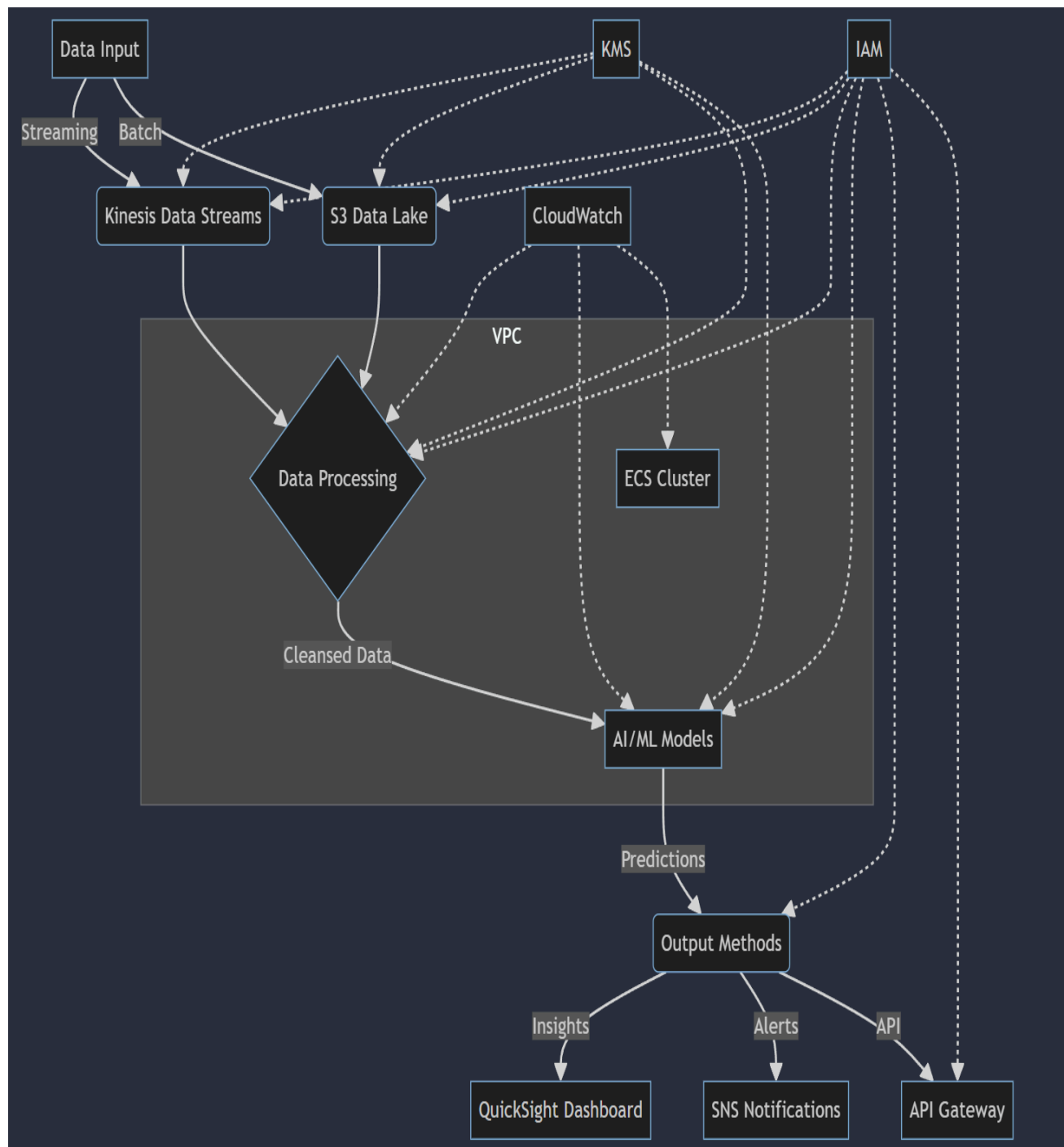


AI Solution for User Feedback Analysis

1. Architecture Diagram



2. Technology Stack and Justification

Cloud Platform: AWS

Justification: Comprehensive set of services, excellent scalability, and robust security features.

