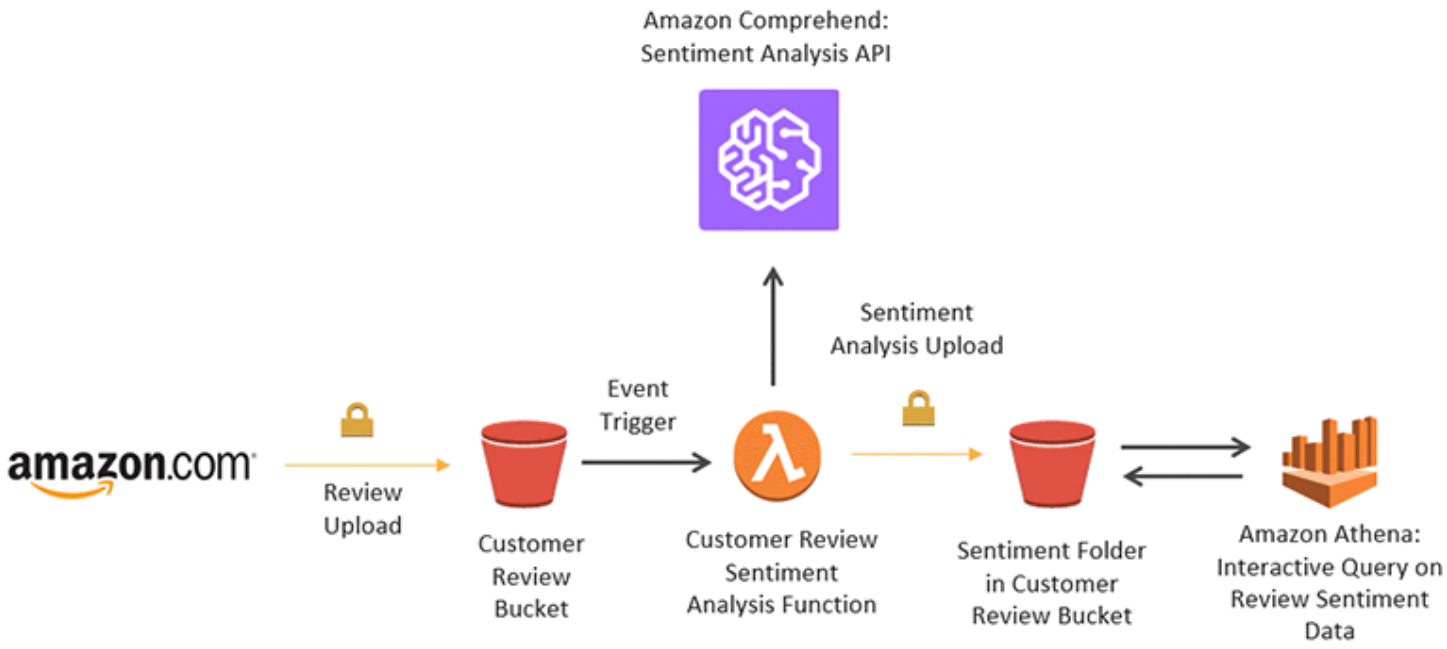
AWS Comprehend Reference Architecture

Amazon Comprehend can be integrated into a cloud workflow to analyze large-scale customer reviews automatically.

The diagram below shows a reference architecture provided by AWS:



Workflow Explanation

1. **Customer Review Upload**: Reviews are uploaded into an Amazon S3 bucket.
2. **Event Trigger**: An event (new file in bucket) triggers a Lambda function.
3. **Sentiment Analysis**: The Lambda function calls the **Amazon Comprehend API** to analyze sentiment.
4. **Results Storage**: The sentiment results are saved into another folder in S3.
5. **Querying**: Amazon Athena allows interactive querying of sentiment results directly from S3.

In this project, the workflow is simplified:

- Data preprocessing and uploads were performed manually.

- Sentiment analysis was handled through **Amazon Comprehend** and traditional ML models in SageMaker.

