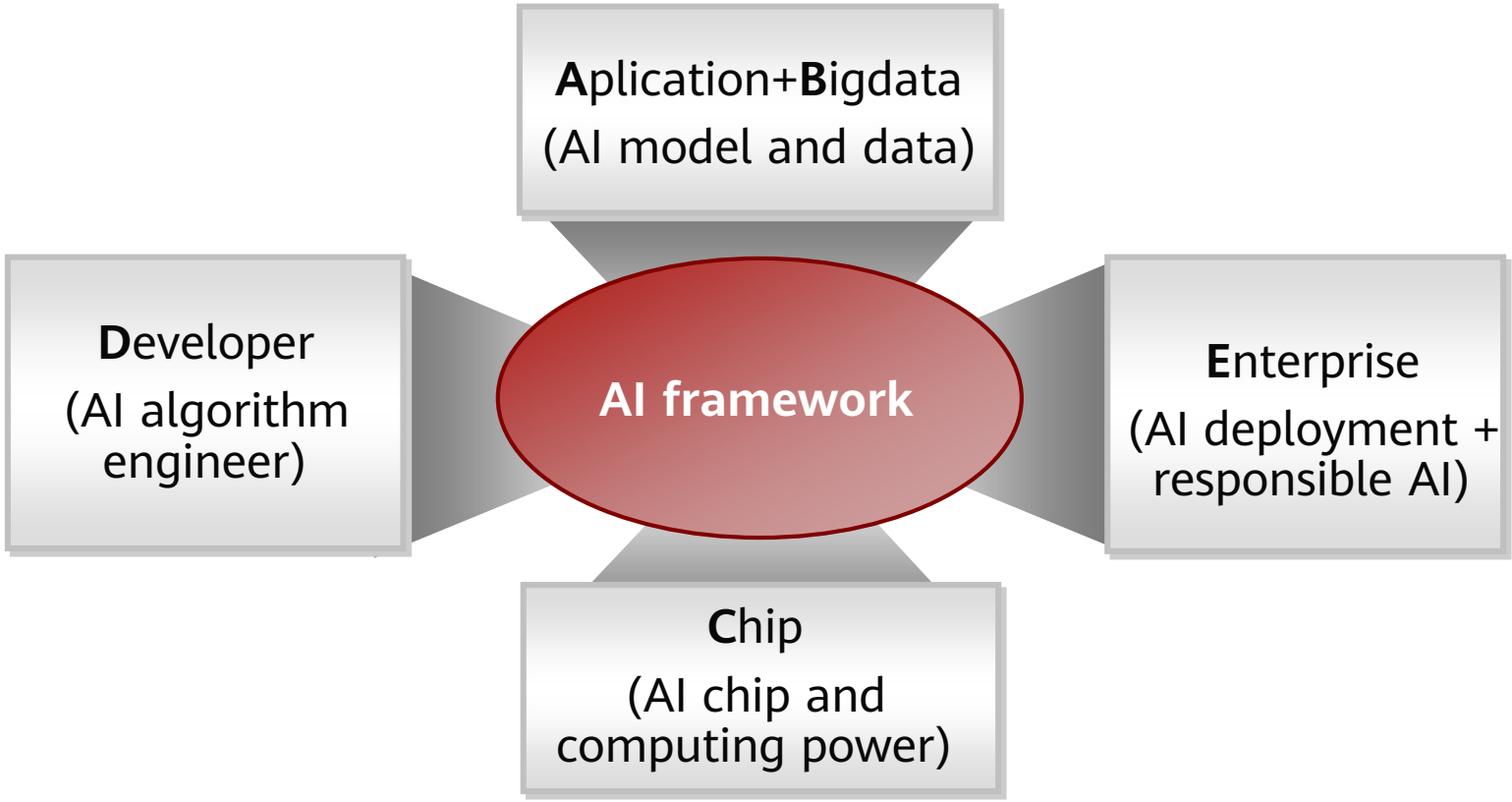


The AI framework technology has not been converged. Google has invested in four different directions to explore technologies. It is estimated that the integration workload in the future is huge.

HUAWEI

"ABCDE": Five Factors Driving the Evolution of the AI Framework

- Increasing model scale and complexity (GPT-3 parameter quantity reaches 175 billion.)
- Evolution from a single NN to general-purpose AI and scientific computing



- New AI programming language
- Unification of dynamic and static graphs

- Model deployment
- Security and privacy protection

- Continuous improvement of chip/cluster performance (Atlas 900 cluster supports a maximum of exabyte-level computing power.)
- Diversified heterogeneous computing power for CPUs, GPUs, and NPUs

HUAWEI

Contents

HUAWEI

Challenge 1: Increasing Model Scale and Complexity

Date	Model	Parameters	Institution
2018.4	ELMO	94m	Ai2
2018.7	GPT	110m	OpenAI
2018.10	BERT-Large	340m	Google
2019.1	Transformer ELMO	465m	Ai2
2019.1	GPT-2	1.5b	OpenAI
2019.7	MegatronLM	8.3b	NVDIA
2020.2	T-NLG	17.5b	Microsoft
2020.5	**GPT-3**	**175b**	**OpenAI**

- **GPT-3:**

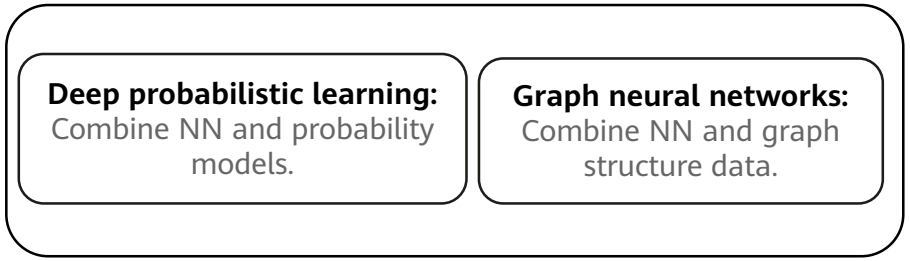
1. Parameters: 175 billion (600 GB+)
2. Datasets (before processing): 45 TB
3. Training cost: tens of millions of dollars; 1024 V100 GPUs; 127 days

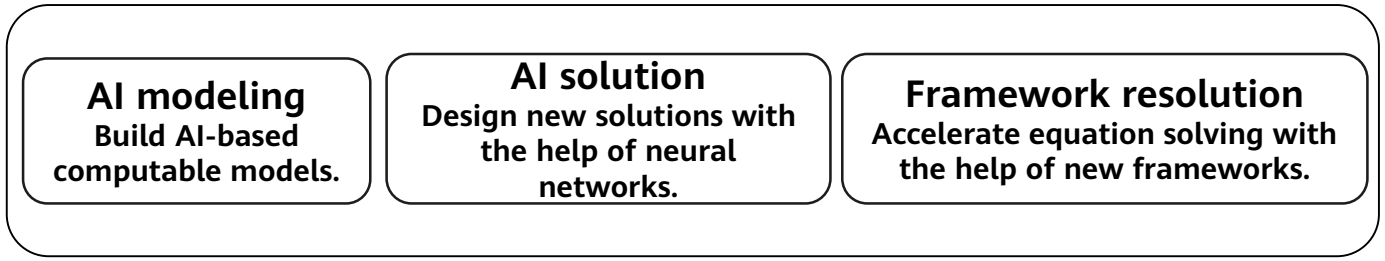
- **Technical challenges and trends:**

1. Performance (memory, communication, and computing usage)
   - Challenges: The single-device memory is insufficient (32 GB). The traffic volume varies greatly due to different parallel partitioning. The computing usage of different parallel partitioning is different. The data preprocessing bottleneck occurs.
   - Trend: memory overcommitment, hybrid parallelism (data parallelism, model parallelism, and pipeline parallelism), and data acceleration.
2. Efficiency
   - Challenges: Manual partitioning is demanding. Parallel logic and algorithm logic are coupled.
   - Trend: automatic parallelism.
3. Accuracy
   - Challenge: Optimizer for large batch sizes
   - Trend: second-order optimization

HUAWEI

Challenge 2: Evolution from Single NN to General-Purpose Tensor Differentiable Computing





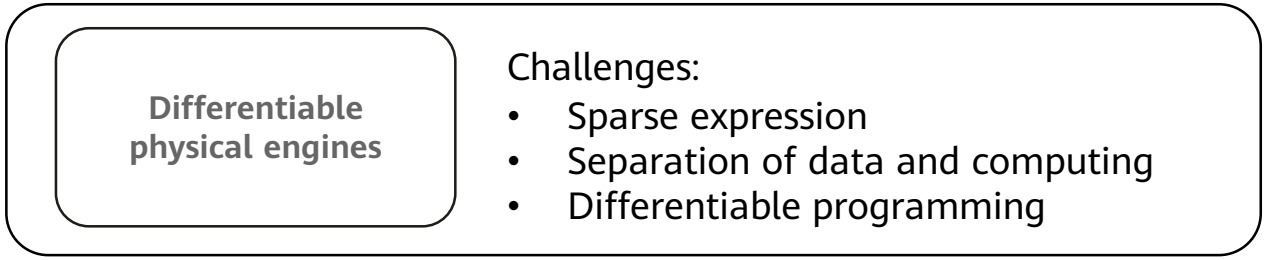
Challenges:

- Integrate NN models and probability models for modeling, reducing the learning difficulty.
- Store, partition, and sample trillions of distributed graph data.
- Support dynamic network structure and elastically distributed training.

Challenges:

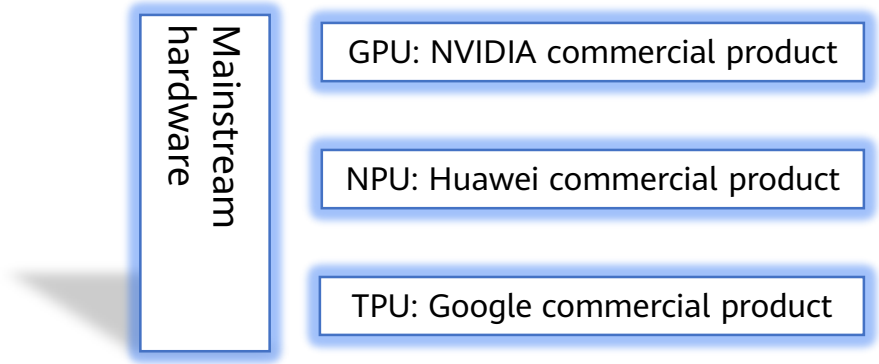
- Equations as code. Users can quickly construct expressions, and the serial coding and parallel coding are consistent.
- Support large-scale heterogeneous parallelism and mixed precision computing.
- Support high-performance higher-order differentiation (the volume of computing higher-order differentiation increases exponentially with the order).

- **Computing graphs (Taichi)**



HUAWEI

Challenge 3: Continuously Increasing Computing Power and Complexity



Hardware development trends

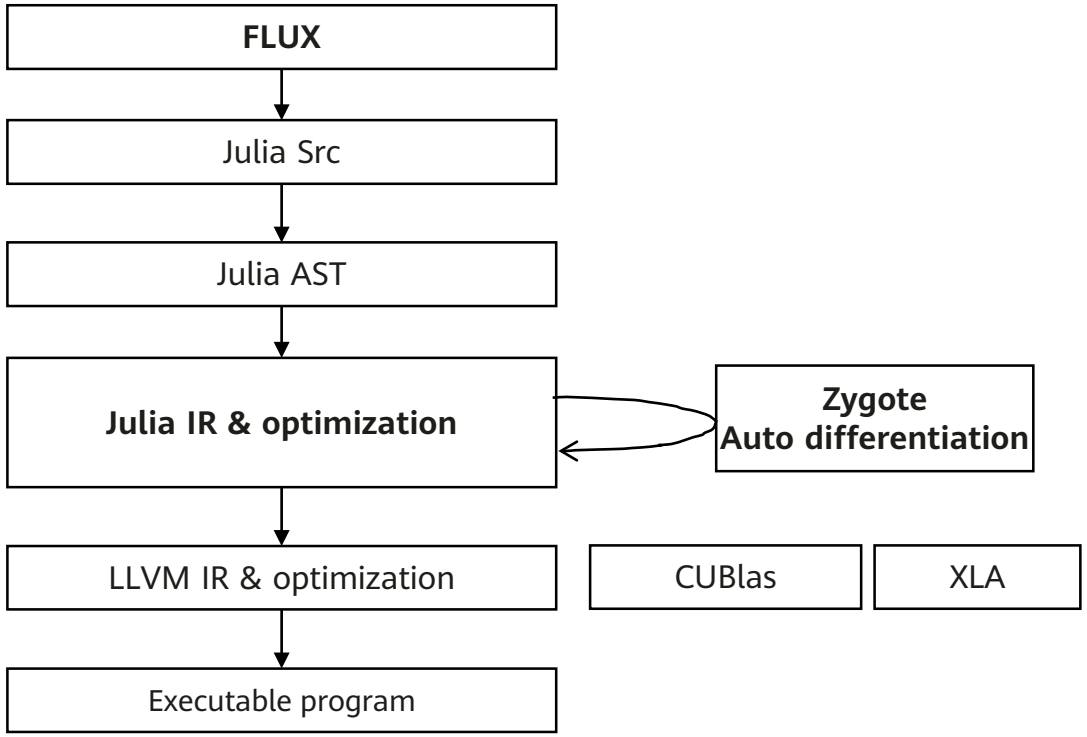
- Increase the computing density of a single core, improve the bandwidth and the process, increase the number of cores, and package more silicon chips.
- Widely use SIMD and increase the tensor core processing scale (4 x 4 → 16 x 16).
- New data types (such as TF32 and BF16), high-speed interconnection between chips, and support for virtualization.

Key challenges to AI framework software during AI chip development:

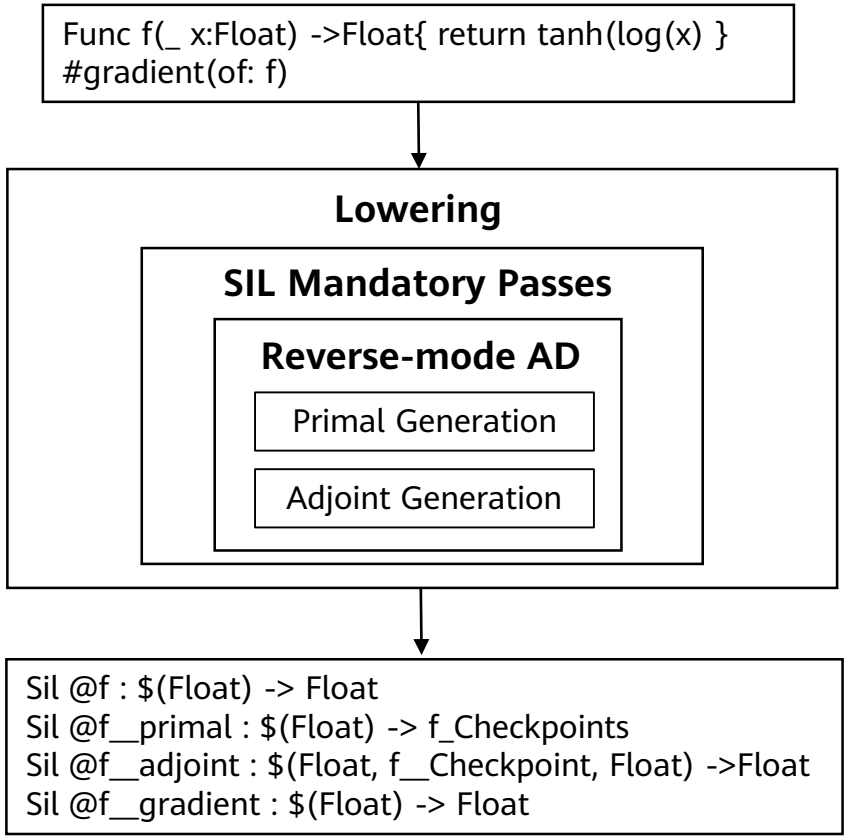
- **Improve the coupling of optimization and hardware, and integrate graph build and operator build.**
  - Fusion optimization at graph layer: Converge hardware-independent optimization, fully utilize the hardware computing power, and break the boundary between the subgraph level and operator level for overall optimization.
  - Optimization at operator layer: Consider hardware capabilities when using operators to implement algorithms.
- **Apply model execution modes to scenarios and hardware.**
  - Mix the graph sink mode and single-operator execution. Use different optimal mode according to the hardware.
  - Use the data flow execution mode to better exert the computing power.
  - Use the SoC-level distributed parallel strategy for packaging more silicon chips.
  - Use virtualization execution mode in SoC.
- **Huge programmability challenges.**
  - The effective computing power is close to the theoretical computing power and has high requirements on the compiler;
  - Sparse acceleration, image preprocessing acceleration module, and complex SIMD acceleration instructions;
  - SoC-level heterogeneous programming: CUBE core, Vector core, and ARM.
  - Multi-chip, single-chip cross-generation, and cross-model compatibility requirements.

HUAWEI

Challenge 4: New Programming Languages Making Breakthroughs in Python



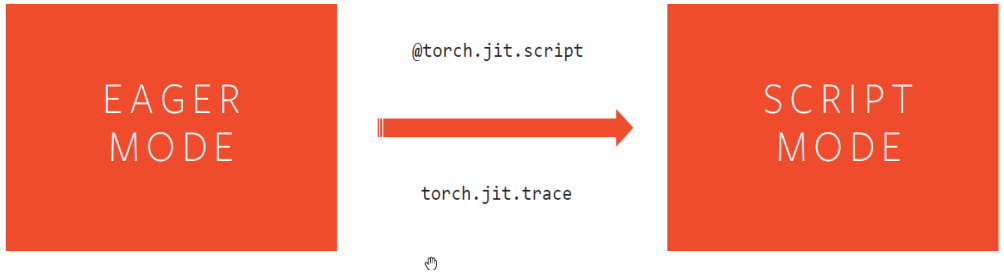
- Julia enters the AI field based on the tensor native expression, IR openness, and high performance as well as the accumulation in the scientific computing and HPC fields.



- Swift for TensorFlow tries to find differentiated competitiveness based on enterprise-class features such as static type, easy deployment, and high performance.

HUAWEI

Challenge 5: Unification of Dynamic and Static Graphs



- Research phase: dynamic graph; Python affinity, flexibility, and usability.
- Production phase: static graph; performance, and deployment capability.

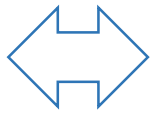
- Pain point: The representation of the dynamic graph is not completely the same as that of the static graph.
- Trend: Optimize JIT to achieve the consistency of the two representations.
- Challenge: It is difficult to fully use JIT to support Python flexibility and dynamics.

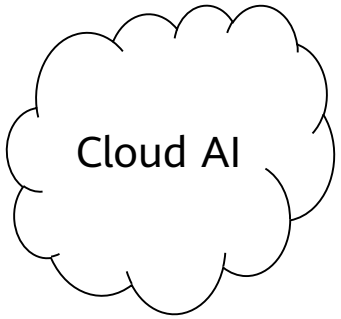
- Industry frameworks use compilers such as accelerated linear algebra (XLA) to work with chips for in-depth optimization.
- Gradually improve the IR from the perspective of optimization to form open AI infrastructure, such as Relay/TVM and MLIR.

HUAWEI

Challenge 6: AI Deployment in All Scenarios

According to the 2019 CIO Agenda survey conducted by Gartner, **the proportion of enterprises that have deployed AI increased from 4% to 14%** from 2018 to 2019. The data is in sharp contrast to the industry's increasing awareness of the value of AI.





Privacy and security

Low latency

High reliability

Low bandwidth required

High computing power

Large model

Big data

High network bandwidth

Tapping mode identification:
Deep learning (90%) vs. Traditional method (60%)

Trend 1: To reduce latency and improve user experience, on-device language model deployment becomes a trend. The challenge is how to reduce the model size and minimize the precision loss.

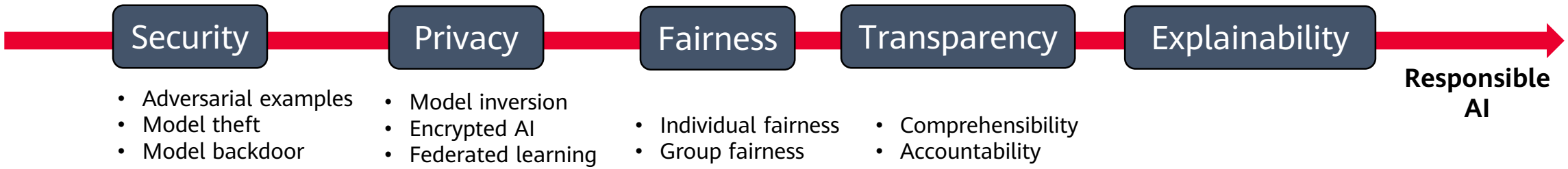
Trend 2: device-cloud synergy

1. Mobile AI = On-Device AI + Smart services, better considering personalization, security, and privacy.
2. Single agent → multiple agent collaboration, implementing real-time perception and decision-making.

Trend 3: Ubiquitous AI is deployed in scenarios where IoT and smart devices have extremely limited resources.

HUAWEI

Challenge 7: Security, Privacy, and Protection



Trend insights:

- In the future, in addition to accuracy and performance, meeting responsible AI will be a key requirement for AI service success.
- The AI framework bears AI services and must have the capability of enabling responsible AI.

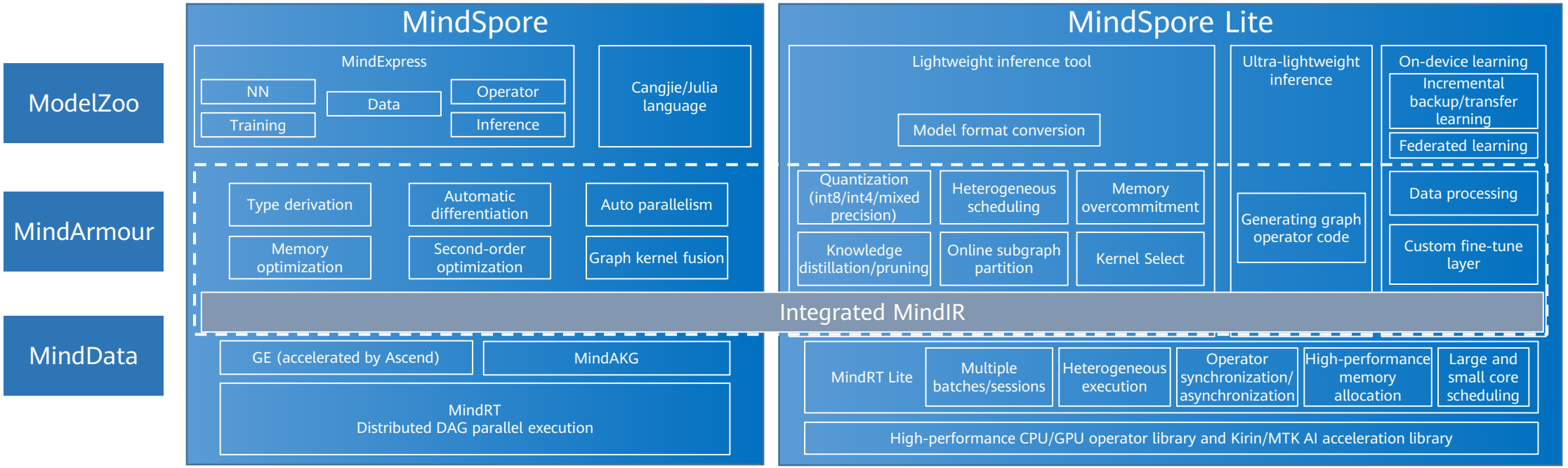
Key challenges:

- There is no general analysis method and measurement system for all aspects of responsible AI, and there is no automatic measurement method for scenario awareness.
- AI model robustness, privacy protection technologies, and encrypted AI have great impact on model performance in actual scenarios.
- Responsible AI is deeply combined with AI explainability and verifiability.

Contents

HUAWEI

MindSpore Open-source Deep Learning Framework



Superior performance

All-scenario support

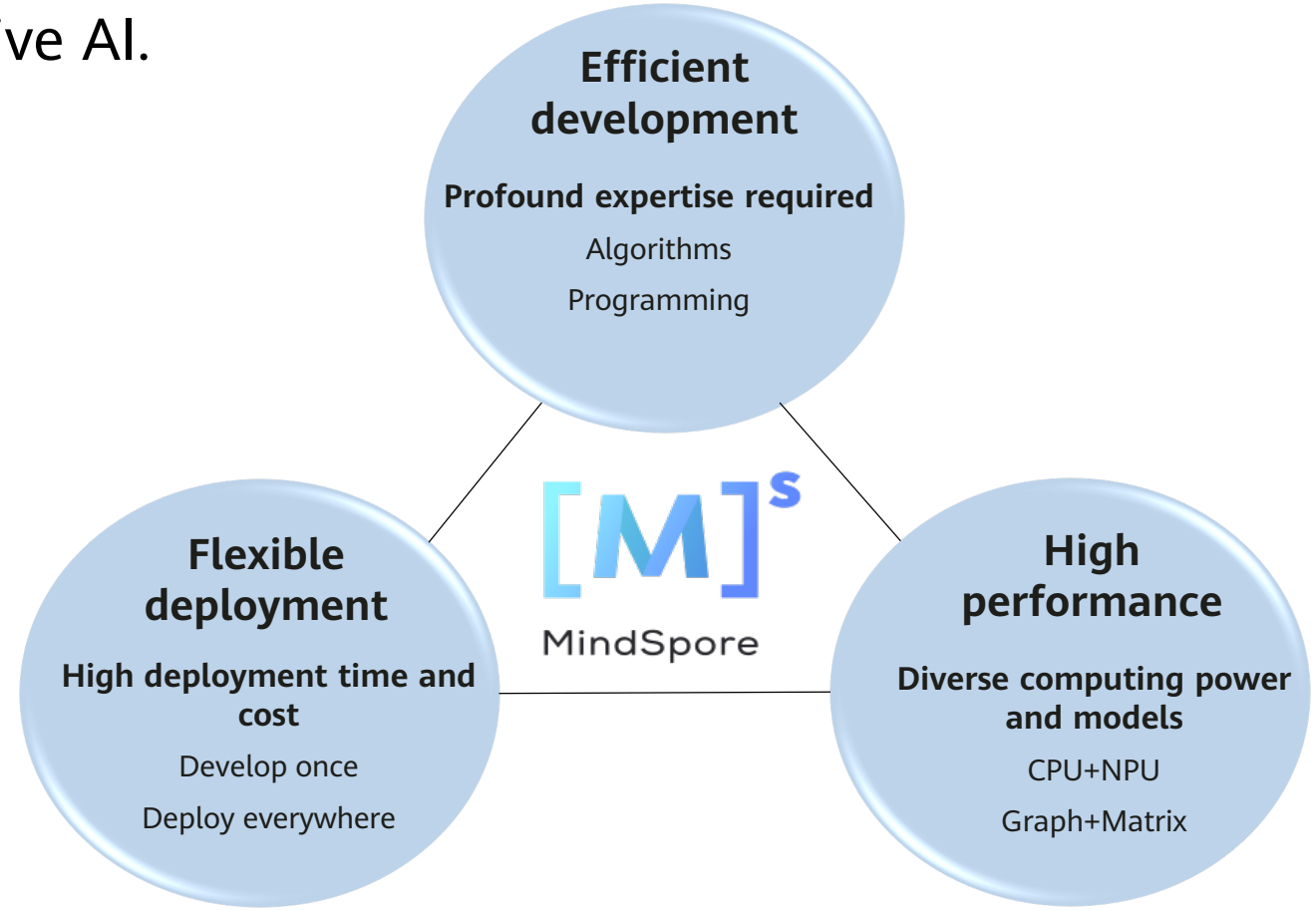
Lightweight

Efficient deployment

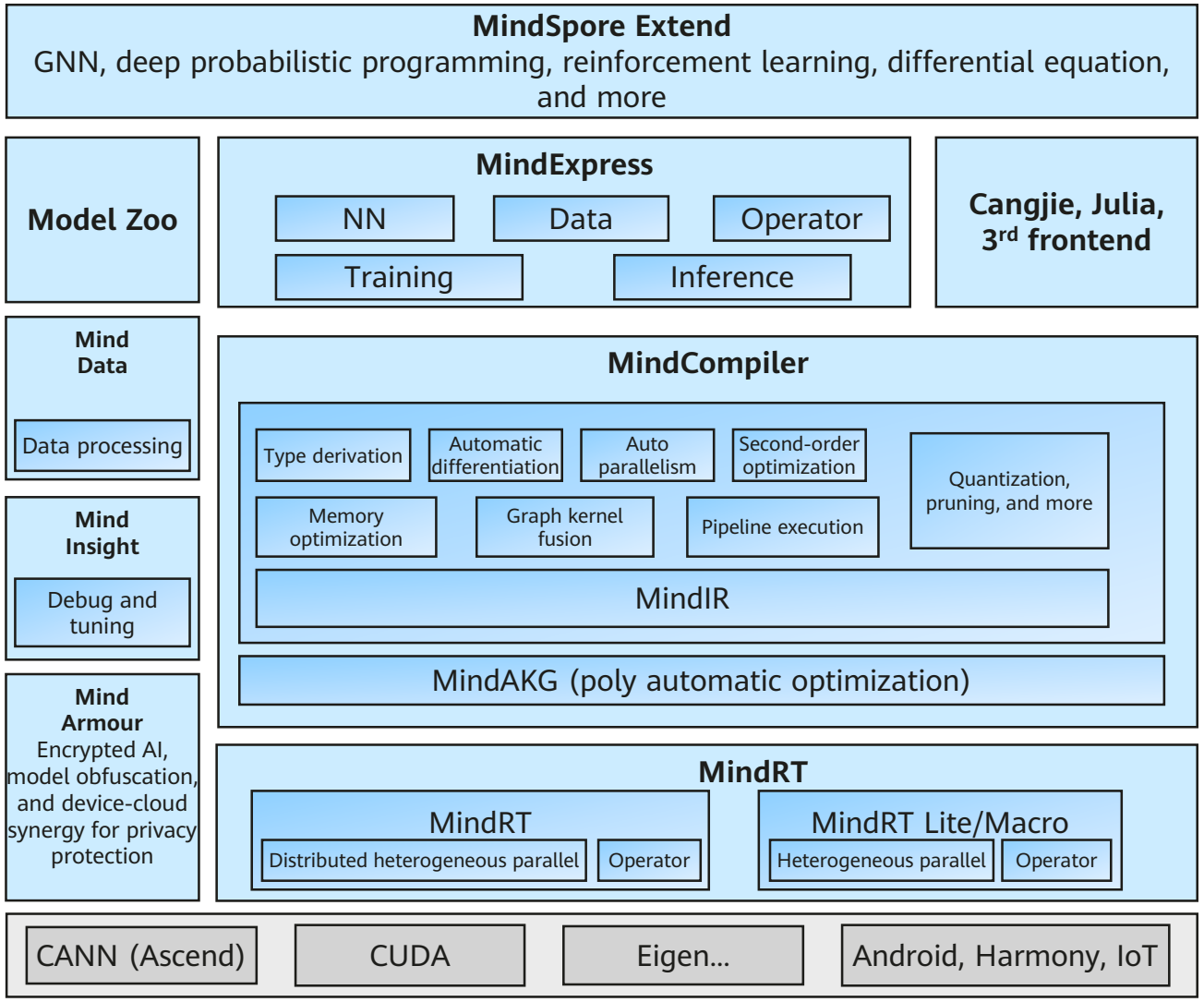
HUAWEI

MindSpore Vision and Value

- Lower the barrier for AI development, maximize Ascend computing power, and empower inclusive AI.



