**12**

**Machine Learning, IoT, and Blockchain in AWS**

Emerging technology such as **Machine Learning** (**ML**), **Artificial Intelligence** (**AI**), blockchain, and the **Internet of Things** (**IoT**) started as experiments by a handful of technology companies. Over the years, major technology companies, including Amazon, Google, Facebook, and Apple, have driven exponential growth by utilizing the latest emerging technology and staying ahead of the competition. With the cloud, emerging technologies have become accessible to everyone. That is another reason why organizations are rushing to adopt the cloud as it opens the door for innovation with tested technology by industry leaders like Amazon, Microsoft, and Google through their cloud platforms.

Today, the most prominent emerging technologies becoming mainstream are ML and AI. IoT is fueling industry revolutions with smart factories, and autonomous cars and spaces. Blockchain has seen tremendous growth recently, with a boom in cryptocurrencies and temper-proof record tracking. Even now, all major cloud providers are investing in quantum computing (which could bring the next revolution in the coming decade) and making it accessible to everyone through their platforms.

In this chapter, you will learn about the following emerging technology platforms available in AWS:

* ML in AWS with Amazon SageMaker
* Using AI in AWS using the readily available trained model
* Building an IoT solution in AWS
* Using **Amazon Managed Blockchain** (**AMB**) to build a centralized blockchain application
* Quantum computing with Amazon Braket
* Generative AI

Let’s start diving deep and learn about these innovative technologies in detail.

**What is AI/ML?**

ML is a type of computer technology that allows software to improve its performance automatically by learning from data without being explicitly programmed. It is a way of teaching computers to recognize patterns and make predictions based on examples. In simple terms, ML is a way for computers to learn from data and make predictions or decisions. There are several types of ML, each with its unique characteristics and use cases. The main types of ML include:

* **Supervised Learning**: Supervised learning is the most widespread form of ML, involving training a model on a labeled dataset to predict the output for new, unseen data. Linear regression, logistic regression, and decision trees are some examples of supervised learning algorithms.
* **Unsupervised Learning**: Unsupervised learning, on the other hand, does not use labeled data and instead discovers patterns and structures in the input data. Examples of unsupervised learning algorithms include clustering, dimensionality reduction, and anomaly detection.
* **Semi-Supervised Learning**: This type of ML is a combination of supervised and unsupervised learning, where the model is given some labeled data and some unlabeled data and must find patterns and structure in the input data while also making predictions.
* **Reinforcement Learning**: Reinforcement learning is used in decision-making and control systems, where an agent interacts with an environment and learns to perform actions that maximize a reward signal. This type of learning is used in decision-making and control systems.
* **Deep Learning**: Deep learning is a subset of ML that employs deep neural networks with multiple layers to learn from data and make predictions or decisions. This method is particularly useful for tasks such as image and speech recognition, **Natural Language Processing** (**NLP**), and decision-making.
* **Transfer Learning**: This type of ML is used when the data or task of interest is different from the data or task on which the model was originally trained. This technique leverages the knowledge learned from a pre-trained model to improve the new model’s performance.

“Artificial intelligence” is a more comprehensive term that encompasses not only ML but also other technologies that empower machines to undertake activities that conventionally necessitate human intelligence, including comprehending natural language, identifying objects, and making decisions. In basic terms, AI is a means for computers to accomplish tasks that ordinarily demand human intelligence, such as understanding spoken language, recognizing facial features, and playing strategic games like chess. AI can be implemented in many ways, from simple rule-based systems to more advanced techniques like ML and deep learning, which allow computers to learn from data and make predictions or decisions. There are several types of AI, each with its characteristics and use cases. The main types of AI include:

* **Reactive Machines**: These types of AI can only react to their environment; they can’t form memories or learn from past experiences. Reactive machines are typically used in applications such as self-driving cars and video game AI.
* **Limited Memory**: These types of AI can take into account past experiences and use that information to make decisions. Examples of limited memory AI include robots that can navigate a room or a self-driving car that can change its driving behavior based on recent experiences.
* **Narrow AI**: These are AI systems designed to perform a specific task, such as image recognition or speech recognition. These systems are not general purpose and can only perform the task they were designed for.
* **Theory of Mind**: This type of AI is designed to understand mental states such as beliefs, intentions, and desires. This type of AI is still in the research phase and has not been fully implemented.
* **Self-Aware**: This is the most advanced type of AI, where the AI has the ability to be aware of its own existence and consciousness. This type of AI is still in the realm of science fiction and has not yet been achieved.
* **General AI**: AI systems that can perform any intellectual task that a human can, also known as artificial general intelligence. These systems do not yet exist but are the ultimate goal of AI research.

In this section, you saw a quick overview of AI/ML. This is a very broad topic, and there are multiple books that explain these concepts in detail. Within the context of this book, let’s focus on AI/ML in AWS.

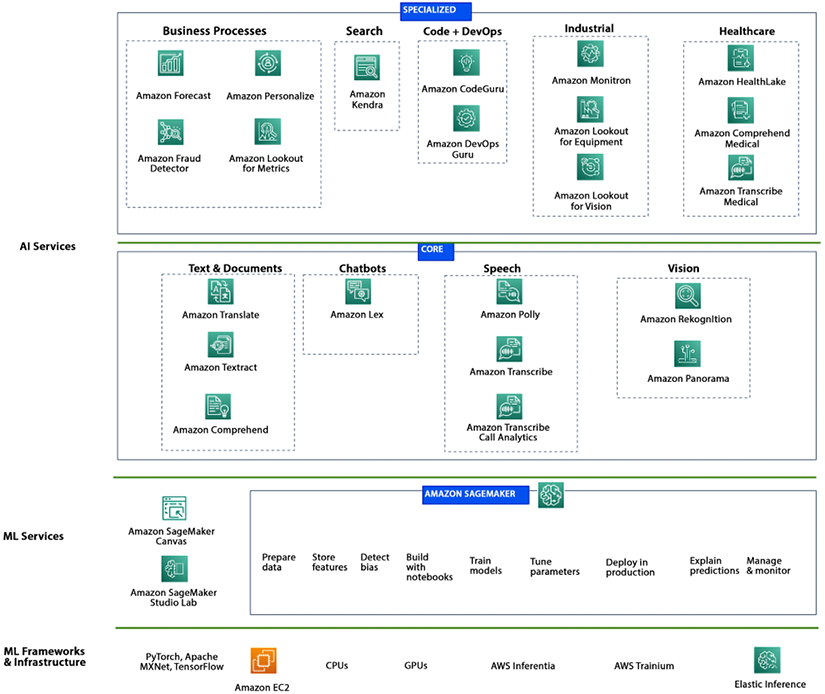
**AI/ML in AWS**

In recent years, ML has rapidly transitioned from a cutting-edge technology to a mainstream one; however, there is still a long way to go before ML is embedded everywhere in our lives. In the past, ML was primarily accessible to a select group of large tech companies and academic researchers. But with the advent of cloud computing, the resources required to work with ML, such as computing power and data, have become more widely available, enabling a wider range of organizations to utilize and benefit from ML technology.

ML has become an essential technology for many industries, and AWS is at the forefront of providing ML services to its customers. Some of the key trends in ML using AWS include:

* **Serverless ML**: AWS is making it easier to build, train, and deploy ML models without the need to manage servers. With services like Amazon SageMaker, customers can build and train models using managed Jupyter Notebook and then deploy them to a serverless endpoint with just a few clicks.
* **Automated ML**: Automating the model-building process is becoming increasingly popular, allowing customers to achieve good results with minimal expertise. AWS offers services like Amazon SageMaker Autopilot, which automatically builds and tunes ML models and selects the best algorithm with hyperparameters for a given dataset.
* **Transfer Learning**: With the amount of data available today, it is becoming increasingly difficult to train models from scratch. Transfer learning allows customers to use a pre-trained model as a starting point and fine-tune it for their specific use case.
* **Reinforcement Learning**: Reinforcement learning is a type of ML that is well suited for problems where the feedback is delayed or non-deterministic. AWS offers services like Amazon SageMaker RL, which allows customers to easily build, train, and deploy reinforcement learning models.
* **Federated Learning**: Federated learning is a distributed ML technique that allows customers to train models on multiple devices while keeping the data private.