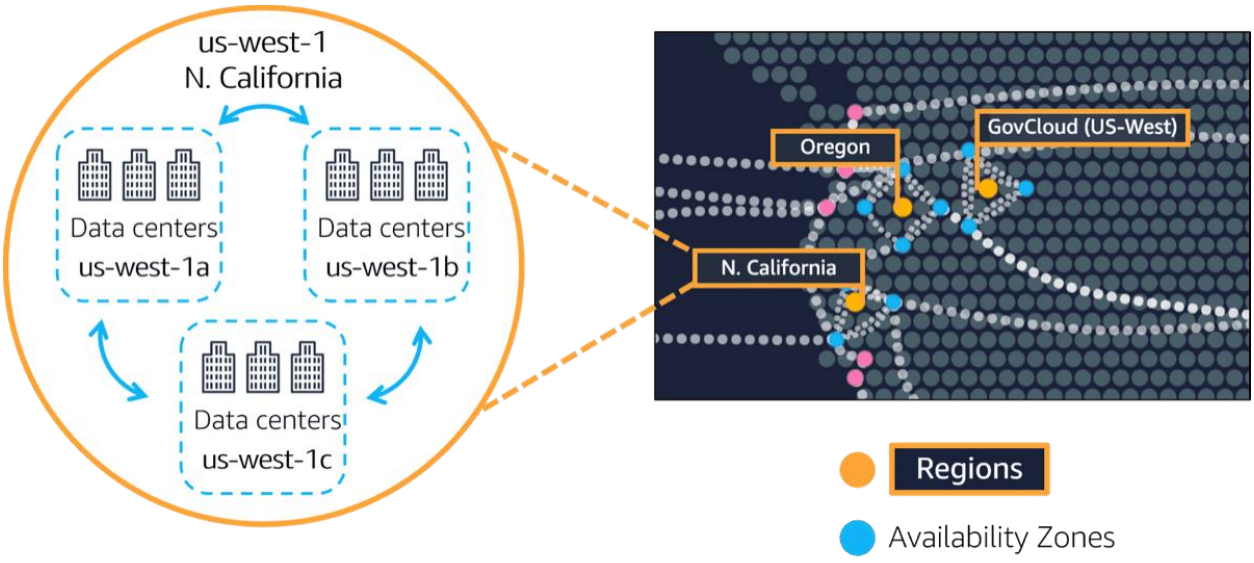
- Visualization and evaluation were completed directly in Google colab notebooks instead of Athena.

AWS Regions and Availability Zones

AWS divides its global infrastructure into **Regions** and **Availability Zones (AZs)**.

- A **Region** is a physical geographical area (e.g., *US West (N. California), Europe (London), Asia Pacific (Tokyo)*).
- Each Region has multiple **Availability Zones (AZs)**. An AZ is essentially a **data center** with independent power, cooling, and networking.

This design ensures **high availability**: if one data center (AZ) fails, others in the same Region can still keep your workloads running.



Why it matters for this project:

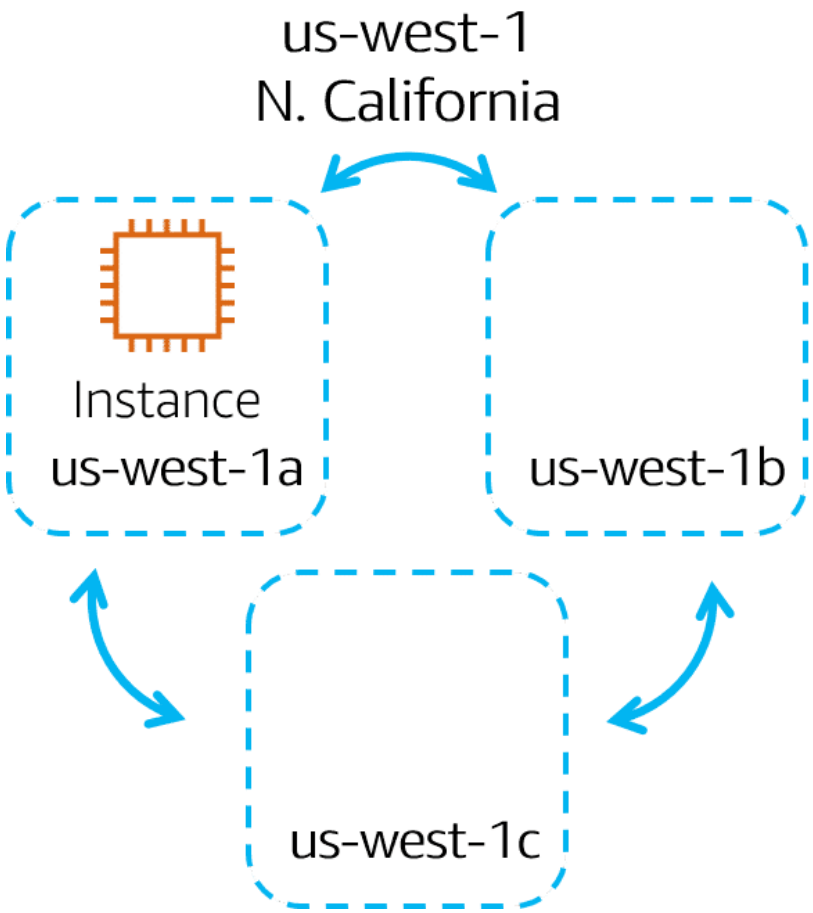
- When creating S3 buckets or SageMaker training jobs, you must choose a Region (e.g., us-west-1 or eu-west-2).
- All services (Comprehend, S3, SageMaker) should be in the same Region to avoid latency and extra costs.