MindSpore Logical Architecture



Design Objectives

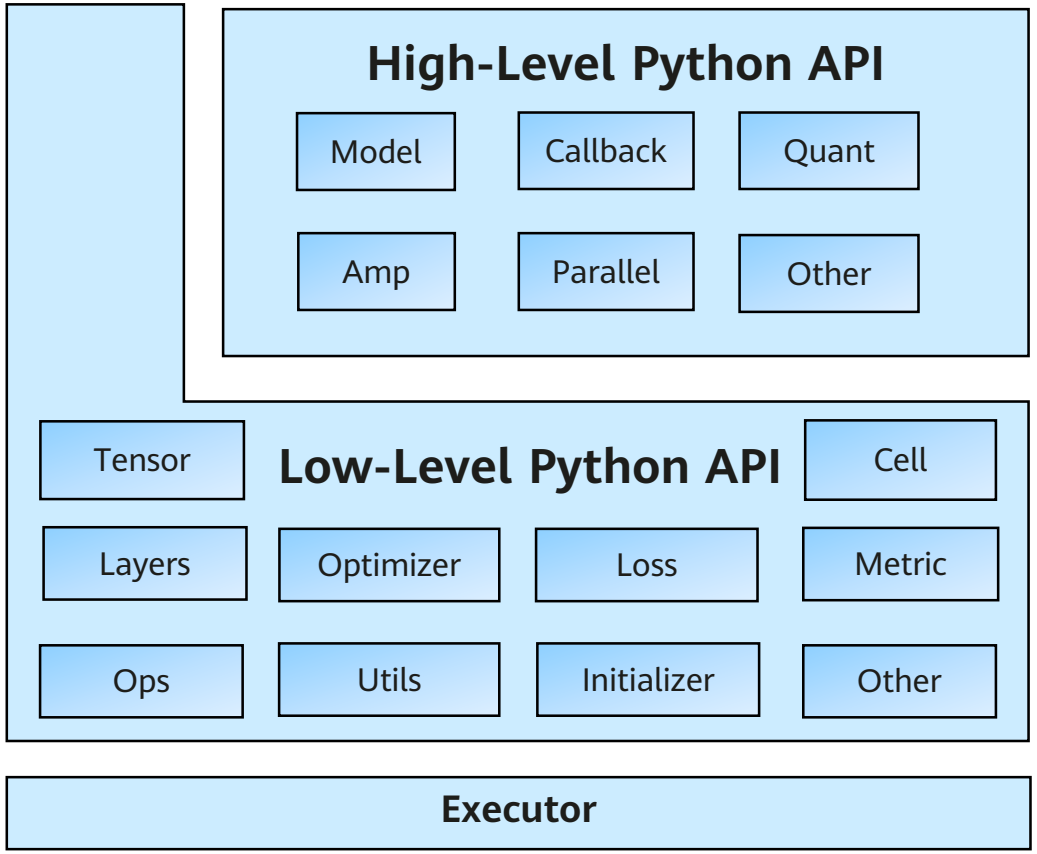
- **Beyond AI**: NN applications → general AI + numerical computation
  - Key feature: general-purpose tensor derivable computing
- **Distributed parallel native**: supporting AI models to go beyond trillions of parameters
  - Key features: automatic parallelism, memory-constrained programming, and second-order optimization
- **In-depth graph kernel fusion**: capitalizing on the computing power of AI chips
  - Key features: joint graph and kernel optimization as well as automatic optimization based on Poly
- **Enterprise-level capabilities in all scenarios**: flexible deployment and collaboration, secure, reliable, and explainable
  - Key features: ultra-lightweight runtime, private training, adaptive model generation, quantitative training, and explainable AI

Design philosophy: AI "JDK"

- **Representation/optimization/operation decoupling:** multi-frontend, cross-chip, and cross-platform
- **Openness:** opening the general graph compilation and running capabilities to third-party frameworks
- **Centralized architecture for all scenarios:** integrated APIs and IRs, enabling smooth AI applications

HUAWEI

Subsystem: MindExpress



Design objectives:

- Design both high-level and low-level APIs for users, supporting network building, entire graph execution, subgraph execution, and single-operator execution.
- Provide integrated APIs for model training, inference, and export, suitable for various scenarios, such as the device, edge, and cloud.
- Provide unified encoding for dynamic and static graphs.
- Provide unified encoding for standalone and distributed training.

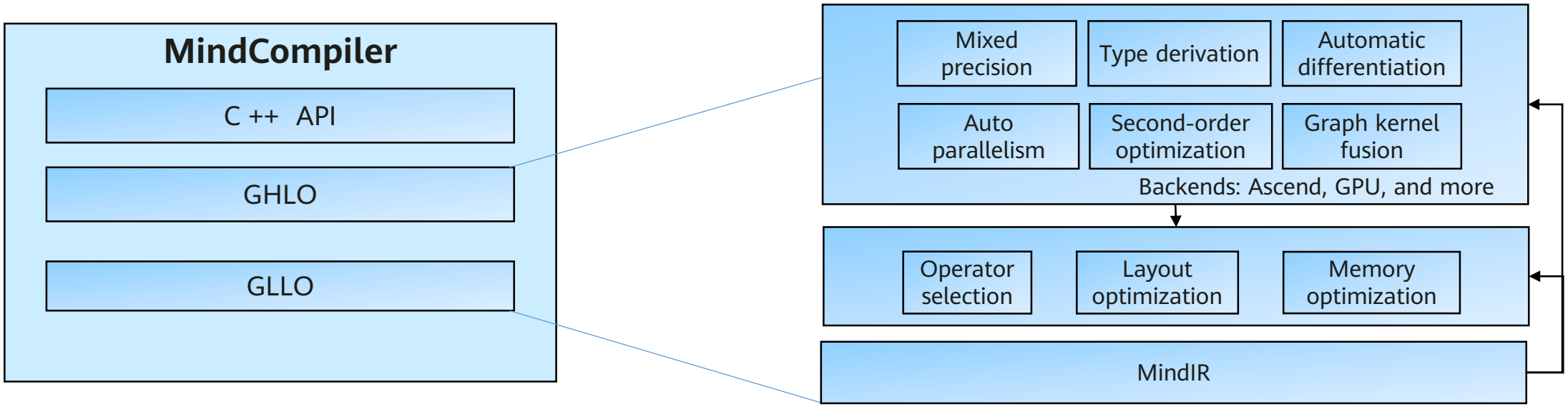
Functional modules:

- High-level APIs provide management, callback, quantization, mixed precision, and parallel control APIs for training and inference, facilitating process control on the entire network.
- Low-level APIs provide basic tensors, cells, NN-layers, optimizers, and initialization, helping users flexibly build networks and control execution processes.
- The executor controls computing execution and interacts with the MindSpore backend.

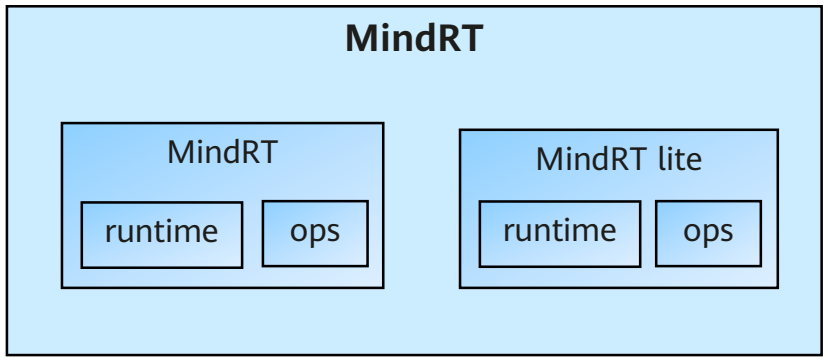
HUAWEI

Subsystem: MindCompiler

- MindCompiler provides the just-in-time compilation capability for MindIR.
  - Graph high level optimization (GHLO) is application-oriented and provides frontend optimization and functions, such as Type derivation, automatic differentiation, second-order optimization, and automatic parallelism.
  - Graph low level optimization (GHLO) is hardware-oriented and performs bottom-layer optimization, such as operator fusion, layout optimization, redundancy elimination, and memory optimization.



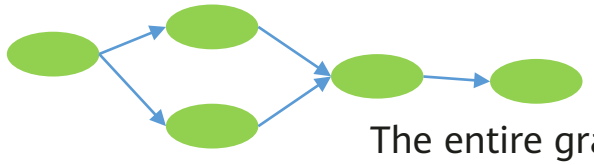
Subsystem: MindRT



The centralized runtime system supports:

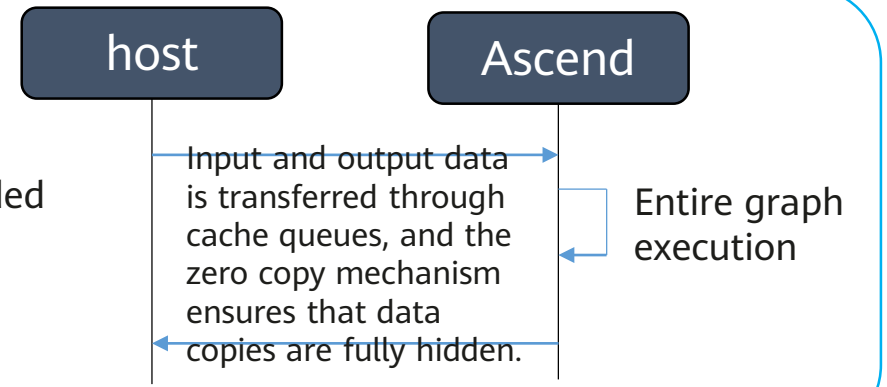
- Multiple device types on the device and cloud
- Scheduling management of multiple hardware platforms, such as Ascend, GPU, and CPU
- Memory pooling management and efficient memory overcommitment
- Asynchronous operators, heterogeneous execution, and multi-flow concurrency

Technology features



The entire graph is offloaded to avoid extra host-device interaction overheads.

Ascend

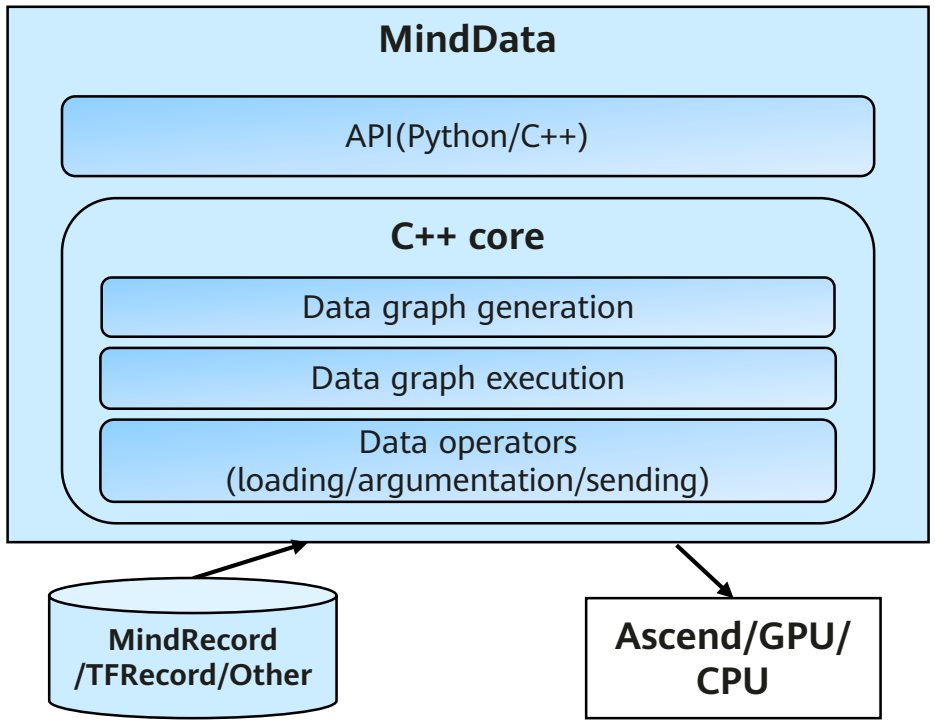


HUAWEI

Subsystem: MindData

MindData is responsible for efficiently executing the training data processing pipeline, forming a pipeline with computing, and promptly importing data for training.

Typical training data processing pipeline



Key functions:

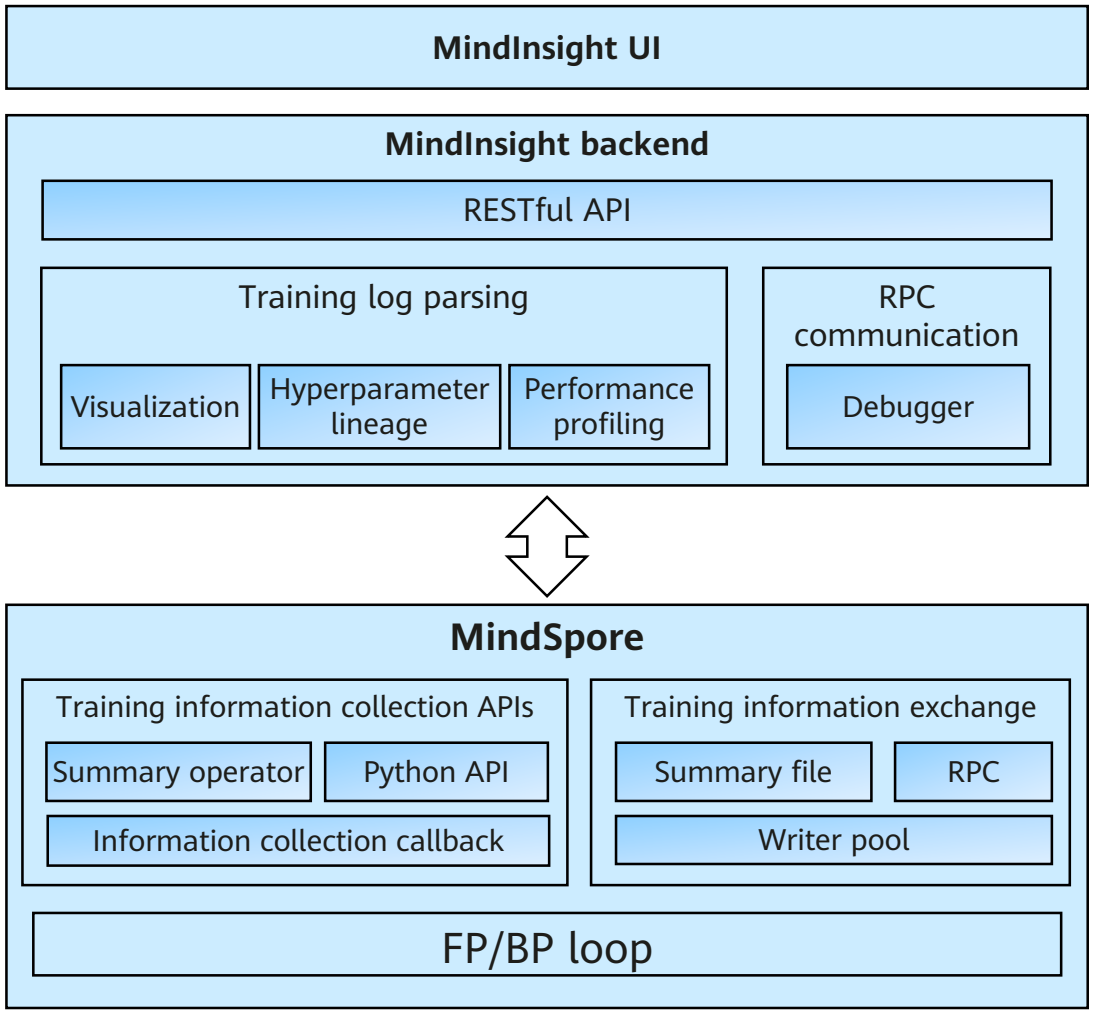
- Pipeline + parallel execution, improving data processing throughput
- Various data operators
- User-defined Python operators and pipelines (data loading, sampling, and argumentation)
- Heterogeneous hardware acceleration (Ascend/GPU/CPU)
- MindRecord: built-in metadata and aggregated storage

Running process:

1. Data graph generation: Data graphs are generated based on Python APIs called by users.
2. Data graph execution: The pipeline executes data operators in a data graph; this happens in parallel to complete dataset loading, shuffle, data argumentation, and batch processing.
3. Importing data to device: The processed data is imported to the device for training.

HUAWEI

Subsystem: MindInsight



MindInsight is the debugging and optimization subsystem of MindSpore. It provides the training process visualization, model lineage, debugger, and performance profiling functions.

Key functions:

- APIs are easy to use, enabling users to easily collect training process metrics, including computational graphs, scalar data (such as loss and accuracy), histogram data (such as gradient and weight), and performance data, and display them on the web UI.
- Collect training hyperparameters, datasets, and data augmentation information to implement model lineage and compare training results.

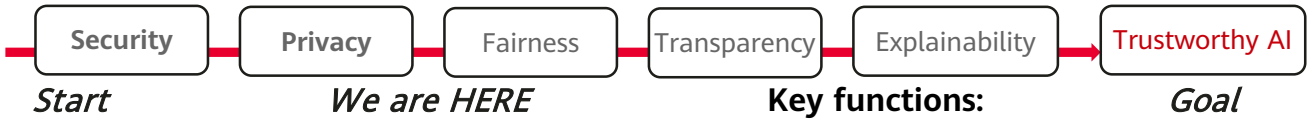
Running process:

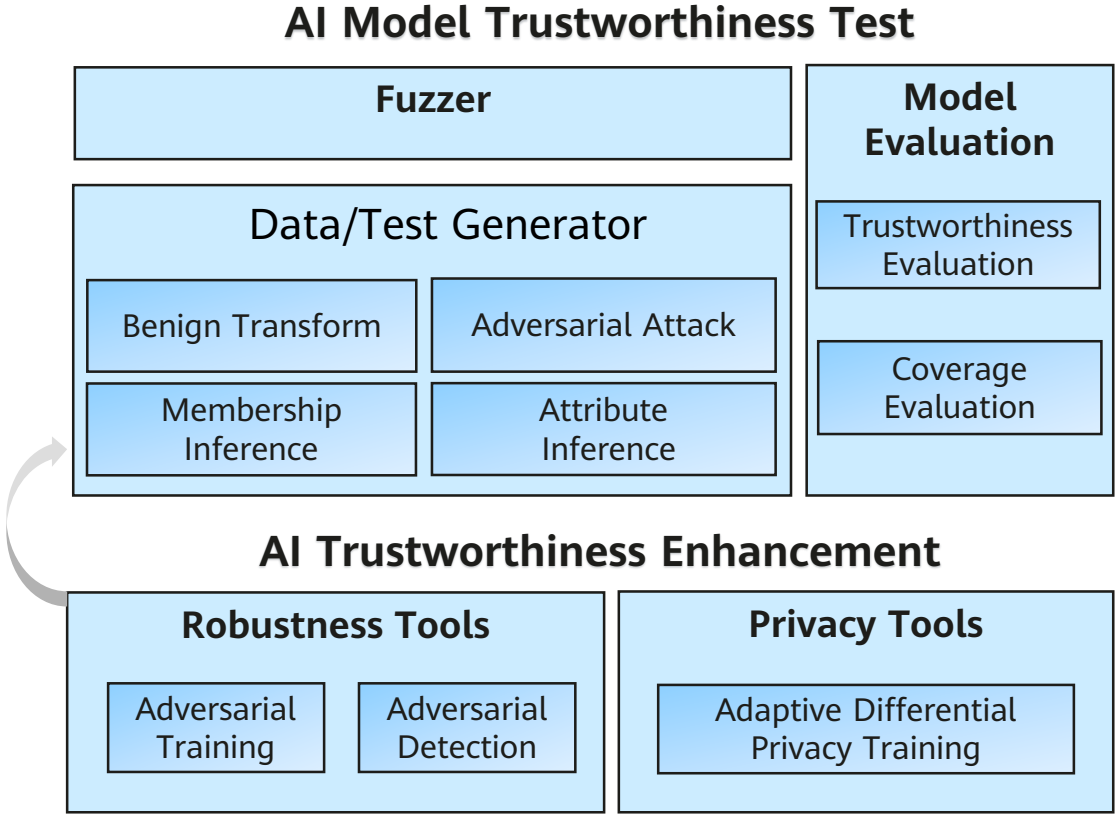
1. Collecting training information: Users collect common training indicators using the callback API, and can decide which information to collect based on their requirements. For example, use the summary operator to collect information about the computational graph and the Python API for information about the Python layer.
2. Generating training logs: Training logs are generated based on the process information collected during training.
3. Displaying training information: MindInsight opens and parses training logs to display the training process information in a graph.

HUAWEI

Subsystem: MindArmour

MindArmour provides comprehensive, effective, and easy-to-use evaluation tools and enhancement methods for AI trustworthiness in each domain.





Key functions:

- Test data generation methods in all scenarios, such as black-and-white box adversarial attacks, member and attribute inference attacks, and data drifts.
- Coverage-based fuzzing test process as well as flexible and customizable test strategies and indicators.
- Common methods of adversarial example detection and model robustness enhancement, including adversarial training and input rebuilding.
- Efficient, adaptive, and differential privacy training and budget statistics algorithms, as well as mathematically proven model for privacy leakage constraints.

Running process:

1. Configuring strategies: Define test strategies based on threat vectors and trustworthiness requirements, and select the appropriate test data generation methods.
2. Executing fuzzing test: Generate trustworthiness test data heuristically based on the model coverage rate and configured strategies.
3. Generating evaluation reports: Generate said reports based on built-in or user-defined trustworthiness indicators.
4. Enhancing trustworthiness: Use preset methods to enhance the trustworthiness of AI models.

HUAWEI

Contents

HUAWEI

MindSpore Feature: Automatic Parallelism

Challenges

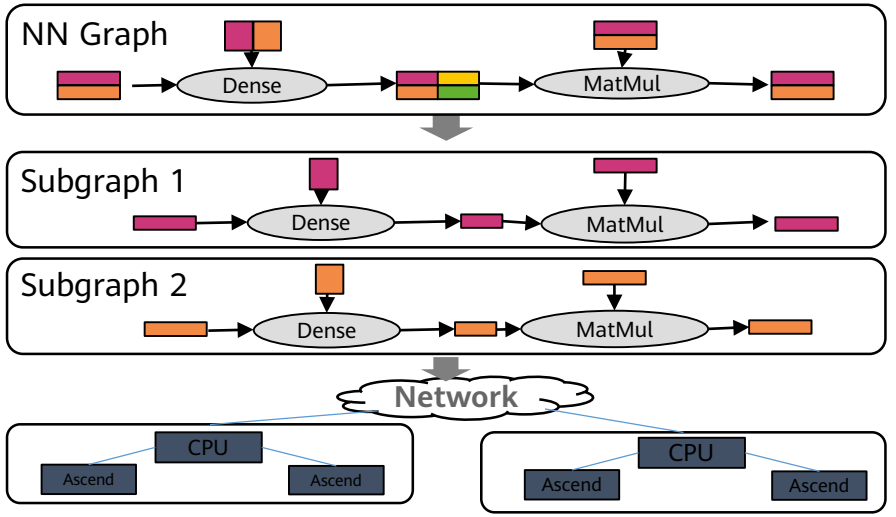
Challenges to efficient distributed training of ultra-large models:

NLP models become larger and larger. The memory overhead for training ultra-large models such as BERT (340 million)/GPT-2 (1542 million) exceeds the capacity of a single device. Therefore, the model needs to be partitioned into multiple devices for execution.

Currently, manual model parallelism requires model partitioning design and cluster topology awareness, which is difficult to develop, and it is hard to ensure high performance and perform tuning.

MindSpore Key Features

Automatically partition an entire graph based on the input and output data of the operator, and integrate data parallelism and model parallelism. **Cluster topology aware scheduling:** The cluster topology is aware, and subgraphs are automatically scheduled to minimize communication costs.



Effect: The standalone code logic is kept to implement model parallelism, improving development efficiency by 10 times compared with manual parallelism!

HUAWEI

MindSpore Feature: Second-order Optimization

Challenges

Deep learning model training requires a large amount of computing power, and training convergence takes a long time.

The second-order optimization method accelerates model convergence and reduces the number of training steps. However, it introduces a large number of complex computation, limiting its application in deep model training.

The second-order optimizer parameters are updated as follows:

$$\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} - \epsilon \boldsymbol{I} \boldsymbol{M}^{-1}_{\boldsymbol{\theta}^{(t)}} \nabla \boldsymbol{g}(\boldsymbol{\theta}^{(t)})$$

Parameter

Learning rate

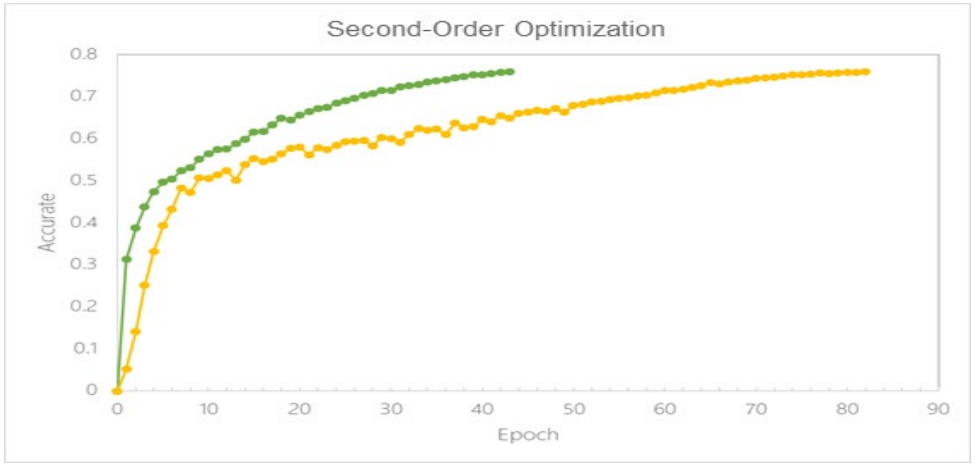
Second-order information matrix

First-order gradient

Core problem: The second-order optimizer needs to compute the inverse matrix of the second-order information matrix. The computation workload is heavy, and it can take hours to solve the second-order matrix directly, creating a technical difficulty.

MindSpore Key Features

The second-order matrix is approximated to reduce the computational complexity, and then the frequency and dimension of the matrix are reduced to accelerate the computation.



Optimizer	Epoch	Convergence Time	Test Scale
SGD+MOMENTUM	About 60	90 minutes	8-device Ascend 910
MindSpore second-order optimization	42	71.5 minutes	8-device Ascend 910

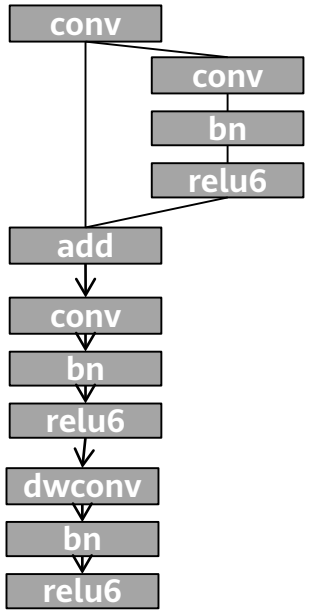
HUAWEI

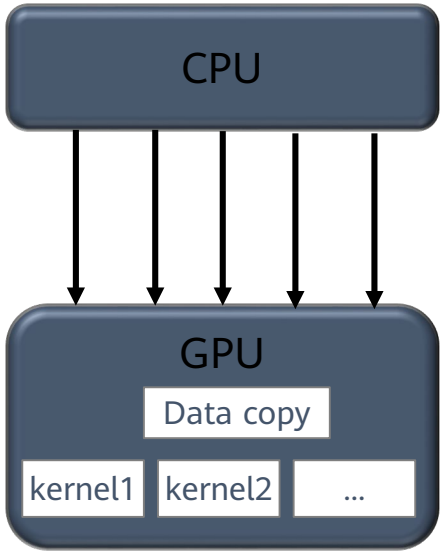
MindSpore Feature: On-Device Execution

Challenges

Challenges to model execution with powerful chip computing power:

Memory wall problems, high interaction overhead, and difficult data supply. Some operations are performed on the host, while others are performed on the device. The interaction overhead is much greater than the execution overhead. As a result, the accelerator usage is low.

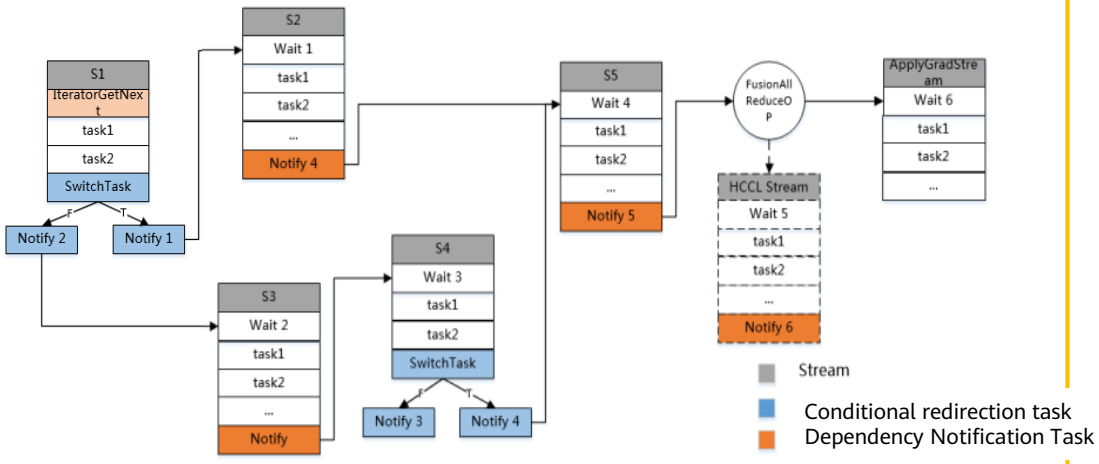




High data interaction overhead and difficult data supply.

MindSpore Key Features

The chip-oriented deep graph optimization is used to reduce synchronization waiting time and maximize the parallelism degree of "data-computing-communication". Data + Entire computational graph to the Ascend chips.



Effect: Compared with the host-side graph scheduling mode, the training performance is improved by 10 times!

HUAWEI

MindSpore Feature: Deployment and Collaboration in All Scenarios

Challenges

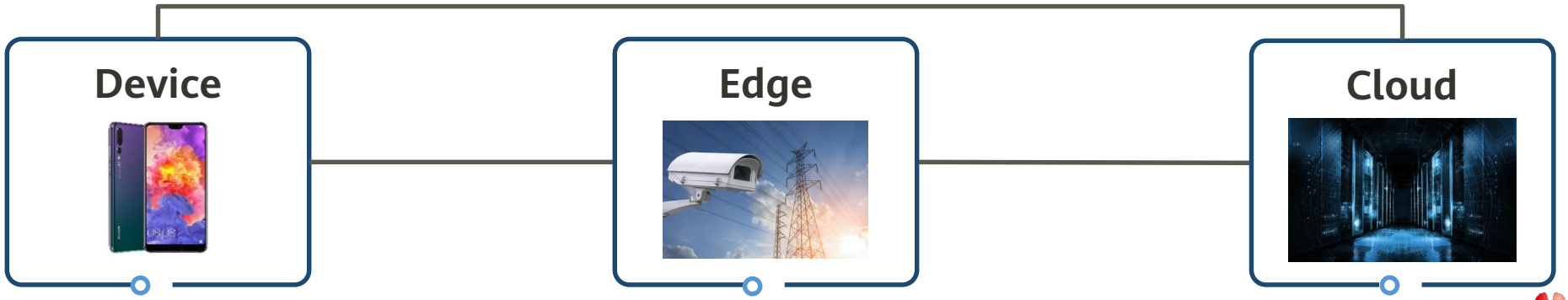
The diversity of hardware architectures leads to deployment differences and performance uncertainties in all scenarios, and the separation of training and inference results in model isolation.

MindSpore Key Features

- **Unified model IR** brings consistent deployment experience.
- The graph optimization technology based on software and hardware collaboration shields scenario differences.
- Federal meta learning based on device-cloud synergy breaks the boundaries of devices and the cloud. The multi-device collaboration model is updated in real time.

Effect: In the unified architecture, the deployment performance of models in all scenarios is consistent, and the accuracy of personalized models is improved!

On-demand collaboration in all scenarios and consistent development experience



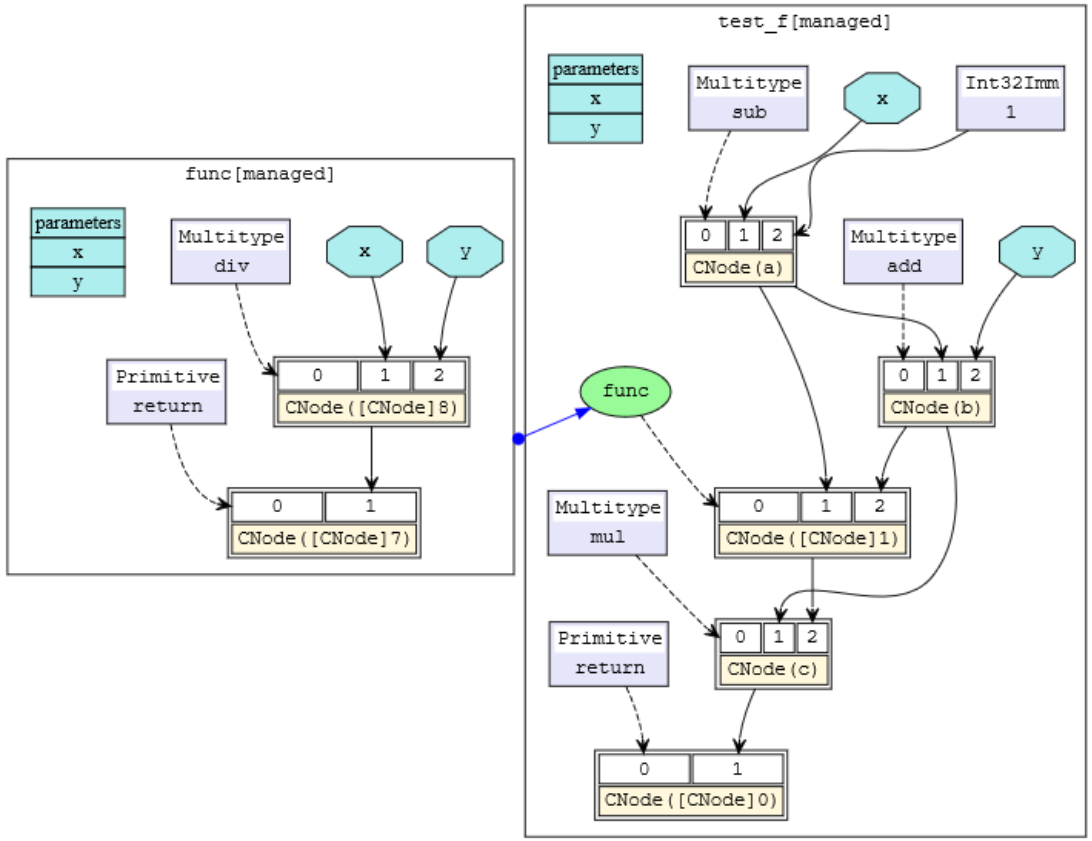
HUAWEI

MindSporeIR

- MindSporeIR is a simple, efficient, and flexible graph-based functional IR that can represent functional semantics such as free variables, higher-order functions, and recursion. It is the program carrier in the process of auto differentiation and compilation optimization.