AWS provides a wide range of services that make it easy for you to build, train, and deploy ML models. With the growing adoption of ML, AWS is well positioned to continue to lead the way in providing ML services to its customers. The following figure represents the services stack for AI/ML in AWS, divided into three parts: **AI services**, **ML services**, and **ML frameworks and infrastructure**.



*Figure 12.1: AWS ML Services Stack*

The diagram illustrates a stack of options for working with ML; starting with the lowest level, ML frameworks and infrastructures require detailed programming and are best suited for field data scientists. This option offers the most flexibility and control, allowing you to work on an ML model from scratch using open-source libraries and languages.

At the top, purpose-built AI services are pre-built and ready to be invoked, such as Amazon Rekognition, which can perform facial analysis on an image without requiring any ML code to be written.

In the middle is Amazon SageMaker, the most feature-rich option, which offers a complete suite of subservices for preparing data, working on notebooks, performing experiments, monitoring performance, and more, including no-code visual ML.

Let’s look into each layer of the above service stack in detail, starting with ML frameworks and infrastructure and moving upwards.

**AWS ML frameworks and infrastructure**

AWS provides a variety of infrastructure services for building and deploying ML models. Some of the key services include:

* **Amazon EC2 for ML workloads**: AWS provides a variety of EC2 instance types that can be used for ML workloads. Depending on the workload’s needs, these instances can be configured with different amounts of CPU, memory, and GPU resources. For example, the P3 and G5 instances are designed explicitly for ML workloads and provide high-performance GPU resources.
* **Amazon Elastic Inference**: This is a service that enables you to attach GPU resources to Amazon EC2 or Amazon SageMaker instances to accelerate ML inference workloads.
* **AWS Inferentia**: AWS provides a custom-built chip called Inferentia, which can be used to perform low-latency, high-throughput inferences on deep learning workloads. It is designed to provide high performance at a low cost and can be used with Amazon SageMaker.
* **AWS Trainium**: AWS Trainium is a chip designed specifically to address the budget constraints development teams face while training their deep learning models and applications. AWS Trainium-based EC2 Trn1 instances provide a solution to this challenge by delivering faster training times and cost savings of up to 50% compared to similar GPU-based instances, allowing teams to train their models more frequently and at a lower cost.

AWS provides a variety of frameworks and libraries for ML development, allowing customers to easily build, train, and deploy ML models. Some of the main ML frameworks and libraries available on AWS include:

* **TensorFlow**: TensorFlow is an open-source ML framework developed by Google, which can be used for a wide variety of tasks, such as NLP, image classification, and time series analysis.
* **Apache MXNet**: MXNet is an open-source ML framework developed by Amazon, which can be used for tasks such as image classification, object detection, and time series analysis.
* **PyTorch**: PyTorch is an open-source ML framework developed by Facebook, which can be used for tasks such as image classification, NLP, and time series analysis.
* **Keras**: Keras is a high-level open-source ML library that can be used as a wrapper for other ML frameworks such as TensorFlow and MXNet, making it easier to build, train, and deploy ML models.

In this section, you learned about the ML frameworks and infrastructure provided by AWS. Let’s learn about the middle layer of the service stack, Amazon SageMaker, which is the key ML service and the backbone of AWS ML. We will also see how to use this infrastructure to train, build, and deploy an ML model.

**AWS ML services: Amazon SageMaker**

Amazon SageMaker is a fully managed service for building, deploying, and managing ML models on the AWS platform. It provides a variety of tools and features for data preparation, model training, and deployment, as well as pre-built algorithms and models. One of the main features of SageMaker is its ability to train ML models in a distributed manner, using multiple machines in parallel, allowing customers to train large models quickly and at scale. SageMaker also provides a variety of pre-built algorithms and models, such as image classification, object detection, and NLP, which can be easily integrated into a customer’s application.

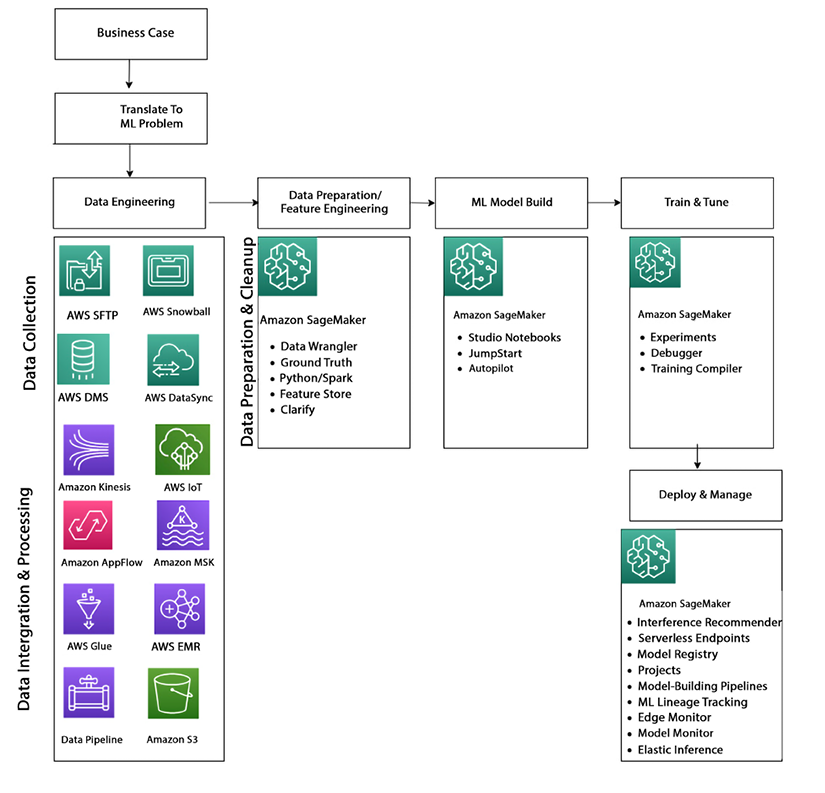
You can use SageMaker to train a model to predict product prices based on historical sales data. A retail company could use SageMaker to train a model on a dataset containing information about past sales, such as the date, product, and price. The model could then be deployed and integrated into the company’s e-commerce platform, allowing it to predict prices for new products and adjust prices in real time based on demand. If you want to improve the accuracy of detecting and classifying objects in images, you can use SageMaker to train object detection models using a dataset of labeled images.

Amazon SageMaker offers SageMaker Studio, an integrated development environment accessible through a web browser, to create, train, and deploy ML models on AWS. The studio provides a single, web-based interface that simplifies the process of building, training, deploying, and monitoring ML models.

The end-to-end ML pipeline is a process that takes a business use case and turns it into a working ML model. The pipeline typically consists of several stages, including data engineering, data preparation, model training, model evaluation, and deployment. AWS SageMaker enables this process by providing a suite of services that support each stage of the ML pipeline. The ML pipeline typically includes the following stages:

1. **Business use case**: Identifying a business problem that can be solved using ML.
2. **Data engineering**: Collecting and integrating the required data, which is often stored in Amazon S3 due to its ability to store large amounts of data with durability and reliability. Other AWS services, such as AWS Data Migration Service, AWS DataSync, and Amazon Kinesis, can also help with data integration.
3. **Data preparation**: Cleansing, transforming, and pre-processing the data to prepare it for model training.
4. **Model training**: Training the ML model using an assortment of algorithms, such as supervised and unsupervised learning, on the prepared data.
5. **Model evaluation**: Evaluating the performance of the trained model using metrics such as accuracy, precision, and recall.
6. **Model deployment**: Deploying the trained model to a production environment, where it can be used to make predictions or decisions.

As shown in the diagram below, SageMaker provides a suite of services for each stage in the ML. SageMaker provides a Jupyter-based notebook environment that allows data scientists and developers to interactively work on their ML models and quickly iterate on their experiments. SageMaker provides various monitoring and debugging tools, such as real-time metrics and logging, allowing customers to easily monitor and troubleshoot their models. SageMaker also provides a variety of pre-built algorithms and models, such as image classification, object detection, and NLP, which can be easily integrated into a customer’s application.



*Figure 12.2: AI/ML Pipeline and Amazon SageMaker*

As shown in the diagram above, Amazon SageMaker’s capabilities can be understood by diving into four main categories: data preparation; model building; training and tuning; and deployment and management.

**ML data preparation**

Data preparation, also known as data pre-processing or feature engineering, is a crucial step in the ML pipeline. It involves cleaning, transforming, and preparing the data for model training. It is essential to understand the data first and figure out what kind of preparation is needed.

