Scalability Document

Introduction

This document describes how the proposed patient mobility forecasting solution can be scaled to support 100,000 patients in a production environment.

1. Big Data Processing Strategy

When the number of patients increases, the volume of step-count and clinical data grows significantly. To handle this, a distributed data processing approach is required. Key tools include Apache Spark (PySpark) for large-scale data processing, Amazon S3 as a data lake for storing raw and processed data, and AWS Glue or AWS EMR for running scalable ETL jobs. Athena and Lake Formation can be used for querying and managing access to large datasets stored in S3.

2. Modeling Strategy

A global model approach is preferred over training one model per patient. This allows the system to learn shared patterns across patients while remaining scalable. Patients can optionally be clustered by disease type to improve performance. Variability in activity levels is handled using normalization, patient-specific offsets, and global feature scaling.

3. Pipeline Architecture

The scalable pipeline consists of data ingestion into S3, feature engineering using Spark jobs, distributed model training, model deployment through batch or API-based services, and large-scale prediction serving. This architecture supports millions of daily predictions efficiently and reliably.

Conclusion

By leveraging cloud-native and distributed technologies, the proposed solution can scale effectively to support 100,000 patients while maintaining performance and interpretability.