- Each graph represents a function definition graph, which consists of ParameterNode, ValueNode, and ComplexNode(CNode).

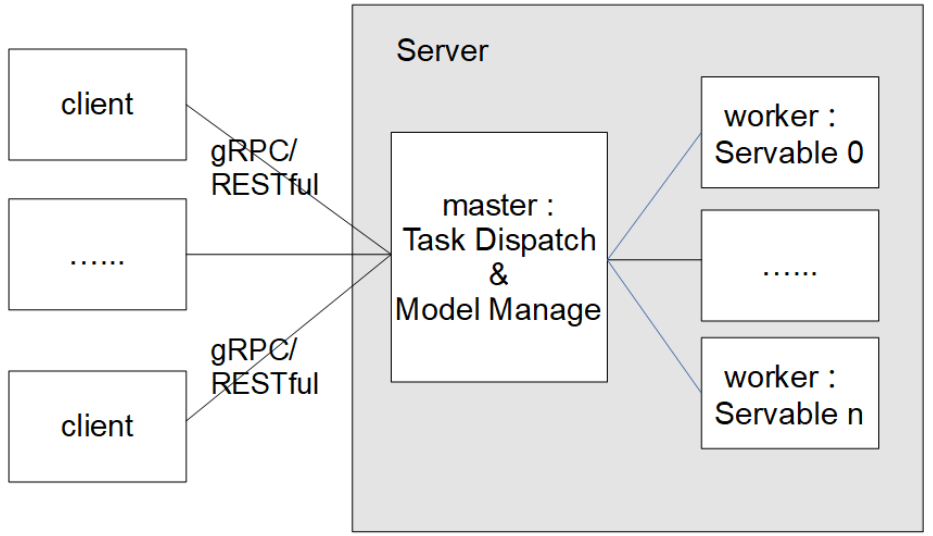
- The figure shows the def-use relationship.



MindSpore Serving: Efficient Deployment of Online Inference Services

MindSpore Serving is a lightweight and high-performance service module that helps MindSpore developers efficiently deploy online inference services in the production environment.

- Easy-to-use
- One-click release and deployment
- Batching
- High performance and scalability



MindSpore Serving structure

https://gitee.com/mindspore/serving/blob/r1.1/README_CN.md#%E9%85%8D%E7%BD%AE%E7%8E%AF%E5%A2%83%E5%8F%98%E9%87%8F

HUAWEI

Contents

HUAWEI

Installing MindSpore

For details about how to install MindSpore, visit https://mindspore.cn/install/en.

MindSpore supports platforms such as Windows, Ubuntu, and CentOS, and hardware such as Ascend 910, CPU, and GPU.

HUAWEI

MindSpore Experience

- In MindSpore, the data storage component is a tensor. Common tensor operations are as follows:
  - asnumpy()
  - size()
  - dim()
  - dtype()
  - set_dtype()
  - tensor_add(other: Tensor)
  - tensor_mul(ohter: Tensor)
  - shape()
  - __str__# Convert into a character string.

ME Module Components

Component	Description
model_zoo	Definition of common network models
communication	Data loading module, which provides the dataloader, dataset definition, and data processing functions such as image and text processing
dataset	Dataset processing module, such as data reading and preprocessing
common	Definitions of tensor, parameter, dtype, and initializer
context	Context class definition, which is used to set parameters for model running, for example, switching to graph or pynative mode
akg	Automatic differentiation and custom operator libraries
nn	Definitions of MindSpore cell, loss function, and optimizer
ops	Basic operator definition and backward operator registration
train	Training model-related and summary function modules
utils	Utilities mainly for parameter validation (for internal framework use)

HUAWEI

MindSpore Programming Concept: Operator

Softmax operator

```
class Softmax(PrimitiveWithInfer):
    r"""
    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
```

1. Operator name and base class

2. Operator comment

3. Operator initialization. The operator attribute values are initialized.

4. Shape derivation

5. data_type derivation

Common MindSpore operators are as follows:

- array: Array-related operators
- **ExpandDims** - **Squeeze**
- **Concat** - **OnesLike**
- **Select** -**StridedSlice**
- **ScatterNd, etc.**

- math: Mathematical operators
- **AddN** - **Cos**
- **Sub** - **Sin**
- **Mul** - **LogicalAnd**
- **MatMul** - **LogicalNot**
- **RealDiv** - **Less**
- **ReduceMean** - **Greater, etc.**

- nn: Network operators
- **-Conv2D** - **MaxPool**
- **-Flatten** - **AvgPool**
- **Softmax** - **TopK**
- **ReLU** - **SoftmaxCrossEntropy**
- **Sigmoid** - **SmoothL1Loss**
- **Pooling** - **SGD**
- **BatchNorm** - **SigmoidCrossEntropy,**
- **etc.**

- control: Control operators
- **ControlDepend**

-random: Random number-related operators

HUAWEI

MindSpore Programming Concept: Cell

- A cell provides basic modules that define computing execution. Cell objects can be directly executed.
  - **__init__**: initializes and verifies components such as parameters, cells, and primitives.
  - **construct**: defines the execution process; in graph mode, a graph is compiled for execution, which is subject to specific syntax restrictions.
  - **bprop** (optional): backward propagation of user-defined modules; if this method is undefined, the framework automatically generates a backward graph to compute the backward propagation of the construct part.
- The following cells are predefined in MindSpore: loss functions (SoftmaxCrossEntropyWithLogits and MSELoss), optimizers (Momentum, SGD, and Adam), network packaging functions (TrainOneStepCell for network gradient calculation and update, and WithGradCell for gradient calculation).

HUAWEI

Contents

HUAWEI

Computer Vision

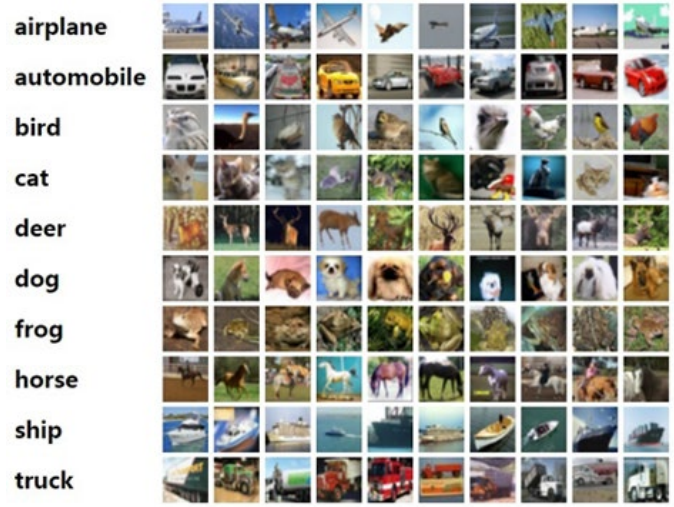
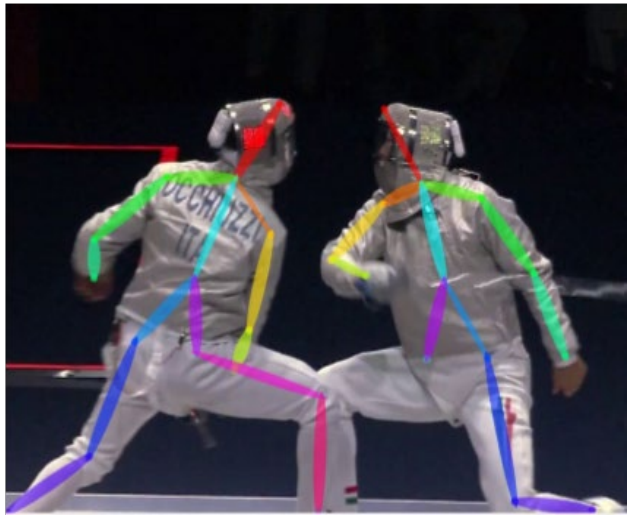


Image classification
arXiv:1512.03385

Object segmentation
arXiv:1703.06870



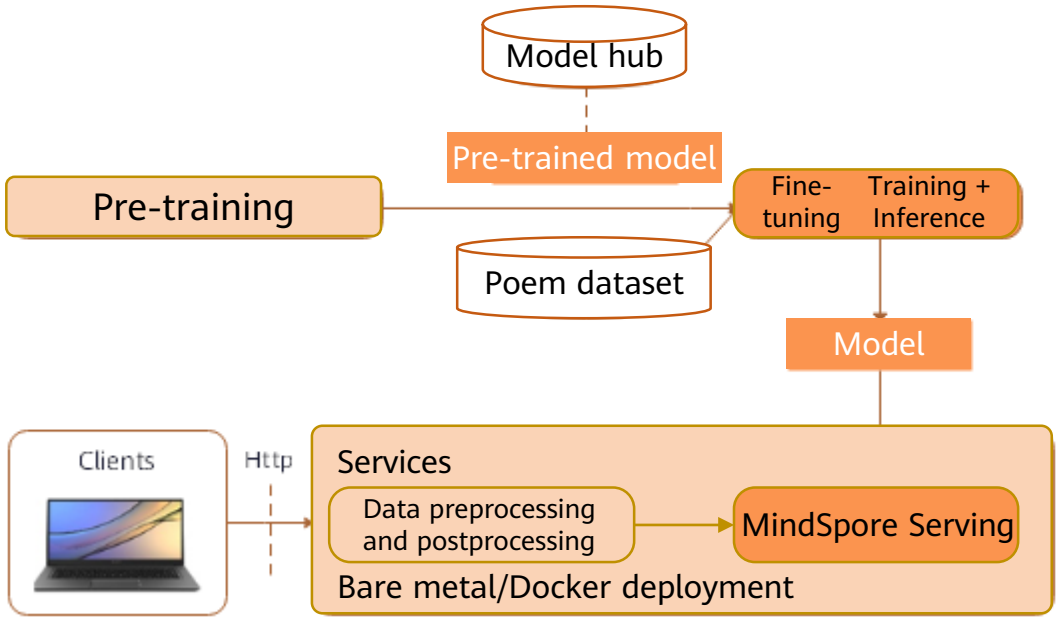
Keypoint detection
arXiv:1611.08050

https://www.mindspore.cn/tutorial/training/zh-CN/r1.1/advanced_use/cv.html#
https://arxiv.org/pdf/1611.08050.pdf
https://arxiv.org/pdf/1706.05587.pdf

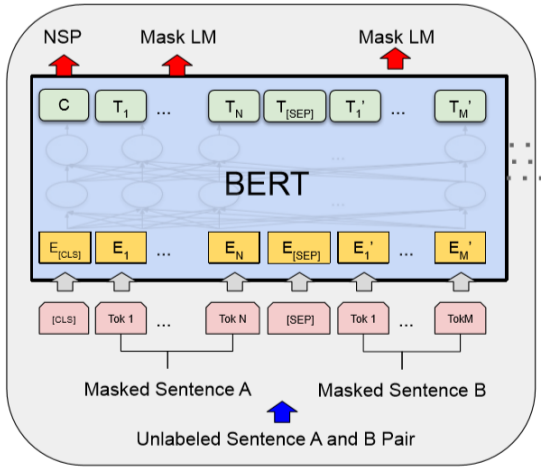
HUAWEI

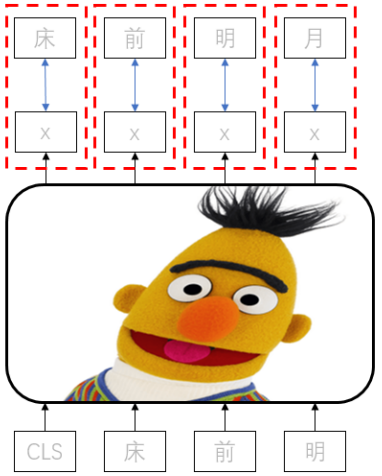
Natural Language Processing (NLP)

Use MindSpore to train intelligent poetry writing models and deploy prediction services



Application flowchart





Bert model and training

https://www.mindspore.cn/tutorial/training/en/r1.1/advanced_use/nlp_bert_poetry.html

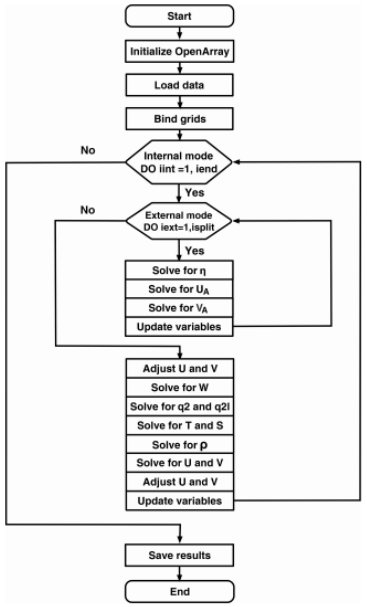
HUAWEI

High Performance Computing (HPC)

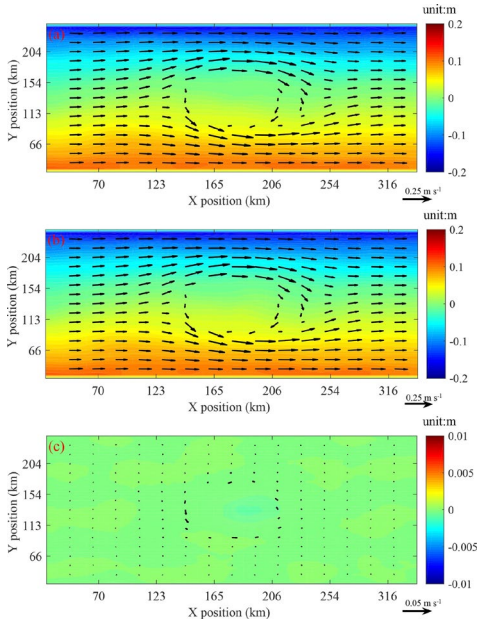
Computing faces unprecedented challenges due to massive amounts of data and access devices. Therefore, AI and HPC are slated to converge in the future, with AI transforming tradition HPC.

The 3D ocean model is key for the entire earth system model. By simulating ocean currents and whirlpools, it can predict typhoons and tsunamis in real time.

However, the code for conventional ocean models is complex and they run on CPUs. However, MindSpore accelerates the GOMO model framework, which runs on a GPU, significantly improving the model performance.



GOMO flowchart



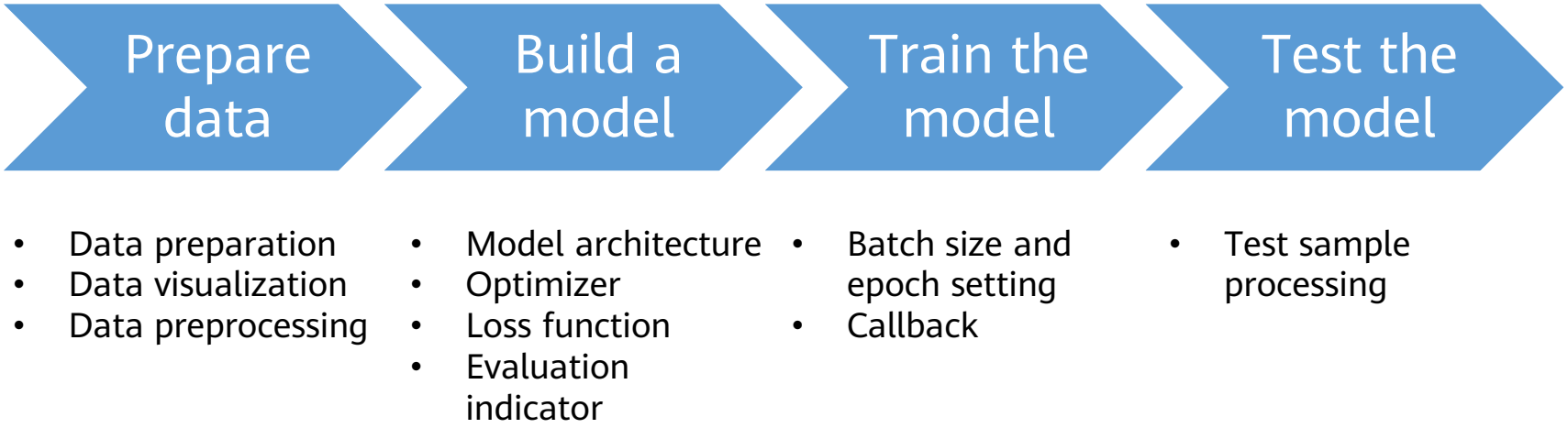
GOMO prediction

https://gmd.copernicus.org/articles/12/4729/2019/gmd-12-4729-2019.pdf

HUAWEI

MindSpore Application Case

- We will use the MNIST handwritten digit recognition to demonstrate the MindSpore-based modeling process.



Quiz

1. Which of the following operators does **nn** belong to in MindSpore? ()

   A. Math-related operators

   B. Network operators

   C. Control operators

   D. Other operators

HUAWEI

Summary

- This chapter introduces the MindSpore framework, design ideas, and features, and describes the MindSpore environment setup process and development procedure.

HUAWEI

More Information

MindSpore official website: https://mindspore.cn/en

Huawei Talent Online website: https://e.huawei.com/en/talent/#/home

WeChat official accounts:

EMUI

Huawei Device Open Lab

Huawei Developer

Contact Huawei Talent Online

HUAWEI

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。

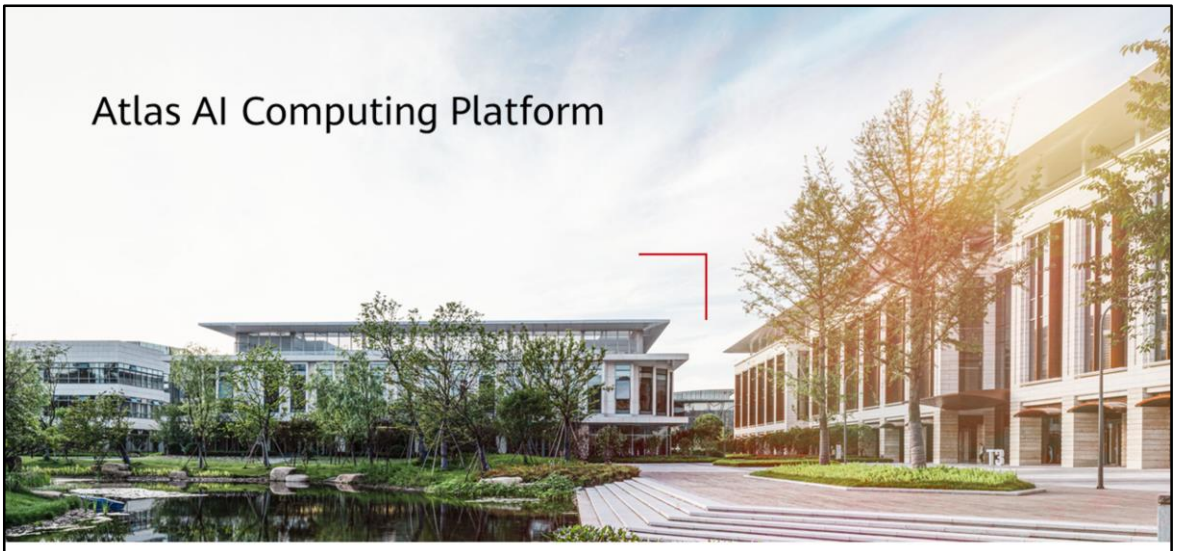
Bring digital to every person, home, and organization for a fully connected, intelligent world.

Copyright©2021 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that
could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.

HUAWEI

Atlas AI Computing Platform



HUAWEI

Foreword

- This chapter describes Huawei's Ascend AI chips, and hardware and software architectures of Ascend chips, and full-stack all-scenario solutions of Ascend AI chips.

HUAWEI

Objectives

On completion of the course, you will be able to:

- Get an overview of AI chips.
- Understand hardware and software architectures of Huawei Ascend chips.
- Learn about Huawei Atlas AI computing platform.
- Understand industry applications of Atlas.

HUAWEI

Contents

HUAWEI

Overview and Objectives

- This section is an overview of AI chips, including the introduction, classification, and status of AI chips, comparison between GPUs and CPUs, and introduction of Ascend AI processors.

HUAWEI

Definition

- Four elements of AI: data, algorithm, scenario, and computing power
- AI chips, also known as AI accelerators, are function modules that process massive computing tasks in AI applications.

HUAWEI

Contents

HUAWEI

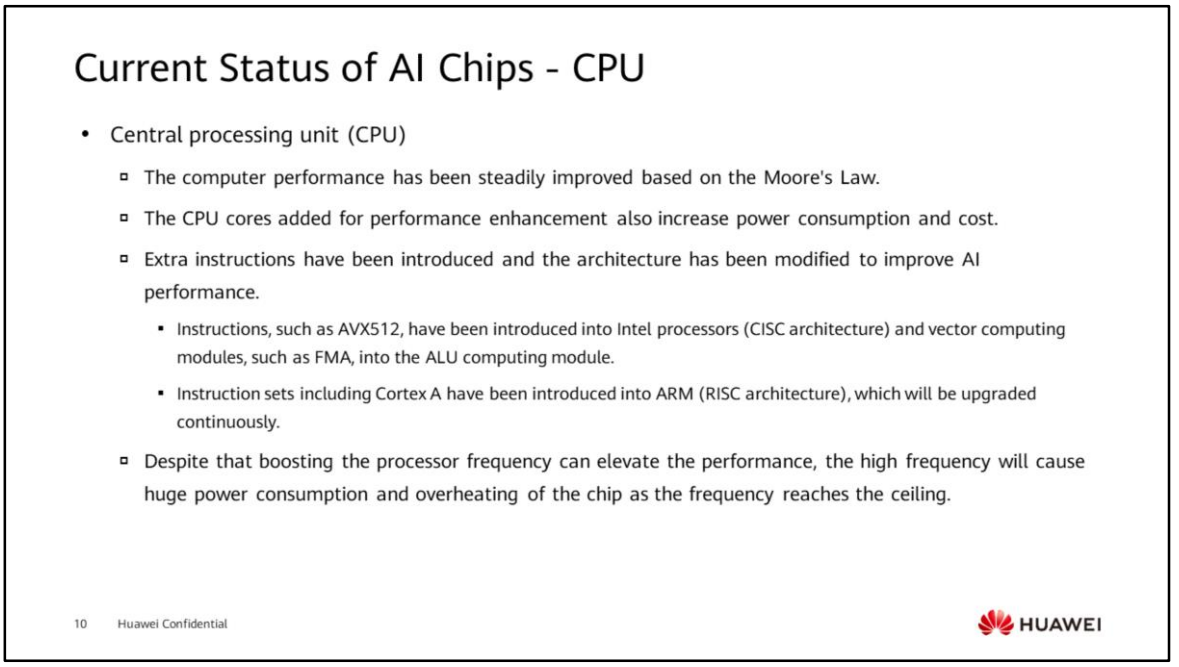
Classification of AI Chips

- AI Chips can be divided into four types by technical architecture:
  - A central processing unit (CPU): a super-large-scale integrated circuit, which is the computing core and control unit of a computer. It can interpret computer instructions and process computer software data.
  - A graphics processing unit (GPU): a display core, visual processor, and display chip. It is a microprocessor that processes images on personal computers, workstations, game consoles, and mobile devices, such as tablet computers and smart phones.
  - An application specific integrated circuit (ASIC): an integrated circuit designed for a specific purpose.
  - A field programmable gate array (FPGA): designed to implement functions of a semi-customized chip. The hardware structure can be flexibly configured and changed in real time based on requirements.

HUAWEI

Contents

HUAWEI



- In earlier years, the computer performance had been steadily improved based on the Moore's Law to meet the needs of users. People impose increasingly high requirements on computer performance, while performance improvement mostly depends on advancement of underlying hardware, which accelerates upper-layer application software. In recent years, improvement brought by the Moore's Law has slowed down. Hardware development gradually encounters physical bottlenecks. Limits on heat dissipation and power consumption make it difficult to further improve the performance of serial programs in the traditional CPU architecture. The current situation drives the industry to constantly look for an architecture and the corresponding software framework more suitable to the post-Moore's Law era.
- The multi-core processor is developed to better meet the hardware speed requirements of software. Intel Core i7 series processors use the parallel instruction processor cores constructed by four independent kernels based on the x86 instruction set, which accelerates the processor programs to some extent. However, the number of kernels cannot increase infinitely, and most traditional CPU programs are written by serial programming. Therefore, a large number of programs cannot be accelerated.

Current Status of AI Chips - GPU

- Graph processing unit (GPU)
  - GPU performs remarkably in matrix computing and parallel computing and plays a key role in heterogeneous computing. It was first introduced to the AI field as an acceleration chip for deep learning. Currently, the GPU ecosystem has matured.
  - Using the GPU architecture, NVIDIA focuses on the following two aspects of deep learning:
    - Diversifying the ecosystem: It has launched the cuDNN optimization library for neural networks to improve usability and optimize the GPU underlying architecture.
    - Improving customization: It supports various data types, including int8 in addition to float32; introduces modules dedicated for deep learning. For example, the optimized architecture of Tensor cores has been introduced, such as the TensorCore of V100.
  - The existing problems include high costs and latency and low energy efficiency.

HUAWEI

- The CPU focuses on logic control in instruction execution, while the GPU has outstanding advantages in large-scale, intensive, and parallel data computing. Program optimization generally requires collaboration of the CPU and GPU.

Current Status of AI Chips - TPU

- Tensor processing unit (TPU)
  - Since 2006, Google has sought to apply the design concept of ASICs to the neural network field and released TPU, a customized AI chip that supports TensorFlow, which is an open-source deep learning framework.
  - Massive systolic arrays and large-capacity on-chip storage are adopted to accelerate the most common convolution operations in deep neural networks.
    - Systolic arrays optimize matrix multiplication and convolution operations to elevate computing power and lower energy consumption.

HUAWEI

- The CPU focuses on logic control in instruction execution, while the GPU has outstanding advantages in large-scale, intensive, and parallel data computing. Program optimization requires collaboration of the CPU and GPU.

Current Status of AI Chips - FPGA

- Field programmable gate array (FPGA)
  - Using the HDL programmable mode, FPGAs are highly flexible, reconfigurable and re-programmable, and customizable.
  - Multiple FPGAs can be used to load the DNN model on the chips to lower computing latency. FPGAs outperform GPUs in terms of computing performance. However, the optimal performance cannot be achieved due to continuous erasing and programming. Besides, redundant transistors and cables, logic circuits with the same functions occupy a larger chip area.
  - The reconfigurable structure lowers supply and R&D risks. The cost is relatively flexible depending on the purchase quantity.
  - The design and tapeout processes are decoupled. The development period is long, generally half a year. The entry barrier is high.

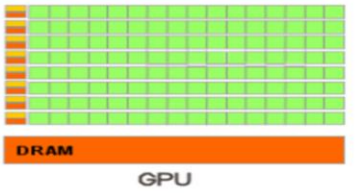
HUAWEI

Contents

HUAWEI

Design Comparison of GPUs and CPUs

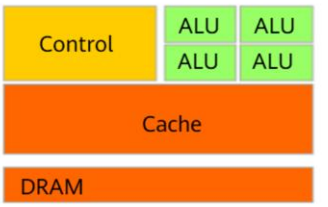
- GPUs are designed for massive data of the same type independent from each other and pure computing environments that do not need to be interrupted.
  - Each GPU comprises several large-sized parallel computing architectures with thousands of smaller cores designed to handle multiple tasks simultaneously.
  - Throughput-oriented design
    - With many ALUs and few caches, which improve services for threads, unlike those in CPU. The cache merges access to DRAM, causing latency.
    - The control unit performs combined access.
    - A large number of ALUs process numerous threads concurrently to cover up the latency.
  - Specialized in computing-intensive and easy-to-parallel programs



HUAWEI

Design Comparison of GPUs and CPUs

- CPUs need to process different data types in a universal manner, perform logic judgment, and introduce massive branch jumps and interrupted processing.
  - Composed of several cores optimized for sequential serial processing
  - Low-latency design
    - The powerful ALU unit can complete the calculation in a short clock cycle.
    - The large cache lowers latency.
    - High clock frequency
    - Complex logic control unit, multi-branch programs can reduce latency through branch prediction.
    - For instructions that depend on the previous instruction result, the logic unit determines the location of the instructions in the pipeline to speed up data forwarding.
  - Specialized in logic control and serial operation



HUAWEI

Contents

HUAWEI

Ascend AI Processors

- Neural-network processing unit (NPU): uses a deep learning instruction set to process a large number of human neurons and synapses simulated at the circuit layer. One instruction is used to process a group of neurons.
- Typical NPUs: Huawei Ascend AI chips, Cambricon chips, and IBM TrueNorth

- Ascend-Mini
- Architecture: Da Vinci
- Half precision (FP16): 8 Tera-FLOPS
- Integer precision (INT8): 16 Tera-OPS
- 16-channel full-HD video decoder: H.264/H.265
- 1-channel full-HD video decoder: H.264/H.265
- **Max. power: 8W**
- 12nm FFC

- Ascend-Max
- Architecture: Da Vinci
- Half precision (FP16): 256 Tera-FLOPS
- Integer precision (INT8): 512 Tera-OPS
- 128-channel full-HD video decoder: H.264/H.265
- Max. power: 350W
- 7nm

HUAWEI

Section Summary

- This section describes AI chips, including classification of AI chips by technologies and functions, AI chip ecosystem, and comparison between GPUs and CPUs.

HUAWEI

Contents

HUAWEI

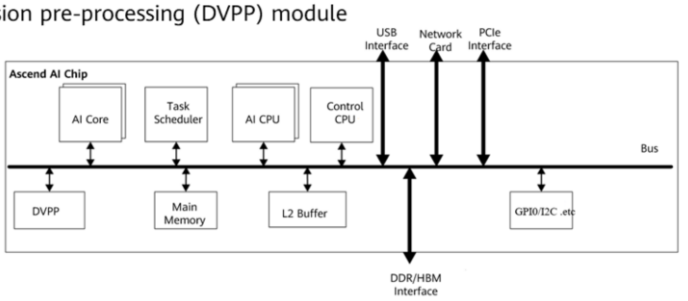
Overview and Objectives

- This section describes the hardware architecture of Ascend chips, including the logic architecture of the Ascend AI processors and Da Vinci architecture.

HUAWEI

Logic Architecture of Ascend AI Processors

- Ascend AI processor consist of:
  - Control CPU
  - AI computing engine, including AI core and AI CPU
  - Multi-layer system-on-chip (SoC) caches or buffers
  - Digital vision pre-processing (DVPP) module



HUAWEI

Contents

HUAWEI

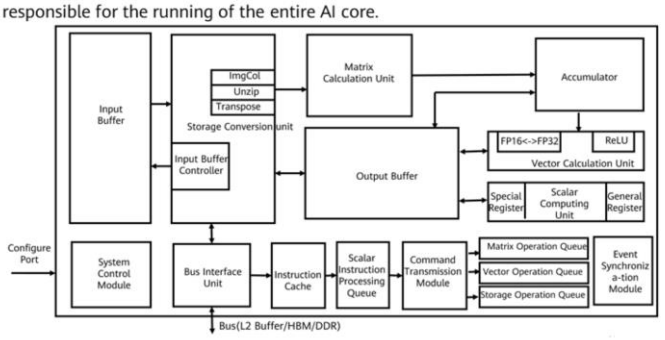
Ascend AI Computing Engine - Da Vinci Architecture

- One of the four major architectures of Ascend AI processors is the AI computing engine, which consists of the AI core (Da Vinci architecture) and AI CPU. The Da Vinci architecture developed to improve the AI computing power serves as the core of the Ascend AI computing engine and AI processor.