One common feature engineering case is dealing with missing data. Ignoring missing data can introduce bias into the model or impact the model’s quality. There are several techniques that can be used to handle missing data, such as imputing the missing values with the mean or median of the data or using ML algorithms that can handle missing data. Another important case is when dealing with imbalanced data, where one class has significantly more samples than others. This can impact the model’s performance and can be addressed by oversampling the small dataset, undersampling the large dataset, or using techniques such as cost-sensitive learning or synthetic data generation.

Outliers can also have a negative impact on the model’s performance and can be handled by techniques such as data transformation, removing the outliers, or using robust models. Text-based data can also be transformed into numerical columns through techniques such as one-hot encoding or word embedding.

Data preparation is a critical step in the ML pipeline, and feature engineering techniques can be used to improve the quality and effectiveness of the model. SageMaker provides a Jupyter-based notebook environment and data preparation tools, such as Data Wrangler, to make it easy to prepare data for ML.

**Data Wrangler** is a tool within SageMaker Studio that allows data scientists and developers to easily prepare and pre-process their data for ML visually and interactively. With Data Wrangler, you can import, analyze, prepare, and add features to your data with no or minimal coding. Data Wrangler provides a simple and intuitive user interface that allows you to perform common data preparation tasks such as filtering, renaming, and pivoting columns, as well as more advanced tasks such as feature engineering, data visualization, and data transformations. Data Wrangler also integrates with other AWS services, such as Amazon S3 and Amazon Redshift, making it easy to import and export data from various data sources. Additionally, Data Wrangler allows you to add custom scripts and data transformations, providing flexibility and extensibility for your data preparation needs. After data preparation, the next step is model building.

**ML model building**

Amazon SageMaker Studio notebooks are a popular service within SageMaker that allow data scientists and ML engineers to build ML models without worrying about managing the underlying infrastructure. With Studio notebooks, data scientists and developers can effectively collaborate on their ML models and promptly refine their experiments by using a Jupyter-based notebook environment that facilitates interactive work.

One of the key features of Studio notebooks is single-click sharing, which makes collaboration between builders very easy. Studio notebooks also support a variety of popular ML frameworks, such as PyTorch, TensorFlow, and MXNet, and allow users to install additional libraries and frameworks as needed. In addition to Studio notebooks, SageMaker also provides other “no-code” or “low-code” options for building models. **SageMaker JumpStart**, for example, provides pre-built solutions, example notebooks, and pre-trained models for common use cases, making it easy for customers to get started with ML.

**SageMaker Autopilot** is another offering that allows customers to automatically create ML models to build classification and regression models quickly. It automatically pre-processes the data, chooses the best algorithm, and tunes the model, making it easy for customers to get started with ML even if they have no experience. These tools make it easy for customers to quickly build, train, and deploy ML models on the AWS platform. Amazon SageMaker provides a variety of built-in algorithms that can be used for various ML tasks such as classification, regression, and clustering. Some of the popular built-in algorithms provided by SageMaker are:

* **Linear Learner**: A supervised learning algorithm that can be used for classification and regression tasks.
* **XGBoost**: A gradient-boosting algorithm that can be used for classification and regression tasks.
* **Random Cut Forest**: An unsupervised learning algorithm that can be used for anomaly detection.
* **K-Means**: A clustering algorithm that can be used to group similar data points.
* **Factorization Machines**: A supervised learning algorithm that can be used for classification and regression tasks.
* **Neural Topic Model**: An unsupervised learning algorithm that can be used for topic modeling.

The algorithms above are designed to work well with large datasets and can handle sparse and dense data. These built-in algorithms can quickly train and deploy models on SageMaker and can be easily integrated into a customer’s application. Additionally, SageMaker allows customers to use custom algorithms or bring their pre-trained models.

After building the model, the next step is to train and tune it.

**ML model training and tuning**

After building an ML model, it needs to be trained by feeding it with training data as input. This process may involve multiple iterations of training and tuning the model until the desired model quality is achieved.

Automated ML training workflows establish a consistent process for managing model development steps to accelerate experimentation and model retraining. With **Amazon SageMaker Pipelines**, you can automate the complete workflow for building models, starting from data preparation and feature engineering, to model training, tuning, and validation. You can schedule SageMaker Pipelines to run automatically, triggered by specific events or on a predetermined schedule. Additionally, you can launch them manually when required.

**SageMaker Automatic Model Tuning** uses thousands of algorithm parameter combinations that are automatically tested to arrive at the most precise predictions, ultimately saving weeks of time and effort.

**Amazon SageMaker Experiments** is a service that helps to capture, organize, and compare every step of the experiment, which makes it easy to manage and track the progress of the model-training process. It allows data scientists and developers to keep track of different versions of the model, compare their performance, and select the best version to deploy.

**Amazon SageMaker Debugger** is another service that helps to debug and profile the training data throughout the training process. It allows data scientists and developers to detect and diagnose issues during training, such as overfitting or underfitting, by providing real-time metrics and alerts on commonly occurring error scenarios, such as too large or small parameter values.

SageMaker Experiments and SageMaker Debugger work together to provide an end-to-end solution for managing and tracking the ML pipeline. You can learn more about model training using SageMaker by visiting the AWS user docs here – <https://aws.amazon.com/sagemaker/train/?>.

Now your ML model is ready and it is time to deploy it in production.

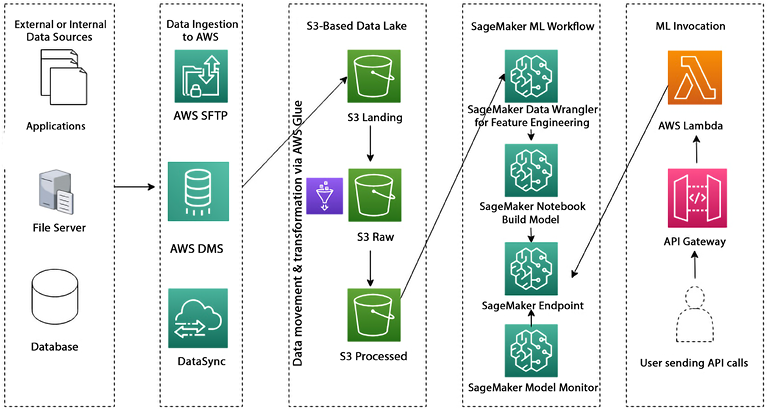
**ML model deployment and monitoring**

Once you are satisfied with the quality of the ML model that you have built, it is time to deploy it in a production environment to realize its business benefits. Amazon SageMaker provides several options for deploying models, including:

* **SageMaker Endpoints**: A serverless option to host your ML model with automatic scaling capability. This means that the number of instances used to host the model will automatically adjust based on the number of requests it receives.
* **SageMaker Projects**: A service that helps to create end-to-end ML solutions with **continuous integration and continuous deployment** (**CI/CD**) capabilities. This allows for easy collaboration and version control of the models and code.
* **SageMaker Model Monitor**: A service that allows you to maintain the accuracy of deployed models by monitoring the model quality, data quality, and bias in production. It also allows you to detect and diagnose issues with the deployed model, such as drift or bias, and take corrective actions.

Amazon SageMaker provides a comprehensive and easy-to-use platform for deploying ML models in production, with built-in support for serverless hosting, CI/CD, and monitoring and debugging capabilities to ensure the accuracy of the deployed models over time.

The following reference architecture example explains an end-to-end ML pipeline implemented using AWS SageMaker services:



*Figure 12.3: End-to-end ML Pipeline in AWS*

This reference architecture is an example of how an end-to-end ML pipeline can be implemented using AWS SageMaker services:

1. **Data ingestion**: Various types of data are ingested into an Amazon S3-based data lake. This source data can originate internally in an organization or come from external source systems. The data is typically stored in three S3 buckets, and AWS Glue is used to integrate and transform the data in the lake.
2. **Data preparation**: Amazon SageMaker then feature engineers the data in the lake using Data Wrangler. This step involves cleaning and pre-processing the data and adding any necessary features to the dataset.
3. **Model building**: Using SageMaker notebooks, data scientists and ML engineers can create and train models using the prepared data. This step involves selecting an appropriate algorithm and instances, adjusting parameters to optimize performance, and training the model.
4. **Model deployment**: Once the model is trained, it is deployed as an endpoint in production using SageMaker Endpoints. The endpoint is then ready to be invoked in real time.
5. **Real-time invocation**: To invoke the SageMaker endpoint, an API layer is created using AWS API Gateway and AWS Lambda functions. This allows the endpoint to be accessed by various applications and systems in real time. You can also use Amazon SageMaker Batch Transform to perform batch inference on large datasets. With Batch Transform, you can easily process large volumes of data and get inference results in a timely and cost-effective manner.