Key Components:

a) Data Input:

- Amazon Kinesis: For real-time data streaming
- Amazon S3: For batch data storage Justification: Scalable, managed services for handling large volumes of data

b) Data Processing:

- AWS Glue: For ETL jobs
- Amazon EMR: For large-scale data processing Justification: Serverless options for data transformation, scalable for big data processing

c) AI/ML Models:

- Amazon SageMaker: For model training and deployment
- Amazon Comprehend: For NLP and sentiment analysis Justification: Managed ML services, reducing operational overhead

d) Compute:

- Amazon ECS: For container orchestration
- AWS Lambda: For serverless computing Justification: Flexible compute options for different workload types

e) Output Methods:

- Amazon QuickSight: For automated reports and dashboards
- Amazon SNS: For real-time alerts
- Amazon API Gateway: For RESTful API Justification: Integrated services for various output needs

f) Additional Services:

- Amazon RDS or DynamoDB: For structured data storage
- AWS CloudWatch: For monitoring and logging
- AWS IAM: For access management
- AWS KMS: For encryption Justification: Comprehensive set of tools for data management, monitoring, and security

Programming Language: Python

Justification: Widely used in data science and ML, excellent libraries for NLP and web development, Docker-friendly

Machine Learning Libraries:

- scikit-learn: For basic ML models and preprocessing
 - NLTK: For NLP tasks and text processing
 - Hugging Face Transformers: For pre-trained NLP models
- Justification: Powerful, widely-used libraries that offer a good balance of functionality and ease of use

Web Framework: Flask

Justification: Lightweight, easy to set up, perfect for creating RESTful APIs quickly

Data Handling: Pandas

Justification: Excellent for data manipulation and analysis in Python

Containerization: Docker

Justification: Ensures consistency across environments and simplifies deployment

Version Control: Git

Justification: Industry-standard for version control, essential for code management

3. Scalability Solutions

- a) Modular Design: Separate components allow for independent scaling
- b) Stateless Application: API designed to be stateless, enabling horizontal scaling
- c) Serverless and Managed Services: Utilize AWS Lambda and managed services for automatic scaling
- d) Auto-scaling: Use ECS with auto-scaling groups for container-based workloads
- e) Distributed Processing: Use EMR for large-scale data processing tasks
- f) Caching: Implement Amazon ElastiCache for frequently accessed data
- g) Database Scalability: Use Amazon RDS with read replicas or Amazon DynamoDB for scalable database solutions

4. Security Considerations

a) Network Security:

- Use Amazon VPC for network isolation
- Implement security groups and network ACLs

b) Data Security:

- Encrypt data at rest using AWS KMS
- Use SSL/TLS for data in transit

c) Access Control:

- Implement least privilege access using IAM roles
- Use multi-factor authentication for all user accounts

d) API Security:

- Implement token-based authentication for API access
- Use API Gateway for rate limiting and throttling

e) Monitoring and Auditing:

- Use CloudTrail for auditing AWS API calls
- Implement CloudWatch for real-time monitoring and alerting

f) Compliance:

- Ensure compliance with relevant standards (e.g., GDPR, HIPAA) based on data types

g) Application Security:

- Implement input validation on all API endpoints
- Regularly update all libraries and dependencies
- Conduct regular security audits and penetration testing

5. Cost Estimation

Rough estimation for processing 100TB monthly:

- Data Storage (S3, RDS): ~\$2,500/month
- Data Processing (Glue, EMR): ~\$6,000-\$12,000/month
- ML Model Training and Inference (SageMaker): ~\$3,500-\$6,000/month
- Compute (ECS, Lambda): ~\$2,000-\$4,000/month
- Other Services (Kinesis, API Gateway, etc.): ~\$1,000-\$2,000/month

Total Estimated Cost: \$15,000 - \$26,500 per month

Note: This is a rough estimate and actual costs may vary based on specific usage patterns, reserved instances, and other optimizations.

6. Handling New Sources and Analysis Types

a) Data Input Flexibility:

- Use schema-less data lake (S3) for raw data storage
- Implement flexible ETL processes in Glue

b) Modular ML Pipeline:

- Design modular SageMaker pipelines for easy addition of new models
- Use feature stores for reusable feature engineering

c) API Extensibility:

- Design API with versioning to allow for new endpoints without breaking existing ones

d) Configurable Dashboards:

- Use QuickSight's dynamic dashboarding capabilities for adaptable visualizations

By implementing these strategies, the system can easily accommodate new data sources and analysis types with minimal changes to the core architecture.