HUAWEI

Da Vinci Architecture (AI Core)

- Main components of the Da Vinci architecture:
  - Computing unit: It consists of the cube unit, vector unit, and scalar unit.
  - Storage system: It consists of the on-chip storage unit of the AI core and data channels.
  - Control unit provides instruction control for the entire computing process. It is equivalent to the command center of the AI core and is responsible for the running of the entire AI core.



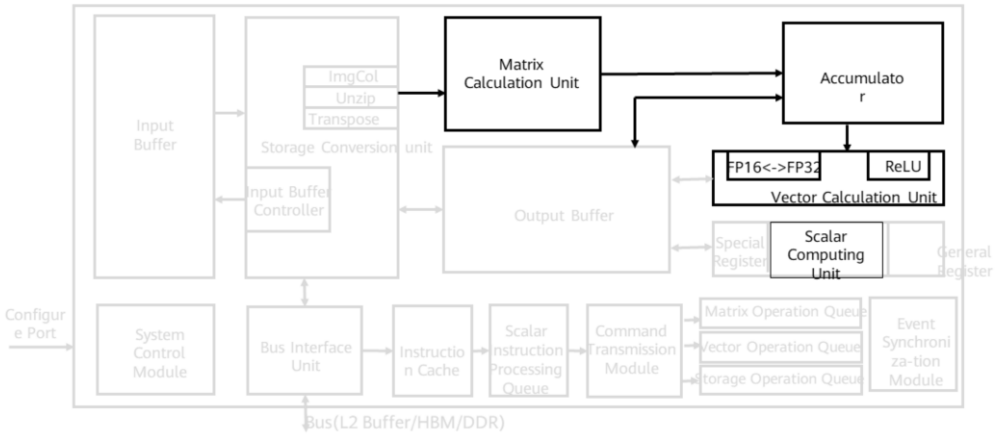
HUAWEI

Da Vinci Architecture (AI Core) - Computing Unit

- Three types of basic computing units: cube, vector, and scalar units, which correspond to matrix, vector and scalar computing modes respectively.
- Cube computing unit: The matrix computing unit and accumulator are used to perform matrix-related operations. Completes a matrix (4096) of 16x16 multiplied by 16x16 for FP16, or a matrix (8192) of 16x32 multiplied by 32x16 for the INT8 input in a shot.
- Vector computing unit: Implements computing between vectors and scalars or between vectors. This function covers various basic computing types and many customized computing types, including computing of data types such as FP16, FP32, INT32, and INT8.
- Scalar computing unit: Equivalent to a micro CPU, the scalar unit controls the running of the entire AI core. It implements loop control and branch judgment for the entire program, and provides the computing of data addresses and related parameters for cubes or vectors as well as basic arithmetic operations.

HUAWEI

Da Vinci Architecture (AI Core) - Computing Unit



HUAWEI

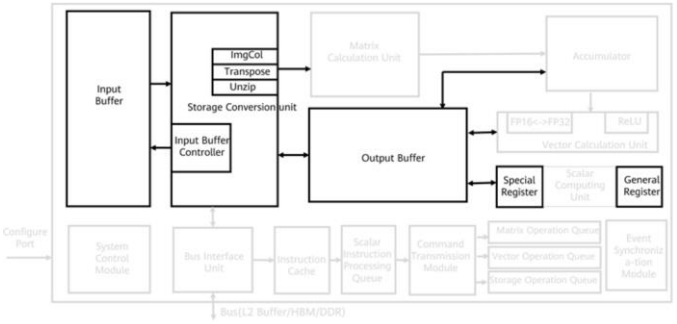
Da Vinci Architecture (AI Core) - Storage System (1)

- The storage system of the AI core is composed of the storage unit and corresponding data channel.
- The storage unit consists of the storage control unit, buffer, and registers:
- Storage control unit: The cache at a lower level than the AI core can be directly accessed through the bus interface. The memory can also be directly accessed through the DDR or HBM. A storage conversion unit is set as a transmission controller of the internal data channel of the AI core to implement read/write management of internal data of the AI core between different buffers. It also completes a series of format conversion operations, such as zero padding, Img2Col, transposing, and decompression.
- Input buffer: The buffer temporarily stores the data that needs to be frequently used so the data does not need to be read from the AI core through the bus interface each time. This mode reduces the frequency of data access on the bus and the risk of bus congestion, thereby reducing power consumption and improving performance.
- Output buffer: The buffer stores the intermediate results of computing at each layer in the neural network, so that the data can be easily obtained for next-layer computing. Reading data through the bus involves low bandwidth and long latency, whereas using the output buffer greatly improves the computing efficiency.
- Register: Various registers in the AI core are mainly used by the scalar unit.

HUAWEI

Da Vinci Architecture (AI Core) - Storage System (2)

- Data channel: path for data flowing in the AI core during execution of computing tasks
  - A data channel of the Da Vinci architecture is characterized by multiple-input single-output. Considering various types and a large quantity of input data in the computing process on the neural network, parallel inputs can improve data inflow efficiency. On the contrary, only an output feature matrix is generated after multiple types of input data are processed. The data channel with a single output of data reduces the use of chip hardware resources.



HUAWEI

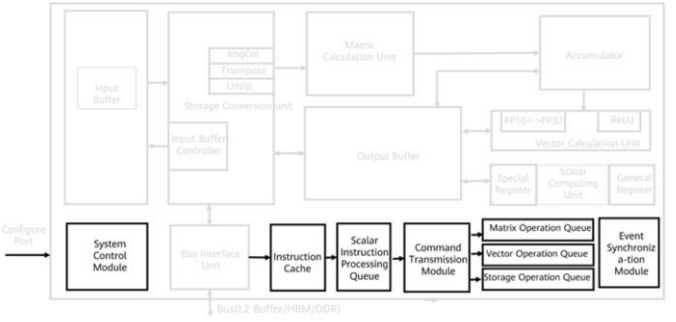
Da Vinci Architecture (AI Core) - Control Unit (1)

- The control unit consists of the system control module, instruction cache, scalar instruction processing queue, instruction transmitting module, matrix operation queue, vector operation queue, storage conversion queue, and event synchronization module.
  - System control module: Controls the execution process of a task block (minimum task computing granularity for the AI core). After the task block is executed, the system control module processes the interruption and reports the status. If an error occurs during the execution, the error status is reported to the task scheduler.
  - Instruction cache: Prefetches subsequent instructions in advance during instruction execution and reads multiple instructions into the cache at a time, improving instruction execution efficiency.
  - Scalar instruction procession queue: After being decoded, the instructions are imported into a scalar queue to implement address decoding and operation control. The instructions include matrix computing instructions, vector calculation instructions, and storage conversion instructions.
  - Instruction transmitting module: Reads the configured instruction addresses and decoded parameters in the scalar instruction queue, and sends them to the corresponding instruction execution queue according to the instruction type. The scalar instructions reside in the scalar instruction processing queue for subsequent execution.

HUAWEI

Da Vinci Architecture (AI Core) - Control Unit (2)

- Instruction execution queue: Includes a matrix operation queue, vector operation queue, and storage conversion queue. Different instructions enter corresponding operation queues, and instructions in the queues are executed according to the entry sequence.
- Event synchronization module: Controls the execution status of each instruction pipeline in real time, and analyzes dependence relationships between different pipelines to resolve problems of data dependence and synchronization between instruction pipelines.



HUAWEI

Section Summary

- This section describes the hardware architecture of Ascend chips, including the computing unit, storage unit, and control unit of the core Da Vinci architecture.

HUAWEI

Contents

HUAWEI

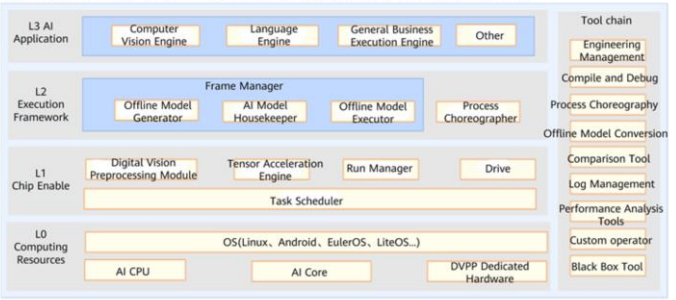
Overview and Objectives

- This section describes the software architecture of Ascend chips, including the logic architecture and neural network software flow of Ascend AI processors.

HUAWEI

Logic Architecture of Ascend AI Processor Software Stack (1)

- L3 application enabling layer: It is an application-level encapsulation layer that provides different processing algorithms for specific application fields. L3 provides various fields with computing and processing engines. It can directly use the framework scheduling capability provided by L2 to generate corresponding NNs and implement specific engine functions.
  - Generic engine: provides the generic neural network inference capability.
  - Computer vision engine: encapsulates video or image processing algorithms.
  - Language and text engine: encapsulates basic processing algorithms for voice and text data.



HUAWEI

- The software stack of the Ascend AI chip consists of four layers and an auxiliary toolchain. The four layers are the application enabling layer (L3), execution framework layer (L2), chip enabling layer (L1), and computing resource layer (L0). The toolchain provides auxiliary capabilities such as program development, compilation and commissioning, application process orchestration, log management, and profiling. The functions of the main components depend on each other in the software stack. They carry data flows, computing flows, and control flows.

Logic Architecture of Ascend AI Processor Software Stack (2)

- L2 execution framework layer: encapsulates the framework calling capability and offline model generation capability. After the application algorithm is developed and encapsulated into an engine at L3, L2 calls the appropriate deep learning framework, such as Caffe or TensorFlow, based on the features of the algorithm to obtain the neural network of the corresponding function, and generates an offline model through the framework manager. After L2 converts the original neural network model into an offline model that can be executed on Ascend AI chips, the offline model executor (OME) transfers the offline model to Layer 1 for task allocation.
- L1 chip enabling layer: bridges the offline model to Ascend AI chips. L1 accelerates the offline model for different computing tasks via libraries. Nearest to the bottom-layer computing resources, L1 outputs operator-layer tasks to the hardware.
- L0 computing resource layer: provides computing resources and executes specific computing tasks. It is the hardware computing basis of the Ascend AI chip.

- L2 execution framework layer: encapsulates the framework calling capability and offline model generation capability. After the application algorithm is developed and encapsulated into an engine at L3, L2 calls the appropriate deep learning framework, such as Caffe or TensorFlow, based on the features of the algorithm to obtain the neural network of the corresponding function, and generates an offline model through the framework manager. After L2 converts the original neural network model into an offline model that can be executed on Ascend AI chips, the offline model executor (OME) transfers the offline model to Layer 1 for task allocation.

- Online framework: uses a mainstream deep learning open source framework (such as Caffe/TensorFlow). Offline model conversion and loading enable the framework to accelerate computing on the Ascend AI chip.
- Offline framework: provides the offline generation and execution capabilities of the neural network, which enables the offline model to have the same capabilities (mainly the inference capability) without the deep learning framework.
- Framework manager: includes the offline model generator (OMG), offline model executor (OME), and offline model inference interface; supports model generation, loading, unloading, inference, and computing.
- OMG: converts the model files and weight files generated in the Caffe or TensorFlow framework into offline model files, which can be independently executed on the Ascend AI chip.

Contents

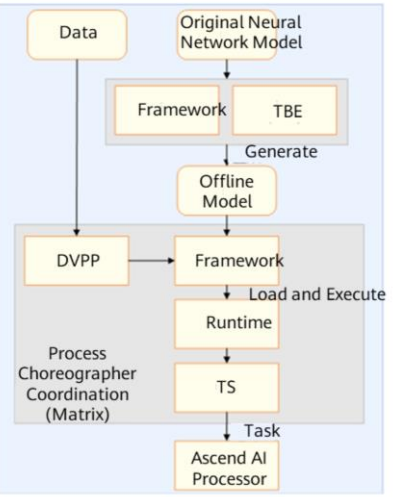
HUAWEI

Neural Network Software Flow of Ascend AI Processors

- The neural network software flow of Ascend AI processors is a bridge between the deep learning framework and Ascend AI chips. It realizes and executes a neural network application and integrates the following functional modules.
- Process orchestrator: implements the neural network on Ascend AI chips, coordinates the whole process of effecting the neural network, and controls the loading and execution of offline models.
- Digital vision pre-processing (DVPP) module: performs data processing and cleaning before input to meet format requirements for computing.
- Tensor boosting engine (TBE): functions as a neural network operator factory that provides powerful computing operators for neural network models.
- Framework manager: builds an original neural network model into a form supported by Ascend AI chips, and integrates the new model into Ascend AI chips to ensure efficient running of the neural network.
- Runtime manager: provides various resource management paths for task delivery and allocation of the neural network.

HUAWEI

Neural Network Software Flow of Ascend AI Processors



HUAWEI

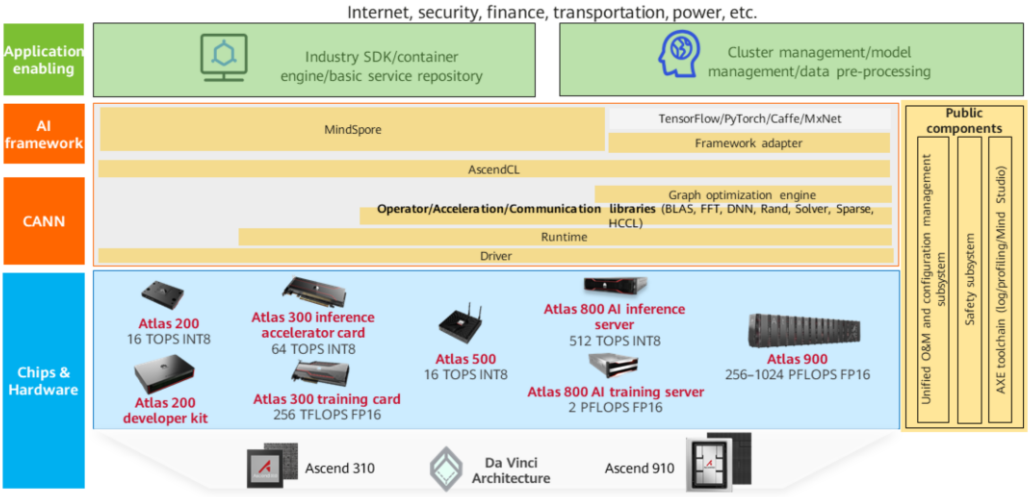
Contents

HUAWEI

Overview and Objectives

- This section describes the main products of Huawei Atlas AI computing platform, including inference and training.

Atlas AI Computing Platform Portfolio



HUAWEI

Atlas Accelerates AI Inference

Ascend 310
AI processor

Performance improved 7x for terminal devices

Atlas 200 Developer Kit (DK) AI developer kit
Model: 3000

Atlas 200 AI accelerator module
Model: 3000

Highest density in the industry (64-channel) for video inference

Atlas 300 AI accelerator card
Model: 3000

Edge intelligence and cloud-edge collaboration

Atlas 500 AI edge station
Model: 3000

Powerful computing platform for AI inference

Atlas 800 AI server
Model: 3000/3010

HUAWEI

Atlas 200DK: Strong Computing Power and Ease-of-Use

Full-Stack AI development on and off the cloud

- **16TOPS INT8 24W**
- **1 USB type-C, 2 camera ports, 1 GE port, 1 SD card slot**
- **8 GB memory**
- **Operating temperature: 0° C to 45° C**
- **Dimensions (H x W x D): 24 mm x 125 mm x 80 mm**

Developers

Set up a dev environment with one laptop
Ultra low cost for local independent environment, with multiple functions and interfaces to meet basic requirements

Researchers

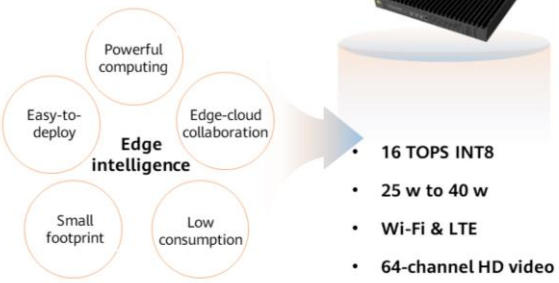
Local dev + cloud training collaboration
Same protocol stack for Huawei Cloud and the developer kit; training on the cloud and deployment at local; no modification required

Startups

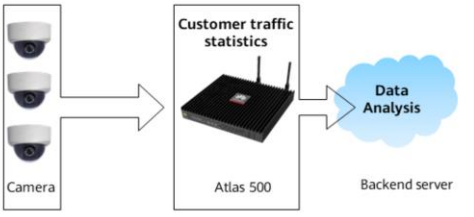
Code-level demo
Implementing the algorithm function by modifying 10% code based on the reference architecture; interaction with the Developer Community; seamless migration of commercial products

HUAWEI

Atlas 500 AI Edge Station



- 16 TOPS INT8
- 25 w to 40 w
- Wi-Fi & LTE
- 64-channel HD video analytics in real time
- Fanless design, stable from -40° C to +70° C



HUAWEI

Atlas 800 AI Server

Model: 3000

Model: 3010

- **An efficient inference platform powered by Kunpeng**
- **Key functions:**
  - 2 Kunpeng 920 processors in a 2U space
  - 8 PCIe slots, supporting up to 8 Atlas 300 AI accelerator cards
  - Up to 512-channel HD video real-time analytics
  - Air-cooled, stable at 5° C to 40° C
- **A flexible inference platform powered by Intel**
- **Key functions:**
  - 2 Intel® Xeon® SP Skylake or Cascade Lake processors in a 2U space
  - 8 PCIe slots, supporting up to 7 Atlas 300/NVIDIA T4 AI accelerator cards
  - Up to 448-channel HD video real-time analytics
  - Air-cooled, stable at 5° C to 35℃

HUAWEI

Atlas Accelerates AI Training

Ascend 910
AI processor

Training card with ultimate computing power

Atlas 300 AI accelerator card
Model: 9000

World's most powerful training server

Atlas 800 AI server
Model: 9000/9010

World's fastest AI training cluster

Atlas 900 AI cluster

HUAWEI

Atlas 300 AI Accelerator Card: Highest-Performing Accelerator Card for AI Training

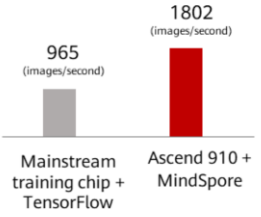
Atlas 300

Model: 9000

2x ↑

Computing power per card

256T FLOPS FP16



70% ↓

Gradient synchronization latency

Direct 100G RoCE

- **Test benchmark:**
  - ResNet 50 V1.5
  - ImageNet 2012
  - Optimal batch size respectively

HUAWEI

Atlas 800 Training Server: Industry's Most Powerful Server for AI Training

Atlas 800

Model: 9000

2.5x ↑

Density of computing power

2P FLOPS/4U

25x ↑

Hardware decoder

16384 (1080p decoding) images/second

1.8x ↑

Perf./Watt

2P FLOPS/5.6kW

HUAWEI

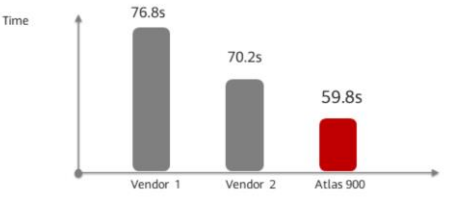
Atlas 900 AI Cluster: Fastest Cluster for AI Training

Atlas 900

256-1024 PFLOPS FP16

Industry-leading computing power | Best cluster network | Ultimate heat dissipation

Shortest time consumption: 59.8s



- **Test benchmark:**
  - Benchmark: ResNet-50 V1.5 model, ImageNet-1k dataset
  - Cluster: 1024 Ascend 910 AI processors
  - Accuracy: 75.9%

HUAWEI

Atlas Deep Learning System Accelerates AI Model Training and Builds Extensive Applications

Atlas 300 training acceleration card
Model: 9000

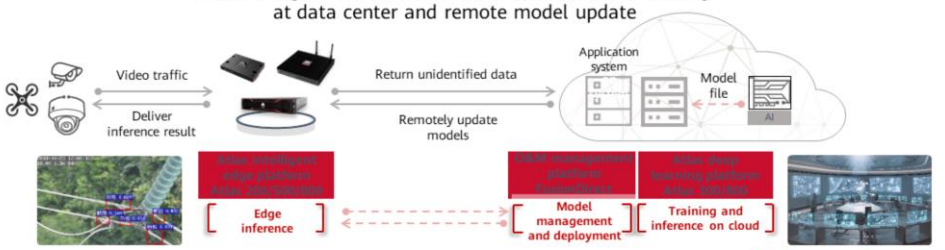
Atlas 800 AI training server
Model: 9000

Atlas 900 AI cluster

HUAWEI

Device-Edge-Cloud Collaboration Enables the Ultimate Development and User Experience

Device-edge-cloud collaboration for continuous training at data center and remote model update



	Centralized development	Centralized O&M	Enhanced security
Atlas	• Centralized development architecture based on Da Vinci and CANN, **develop once, deploy everywhere**	• FusionDirector manages up to **50,000 nodes**, **manages central and edge devices**, and remotely pushes models and upgrades devices	• Transmission channel encryption • **Model encryption, double assurance**
Industry solution	• Edge and data centers use different development architectures. **Models cannot transfer freely, requiring secondary development.**	• **No O&M management tools**; provides only APIs, so customers need to develop APIs by themselves.	• **No encryption/decryption engine**; models are not encrypted.

HUAWEI

Section Summary

- This section introduces products of Huawei Atlas AI computing platform, mainly inference products including Atlas 200 DK, Atlas 200 AI accelerator module, Atlas 300 AI accelerator card, Atlas 500 AI edge station, and Atlas 800 AI server. The computing platforms used for training include Atlas 300 AI accelerator card, Atlas 800 AI server, and Atlas 900 AI cluster.

HUAWEI

Contents

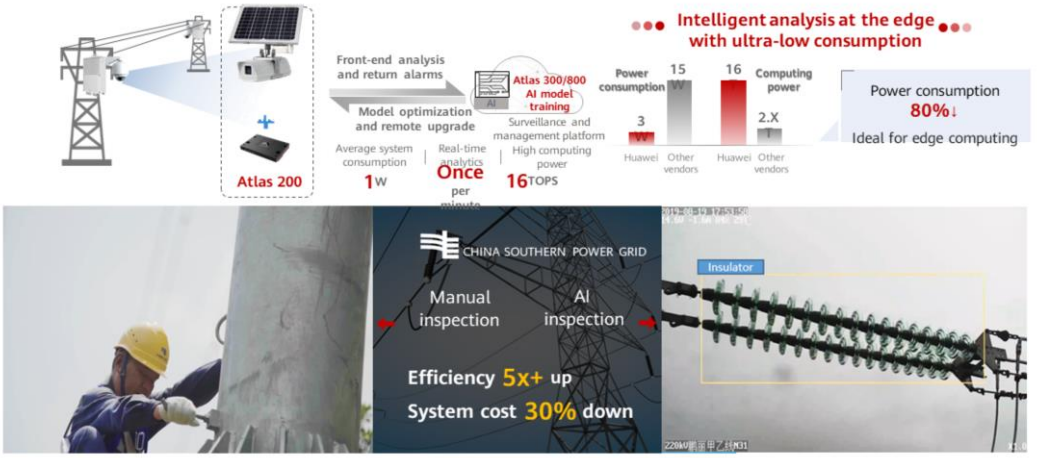
HUAWEI

Overview and Objectives

- This section introduces industry application scenarios of the Atlas AI computing platform.

HUAWEI

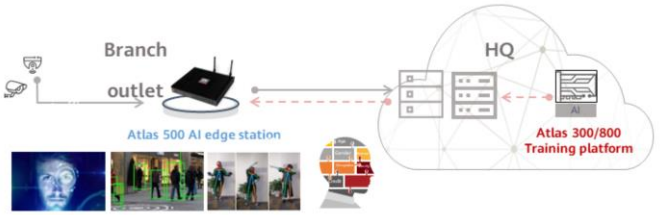
Electric Power: Industry's First Intelligent Unattended Inspection Solution, with 5x Efficiency



HUAWEI

Finance: AI Enables Smart Branches for Banks

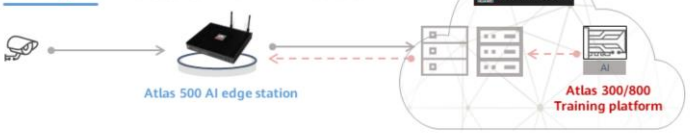
Past: Customer experience needs improvement **Now:** cloud-edge collaboration, smart finance



HUAWEI

Manufacturing: AI Enables Intelligent Production Line with Machine Vision

Cloud-edge collaboration enables intelligent EL detection for PV modules



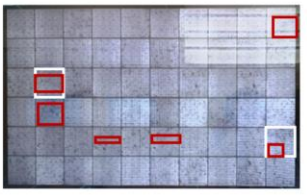
Manual inspection

Unstable results | Low production efficiency | Discontinuous process | High labor costs

Number of defective battery chips: *2*

Detection duration: *about 5s*

Accuracy: *33.33%*



Smart inspection

Zero omission | High production efficiency | Cloud-edge synergy | Reduced labor costs

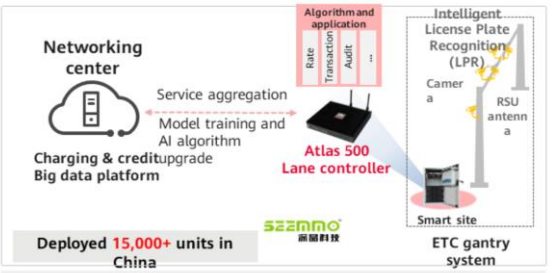
Defective battery chip: *6*

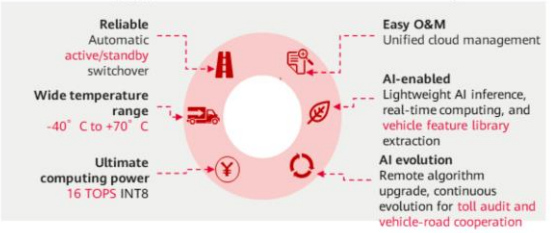
Detection duration: *1.36s*

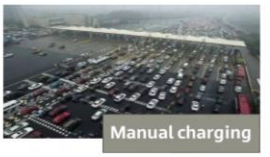
Accuracy: *100%*

HUAWEI

Transportation: AI Smooths Highways with 5x Efficiency Boost







- Low efficiency
- Long queuing time

- Passing efficiency 5X↑
  Saving energy and reducing emission

- Proactive security control
- Road cooperation management
- Autonomous vehicle driving

HUAWEI

Supercomputing: Atlas Helps PCL Build CloudBrain Phase II

Peng Cheng Laboratory (PCL)
Innovative basic platform for national missions

Application layer
Automated driving | City brain | Smart healthcare | Speech recognition | Natural language processing

Basic platform layer
Critical infrastructure for AI

Physical resource layer
Peng Cheng CloudBrain Phase I 
Peng Cheng CloudBrain Phase II 

Peng Cheng CloudBrain Phase II mainly built **Atlas 900, the world's fastest training cluster**

Ultimate computing power
Level-E AI computing power

Top cluster network
HCCL communication supports 100 TB/s non-blocking parameter plane networking

Ultimate energy efficiency
AI cluster PUE < 1.1

HUAWEI

Attract More Developers Based on the Ascend Developer Community

+

- **Developer-centric enabling platform**
- **https://ascend.huawei.com/home**

Annual technical salon	• Held in **10+** cities • 20+ senior trainers • Dozens of speeches
Developer contest	• **1,500+** teams • Annual prize of over RMB1 million • Equal opportunities for enterprises and universities
Developer support	• Public cloud vouchers • Free certification course tickets • Free Atlas developer kits

HUAWEI

Summary

- This chapter describes the products of Huawei Atlas computing platform and helps you to understand the working principles of Huawei Ascend chips. It focuses on the hardware and software architectures of Ascend chips and application scenarios of the Atlas AI computing platform.

HUAWEI

Quiz

1. What are the main applications of Ascend 310? ( )
   A. Model inference
   B. Model training
   C. Model building

HUAWEI

- Answer: A

Recommendations

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

HUAWEI

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.

Copyright©2020 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.

Huawei Open AI Platform for Smart Devices

HUAWEI

Foreword

- Huawei HiAI is an open artificial intelligence (AI) capability platform for smart devices, which adopts a "chip-device-cloud" architecture, opening up chip, app, and service capabilities for a fully intelligent ecosystem. This assists developers in delivering a better smart app experience for users by fully leveraging Huawei's powerful AI processing capabilities.

HUAWEI

Objectives

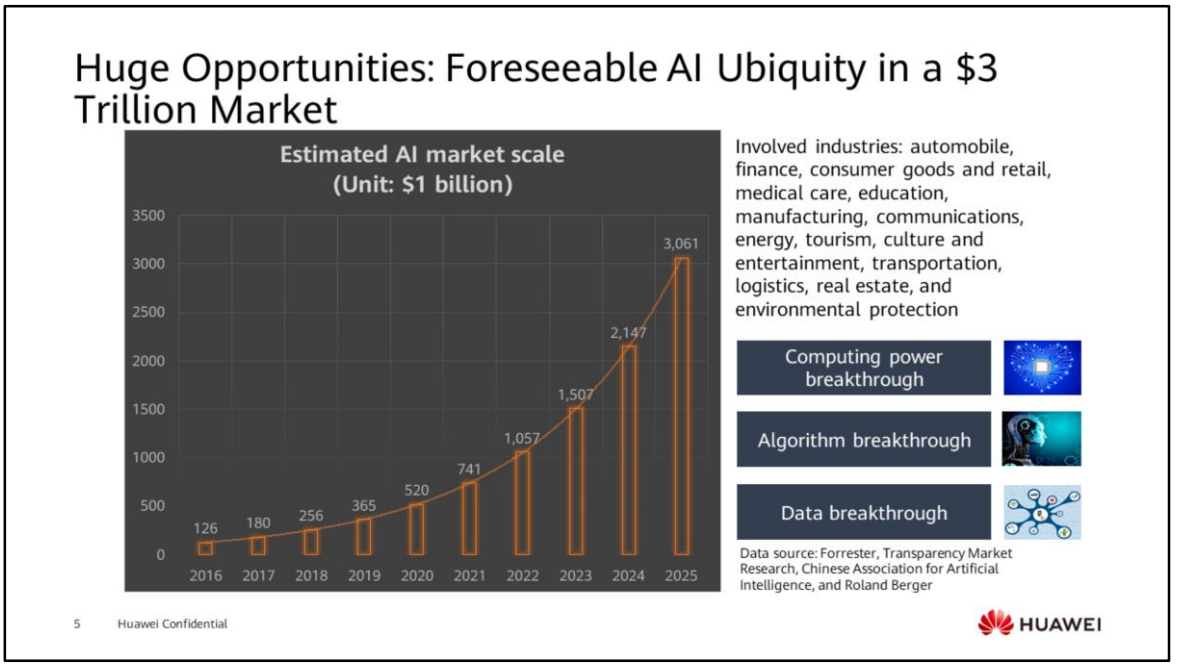
After this course, you will be able to:

- Master the usage of Huawei HiAI platform.
- Understand the powerful functions of Huawei HiAI platform.

HUAWEI

Contents

HUAWEI

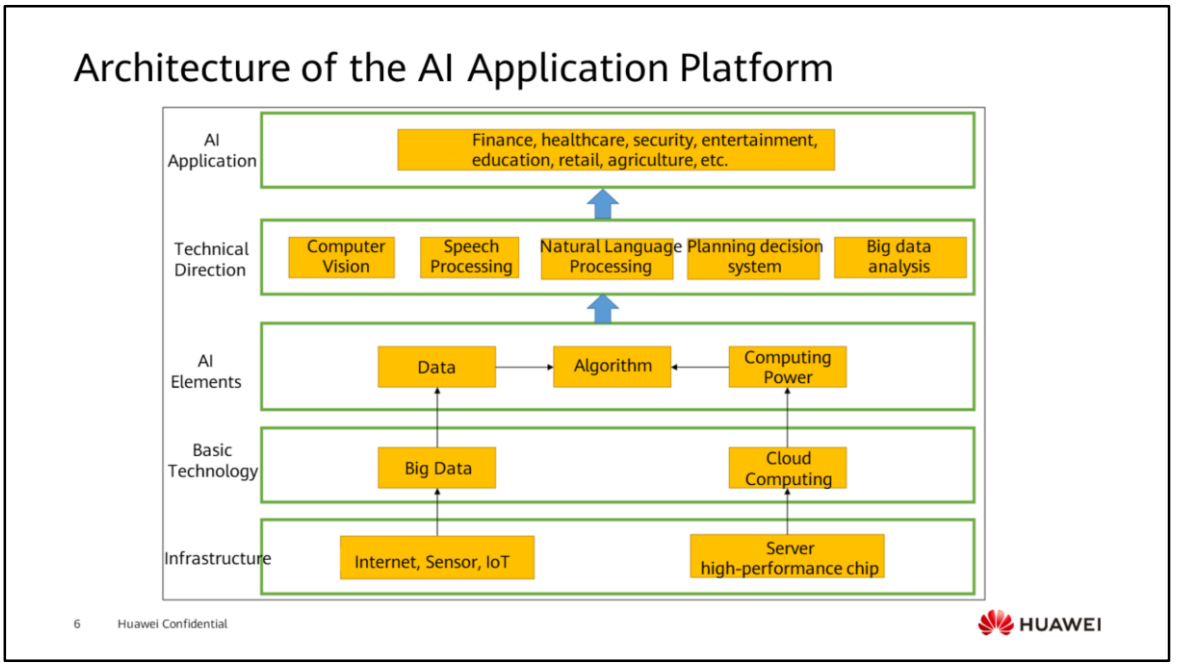


- In the past 50 years, we have experienced three AI upsurges, which were represented by man-machine games. The first one occurred in 1962 when the checkers program developed by Arthur Samuel from IBM beat the best checkers player in the United States. The second one occurred in 1997 when IBM Deep Blue beat Gary Kasparov, the world champion of chess, at 3.5:2.5. The third one broke out in 2016 when AlphaGo, a robot developed by Google DeepMind defeated the Go world champion Lee Sedol who is a player of 9 dan rank in South Korea.