This reference architecture demonstrates how AWS SageMaker services can be used to create an end-to-end ML pipeline, from data ingestion and preparation to model building, deployment, and real-time invocation. Let’s look at AWS AI services and where to use them; you only need a little ML knowledge.

**AWS AI services**

In *Figure 12.1*, AI services are at the top of the stack, representing pre-trained AI services that can be easily integrated into your applications or workflows with just a few API calls without the need for ML, specialized skills, or training. These services are divided into several categories as you will learn in the subsections below.

**Vision**

Services such as Amazon Rekognition and Amazon Textract provide image and video analysis, object and scene recognition, and text extraction capabilities.

Amazon Rekognition is a deep learning-based computer vision service that allows you to analyze images and videos and extract information from them, such as objects, people, text, scenes, and activities. This service can be used for a range of use cases, such as object recognition, image search, facial recognition, and video analysis.

With Rekognition, you can upload an image or video to the API, which will analyze it and return information about the content. The service can detect and identify objects, people, text, and scenes in the image or video. It can also detect and recognize activities, such as a person riding a bike or a dog playing fetch. It can detect inappropriate content in an image or video, making it suitable for use in sensitive applications such as child safety.

One of the most powerful features of Rekognition is its facial recognition capabilities. The service can perform highly accurate facial analysis, face comparison, and face searching. It can detect, analyze, and compare faces in both images and videos, making it well suited for various applications, such as user verification, cataloging, people counting, and public safety.

You can use it to build applications for facial recognition-based security systems, photo tagging, and tracking people in a video.

**Amazon Textract** extracts text and data from nearly any type of document. It can process documents in various formats, including PDFs, images, and Microsoft Office documents. The service can automatically detect and extract text, tables, forms, and even handwriting and symbols, which can greatly reduce the manual tasks of reading, understanding, and processing thousands of documents.

The service can extract text and data from various documents such as invoices, contracts, and forms. The extracted data can be used to automate document-based workflows, such as invoice processing and data entry. It can also be used to extract structured data from unstructured documents to improve business processes and decision-making.

Textract can be integrated with other AWS services, such as Amazon SageMaker, to build ML models that can automatically classify and extract data from documents. It can also be integrated with Amazon Comprehend to extract insights from text, such as sentiment analysis and entity recognition. Additionally, it can be integrated with Amazon Translate to translate text automatically.

Textract can greatly improve the efficiency and accuracy of document-based workflows by automating the process of extracting text and data from documents. This can help organizations to save time, reduce costs, and improve the accuracy of their business processes.

**Speech**

Services such as Amazon Transcribe and Amazon Polly provide automatic speech recognition capabilities, including transcription and translation of audio files in multiple languages and formats.

**Amazon Polly** is a text-to-speech service that uses deep learning-based technologies to convert text into lifelike speech. Polly enables you to develop applications with speech capabilities, which allows you to create novel speech-enabled products and services.

The service supports a wide range of natural-sounding voices in multiple languages and dialects, including English, Spanish, Italian, French, German, and many more. You can also customize the pronunciation of words and use **Speech Synthesis Markup Language** (**SSML**) to control the generated speech’s pitch, speed, and volume.

Amazon Polly is particularly useful for creating applications that are accessible to visually impaired users or for situations where reading is not possible, such as while driving or working out. It can be used to create voice-enabled applications such as audiobooks, language learning applications, and voice assistants.

Polly can be integrated with other AWS services to create more advanced applications. Polly is a powerful service that allows you to add natural-sounding speech to your applications, making them more accessible and user-friendly.

**Amazon Transcribe** employs automatic speech recognition technology to transcribe speech into text. It can transcribe audio files in various languages and formats, including MP3, WAV, and OGG. Additionally, the service can automatically detect the primary language in an audio file and produce transcriptions in that language. You can also add custom vocabulary to support generating more accurate transcriptions for your use case.

Transcribe can be used to transcribe audio files in a wide range of applications, such as media and entertainment, education, and business. For example, it can be used to transcribe audio from podcasts, videos, and webinars to make them more accessible to a wider audience. It can also be used to transcribe speech in customer service calls, focus groups, and interviews to gain insights into customer needs and preferences.

One of the notable capabilities of Transcribe is the capacity to construct and train your own custom language model tailored to your specific use case and domain. Custom language models allow you to fine-tune the transcription engine to your specific use case by providing additional context about the vocabulary, grammar, and pronunciation specific to your domain. It allows you to improve transcription accuracy, even in noisy or low-quality audio environments.

Amazon Transcribe is a powerful service that allows you to transcribe audio files into text quickly and accurately. It can help you to make audio content more accessible, improve customer service, and gain insights into customer needs and preferences. For example, you can use Amazon Transcribe to transcribe speech to text and then use Amazon Comprehend to analyze the text to gain insights, such as sentiment analysis or key phrase extraction.

**Language**

Services such as Amazon Comprehend and Amazon Translate provide NLP capabilities, including sentiment analysis, entity recognition, topic modeling, and language translation.

**Amazon Translate** is an NLP service that uses deep learning-based neural machine translation to translate text from one language in to another. The service supports a wide range of languages, including English, Spanish, Chinese, French, Italian, German, and many more. One of the key features of Translate is its ability to support various content formats, including unstructured text, PowerPoint presentations, Word documents, Excel spreadsheets, and more. This makes it easy to integrate into a wide range of applications and workflows.

Amazon Translate can be used in a variety of applications, such as e-commerce, customer service, and content management. For example, it can be used to automatically translate product descriptions on a website to make them more accessible to a global audience. It can also be used to translate customer service emails and chat transcripts to improve communication with customers who speak different languages.

Translate can be integrated with other AWS services, such as Amazon Transcribe, Amazon Comprehend, and Amazon Transcribe Medical, to create more advanced applications. It can help you to expand your reach to a global audience, improve customer service, and streamline content management.

**Amazon Comprehend** is an NLP service that allows you to extract insights from unstructured text data. It uses ML algorithms to automatically identify key phrases, entities, sentiments, language, syntax, topics, and document classifications.

This service can be used in a wide range of applications, such as social media monitoring, sentiment analysis, content classification, and language detection. For example, you can use Amazon Comprehend to analyze customer feedback in social media posts to understand customer sentiment and identify common themes. It can also be used to process financial documents, such as contracts and invoices, to extract key information and classify them into different categories.

One of the key advantages of Comprehend is its ability to process text in multiple languages. The service supports many languages, including English, Spanish, Chinese, French, Italian, German, and many more. This allows you to analyze text data in its original language without requiring manual translation. Comprehend can help you to understand customer sentiment, identify common themes, classify content, and process text data in multiple languages. With this service, you can gain a deeper understanding of your text data to make data-driven decisions.

**Chatbots**

Services such as Amazon Lex provide NLP and automatic speech recognition capabilities for building chatbots and voice-enabled applications.

**Amazon Lex** allows you to build conversational chatbots for your business using natural language understanding and automatic speech recognition technologies. With Amazon Lex, you can create chatbots that understand the intent of the conversation and its context, allowing them to respond to customer inquiries and automate simple tasks. One of the key benefits of Lex is that it makes it easy to build chatbots without requiring specialized skills in ML or NLP. The service has a visual drag-and-drop interface, pre-built templates, and pre-trained models to help you quickly create and deploy chatbots.

Lex chatbots can be integrated with a variety of platforms, including mobile and web applications, messaging platforms, and voice assistants. They can also be integrated with other AWS services, such as Amazon Lambda and Amazon Connect, to perform tasks like data lookups, booking appointments, and more. Some common use cases for Lex include customer service, e-commerce, and lead generation. For example, you can use Lex to create a chatbot that can answer customer questions, help customers find products, and even place orders.

Amazon Lex makes it easy to create chatbots that understand the intent of the conversation and automate simple tasks. With Amazon Lex, you can improve customer service, increase sales, and streamline business processes.

**Forecasting**

Services such as Amazon Forecast provide time series forecasting capabilities, allowing you to predict future events or trends based on historical data.