AWS Region, Availability Zones, and SageMaker Instance Resilience

AWS regions and availability zones (AZs) underpin the global infrastructure that ensures high reliability, performance, and fault tolerance.

- A **Region** is a geographic location that contains multiple, physically separated, **Availability Zones (AZs)**.
- Each AZ operates independently with its own power, networking, and cooling systems.
- AWS recommends deploying resources across multiple AZs to enhance application availability—if one AZ fails due to an outage or hardware issue, the others remain operational.

Single Instance in One Availability Zone

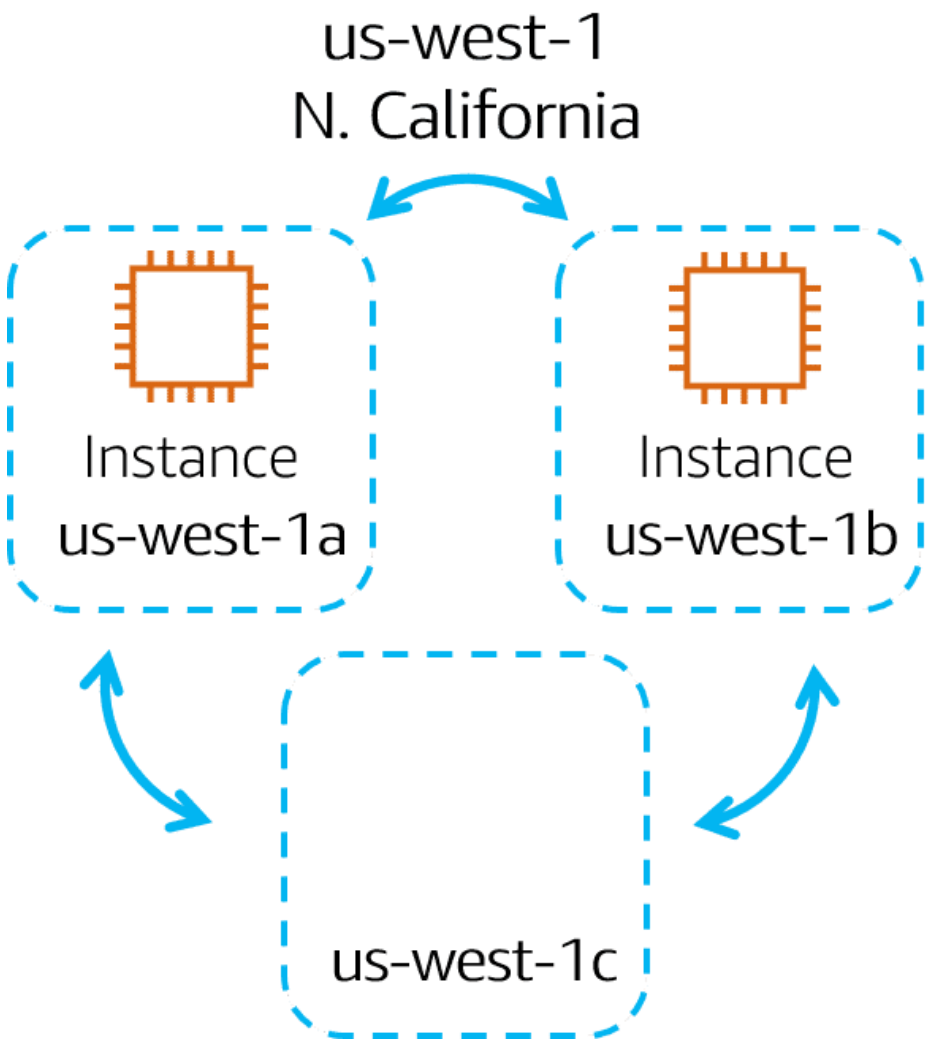


This diagram shows a single SageMaker instance running in us-west-1a (N. California).If that Availability Zone (AZ) goes down, the instance becomes unavailable, creating a single point of failure.

Multiple Instances Across AZs

Here, SageMaker is running **multiple instances** across different Availability Zones (e.g., us-west-1a and us-west-1b).