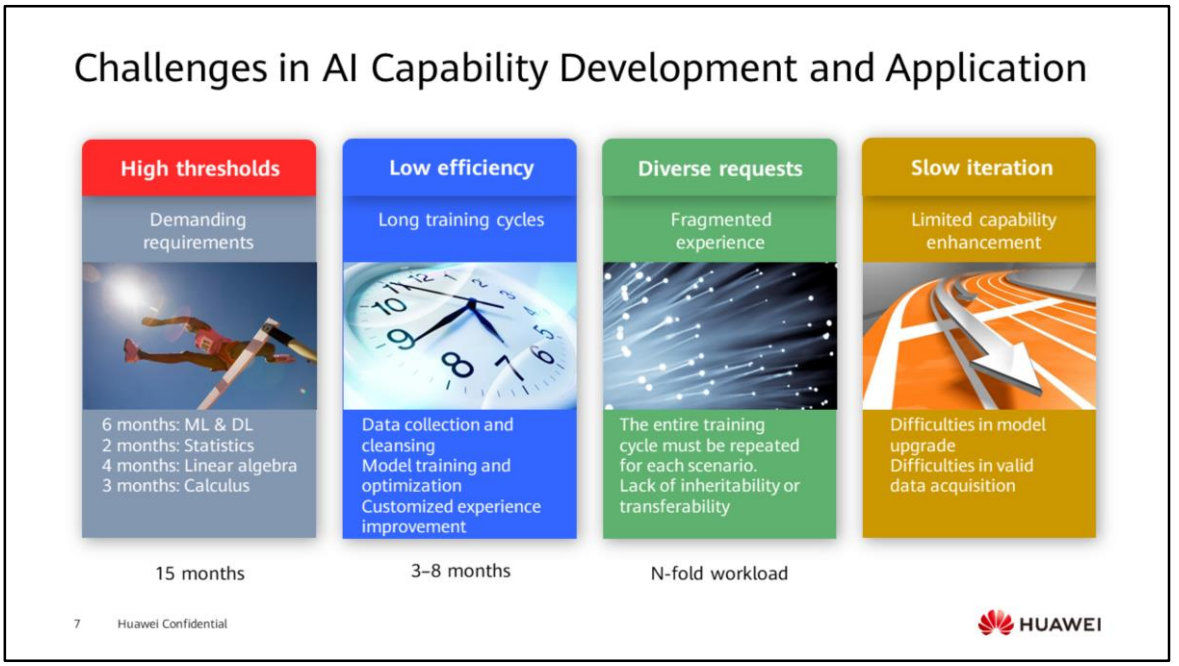
- Computing capability and chip improvement
- Enhancement of machine learning algorithm capabilities, such as AlphaGo
- Improvement in industry application and data quality

- Requirement: How to bridge the gap between requirements and capabilities?
- I. Opportunities
- II. Challenges
- 1. How does the device collaborate with the cloud-side AI? (Highlight the advantages of the device.)
- 2. How do developers develop open AI models?
- 3. How can developers quickly develop AI-based applications?

- 4. How can we help customers improve experience through AI?



- The four elements of AI are data, algorithm, computing power, and scenario. To meet requirements of these four elements, we need to combine AI with cloud computing, big data, and IoT to build an intelligent society.

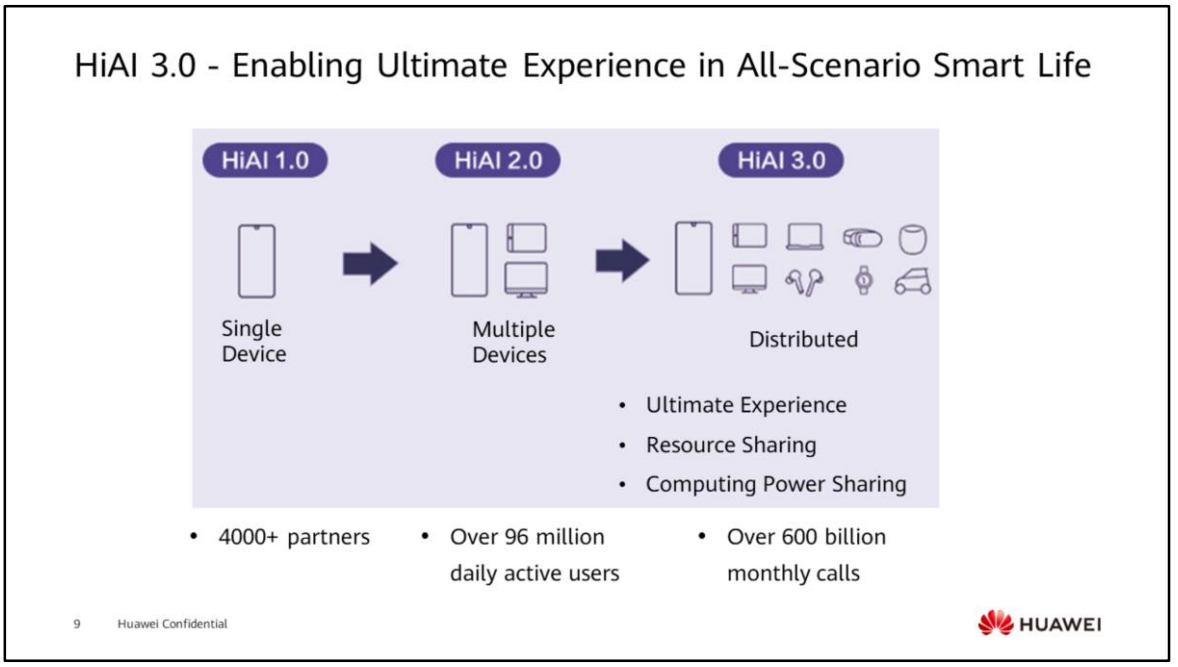


- Computing capability and chip improvement
- Enhancement of machine learning algorithm capabilities, such as AlphaGo
- Improvement in industry application and data quality

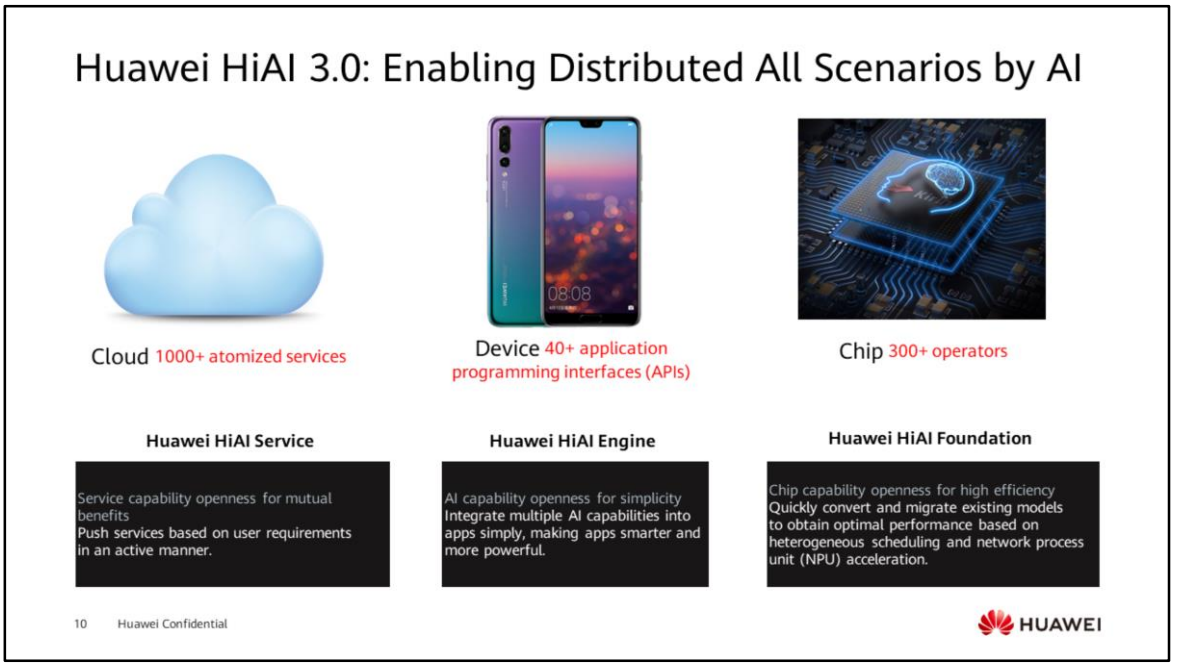
- Requirement: How to bridge the gap between requirements and capabilities?
- I. Opportunities
- II. Challenges
- 1. How does the device collaborate with the cloud-side AI? (Highlight the advantages of the device.)
- 2. How do developers develop open AI models?
- 3. How can developers quickly develop AI-based applications?
- 4. How can we help customers improve experience through AI?

Contents

HUAWEI



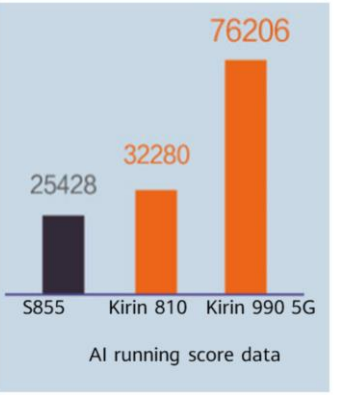
- On November 19, Dr. Wang Chenglu, President of Huawei Consumer BG Software Engineering Dept, officially released HUAWEI HiAI 3.0 at the Software Green Alliance Developer Conference. Evolving from device-side AI to distributed AI, what will HUAWEI HiAI 3.0 bring to all-scenario smart life? This time, we will define it with "superb".
- Currently, consumers are exposed to a large number of AI applications, such as voice assistant, AI photography, and image beautification. The application scenarios are limited. In fact, with the evolution from device-side AI to distributed AI, resource and computing power sharing among multiple devices will greatly expand application scenarios of device-side AI, further enabling developers to achieve more smart innovations and bringing superb experience to consumers.
- HUAWEI HiAI 3.0 provides the distributed computer vision (CV) and automatic speech recognition (ASR) capabilities, which can help people exercising at home achieve an effect similar to that under personal training guidance in the gym. The distributed CV can be used to identify key elements of a human body in 3D. User can capture motion postures of multiple angles in real time by using cameras at different locations, and correct the postures by using multiple screens. With the open ASR capability, users can use the smart speaker to control the movement pace through voice interaction and further assist consumers in personal training at home.
- As HUAWEI HiAI 3.0 is combined with distributed technologies, users can connect a smartphone to their car, use the camera inside the car to detect driving behavior, and use the AI chip computing power of the smartphone to remind them of dangerous behaviors such as fatigue driving. All these can be performed in an in-vehicle network environment. The lower-latency local data computing helps drivers better protect themselves.
- HUAWEI HiAI 3.0 provides one-time service access and multi-device adaption experience. Users can enjoy services, such as voice assistant and HiBoard on devices including the mobile phones, tablets, smart screens, and smart speakers.



- HiAI is an AI computing platform that is designed for mobile devices. It builds a three-layer AI ecosystem for greater openness of service, app, and chip capabilities. This three-layer open platform utilizes features of the chips, devices, and clouds to deliver an extraordinary experience to both users and developers.
- Cloud: created once and reused multiple times;
- Device: distributed and all-scenario;
- Chip: stronger computing power, more operators and frameworks, and smaller models.

HiAI Foundation

- HiAI Foundation APIs constitute an AI computing library of a mobile computing platform, enabling developers to efficiently compile AI apps that can run on mobile devices.
  - Leveraging high performance and high precision of Kirin chips, better device-end AI performance will be delivered by more powerful computing power.
  - Support the largest number of operators (300+) in the industry and more frameworks, greatly improving flexibility and compatibility.
  - The Honghu, Kirin, and AI camera chips enable AI capabilities for more devices.



HUAWEI

HiAI Engine

- HiAI Engine opens app capabilities and integrates multiple AI capabilities into apps, making apps smarter and more powerful.
  - Provide handwriting recognition and dynamic gesture recognition capabilities, with 40+ underlying APIs.
  - Computer vision and speech recognition will develop toward a distributed mode, assisting developers in delivering more all-scenario smart life experience.

Distributed AI helps sports healthy and safe driving

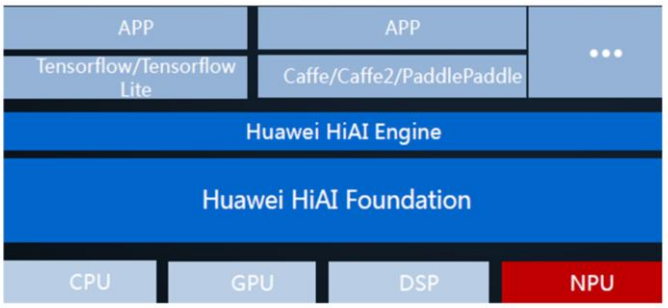
HUAWEI

HiAI Service

- HiAI Service enables developers to reuse services on multiple devices, such as mobile phones, tablets, and large screens, with only one service access, efficiently implementing distribution.

HUAWEI

Architecture of the HiAI Mobile Computing Platform



Supports diverse mainstream front-end frameworks.

Provides various upper-layer function service APIs to ensure efficient running on mobile devices.

Allows flexible scheduling of heterogeneous resources, meeting developers' demand to accelerate neural network model computing and operator computing

Tool chain

Comprehensive documents

Different types of APIs

Source codes enabling quick start

HUAWEI

What can apps benefit from Huawei HiAI?

Real time

Ready-to-use

Stability

Security

Lower cost

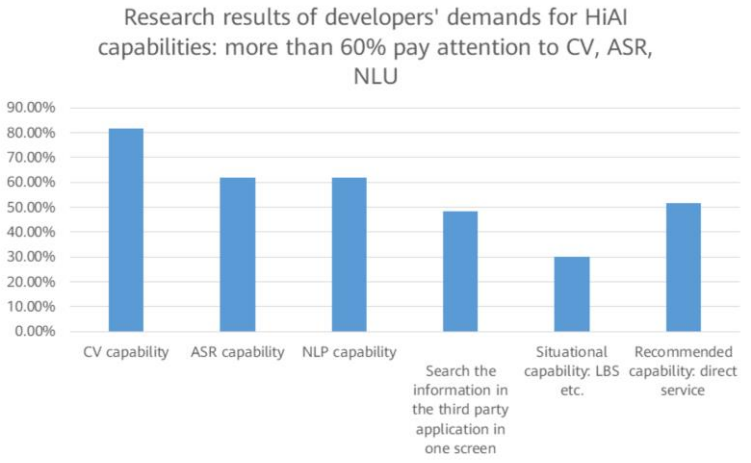
HUAWEI

Huawei HiAI + Ctrip Help You Take Poetic Photos

HUAWEI

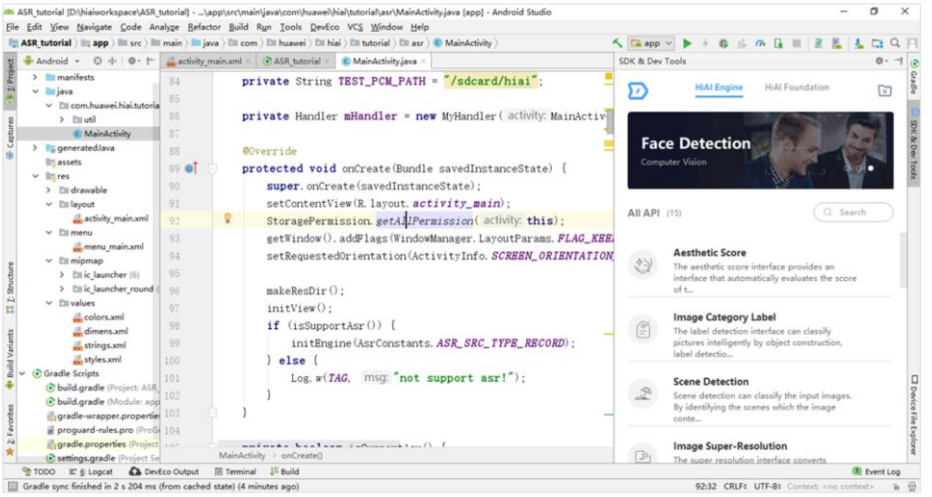
-

AI Capability Provider, Accelerating Application Development



HUAWEI

Comprehensive Tools for Developers



HUAWEI

Contents

HUAWEI

HUAWEI

Huawei HiAI Helps Deaf and Mute People

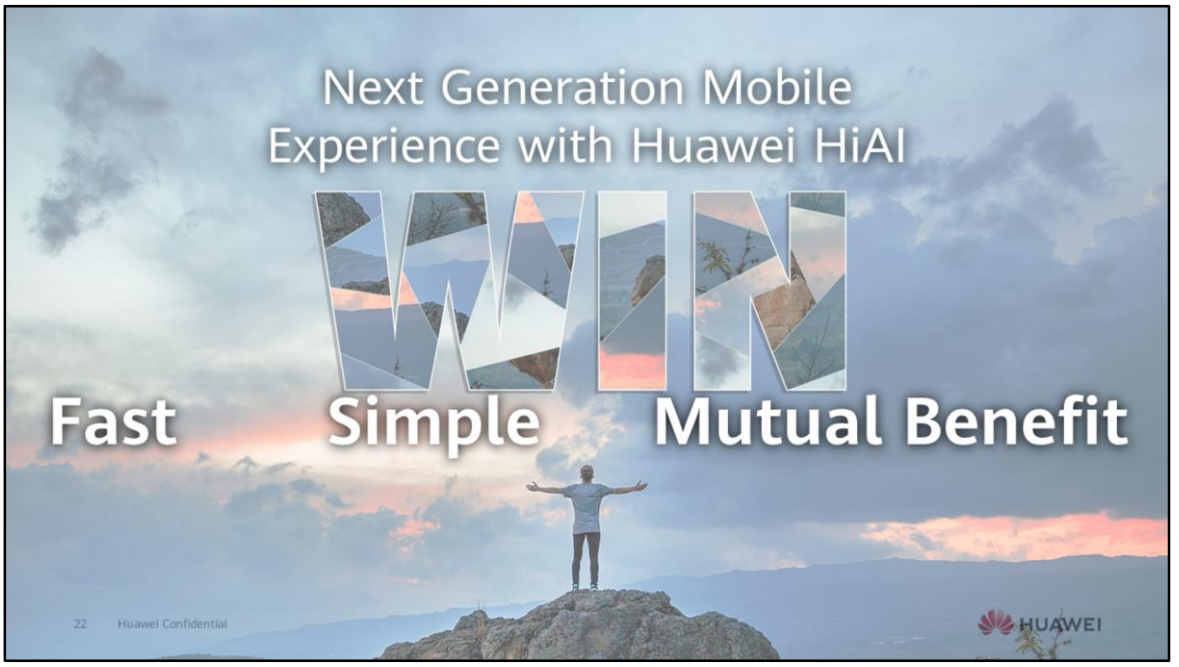
HUAWEI

- The picture is from The Silent Child, a film that won the Oscar for Best Live Action Short Film in 2018. This film is about a deaf girl called Libby, played by Maisie Sly, a little girl with hearing impairments in real life. Her family's hearing was normal. She was always the one that was ignored at home due to lack of interaction with her family. In school, she often sat alone. Even though she wants to communicate with others, she failed to make any friend.
- Children with hearing disabilities cannot enjoy a good time because of physical obstacles. They cannot hear the greetings of their relatives and friends, nor can they read the words in the books. The world is silent and lonely to them. There are about 32 million deaf children around the world. They cannot hear the wonderful voice or verbally express their ideas. The way they communicate with the world is full of challenges.
- It is cruel that 90% of children with hearing disabilities have healthy parents, 78% of whom are unable to communicate with their children. For children who are severely or profoundly deaf, learning and reading can be an overwhelming challenge.

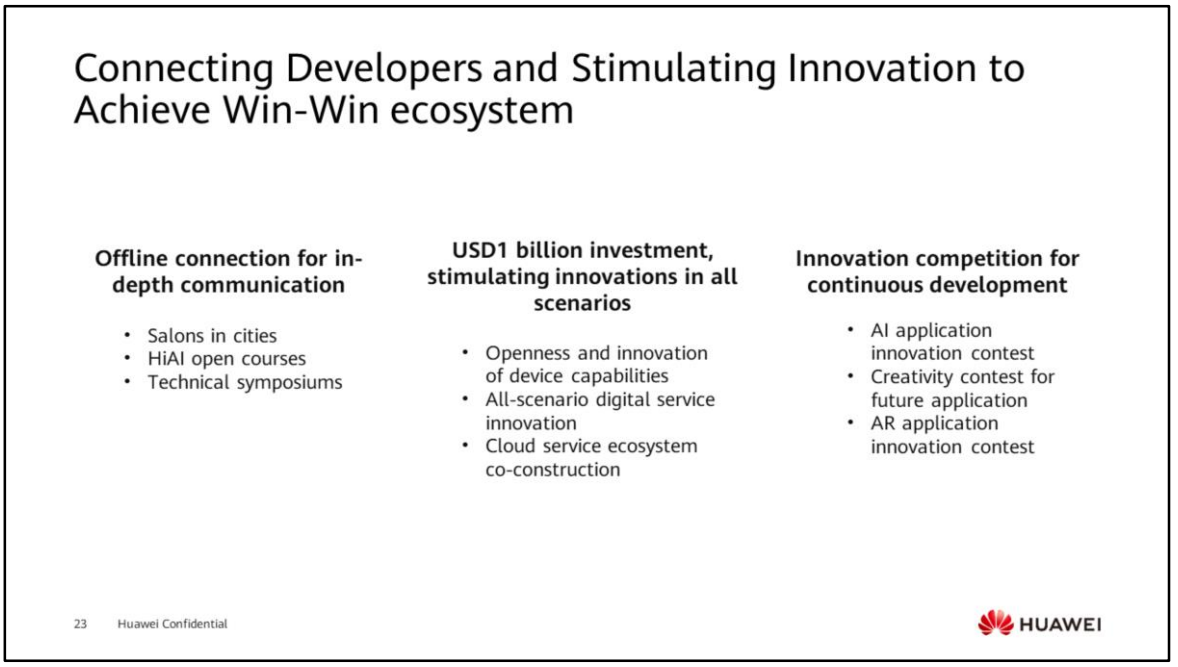
- Languages are learned by listening, speaking, reading, and writing. Listening is a key part for language learning. When encountering a strange word, a normal child can understand its meaning with their family's explanation, and master it by speaking the word continuously. Children with hearing disabilities, however, can only learn languages through sign language. Without the help of professional sign language teachers, they cannot communicate with normal people.

- To address this issue, Huawei developed StorySign in partnership with the European Union of the Deaf, the British Deaf Association, Penguin Books and Aardman Animations among others. With Huawei HiAI's open image recognition and optical character recognition (OCR) capabilities, the lovely Star translates a

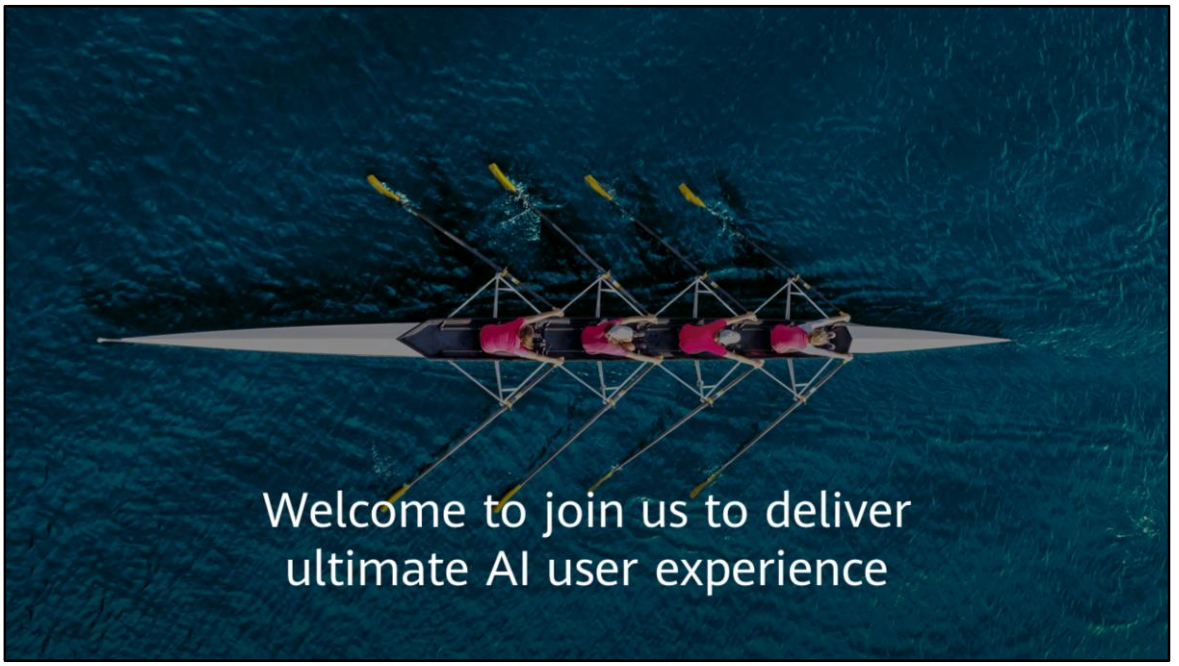
text into sign language when the mobile phone reads the text from a book.



- Users with AI development capabilities can use HiAI Foundation.
- Users without AI development capabilities can use HiAI Engine.
- Common apps can act as service providers to use HiAI Service.



- According to the official introduction, Huawei has more than 4,000 HiAI partners, more than 96 million daily active users, and more than 600 billion monthly calls.


Welcome to join us to deliver
ultimate AI user experience

Summary

- We believe that AI can make life better by bringing unprecedented convenience for both back end and devices. However, this requires actual application scenarios that allow more enterprises and developers to play a part in improving user experience substantially. Huawei is willing to work with partners to jointly promote intelligent transformation of industries with more developers and enterprises based on the HiAI3.0 platform.

HUAWEI

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.

Copyright©2020 Huawei Technologies Co., Ltd.
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.

HUAWEI CLOUD Enterprise Intelligence Application Platform

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Objectives

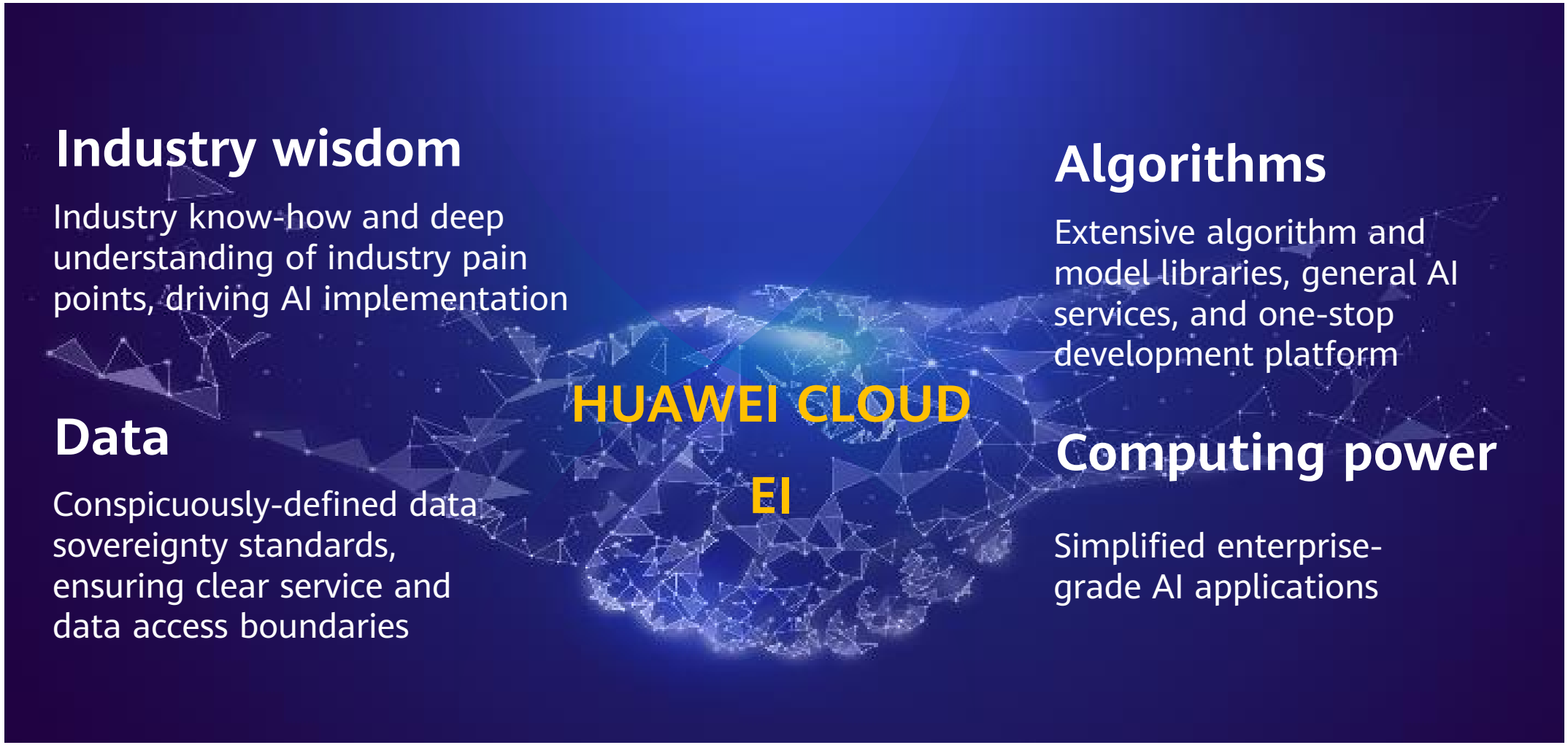
- On completion of this course, you will be able to:
  - Understand the HUAWEI CLOUD Enterprise Intelligence (EI) ecosystem and EI services.
  - Understand the Huawei ModelArts platform and know how to use it.
  - Understand the application fields of HUAWEI CLOUD EI.

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Contents

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

HUAWEI CLOUD EI



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

HUAWEI CLOUD EI

General APIs

ASR TTS Image NLP

OCR Face Moderation AIS

Advanced APIs

CBS ImageSearch VCM VCT

IDS VGS VCC VCR

Pre-integrated solutions


City


Internet


Home


Internet of Vehicles (IoV)


Logistics


Healthcare


Campus


Manufacturing

Essential platform services

ModelArts ExeML GES DLS MLS Batch UPredict RLS

MindSpore TensorFlow mxnet PyTorch Keras

Ascend Ascend 310 Ascend 910

Kunpeng

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Development of HUAWEI CLOUD EI

- 59 cloud services and 159 functions
- Multi-domain Intelligent Twins
- Dedicated to inclusive AI

- IDC: Huawei ranked No. 1 in China's big data market
- 190+ patents

- PMCs and committers of core projects
  1. Hadoop Core/HBase: 7
  2. Spark+CarbonData: 8
- CarbonData: top Apache project

Cloud EI services

Cloud EI services

Cloud EI services

Enterprise big data platform (FusionInsight)

Reliable and secure self-management

Telecom big data solution

(Telco industry)

Performance-oriented and equipment-based

Big data technology research

Hadoop kernel optimization and community contributions

Traditional BI (Telco industry)

ETL & analytics technology

AI technology research

AI practice: Focus on the inside of Huawei and support intelligent upgrade.