**Amazon Forecast** is a fully managed service that uses ML to make accurate predictions based on time-series data. The service allows you to quickly create, train, and deploy forecasts without any prior ML experience. Amazon Forecast can be used to predict a wide range of business outcomes, such as demand forecasting, inventory management, and sales forecasting. For example, a retail company could use Amazon Forecast to predict demand for its products, allowing it to optimize inventory levels, reduce stockouts, and improve customer satisfaction.

One of the key benefits of Forecast is its ability to handle large amounts of data and make predictions at scale. The service can automatically process data from different sources, such as Amazon S3, Redshift, and Glue. It can handle data in various formats, such as CSV, JSON, and Parquet. Forecast also provides an easy-to-use web interface and APIs, which allow you to create, train, and deploy forecasts with just a few clicks. Additionally, it also provides pre-built models for everyday use cases such as demand forecasting, which eliminates the need for data scientists to build models from scratch.

Amazon Forecast is a powerful service that helps you make accurate predictions based on time-series data. With Forecast, you can improve demand forecasting and planning, optimize inventory management, and make data-driven decisions to drive business growth.

**Recommendations**

Services such as Amazon Personalize provide personalized recommendations based on user behavior and interactions.

**Amazon Personalize** is a fully managed service that makes it easy to create and deliver real-time personalized recommendations for your customers. This service utilizes ML to understand your customers’ behavior and predict the items they are most likely interested in.

With Personalize, you can easily build personalized experiences for your customers across a wide range of use cases, such as personalized product re-ranking, product recommendations, and customized direct marketing. For example, an e-commerce company can use Amazon Personalize to provide personalized product recommendations to its customers, increasing their likelihood of purchasing.

The service has a simple and intuitive interface, making it easy to use even for those without ML experience. It also includes pre-built models easily customizable to fit your business use cases. Personalize can process data from a variety of sources, including Amazon S3, Amazon DynamoDB, and Amazon Redshift. It also integrates seamlessly with Amazon Personalize campaigns, to deliver personalized recommendations in real time to your customers.

With its easy-to-use interface, customizable pre-built models, and the ability to process data from various sources, Personalize makes it easy to deliver a personalized experience to your customers, increasing engagement and driving sales.

You’ve now learned about various AWS AI/ML stacks and their use cases. It is essential to launch your model in production seamlessly and take action when there is any model drift. Let’s learn about **Machine Learning Operations** (**MLOps**) to understand how to put an ML model in production using AWS offerings.

**Building ML best practices with MLOps**

MLOps are the practices and tools used to manage the full lifecycle of ML models, from development to deployment and maintenance. The goal of MLOps is to make deploying ML models to production as seamless and efficient as possible.

Managing an ML application in production requires a robust MLOps pipeline to ensure that the model is continuously updated and relevant as new data becomes available. MLOps helps automate the building, testing, and deploying of ML models. It manages the data and resources used to train and evaluate models, apply mechanisms to monitor and maintain deployed models to detect and address drift, data quality issues, and bias, and finally enables communication and collaboration between data scientists, engineers, and other stakeholders.

The first step in implementing MLOps in AWS is clearly defining the ML workflow, including the data ingestion, pre-processing, model training, and deployment stages. The following are the key MLOps steps for managing an ML application in production using AWS:

1. **Set up a data pipeline**: AWS offers a wide range of services for data pipeline management, such as AWS Glue, Amazon Kinesis, and Amazon S3, which can be used to automate data ingestion, pre-processing, and storage. Use Amazon SageMaker Data Wrangler for data engineering.
2. **Use SageMaker for model training**: Use SageMaker for ML model training and deployment. As you learned, it provides a variety of built-in algorithms and tools for feature engineering, model training, and hyperparameter tuning. Use Amazon SageMaker Pipelines to build a training pipeline.
3. **Automate model testing and validation**: Use SageMaker Ground Truth, SageMaker Debugger, and SageMaker Experiments to automate the testing and validation of your models.
4. **Implement CI/CD**: Use AWS CodePipeline and CodeBuild to automate the continuous integration and deployment of your ML models so that you can quickly and easily update your models as new data becomes available. Use source control management tools like Git to store and manage your ML code and version control.
5. **Monitor and maintain your models**: Use Amazon CloudWatch and Amazon SageMaker Model Monitor to monitor the performance of your models in production and take action when there is any model drift.
6. **Deploy models in real time**: Use Amazon SageMaker endpoints to deploy your models and make real-time predictions.
7. **Use auto-scaling**: Use auto-scaling to adjust the number of instances based on the traffic automatically.
8. **Security and Compliance**: Use SageMaker’s built-in security features to ensure that your data and models are protected, and comply with relevant industry and regulatory standards.

By following the above best practices, you can ensure that your ML models are built, trained, and deployed as efficiently and effectively as possible and perform well in production. You can learn more about how to build MLOps using Amazon SageMaker by referring to the AWS page here – <https://aws.amazon.com/sagemaker/mlops/>.

As you learned about AI/ML in this section, now let’s learn about the next technology trend, IoT, which is becoming mainstream and driving the modern-day industrial revolution.

**What is IoT?**

IoT stands for “Internet of Things,” and it refers to the idea of connecting everyday devices to the internet so that they can share data and be controlled remotely. A simple example of this would be a smart thermostat in your home. A smart thermostat is a device you can control from your phone; it learns your temperature preferences and can even detect when you’re away and adjust the temperature accordingly to save energy.

So, instead of manually adjusting the temperature, you can control it remotely using your phone or voice commands. This is just one example of the many ways that IoT can make our lives more convenient and efficient. Another example is a smart fridge, which can keep track of your groceries and alert you when you’re running low on certain items or even order them for you automatically.

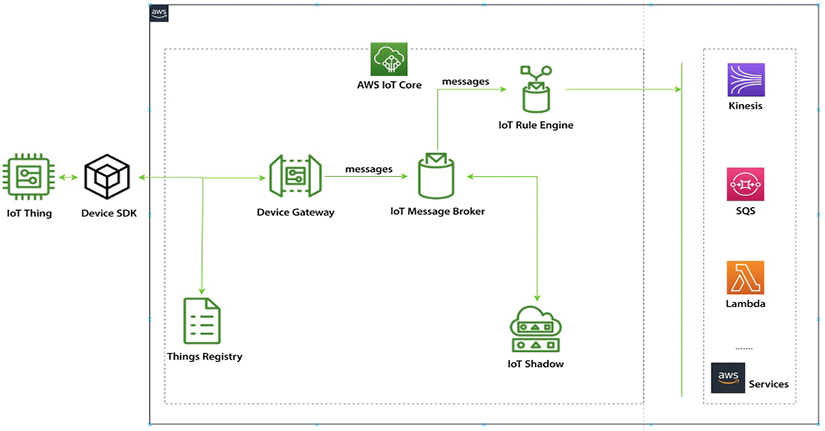
IoT refers to the interconnectedness of everyday physical objects, such as devices, vehicles, and buildings, to the internet through sensors and other technologies. These connected devices are able to collect and share data with each other and with other systems, such as cloud-based servers and analytics platforms. This allows for the creation of smart systems and applications that can be used for a wide range of use-cases, such as monitoring, control, automation, and prediction.

Another real-life example is fleet management; you can connect your fleet of trucks to the internet. This allows you to track and monitor them remotely, giving you insights and control over them. You can put IoT sensors on the rented trucks, which send information such as location back to the company. This allows the company to easily track the location of their trucks in near real time without having to check on them physically. This can help with logistics, maintenance, and security.

IoT can also be used in many industries, such as manufacturing, healthcare, transportation, and retail. For example, IoT can monitor and control machines and equipment in manufacturing, predict when maintenance is needed, and improve overall efficiency. In healthcare, IoT can monitor patients remotely and improve patient outcomes. In transportation, IoT can be used to optimize routes and improve fleet management. And in retail, IoT can be used to track inventory and optimize supply chain logistics. Overall, IoT has the potential to revolutionize many industries by providing valuable insights and automating processes. Let’s learn how AWS can help to implement IoT workloads.

**Building IoT applications in AWS**

AWS IoT is a platform that allows you to connect, monitor, and control millions of IoT devices. It provides services that allow you to easily and securely collect, store, and analyze data from IoT devices. Let’s understand AWS IoT services by looking at the following architecture diagram.