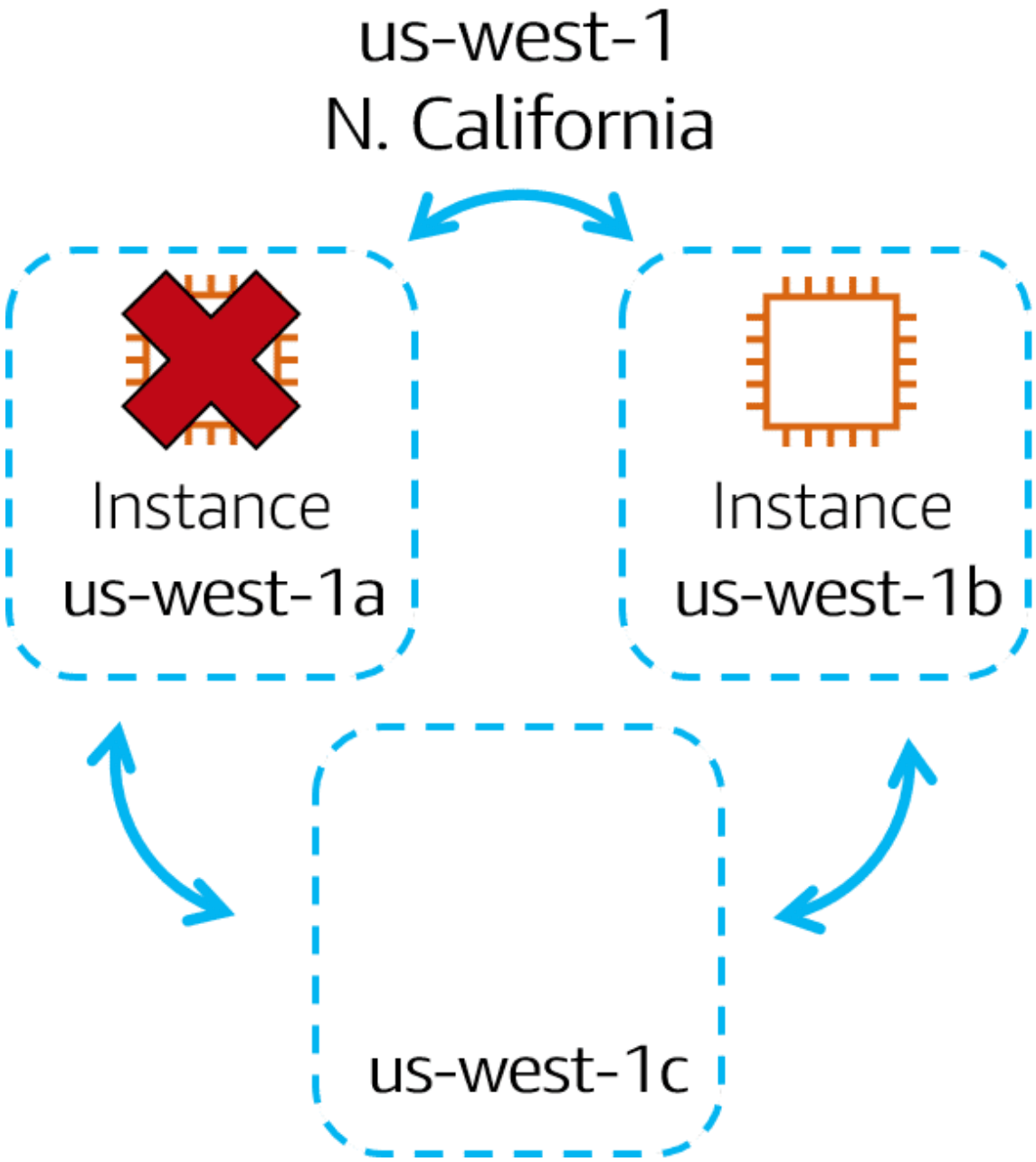
If one AZ fails, the training job or inference service can continue running in the other AZs. This setup provides **higher availability and fault tolerance**.



Fault Scenario Example

In this case, if the instance in **us-west-1a** fails (shown with a red ✖), the workload can automatically failover to other instances (such as in us-west-1b or us-west-1c).

This illustrates the **resilience of distributed training** across multiple Availability Zones.



Deployment Strategy	Description	Pros	Cons
Single-AZ Deployment	Run all training/inference in one Availability Zone (e.g., us-west-1a).	Lower cost, simpler setup.	Single point of failure — downtime if AZ fails.
Multi-AZ Deployment	Distribute instances across multiple AZs in a Region (e.g., us-west-1a + us-west-1b).	High availability, fault tolerance.	Higher cost, slightly more complex.
Distributed Training	Training job split across multiple instances in multiple AZs.	Faster training, resilient to failures.	Cost increases with instance count.