2002 2007 2011 2013 2015 2017 2019

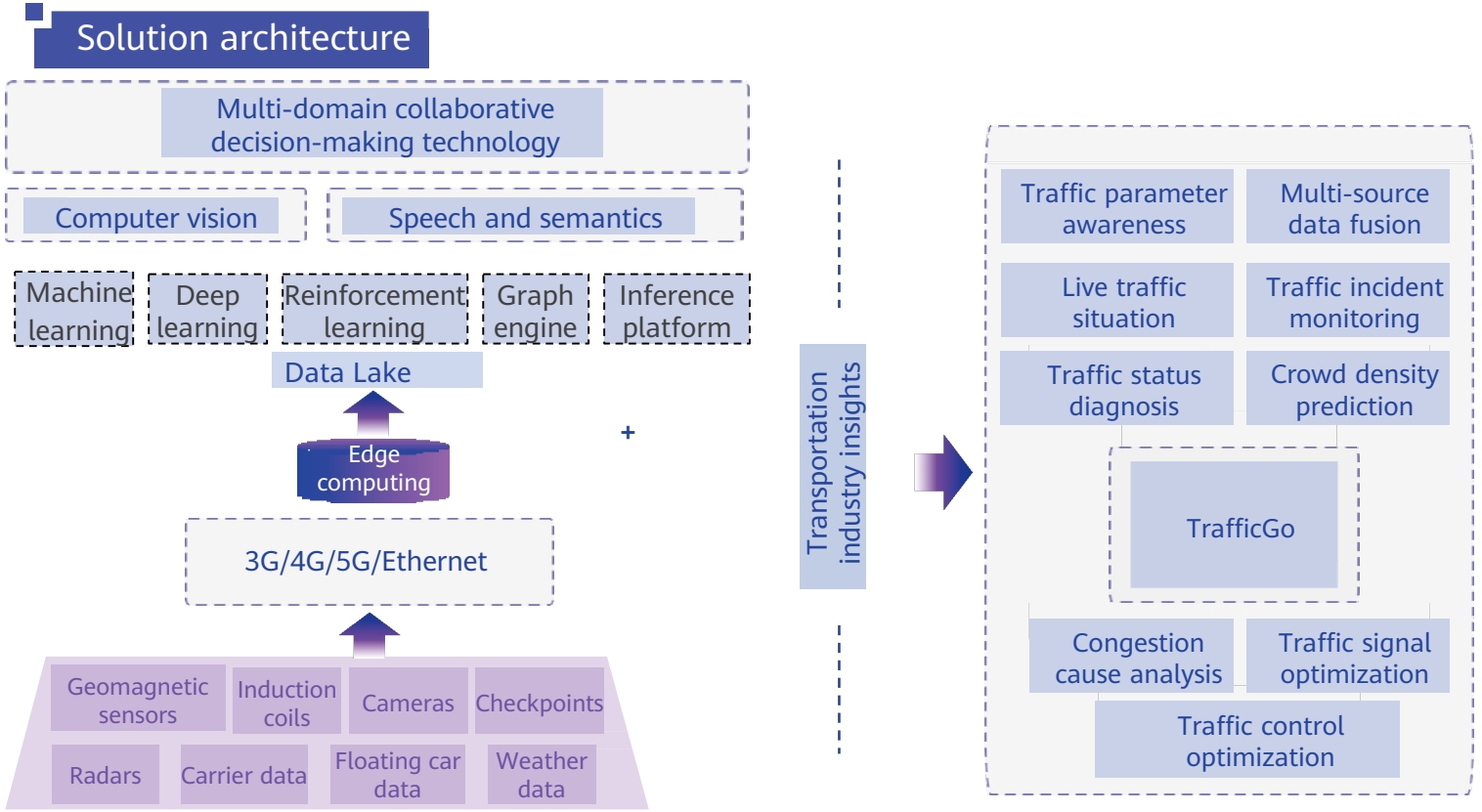
Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Contents

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

TrafficGo

- Traffic Intelligent Twins (TrafficGo) is a comprehensive urban traffic governance solution. Powered by the big data platform, AI algorithms, and expert experience in the transportation industry, TrafficGo builds a network for urban traffic governance to implement intelligent awareness, diagnosis, and optimization in all domains. TrafficGo enables 24/7 traffic condition monitoring in all areas, traffic incident detection, real-time regional traffic signal scheduling, traffic situation display, and key vehicle management. This makes transportation more efficient and safer while yielding new levels of energy-efficiency.

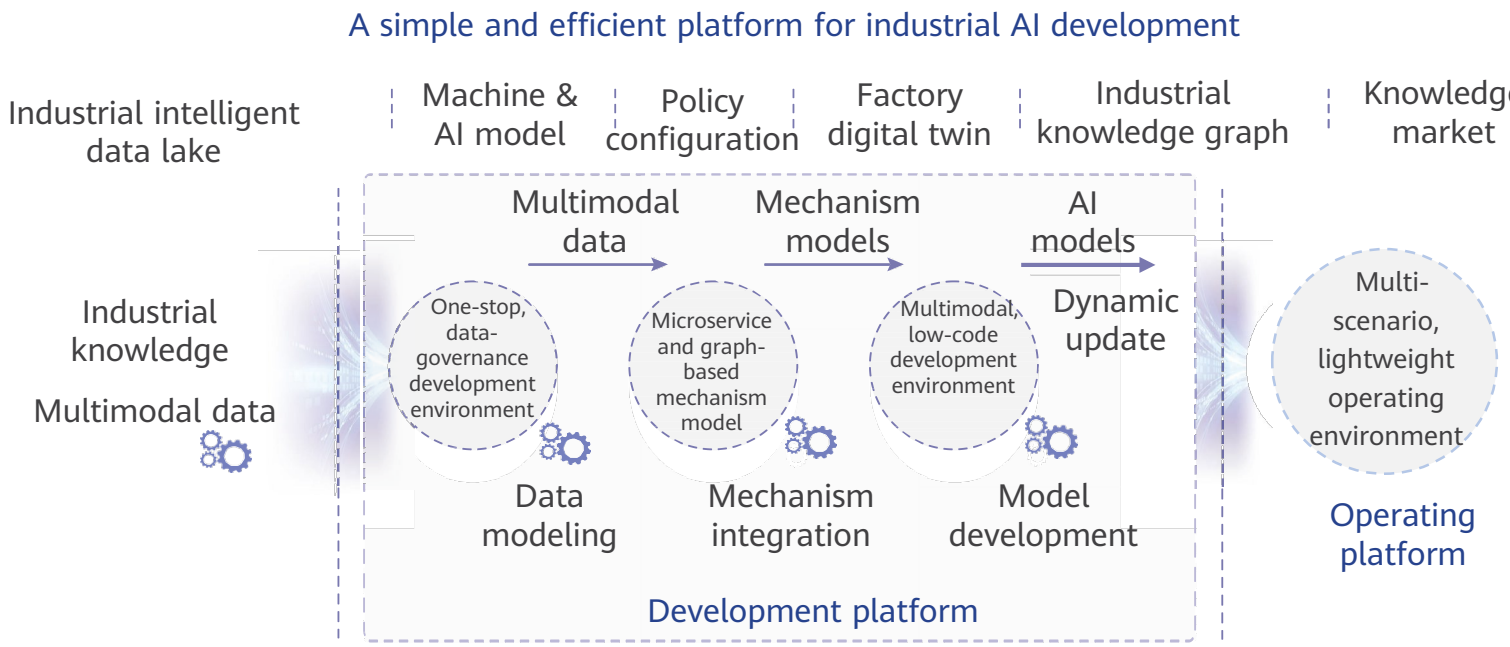


Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Industrial Intelligent Twins

- HUAWEI CLOUD Industrial Intelligent Twins builds an industrial intelligent platform that deeply integrates industrial knowledge and AI to adapt to frequent changes of working conditions and break the application limit of the industrial mechanism. In addition, Huawei works with industry-leading know-how customers and partners to build an open and win-win industrial intelligent ecosystem to make enterprises more intelligent and promote industrial upgrade.

Solution architecture



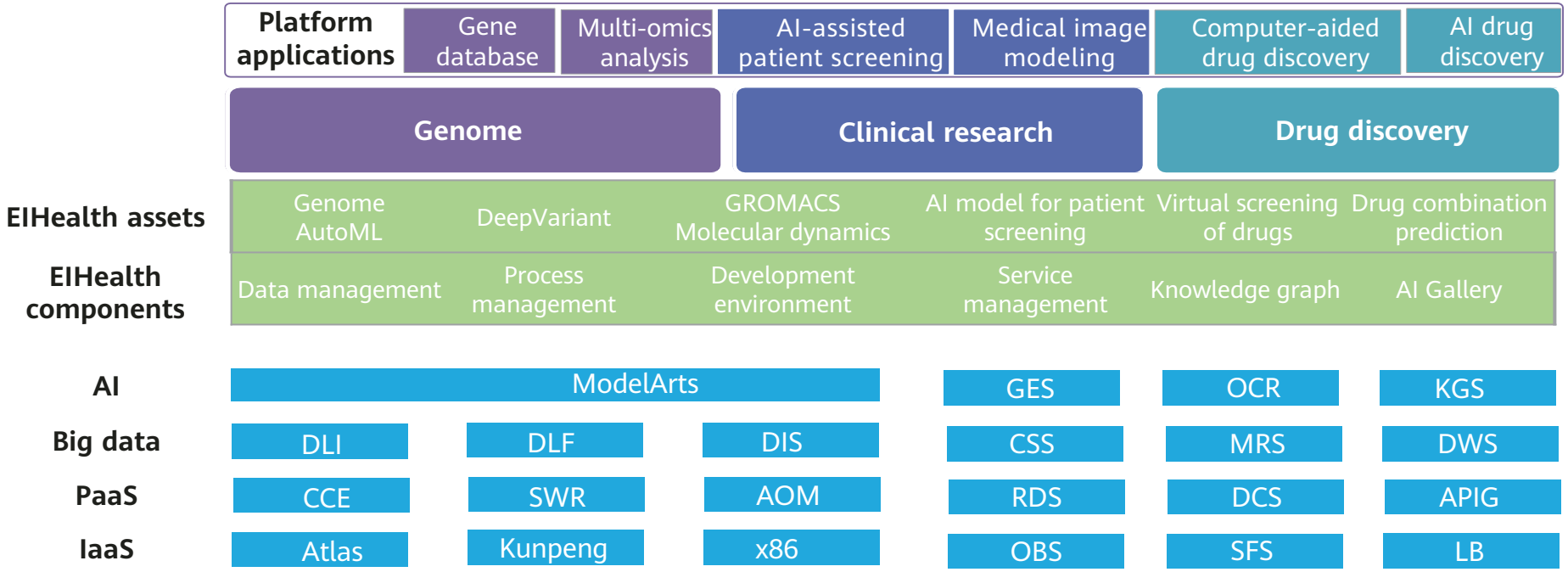
Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

EIHealth

- HUAWEI CLOUD EIHealth covers genome, clinical research, and drug discovery. It focuses on people's health and promotes the combination of AI and healthcare to improve the service capability and coverage of healthcare. With Huawei's powerful computing, storage, and AI algorithms, EIHealth enables the genome, disease diagnosis, and pharmaceutical fields.

Solution architecture

1 + 3 + X: 1 EIHealth cloud platform + 3 sub-domain platforms + X enterprise/organization customized platforms (delivered with ISVs)



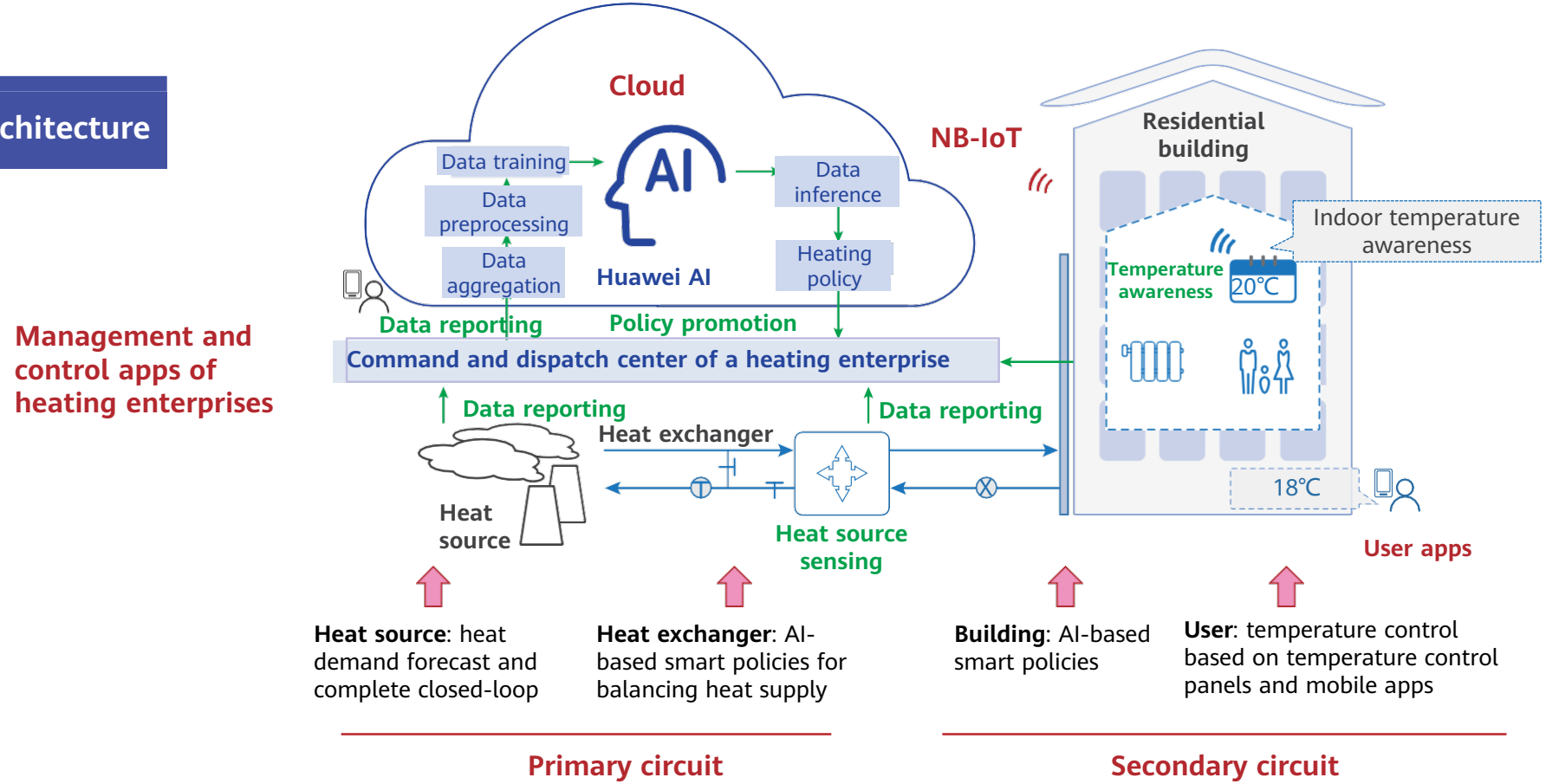
Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

HeatingGo

Based on the existing auto control facilities and technologies, HeatingGo uses technologies such as AI and big data to implement intelligent heat supply supervision, operations, and control for heat sources, networks, stations, and users.

It provides a heating system that is inclusive, inexpensive, and controllable.

Solution architecture

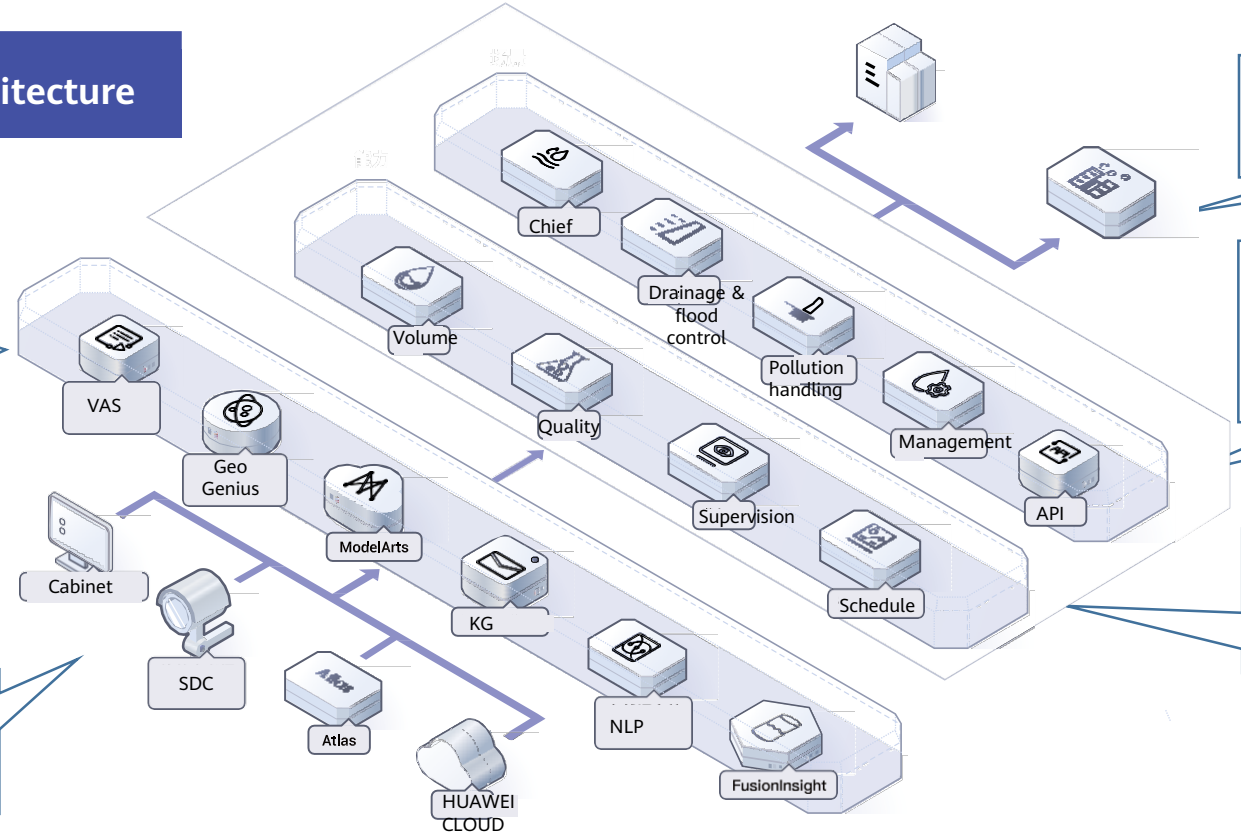


Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

WaterGo

With technologies including AI and edge computing, Huawei is transitioning cameras into universal sensors capable of sensing all elements related to water, such as its volume and quality, making them "eyes" for river and lake monitoring. The cameras are widely distributed, intelligent, and responsive. Leveraging AI technologies, they improve the existing prediction and scheduling system, making water resource management more efficient and intelligent.

Solution architecture



The universal AI platform provides services such as Video Analysis Service (VAS), ModelArts, Knowledge Graph, Natural Language Processing (NLP), FusionInsight, and GeoGenius.

Various edge devices and terminals can access the network.

Oriented to digital government and smart city sectors

Suitable for drainage and flood control, water pollution control, and water resource management

Capable of monitoring and scheduling water volume and quality

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

GeoGenius

Powered by HUAWEI CLOUD's accumulation in cutting-edge technologies such as AI and big data, GeoGenius builds a one-stop, full-process intelligent development cloud platform for remote sensing. This cloud platform functions as both a data platform and an intelligent computing platform to help you focus on mining core values of data and developing application algorithms. It enables rapid application innovation in the industry and provides technical support for key tasks such as investigation, monitoring, evaluation, supervision, and law enforcement of natural resources, ecology, meteorology, environmental protection, and oceans.

Natural resource survey

Ecological environment monitoring

Agriculture and forestry monitoring

Weather forecast

Marine conservation

Emergency response and disaster prevention

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Smart Logistics Solution

The smart logistics solution provides AI-enabled 3D packing services to improve container loading rate. In addition, it provides the vehicle route optimization service to reduce customers' transportation costs.

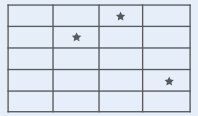
Customers
Send a packing request.
(RESTful API)

HUAWEI CLOUD
Work out a 3D packing plan with a high loading rate.

Customers
In the response message of the packing service, obtain the packing plan to facilitate packing.

Customer API requests

1. Container information
2. Goods information



HUAWEI CLOUD

3D packing service

Customers
Packing plan

Improve container space utilization, reduce the number of boxes

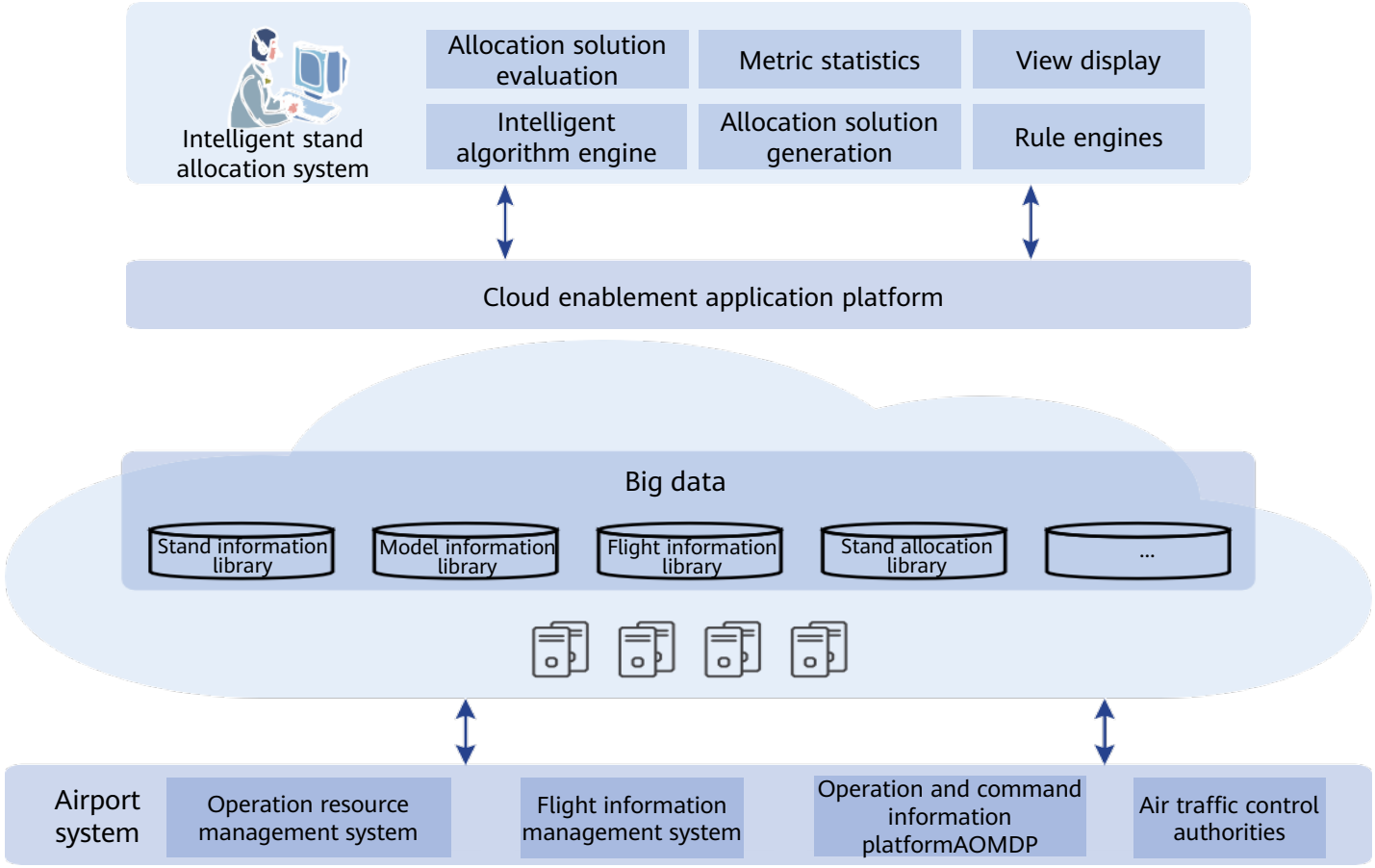
Reduce the operational cost

Improve the overall logistics operation efficiency

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Intelligent Stand Allocation Solution

Based on AI algorithms and scenario understanding, the intelligent stand allocation system provides the optimal stand allocation solution powered by Huawei big data platform and ModelArts, improving the docking rate, passenger experience, and airport operation efficiency.



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Contents

 Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

HUAWEI

Essential AI Platforms

ModelArts (AI development)

One-stop AI development platform

Huawei HiLens

Multimodal AI development application platform featuring device-cloud synergy

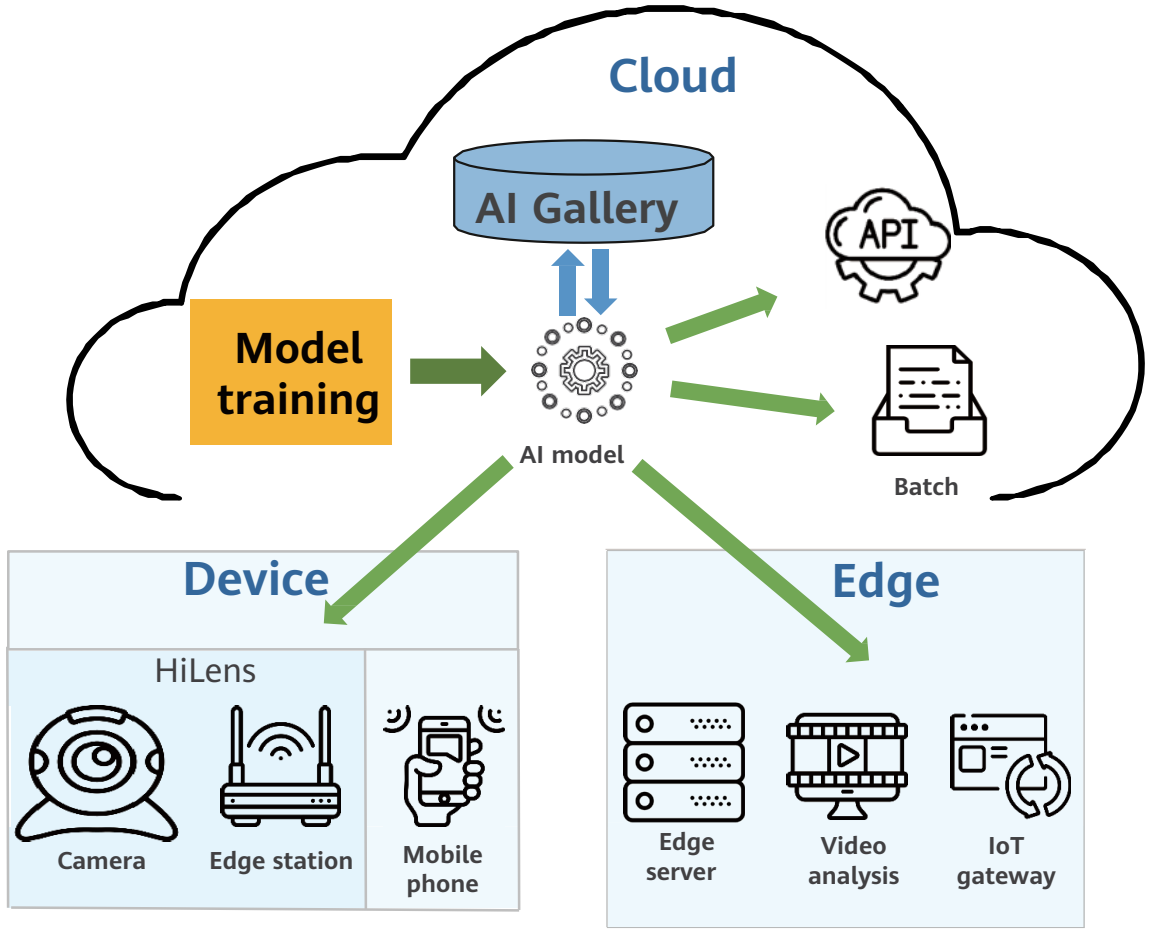
Graph Engine Service (GES)

China's first commercial distributed native graph engine with independent intellectual property rights

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

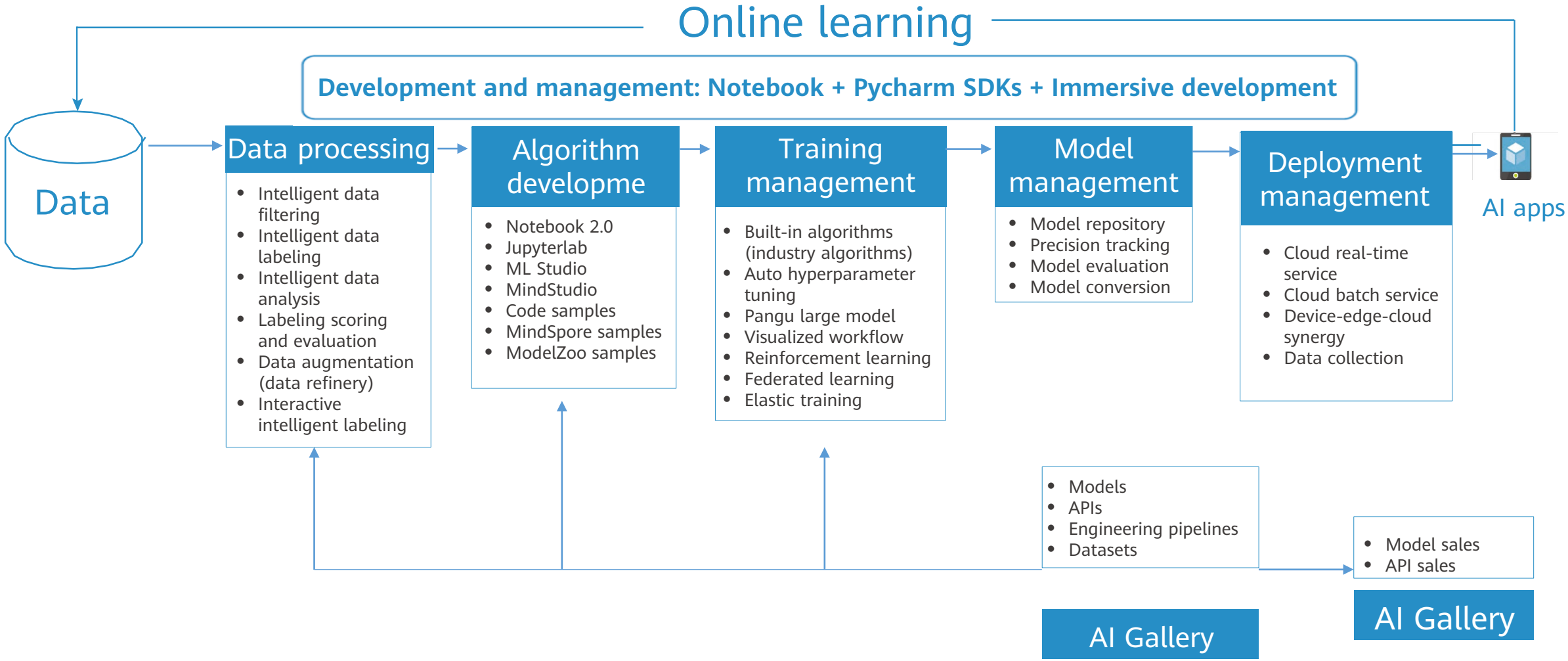
ModelArts

- ModelArts is a one-stop AI development platform. It provides data preprocessing, semi-automated data labeling, distributed training, automated model building, and model deployment on the device, edge, and cloud to help AI developers build models quickly and manage the AI development lifecycle.



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

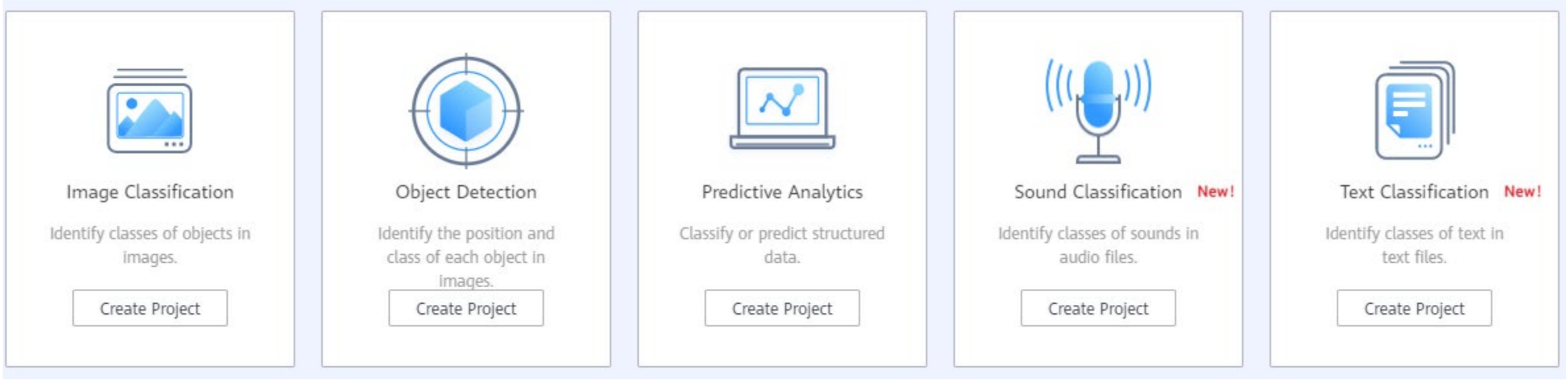
ModelArts Functions



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.
HUAWEI

ExeML

ExeML is an entry-level service provided by ModelArts for beginners. Powered by the ExeML engine, ExeML enables even beginners to build, train, verify, and publish models.