*Figure 12.4: AWS IoT Services*

As shown above, the process of connecting IoT devices to AWS typically starts with connecting the devices to AWS IoT Core. This can be done using different protocols such as MQTT, HTTP, and WebSocket. Once connected, the devices send data to the IoT Core service, which then securely transmits it to the IoT Message Broker using the Device Gateway. The IoT Rules Engine filters and processes the data, which can then be sent to other AWS services for storage, analysis, and visualization. Even when the devices are offline, the IoT Shadow service can be used to track their status. This architecture allows for real-time data processing and analysis, as well as the secure communication and management of IoT devices in the cloud.

AWS provides several IoT services to help developers and businesses connect, manage, and analyze IoT devices and data. Let’s learn about some of the main AWS IoT services here.

**AWS IoT Core**

IoT Core is the foundation of the AWS IoT platform and allows you to securely connect, manage, and interact with your IoT devices. The various components of AWS IoT Core are:

* **Things Registry**: A database to store metadata about the devices, such as the device type, serial number, location, and owner.
* **Device Gateway**: A gateway to manage bidirectional connections and communications to and from IoT devices using protocols such as MQTT, WebSocket, and HTTP.
* **IoT Message Broker**: A component used to understand and control the status of devices at any time, providing two-way message streaming between devices and applications.
* **IoT Rules Engine**: A rules engine to connect IoT devices to other AWS services such as Kinesis, SQS, SNS, Lambda, and more, with data pre-processing capabilities such as transform, filter, and enrich.
* **IoT Shadow**: A virtual representation of a device at any given time, used to report the last known status of a device, change the state of a device, and communicate with devices using REST APIs.

**AWS IoT Device Management**

IoT Device Management is a service that helps to manage, organize, and monitor your IoT devices at scale. The service allows you to onboard and register your devices, organize them into groups, and perform actions on those groups. The first step in using the service is to onboard your devices. This process involves creating and registering device identities with the service, such as certificates and policies. You can also add metadata to each device, such as location, owner, and device type. This makes it easy to find and manage your devices later on.

Once your devices are registered, you can use the service to organize them into groups. These groups can be based on any combination of device attributes, such as location, hardware version, or device type. This makes it easy to perform actions on a specific subset of devices, such as sending a firmware update to all devices in a particular location. You can also use the service to monitor your fleet of devices. You can use Amazon CloudWatch to view the aggregate state of your devices periodically. This can help you identify and troubleshoot issues with your devices, such as low battery levels or connectivity issues.

In addition to these features, the service also allows you to create jobs that can be used to perform actions on a group of devices. For example, you can create a job to send a firmware update to all devices in a specific group or to reboot a group of devices. This can help to automate common device management tasks, making it easier to manage your fleet of devices at scale.

IoT Device Management allows you to easily onboard, organize, and manage your IoT devices at scale. It provides a range of tools and features that can help you to automate common device management tasks, such as device registration, grouping, monitoring, and job management. This service helps you to keep your IoT devices up to date, healthy, and responsive while keeping track of them.

**AWS IoT Analytics**

AWS IoT Analytics is managed entirely by AWS, which means that you don’t have to worry about the cost and complexity of building your own analytics infrastructure. With IoT Analytics, you can collect, pre-process, enrich, store, and analyze massive amounts of IoT data, which enables you to identify insights and improve your business. This service is designed to handle sophisticated analytics on IoT data, making it easier for you to act on the insights you gain. The service includes several key features that make it easy to work with IoT data, including:

* **Data collection**: IoT Analytics allows you to collect data from IoT devices and other sources, such as AWS IoT Core, Amazon Kinesis Data Streams, and Amazon S3. It also supports the ability to filter, transform, and enrich the data before storing it.
* **Data storage**: IoT Analytics automatically stores your data in a highly secure, scalable, and low-cost data store that is optimized for IoT data. It also provides a SQL-like query language for easy data access and exploration.
* **Data analysis**: IoT Analytics provides a set of pre-built IoT analytics models and tools, such as anomaly detection and forecasting, that you can use to gain insights from your data. It also supports the ability to run custom SQL queries, and R and Python code to analyze your data.
* **Data visualization**: IoT Analytics provides a built-in visualization tool that allows you to easily create visualizations of your data and share them with others.
* **Data pipeline automation**: IoT Analytics allows you to automate the entire data pipeline, from data collection to analysis and visualization, using AWS Step Functions and AWS Lambda.

AWS IoT Analytics also integrates with other AWS services, such as AWS IoT Core, Amazon Kinesis Data Streams, Amazon S3, Amazon QuickSight, and Amazon Machine Learning, to make it easy to build complete IoT solutions. IoT Analytics makes it easy to process and analyze IoT data at scale, so you can gain insights and make better decisions for your business.

**AWS IoT Greengrass**

IoT Greengrass is a service that allows you to run AWS Lambda functions and access other AWS services on connected devices securely and reliably. This enables you to perform edge computing, where you can perform local data processing and respond quickly to local events, even when disconnected from the cloud.

With IoT Greengrass, you can easily create and deploy Lambda functions to your connected devices and have them securely communicate with other devices and the cloud. The service also provides a local data store for your device data, allowing you to perform local analytics and ML on the data without needing to send it to the cloud constantly.

IoT Greengrass also provides a feature called Greengrass Discovery, which makes it easy for devices to discover and communicate with each other securely, regardless of their location. This is particularly useful for large-scale IoT deployments where devices are distributed across multiple locations.

In addition, IoT Greengrass provides a feature called Greengrass Core, which is a software component that runs on the connected device and provides the runtime environment for your Lambda functions and other AWS services. Greengrass Core provides a secure communication channel between the device and the cloud using mutual authentication and encryption.

IoT Greengrass allows you to build IoT solutions that can perform local data processing and respond quickly to local events while still being able to communicate with the cloud and other devices securely. This enables you to build IoT solutions that are highly responsive, secure, and reliable.

**AWS IoT Device Defender**

IoT Device Defender is a fully managed service that makes it easy to secure your IoT devices. It helps you to detect and mitigate security threats by continuously monitoring your IoT devices and alerting you to potential security issues. The service includes a set of security checks that are performed on your devices to ensure they meet best practices for security. These checks include ensuring that devices use the latest security protocols, have strong authentication and access controls, and are free from malware and other threats.

IoT Device Defender also allows you to set-up custom security policies that can be used to detect and mitigate security threats. These policies can be configured to check for specific security issues, such as devices communicating with known malicious IP addresses or sending sensitive data to unauthorized recipients. When a security threat is detected, AWS IoT Device Defender can take a variety of actions to mitigate the threat. This can include things like blocking traffic from a device, shutting down a device, or sending an alert to the device administrator.

In addition to these security features, AWS IoT Device Defender also provides a range of reporting and analytics capabilities that allow you to track the security status of your devices over time.

This can help you to identify trends and patterns in security threats and to take proactive steps to improve the security of your IoT devices.

IoT Device Defender helps to secure your IoT devices and protect your business from security threats. It is easy to use and can be integrated with other AWS services to provide a comprehensive security solution for your IoT devices.

**AWS IoT Things Graph**

AWS IoT Things Graph is a service that makes it easy to visually connect devices, web services, and cloud-based systems using a drag-and-drop interface. The service allows you to quickly create and deploy IoT applications without writing any code, making it easy to connect devices, cloud services, and web services to create IoT workflows. One of the key features of IoT Things Graph is its ability to connect different devices and services easily. This is achieved through Things, which are pre-built connectors for various devices and services. These Things can be dragged and dropped onto a graph and wired together to create a workflow.

IoT Things Graph also provides a set of pre-built flow templates that can quickly create common IoT workflows such as device provisioning, data collection, and alerting. These templates can be customized to suit the specific needs of your application. In addition to its visual interface, IoT Things Graph also provides a set of APIs that can be used to create and manage IoT workflows programmatically. This allows developers to integrate IoT Things Graph into their existing applications and systems.

IoT Things Graph also provides a set of security and compliance features to help ensure the security of your IoT deployments. These include role-based access controls, encryption of data in transit and at rest, and the ability to meet compliance requirements such as SOC 2 and PCI DSS. IoT Things Graph makes it easy to connect devices and services to create IoT workflows without coding. Its visual interface and pre-built templates make it easy for developers of all skill levels to create IoT applications. At the same time, its security and compliance features help ensure the safety of your data and systems.