Zero coding

No AI experience required

Step 1:
Upload data and label it.

Step 2:
Train the model.

Step 3:
Check and publish the model.

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Data Management

A wide range of data formats

- Five types of data (image, audio, video, text, and table)
- Custom data formats

Team labeling

- Great for ultra-large-scale labeling

Iterative intelligent labeling framework

- Adaptive to data and algorithm changes
- Intelligent data filtering and auto pre-labeling

Intelligent data filtering

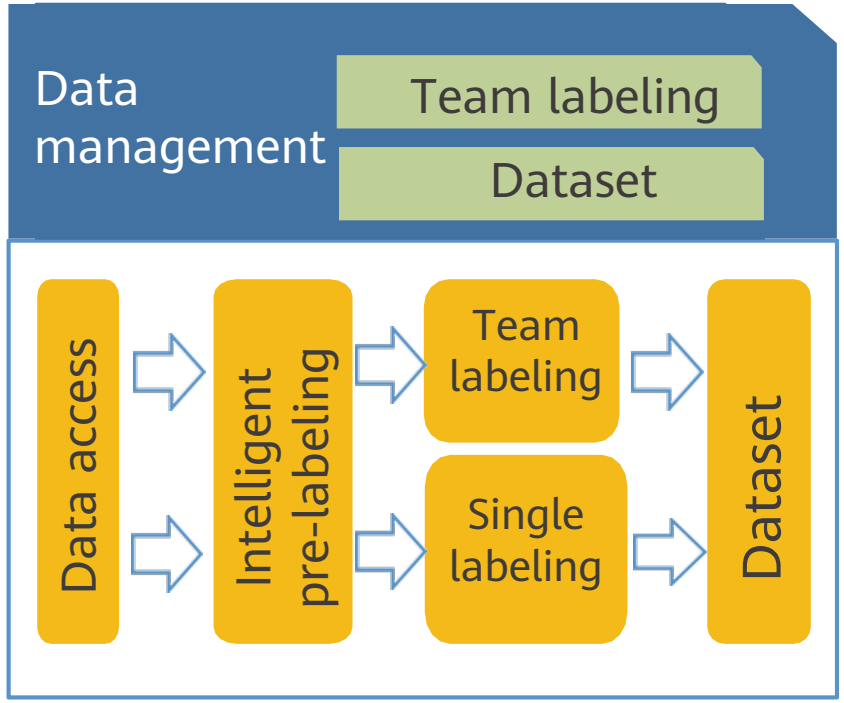
- Automatic image clustering
- Recognition rate of invalid images > 80%

Automatic pre-labeling

- Iterative semi-auto data pre-labeling
- Labeling efficiency up by 5 times

Auto feature mining

- In-depth data optimization suggestions in 30+ dimensions
- General models for various scenarios



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.
HUAWEI

Training Platform

Flexible, high-efficiency and cost-effective

- Multiple built-in algorithms, import of partner algorithms, custom training logic and images, and quick switchover between heterogeneous resources
- Linearly improved parallel training capability, auto parameter tuning, and a wide array of development modes
- Elastic training, economical mode, self-developed chips and software, and ultimate cost-effectiveness

Built-in training models, accelerating AI implementation

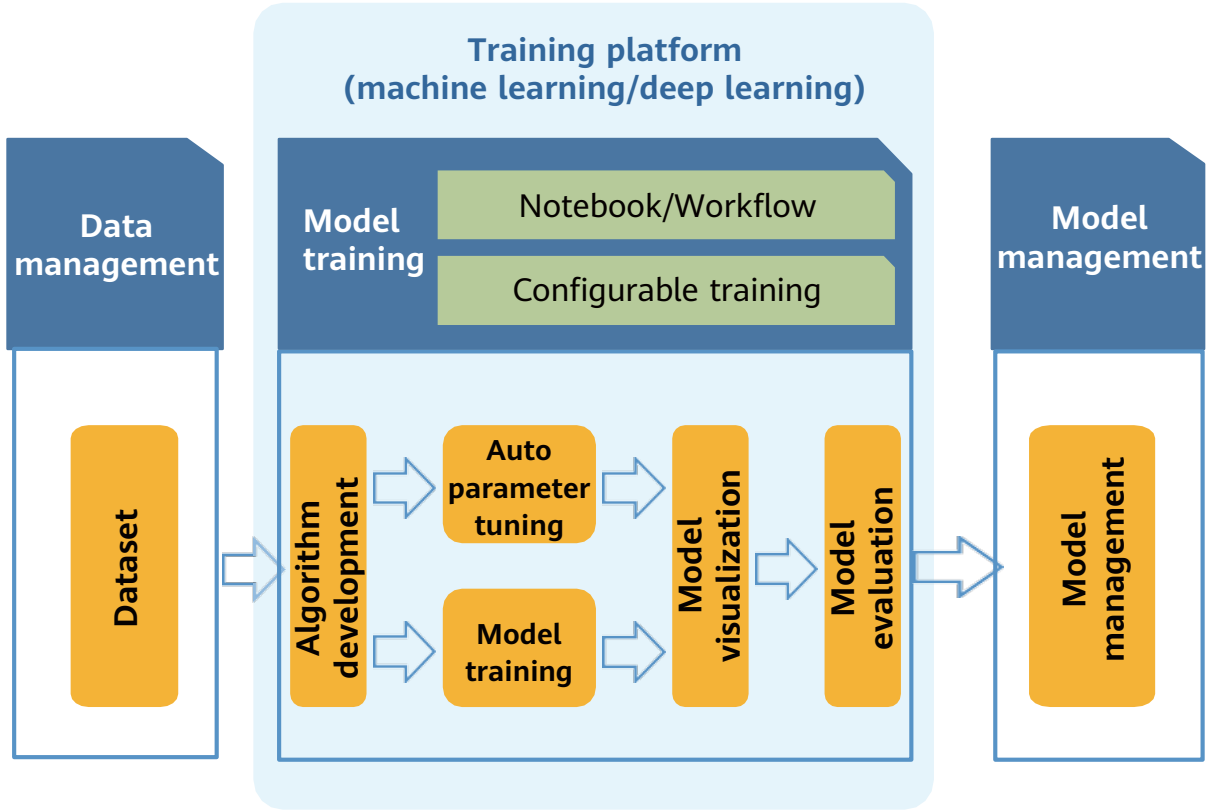
- 100+ algorithms, including image, text, time series, and reinforcement learning algorithms
- One-click training requiring only data source configuration

Multiple development modes, meeting different requirements

- In-cloud development (Notebook+SDK)
- On-premises development (IDE+PyCharm ToolKit)

Elastic training, improving training efficiency

- Turbo mode: Resources are automatically expanded during off-peak hours, accelerating the training speed by 10 times (from 2 hours to 10 minutes).
- Economic mode: Optimized scheduling reduces the training cost by 30%.



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Inference Platform

Unified management

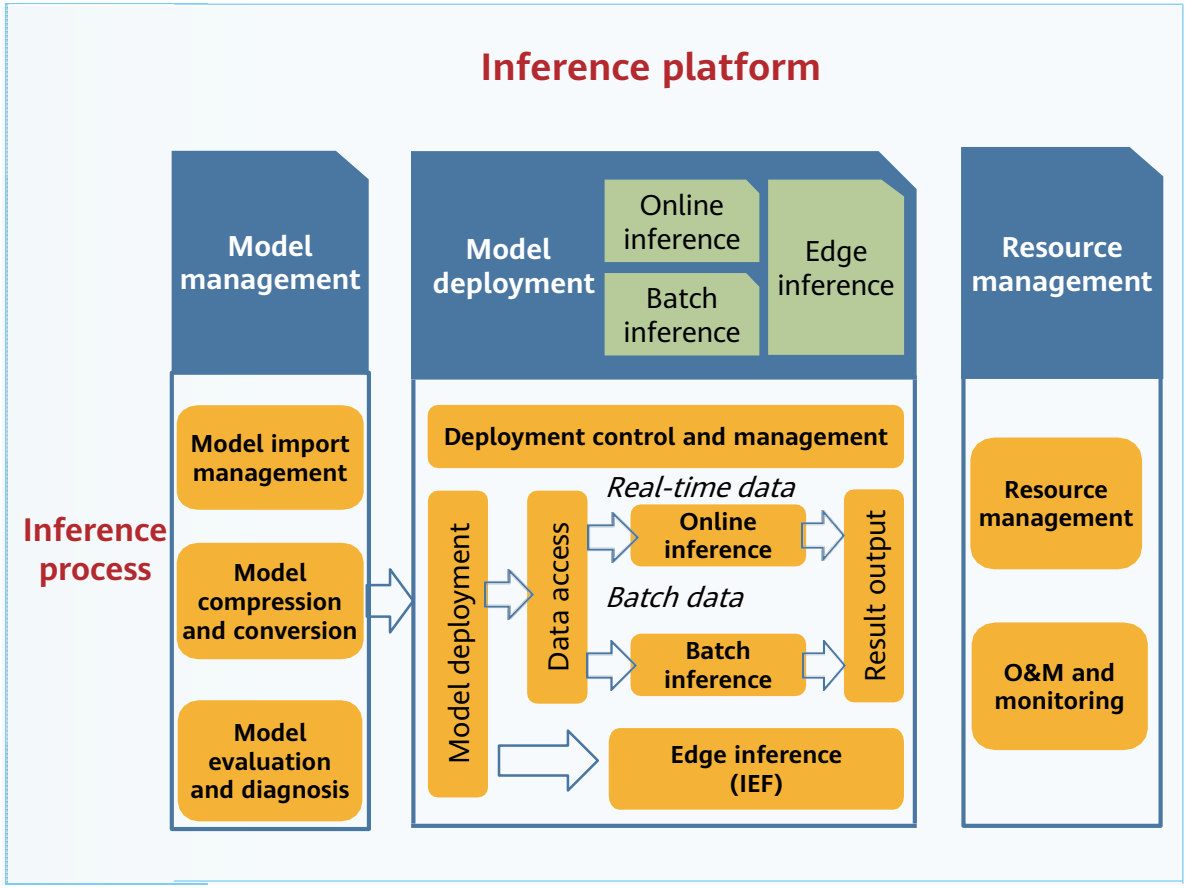
- Unified management of models of different vendors, frameworks, and functions
- High-concurrency model deployment, low-latency access, auto scaling, grayscale release, and rolling upgrade

Flexible deployment

- Models can be deployed as real-time and batch inference services on the cloud, edge, and devices.

Iterative model update

- Hard example mining, automatic identification of hard examples, and quick adaptation to data changes



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

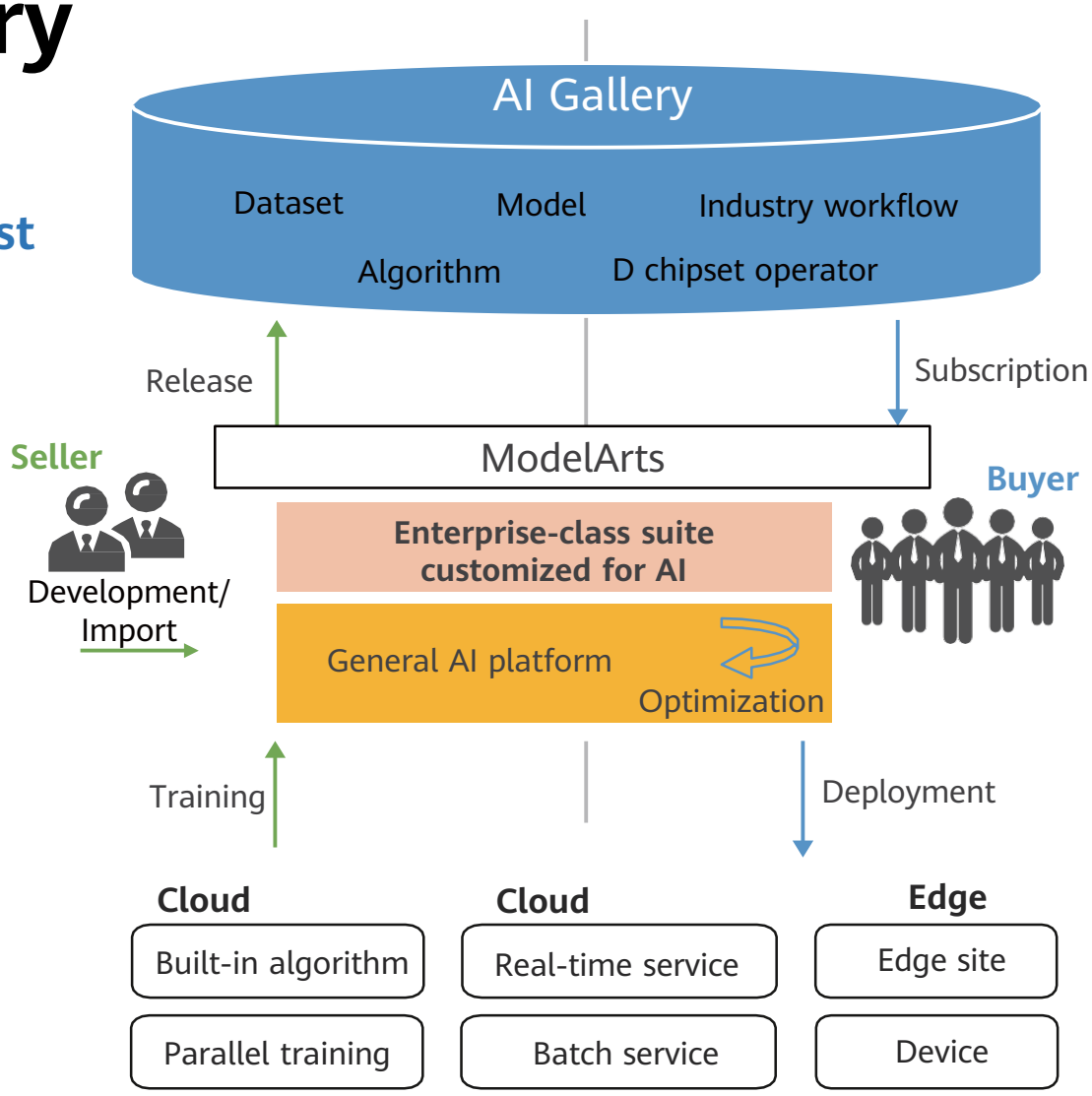
AI Gallery

Publishing at zero cost

- No fees for sellers to publish a model because no resources are consumed

Model finetuning

- Portfolios of cloud models and algorithms for buyers to optimize models based on differentiated data



On-demand subscription and quick deployment

- On-demand subscription, deployment, and charging for buyers
- Quick service deployment and low-cost O&M because of a full set of deployment capabilities

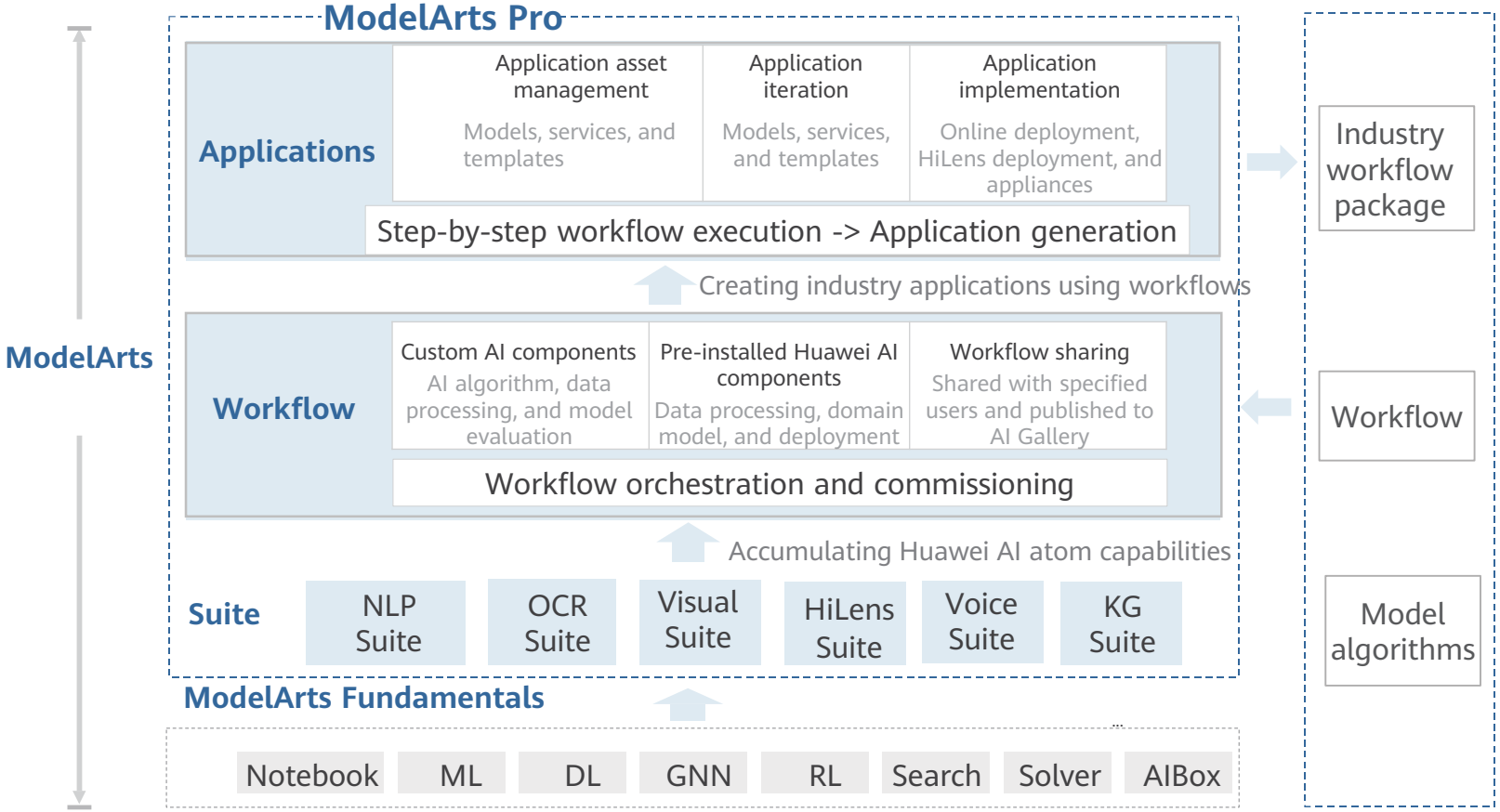
Encrypted deployment

- Confidentiality of container deployment protects intellectual property

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

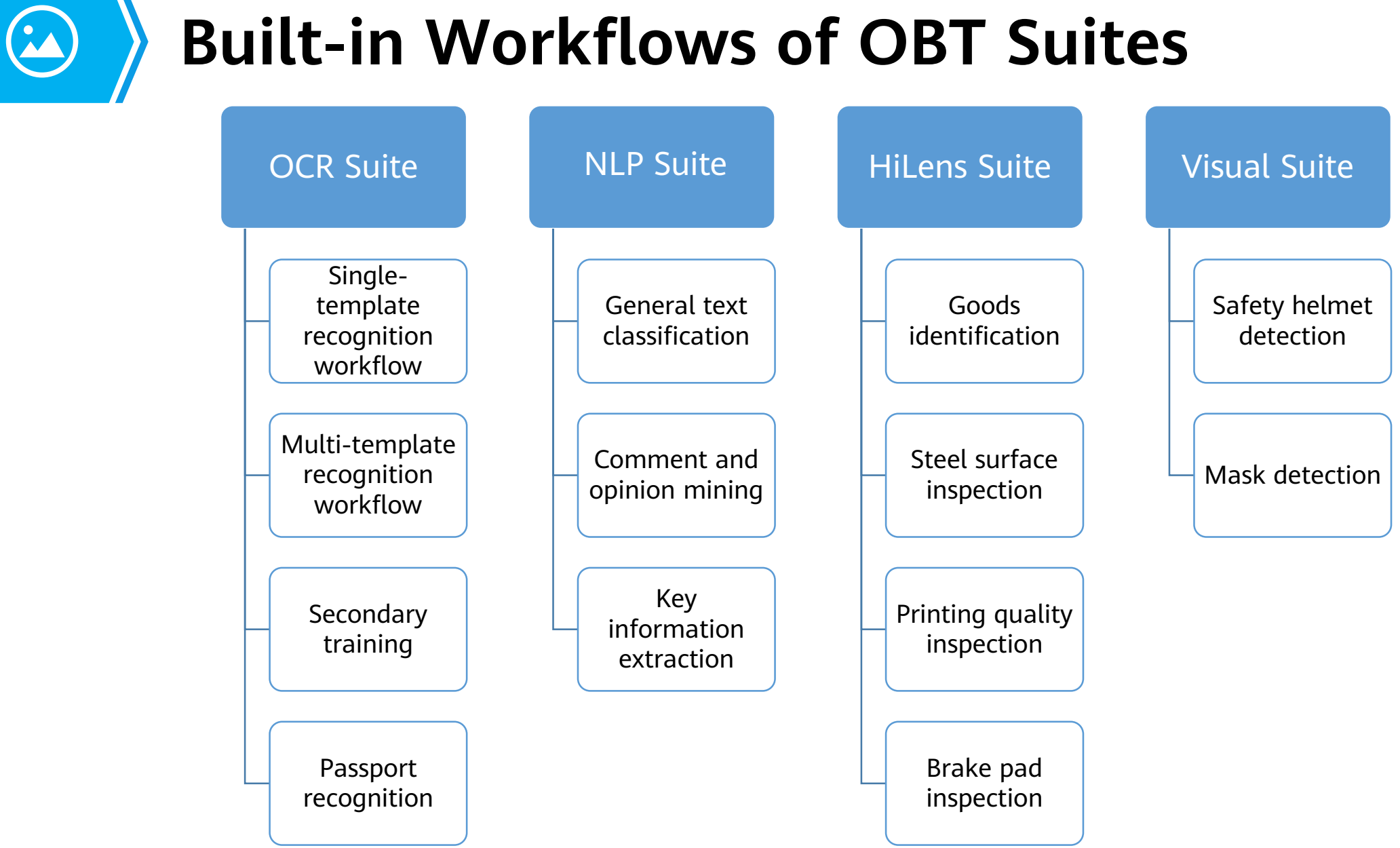
ModelArts Pro

ModelArts Pro is a professional development suite for enterprise-class AI applications. AI development is simplified with the advanced algorithms, quick training, and built-in workflows and models provided by HUAWEI CLOUD. In addition, customers can quickly develop, share, and launch applications through custom workflow orchestration. With these, HUAWEI CLOUD aims to create an open AI ecosystem for Inclusive AI. The ModelArts Pro suite consists of NLP Suite, OCR Suite, and Visual Suite, which allows it to meet AI implementation requirements in different scenarios.



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

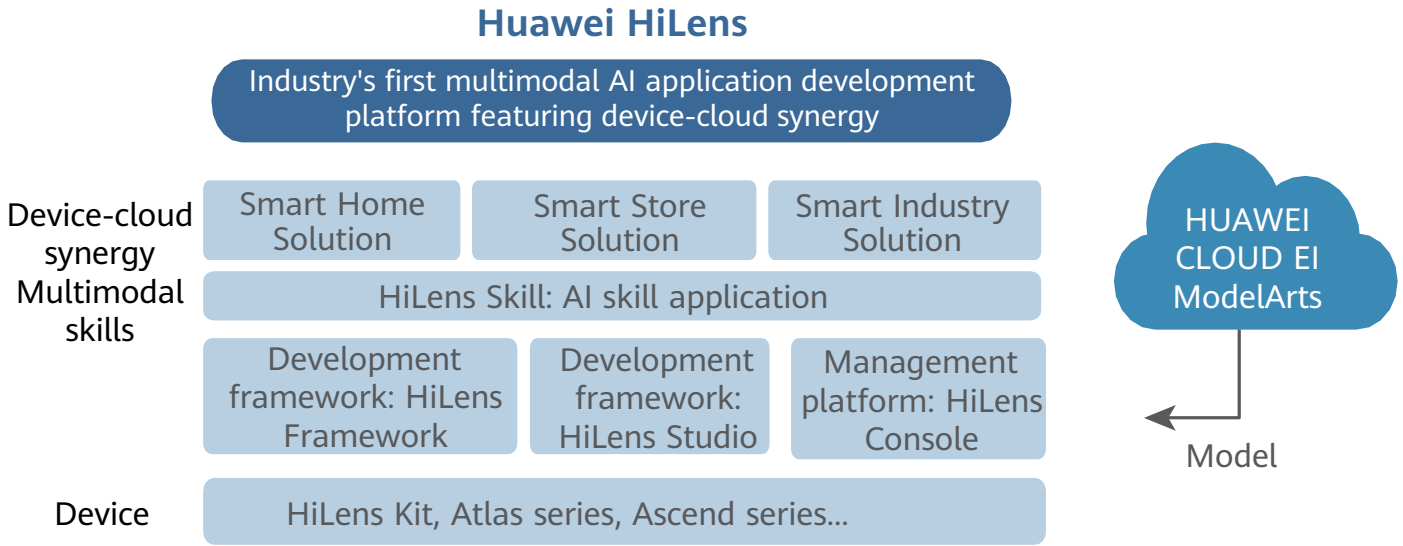
Built-in Workflows of OBT Suites



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.
HUAWEI

Huawei HiLens

- Huawei HiLens, featuring device-cloud synergy, enables you to develop AI applications in multiple modes. It has an easy-to-use development framework, with an out-of-the-box development environment. It also boasts a skill market with extensive sets of AI skills and a cloud management platform. Connecting to various computing devices, Huawei HiLens allows you to develop visual and auditory AI applications, deploy AI applications in real time, and manage a large number of devices.

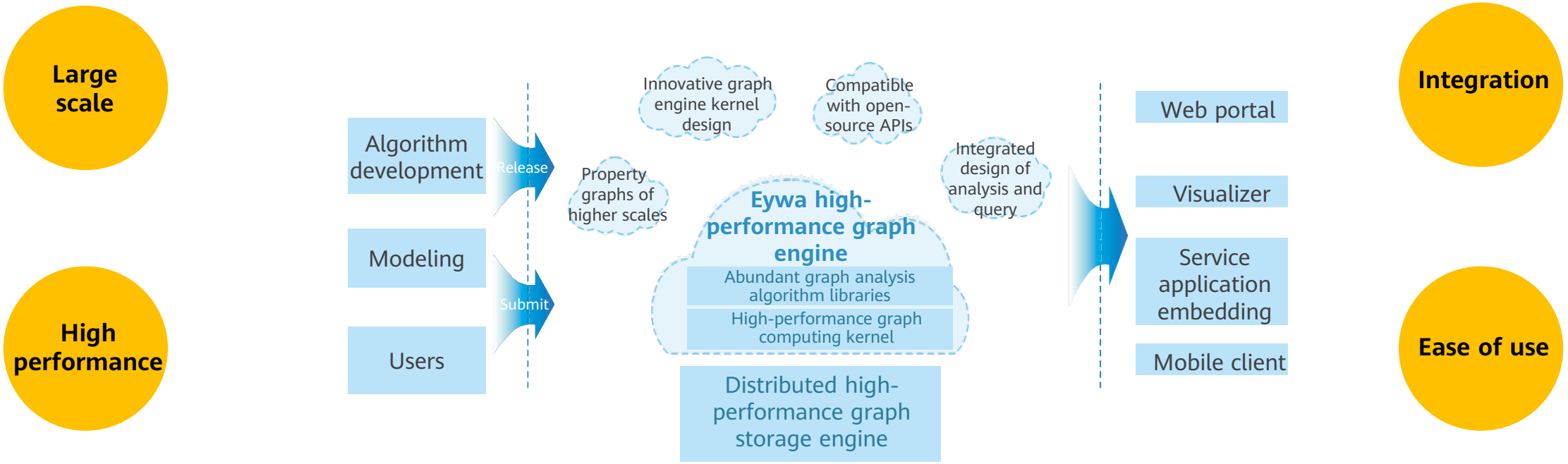


The HiLens platform has advantages such as device-cloud synergy for inference, rich skill market, convenient development, and powerful developer ecosystem.

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Graph Engine Service (GES)

- GES is the first commercial distributed native graph engine with independent intellectual property rights in China. It facilitates query and analysis on graph structure data based on relationships. It is specifically suited for scenarios involving social applications, enterprise relationship analysis, knowledge graph, risk control, and recommendations.



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

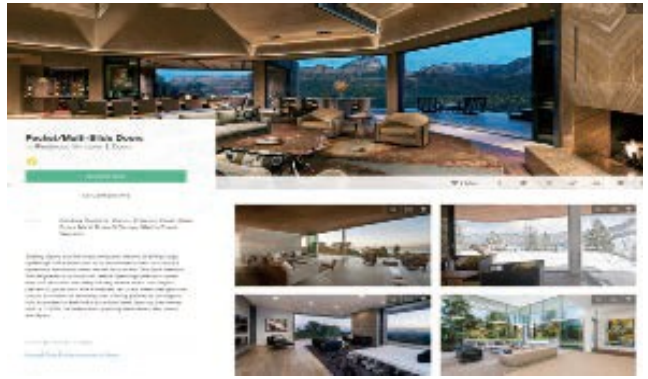
ImageSearch

Clothing search and recommendation

Piracy identification and copyright protection

Parts search and boost efficiency

Design search and comparison



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.
HUAWEI

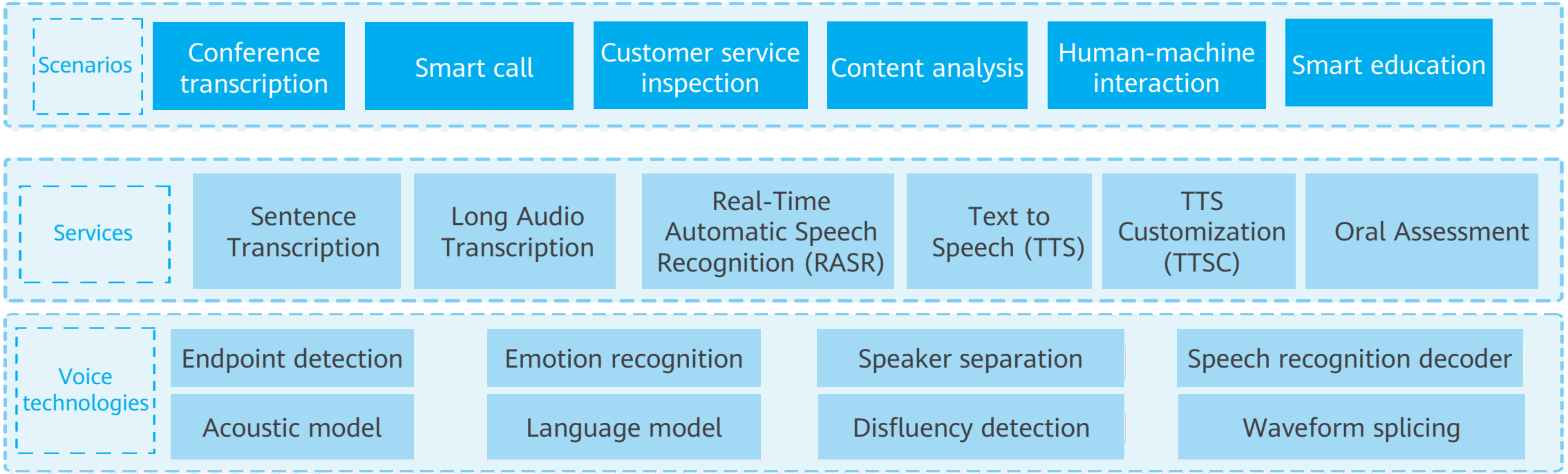
Speech Interaction Service (SIS)

Intelligent algorithms

- The speech interaction engine uses innovative algorithms and integrates traditional algorithms and deep learning models to achieve high recognition accuracy.