**AWS IoT SiteWise**

AWS IoT SiteWise is managed by AWS and designed to simplify the process of collecting, storing, processing, and analyzing large amounts of data from industrial equipment. With IoT SiteWise, businesses can easily manage and analyze data from their industrial equipment at scale.

It allows you to create a centralized industrial data model to organize and analyze data from multiple industrial gateways and devices and provides pre-built connectors for standard industrial protocols such as OPC-UA and Modbus.

One of the key features of IoT SiteWise is its ability to create a standardized data model for industrial equipment. This allows users to easily access and analyze data from multiple devices and gateways, regardless of their individual data formats or protocols. It also allows users to create custom data models that align with their industrial use case.

SiteWise can also be used alongside other AWS services to collect, store, and process data in near real time using AWS IoT Core and AWS Lambda and easily visualize and analyze the data using Amazon QuickSight. Additionally, it also enables you to perform asset modeling and create hierarchical structures of industrial equipment, so you can easily navigate and understand the relationships between different assets. AWS IoT SiteWise also provides a set of pre-built connectors for common industrial protocols such as OPC-UA and Modbus, which allow you to connect industrial gateways and devices to the service easily.

AWS IoT SiteWise can also be integrated with other AWS services like Amazon S3, Amazon Kinesis, and Amazon SageMaker to build more advanced analytics and ML models and to perform more complex data processing and analysis.

AWS IoT SiteWise is a highly capable and adaptable service that simplifies the collection, storage, processing, and analysis of data from industrial equipment at scale. This is made possible through the use of a centralized industrial data model, pre-built connectors, and integration with other AWS services. With these tools, businesses can easily handle the complexities of industrial data, and quickly gain insights that can help them optimize their operations and improve their bottom line.

**AWS IoT TwinMaker**

IoT TwinMaker is a service that allows customers to create digital twins of their physical equipment and facilities. A digital twin is a digital replica of a physical asset that can be used to monitor, control, and optimize the performance of the physical asset. AWS IoT TwinMaker allows you to develop digital replicas of your equipment and facilities, which can be utilized to visualize forecasts and insights derived from data collected by connected sensors and cameras.

Moreover, the service incorporates pre-built connectors that make it effortless to access and use data from various sources, including equipment sensors, video feeds, and business applications.

You can also import existing 3D visual models to produce digital twins that update instantly with data from connected sensors and cameras. This service can be used across various industries, such as manufacturing, energy, power and utilities, and smart buildings.

**AWS Industrial IoT (IIoT)**

The **Industrial Internet of Things** (**IIoT**) refers to the use of IoT technology in industrial settings, such as manufacturing, energy, and transportation. It involves connecting industrial equipment, machines, and devices to the internet to collect and analyze data, automate processes, and improve efficiency and productivity.

AWS provides a variety of services for building and deploying IIoT solutions, such as AWS IoT Core for securely connecting and managing devices, AWS IoT Device Defender for securing device connections, AWS IoT Greengrass for running local compute, messaging, and data caching on IoT devices, and AWS IoT Things Graph for building IoT applications with pre-built connectors to AWS and third-party services. AWS provides industry-specific services for IIoT, such as Amazon Monitron for monitoring industrial equipment.

One use case of IIoT is in the manufacturing industry. A manufacturing company can use IoT sensors on their production line machines to collect data on machine performance, such as temperature, vibration, and power consumption. This data can be analyzed to identify patterns and anomalies, which can indicate potential maintenance issues. The company can proactively reduce downtime and increase productivity by proactively addressing these issues. The data can also be used to optimize the production process, resulting in improved efficiency and reduced costs.

Another use case of IIoT is in the energy industry. An energy company can use IoT sensors on their power generators to collect data on generator performance and energy usage. This data can be analyzed to identify patterns and anomalies, which can indicate potential maintenance issues. The company can reduce downtime and increase power generation efficiency by proactively addressing these issues. The data can also be used to optimize the energy distribution process, resulting in improved efficiency and reduced costs.

AWS provides a comprehensive and flexible set of services for building and deploying IIoT solutions, allowing customers to easily collect, analyze, and act on data from industrial equipment and devices, improving their operations’ efficiency and performance.

Let’s learn about best practices when building AWS IoT applications.

**Best practices to build AWS IoT applications**

When building an IoT application on AWS, you should keep the following best practices in mind:

* **Secure your devices**: Ensure that all your devices are correctly configured and have the latest security updates. Use AWS IoT Device Defender to monitor and secure your devices against potential security threats.
* **Use MQTT or HTTPS for communication**: These protocols are designed for low-bandwidth, low-power devices and are well suited for IoT applications.
* **Use AWS IoT Analytics to process and analyze your data**: This service provides tools for cleaning, filtering, and transforming IoT data before it is analyzed.
* **Store your data in the right place**: Depending on your use case, you may want to store your data in a time-series database like Amazon Timestream or a data lake like Amazon S3.
* **Use AWS IoT Greengrass for edge computing**: With Greengrass, you can run AWS Lambda functions on your devices, allowing you to process and analyze data locally before sending it to the cloud.
* **Use AWS IoT Things Graph to create visual, drag-and-drop IoT workflows**: This service allows you to quickly connect devices and AWS services to create IoT applications without writing any code.
* **Use AWS IoT Device Management to manage your fleet of devices**: This service allows you to easily onboard and organize your devices and trigger actions on groups of devices.
* **Use AWS IoT SiteWise to handle data from industrial equipment**: This service allows you to collect and organize data from industrial equipment and create a digital twin of your physical assets, helping you to improve your operations.
* **Use AWS IoT Device Defender to secure your IoT devices and data**: This service allows you to monitor and secure your devices against potential security threats.
* **Use AWS IoT EventBridge to route and process IoT data**: This service allows you to route and process IoT data efficiently and integrates with other AWS services like AWS Lambda, Amazon Kinesis, and Amazon SNS.

You learned about various IoT services and their use cases in this section. Now let’s learn about the next technology evolution in progress with blockchain and how AWS facilitates that with their platform.

**Blockchain in AWS**

Blockchain is a digital ledger that is used to record transactions across a network of computers. It is a decentralized system, which means that it is not controlled by any single entity, and it is highly secure because it uses cryptography to secure and validate transactions and keep them private. Each block in the chain contains a record of multiple transactions, and after a block has been added to the chain, it cannot be altered or deleted. This makes blockchain technology useful for a variety of applications, including financial transactions, supply chain management, and secure record-keeping.

Blockchain allows multiple parties to securely and transparently record and share information without a central authority. The most well-known use of blockchain technology is in creating digital currencies like Bitcoin, but it can be used for a wide range of applications, such as supply chain management, smart contracts, and voting systems. Blockchain technology is based on a network of computers that all have copies of the same data, which makes it difficult for any single party to manipulate or corrupt the information.

AWS offers AMB,which enables customers to easily create and manage blockchain networks, regardless of their chosen framework.

**AMB**: AMB is a service that is fully managed by AWS, which means that it simplifies the process of creating and managing scalable blockchain networks using popular open-source frameworks such as Hyperledger Fabric and Ethereum. This service enables customers to easily set up and manage a blockchain network using just a few clicks in the AWS Management Console, without requiring specialized expertise in blockchain technology. In short, AMB takes the complexity out of blockchain network management, allowing customers to focus on their core business activities.

The service also provides built-in security, scalability, and performance optimizations, enabling customers to easily create secure and reliable blockchain networks that can scale to meet the needs of their applications.

One of the key features of AMB is its ability to easily add new members to the network, enabling customers to easily collaborate with other organizations and share data securely and transparently. This can be done through a self-service portal, and customers can also use the service to set fine-grained permissions for different network members.

AMB also integrates with other AWS services such as **Amazon Elastic Container Service** (**ECS**) for data storage and container management, respectively.

This allows you to easily store and manage the data on your blockchain network and also perform analytics on the data using other AWS services such as Amazon QuickSight. It allows you to store and track data changes without the need for a central authority.

AMB also provides a set of APIs and SDKs that allow customers to interact easily with their blockchain network and integrate it with their applications and workflows. Some potential use cases for AMB include:

* **Supply chain traceability**: By using blockchain to record and track the movement of goods throughout the supply chain, companies can increase transparency and reduce the risk of fraud and errors.
* **Digital identity management**: Blockchain can create a decentralized system for managing digital identities, making it more secure and private than traditional centralized systems.
* **Tokenization**: Blockchain can create tokens representing assets such as company shares or a certificate of authenticity for a piece of art.
* **Smart contracts**: Blockchain can create “smart contracts” that execute automatically when certain conditions are met. This can automate processes such as insurance claims or real estate transactions.
* **Payment and financial services**: Blockchain can be used to create secure and efficient payment networks and facilitate cross-border transactions and remittances.

AMB makes it easy for customers to create and manage scalable blockchain networks without needing specialized expertise in blockchain technology. It enables you to collaborate with other organizations easily, share data securely and transparently, and also integrate your blockchain network with other AWS services for data storage and analytics.

In this section, you had a brief introduction to blockchain and AMB, which provides a platform to address blockchain-based use cases.

Quantum computing is another emerging technology waiting to revolutionize the world, but it may take the next 5 to 10 years to make it stable enough to solve daily use cases in production. AWS makes quantum computing accessible through AWS Braket. Let’s see an overview of it.

**Quantum computing with AWS Braket**

Quantum computing is a computing method that employs quantum-mechanical phenomena to execute data operations. In quantum computing, data is expressed as qubits, or quantum bits, which can simultaneously exist in multiple states.

This unique attribute empowers quantum computers to perform specific calculations much more rapidly than classical computers. Despite its potential, this technology is still in its infancy and necessitates specialized hardware and expertise to operate. Some of the key use cases where quantum computing can be very efficient are:

* **Drug discovery and materials science**: Quantum computing can be used to simulate complex chemical and biological systems, which can help in the discovery of new drugs and materials.
* **Financial modeling**: Quantum computing can solve complex financial problems such as portfolio optimization, option pricing, and risk analysis.
* **ML**: Quantum computing can be used to develop new algorithms for ML and AI, which can help solve complex tasks such as image recognition and NLP.
* **Supply chain optimization**: Quantum computing can optimize logistics and supply chain management by predicting demand, optimizing routes, and scheduling deliveries.
* **Cybersecurity**: Quantum computing can be used to develop new encryption methods that are resistant to hacking and to break existing encryption methods.

AWS Braket is a fully managed service from AWS that allows developers and scientists to experiment with quantum computing. It provides a development environment that allows users to explore and test quantum algorithms, circuits, and workflows using different quantum hardware technologies.

Braket supports quantum hardware from leading quantum hardware providers such as D-Wave, IonQ, and Rigetti Computing and allows users to access these devices through a unified interface. This enables users to test their quantum algorithms and workflows on different quantum hardware technologies and compare the results.

AWS Braket also includes a quantum development kit that allows users to write, simulate, and debug quantum algorithms using Python or Q#, a domain-specific quantum programming language developed by Microsoft. This allows users to test their algorithms on a simulator before running them on quantum hardware.

AWS Braket also has a built-in Jupyter notebook interface for running and visualizing quantum algorithms. It also integrates with other AWS services, such as Amazon S3, Amazon SageMaker, and Amazon ECS, to provide a complete development environment for quantum computing. Braket makes it easy for developers and scientists to experiment and explore the possibilities of quantum computing without significant investments in hardware and infrastructure.

Quantum computing is a complex topic and requires lots of details to understand the basics. To learn about quantum computing in simple language, please refer to *Solutions Architect’s Handbook* available on Amazon – <https://www.amazon.com/Solutions-Architects-Handbook-Kick-start-architecture/dp/1801816611/>.

The final emerging technology we will discuss in this chapter has gained massive momentum over recent months; let’s take a quick look at generative AI and AWS’s offerings in this domain.

**Generative AI**

With the launch of Chat**GPT** (**Generative Pre-trained Transformer**), generative AI has become the talk of the town. It has opened endless possibilities for revolutionizing the way we work today. This revolution is comparable to the innovation brought about by computers, and how the world moved from typewriters to shiny new computers, which made things more efficient. ChatGPT is just one dimension that shows the world the art of possibility and brings much-needed innovation that the world has been waiting for for a long time. Over the last two decades, you might have wondered who can challenge the position of Google in the AI market, especially Google Search. But, as you know, there is always a disrupter; if you don’t innovate fast enough, someone else will do it. ChatGPT has brought that innovation to the hands of everyone.

Let’s first understand what generative AI is. Generative AI uses AI algorithms to create new content that resembles content from a particular domain. This type of AI differs from other types of AI designed to recognize patterns or make predictions based on existing data. Generative AI is focused on creating new data that did not exist before. Generative AI can be used in various applications, from creating realistic images and videos to generating text and audio. For example, generative AI can be used to create realistic images of people, animals, or landscapes and generate new pieces of music or poetry that are similar to existing works. One of the advantages of generative AI is its ability to create new content that is personalized and unique. With generative AI, it is possible to create custom content tailored to a specific audience or user based on their preferences or other data.

Generative AIs using **LLMs** (**Large Language Models**) use these language models to generate new text that is similar in style and content to an existing text. These models are trained on large amounts of text data and can generate coherent, natural-sounding text in various contexts. It has many applications, from creating chatbots and virtual assistants, to understanding and responding to natural language queries, to generating text for marketing campaigns, and other content-creation tasks. With generative AIs that use LLMs, it is possible to create personalized, engaging content that resonates with users and drives engagement.

In recent years, everyone has been jumping onto the bandwagon of generative AI: whether it is Google with their LLM called **BERT** (**Bidirectional Encoder Representations from Transformers**), or Microsoft putting their weight behind OpenAI’s ChatGPT and embedding it in their Office 365 products and Bing search engine. While big tech companies are rolling out their offerings and launching hundreds of AI tools every day, Amazon has not been left behind; Amazon recently launched its own offering in this space with **Amazon Bedrock**.

Amazon Bedrock is an AWS service that helps you choose from different pre-built AI models that other companies make. These models can help you with tasks like making predictions or recognizing patterns. With Bedrock, you can easily customize these models with your data and put them into your applications without worrying about managing any technical stuff, such as model training using SageMaker or figuring out how to use IT infrastructure to train the model at scale. It’s easy to use, and you can integrate it with other AWS services you are already familiar with. You can learn more about Amazon Bedrock by referring to the AWS page here: <https://aws.amazon.com/bedrock/>.

The other generative code tool launched by AWS is **Amazon Codewhisperer**. It is an AI service trained on billions of lines of code. Based on your comments and existing code, it can quickly suggest code snippets or even entire functions in real time. This means you can save time on coding tasks and speed up your work, especially when working with unfamiliar APIs. You can learn more about Codewhisperer by visiting the AWS page here: <https://aws.amazon.com/codewhisperer/>.

However, generative AI also presents challenges, particularly when ensuring the generated content is high-quality and free from biases or other errors. As with all AI algorithms, it is crucial to carefully evaluate and test generative AI models to ensure they produce accurate and reliable results. In particular, generative AI models may sometimes produce biased or inappropriate content. It is essential to carefully review and edit the generated text to ensure it meets ethical and legal standards.

**Summary**

Organizations must drive innovation and stay agile by using emerging technologies to stay ahead of the competition. With cloud providers like AWS, these technologies are easily accessible for you to experiment with and add to your use case.

In this chapter, you began by learning about ML and AI. You learned how AWS services help build an end-to-end ML pipeline, taking an ML workload from ideation to production. You learned about three layers of AWS AI/ML services, starting with the ML infrastructure provided by AWS to train your model.

After that, you learned about Amazon SageMaker, which is at the center of the AWS ML tech stack to build, train, deploy, tune, and monitor ML models. Next, you learned about the top stack where AWS AI services reside, providing pre-trained models that can be used with simple API calls without any knowledge of ML. This AI service is available to address multiple use cases involving vision, speech, chatbots, forecasting, and recommendations. Finally, you learned about applying ML best practices using MLOps to productionize the ML workload.

You then dove deep into the next emerging technology, IoT, and how AWS helps to build IoT applications with services like IoT Core, IoT Analytics, IoT Device Management, IoT SiteWise, and so on. You learned about the 10 best practices for building IoT applications in AWS.

Next, you learned about AMB and how it helps build blockchain applications. You then learned about the next technology revolution in the making, quantum computing, and how AWS Braket makes it accessible for everyone to experiment with quantum computing. Finally, you had a brief look at generative AI, and the AWS tools on offer in this domain.

Containers are becoming key to optimizing cost and increasing efficiency. In the next chapter, you will learn about container patterns and AWS services for container workloads.