Self-service optimization

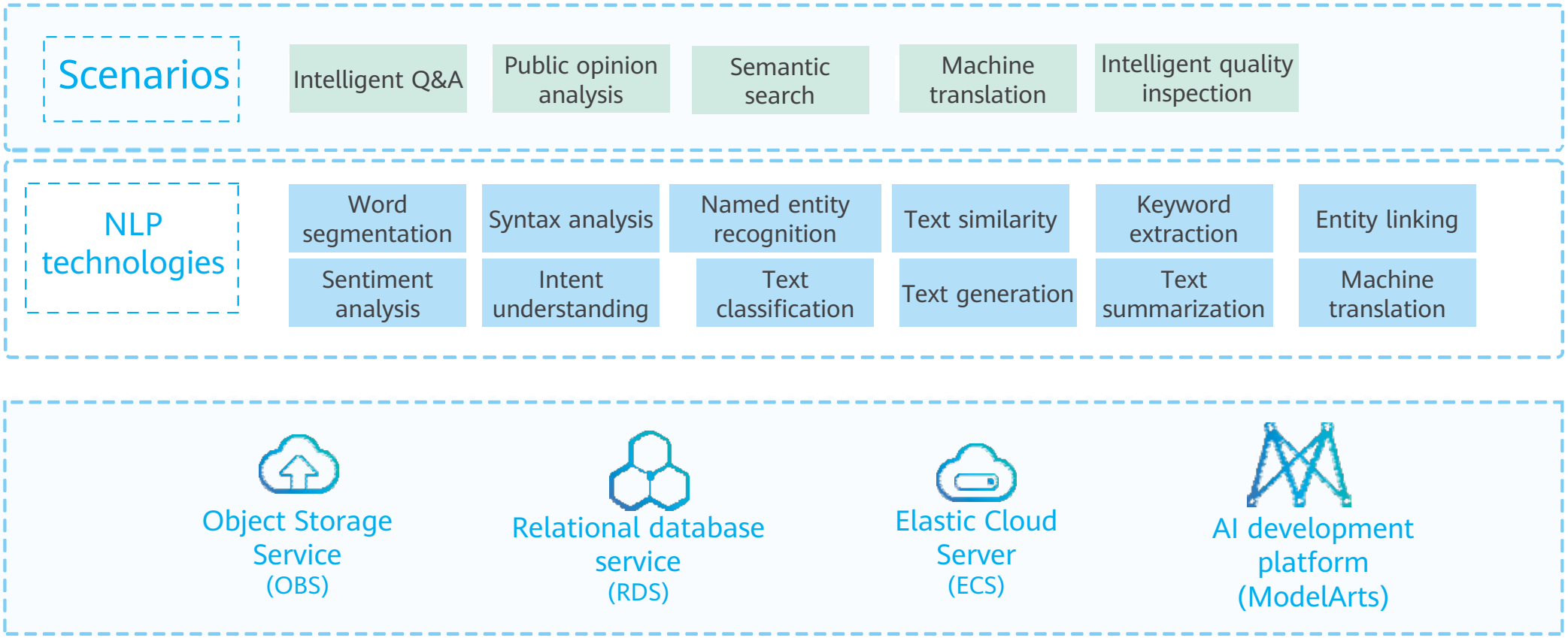
- The hot word function allows you to import hot words by yourselves to optimize the recognition effect in specific domains.



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Natural Language Processing (NLP)

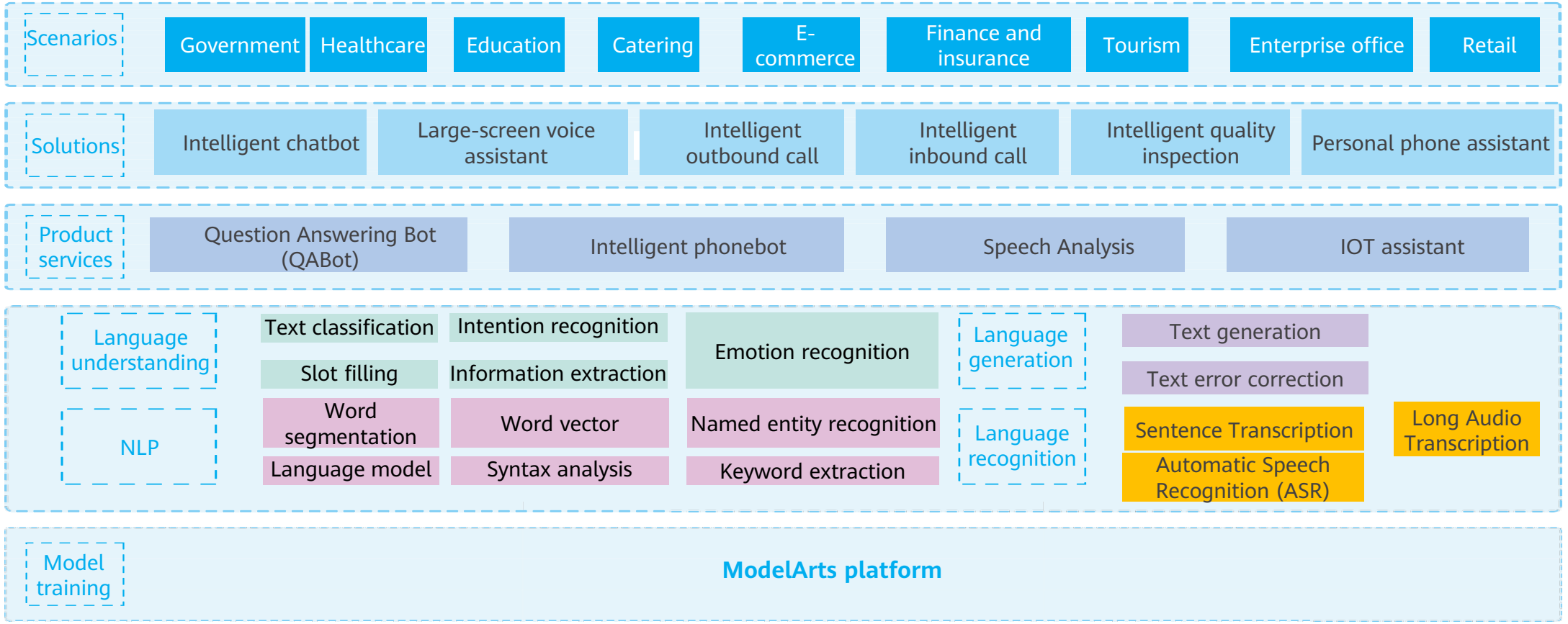
- NLP provides users with APIs related to natural language processing, including word segmentation, named entity recognition, sentiment analysis, and machine translation. It can be used in multiple application scenarios, such as intelligent Q&A, semantic search, public opinion analysis, and intelligent quality inspection.



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

CBS Customization (CBSC)

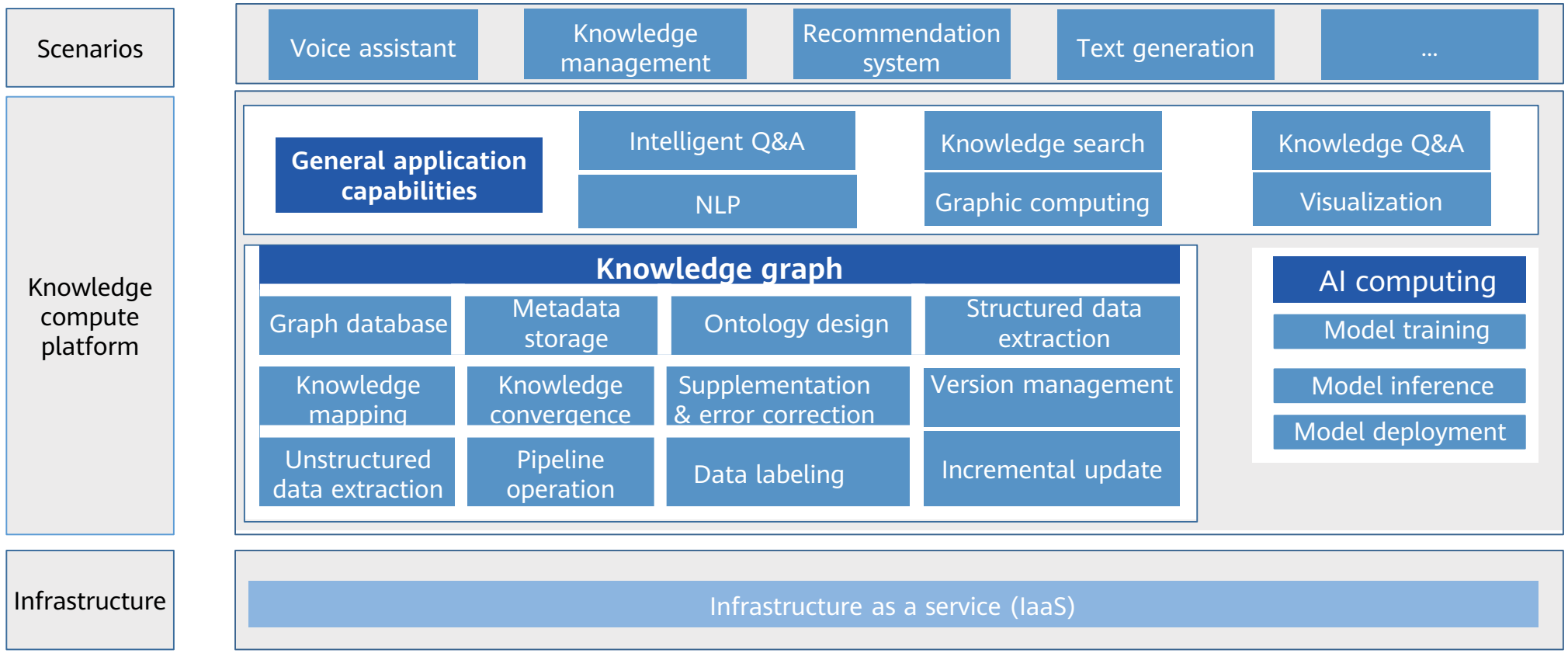
- Conversational Bot Service (CBS) is a cloud service based on NLP technology and enables dialog AI for different scenarios. With speech recognition, CBS enables product services to upgrade from graphical user interface (GUI) to voice interaction (VUI) to improve interaction efficiency and enhance interaction experience.



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Knowledge Graph

Knowledge Graph is a one-stop knowledge graph building platform that provides functions such as ontology design, information extraction, knowledge mapping, multi-source convergence, and incremental update.



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Optical Character Recognition (OCR)

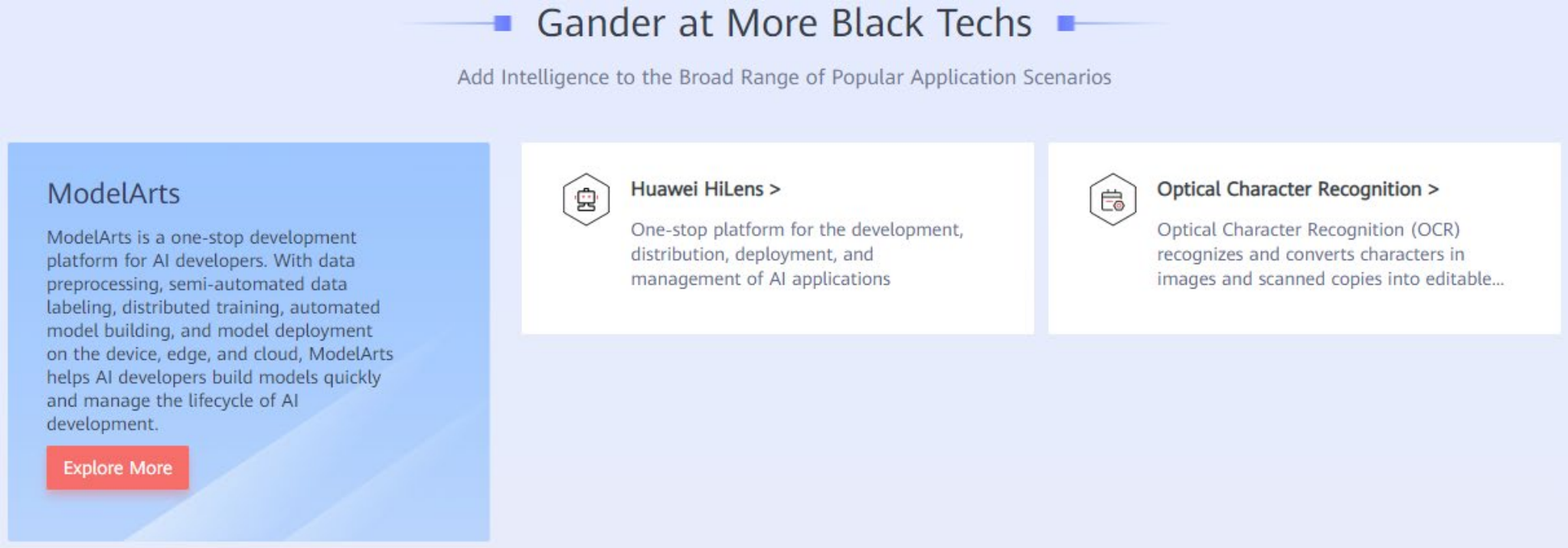
OCR is an online character recognition service that recognizes characters in images or scanned copies and converts them into editable text. OCR supports certificate recognition, receipt recognition, custom template recognition, and general table text recognition.

General OCR	Receipt OCR	Card OCR	Domain OCR	Custom OCR
General Text OCR	VAT Invoice OCR	ID Card OCR	Electronic Waybill OCR	User-defined templates
General Table OCR	Vehicle Sales Invoice OCR	Driving License OCR	Insurance Policy OCR	Dedicated API customization and development
Web Image OCR	Taxi Invoice OCR	Vehicle License OCR		
Auto Classification OCR	Train Ticket OCR	Passport OCR		
Handwritten Letters & Digits OCR	Quota Invoice OCR	Bank Card OCR		
Signature & Seal OCR	Toll Invoice OCR	Business License OCR		
	Flight Itinerary OCR	Transportation License OCR		
	Chinese and English Customs Form OCR	Plate Number OCR		
		Business Card Recognition		
		Vehicle Identification Number OCR	Available	
		Document Check Classification OCR	Coming Soon	

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

EI Experience Center

The EI Experience Center allows you to experience AI. It is dedicated to lowering the expertise requirements for using AI and making AI ubiquitous.



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

EI Training Camps

EI Activities

21-Day Training Camp for AI Novices

Coming soon

Become an AI developer with ease with the help of HUAWEI CLOUD.

21-Day Training Camp for Big Data Enterprise Application

Coming soon

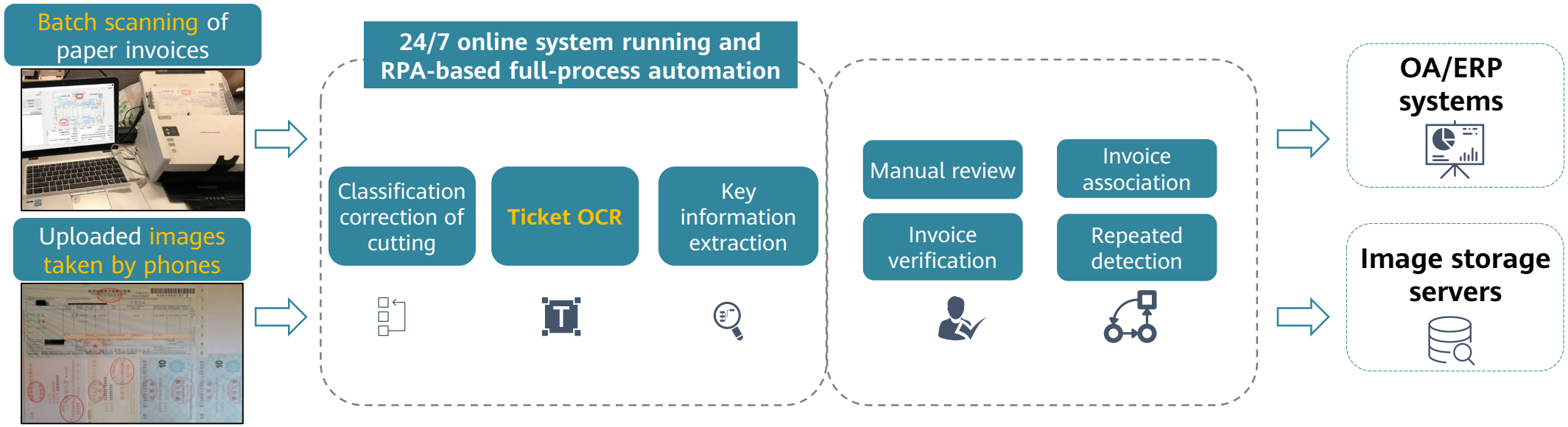
Learn how to get the most out of big data through lessons from eight typical enterprise scenarios.

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

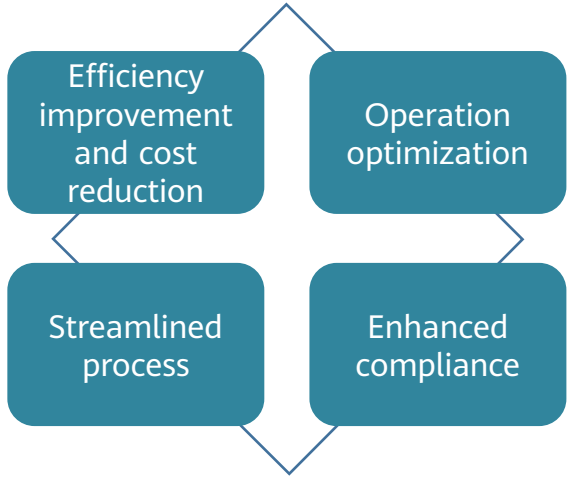
Contents

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Financial Reimbursement

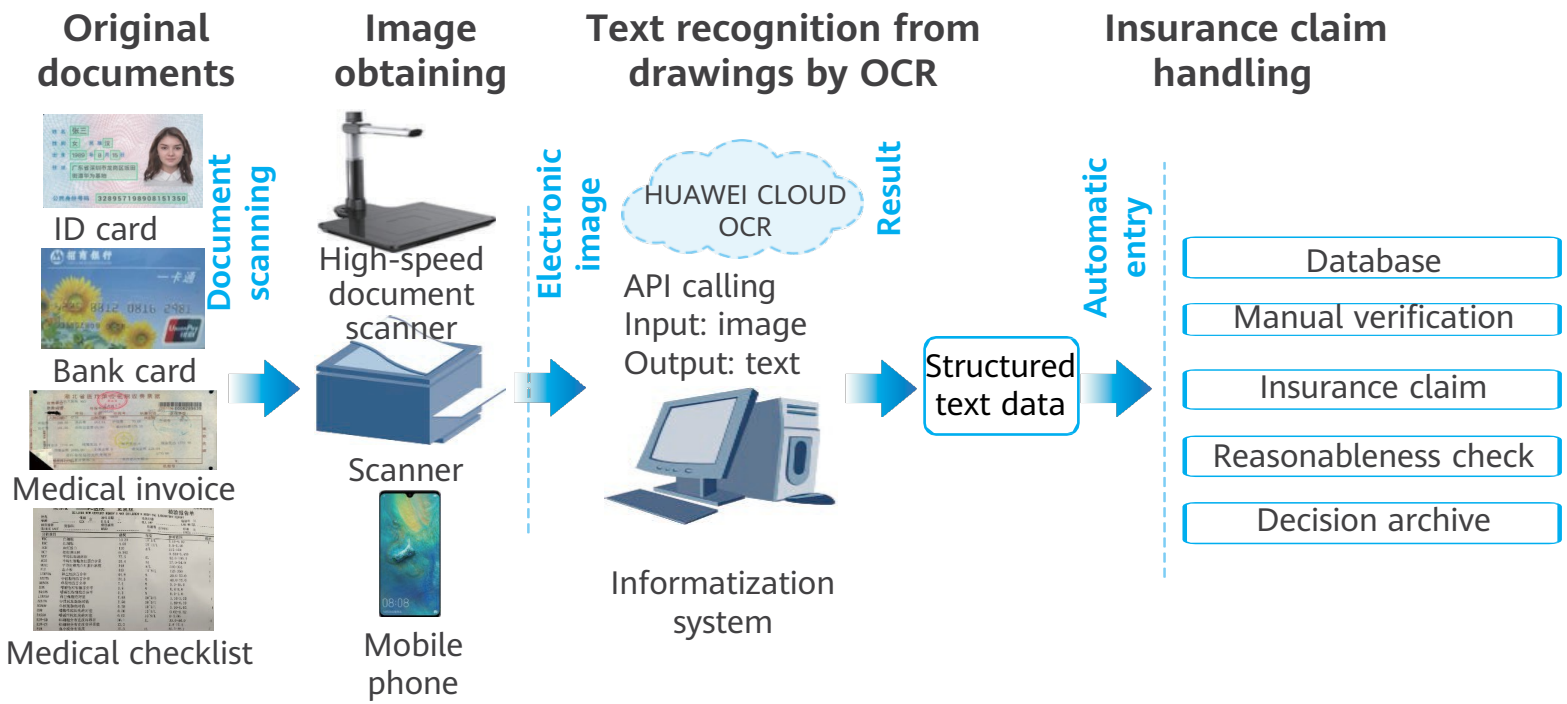


- **Multiple access modes**: Automatically connects to scanners, high-speed document scanners, and mobile phones to obtain images in batches.
- **Flexible deployment**: Supports multiple deployment modes, such as public cloud, HCS, and appliance, and provides unified standard APIs.
- **Support for various invoices**: VAT invoices/special invoices/electronic invoices/ETC invoices/bills, taxi invoices/train invoices/travel invoices/fixed-amount invoices/toll invoices
- **Multiple invoices in an image**: Automatically classifies and recognizes multiple types of invoices.
- **Visualized comparison**: The location information is returned and converted into an Excel file for statistics and analysis.



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Medical Insurance Claims



- Greatly accelerates data recording and verification, improves user experience, and reduces labor costs.
- Extracts structured information about images, such as the car sales invoice and effectively handles the image rotation, error line, blur, and deformation issues. The accuracy rate is higher than 98%.
- Automatically detects contract signatures to ensure compliance.

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Intelligent Customer Service Solution

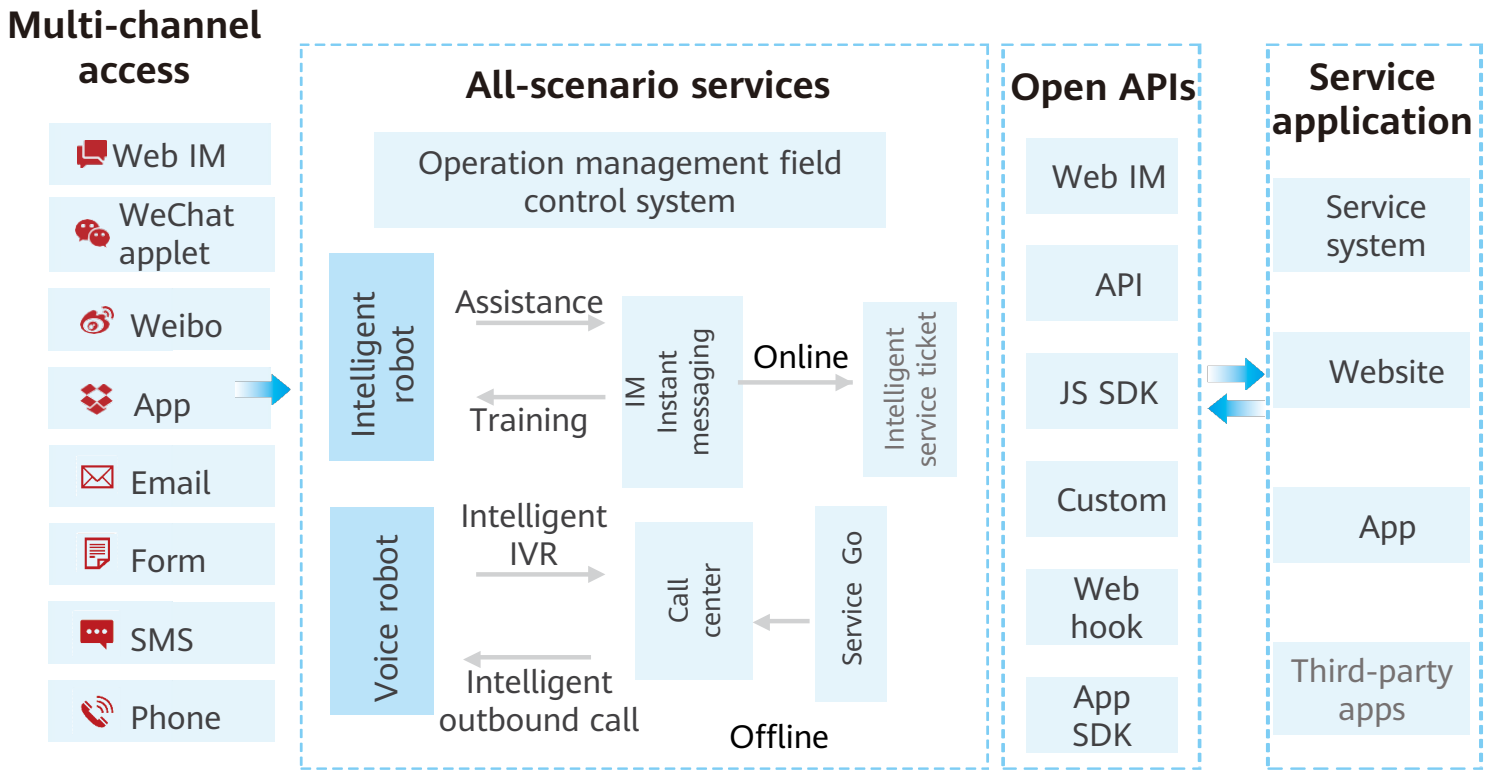
Machines assisting humans
- The robot preliminarily filters user problems and transfers the problems that cannot be solved to the customer service personnel to reduce the workload of. the customer service personnel.

Robots replace humans to answer questions
- The robot answers questions more accurately, comprehensively, and quickly, reducing the training workload of customer service personnel.

Intelligent routing
- The robot identifies the department or business line corresponding to a specific problem through dialogs with users, and then transfers the call to the target manual service or invokes the target sub-robot to facilitate subsequent customer service data statistics.

Intelligent quality inspection
- The robot performs offline inspection on the service quality of the manual customer service in batches to identify user emotions and help seize the blind spots of the customer service to effectively improve the service quality.

Hot issue mining
- Based on the questions raised by users, the robot clusters the questions, calculates the frequency, and sorts out hot questions to learn about user feedback in a timely manner.



Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Large-Screen Voice Assistant in Longzihu, Zhengdong New Area

Challenges

- On the large screen, it is complex to select the content to display.
- It is difficult to quickly extract required information from a large amount of data.
- Onsite presentation and large-screen interaction cannot be automated.

Solution

- The large-screen voice assistant solution combines the large-screen control system with bot data.
- Voice navigation is used to display specific pages on the large screen through conversations.
- Voice Q&A is available, so you can interact with the big screen to query metrics and trends through conversations.

Benefits

- Frees your hands and simplifies operations. You do not need to operate the large screen on your computer;
- Data can be queried in a more timely manner without manual sorting and searching.

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Return Visits to the Insurance Industry

Recognition rate

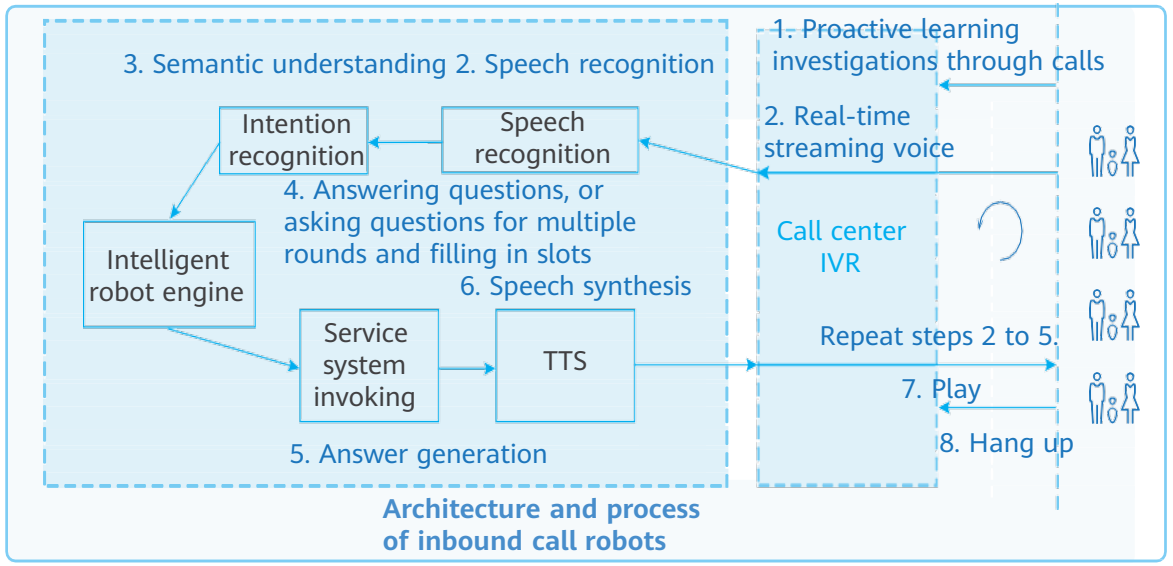
- High recognition accuracy and voice recognition technology that has been verified on a large scale.

Hot words

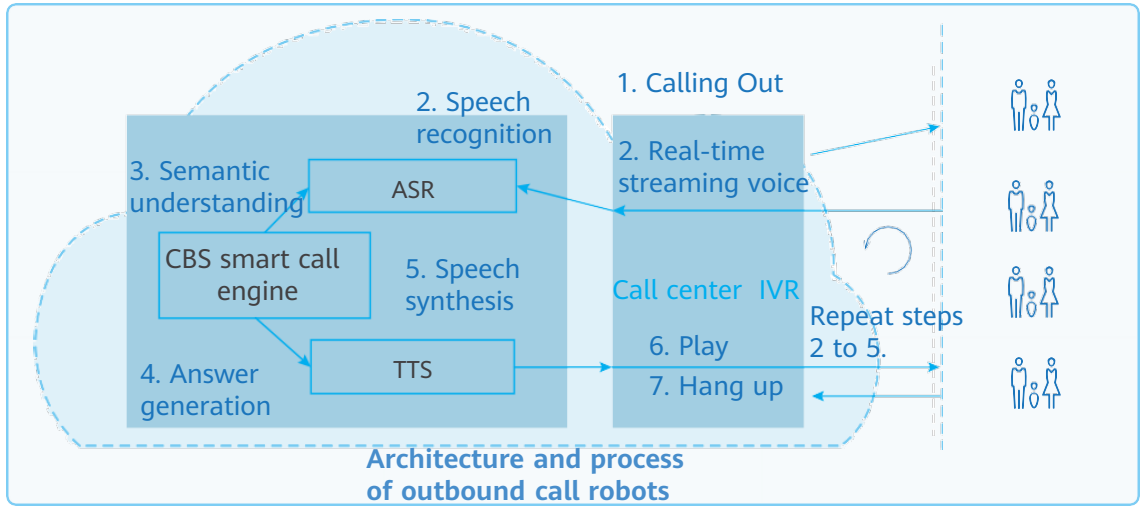
- Use hot words to improve the recognition accuracy of professional words.

High concurrency and low latency

- Background services can be elastically scaled to meet high concurrency requirements.
- Real-time stream identification has a low delay and low IVR process customization.
- Multiple concurrent calls are supported, and the response time is within milliseconds.



Architecture and process of inbound call robots



Architecture and process of outbound call robots

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

AI-Assisted Diagnosis of COVID-19

Scenario

CT imaging is an important means to evaluate the condition of COVID-19 patients. Manual screening takes at least 12 minutes, and expert resources are scarce.

HUAWEI CLOUD works with Huazhong University of Science and Technology and BlueNet to launch the AI-assisted COVID-19 screening service to deliver AI medical image analysis capabilities, improving the quality and efficiency of auxiliary diagnosis.

Solution advantages

- Second-level response: Outputs screening results in seconds, improving doctors' image reading efficiency;
- 3D reconstruction of lesions: Detects the lesions and their affected positions, automatically measures the volume, and outputs anatomy and location through 3D reconstruction.
- Comparison of follow-up positioning: Helps doctors effectively evaluate the progress of patients' illnesses and the efficacy of medication.

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Summary

- This course first introduces the HUAWEI CLOUD EI ecosystem, helping you understand HUAWEI CLOUD EI services. Then, it focuses on ModelArts and AI services, helping you quickly understand ModelArts. Finally, it introduces success stories of EI.

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.

Quiz

1. Which of the following scenarios is EI suitable? ( )

   A. Smart government

   B. Smart city

   C. Smart manufacturing

   D. Smart finance

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.
HUAWEI

More Information

- Huawei Talent Online

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

- WeChat official accounts:

EMUI

Huawei Device Open Lab

Huawei Developer

Contact Huawei Talent Online

Copyright © 2021 Huawei Technologies Co., Ltd. All rights reserved.


Thank You
www.huawei.com