TECHNOLOGY STACK

Technical Architecture: AI System That Verifies User Identities Based On Their Online Behavior Patterns
Introduction

This document describes the technical architecture for an AI system that verifies user identities based on their online behavior patterns. The system is designed to add an extra layer of security to applications by making it more difficult for attackers to gain unauthorized access.

System Overview

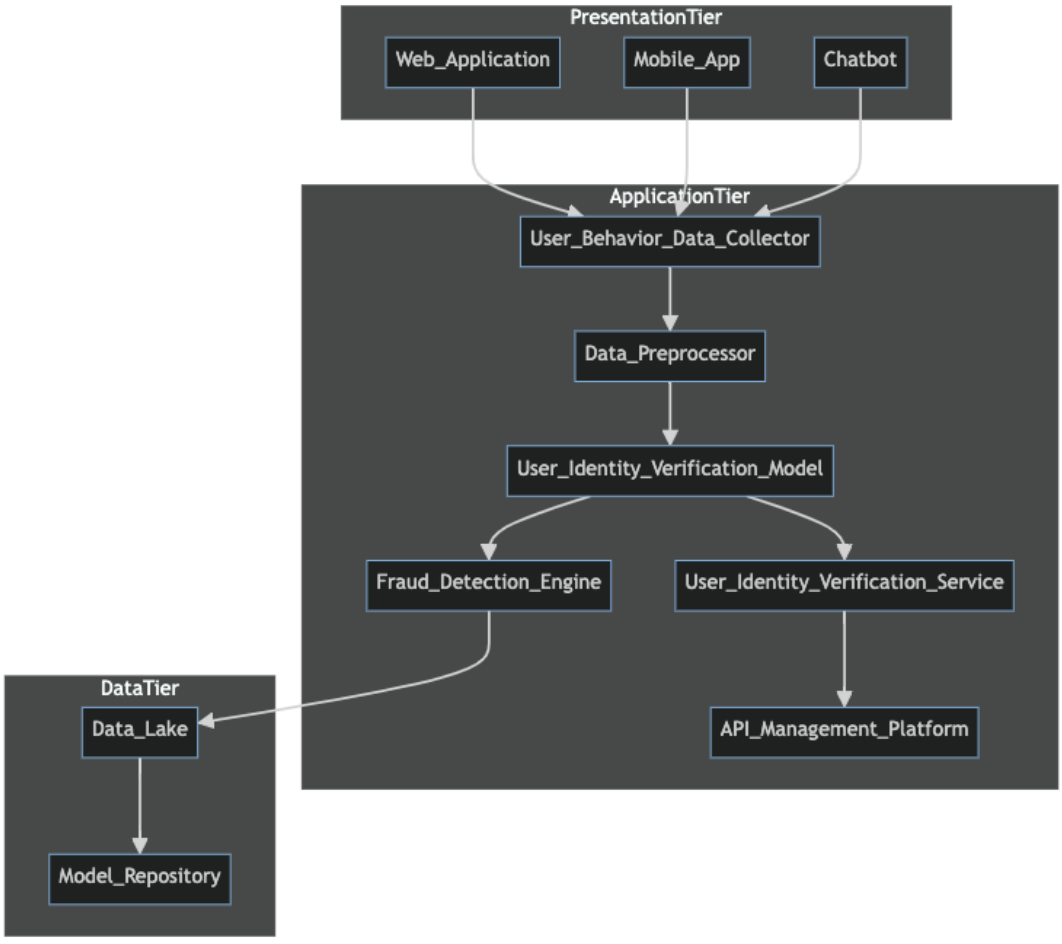
The system consists of the following components:

- User Behavior Data Collector: Collects user behavior data from various sources, such as website visits, app usage, and social media activity.
- User Behavior Data Preprocessor: Preprocesses the user behavior data to clean and prepare it for machine learning.
- User Identity Verification Model: Machine learning model that verifies user identities based on their online behavior patterns.
- User Identity Verification Service: Service that exposes the user identity verification model to other applications.
-

Fraud Detection Engine: Machine learning model that detects fraudulent activity.

Technical Architecture Diagram

The following diagram shows the technical architecture of the system:



Technical Architecture Diagram

Components and Technologies

The following components and technologies are used in the technical architecture diagram:

- User Behavior Data Collector: This component collects user behavior data from various sources, such as website visits, app usage, and social media activity. The data can be collected using a variety of methods, such as APIs, web scraping, and cookies.
- Data Preprocessor: This component preprocesses the user behavior data to clean and prepare it for machine learning. This may involve tasks such as removing outliers, normalizing the data, and converting it to a format that is compatible with the machine learning model.
- User Identity Verification Model: This machine learning model is trained on a dataset of user behavior data that is labeled with the user's identity. The model learns to identify patterns in the data that are unique to each user.
- User Identity Verification Service: This service exposes the user identity verification model to other applications. Applications can use the service to verify the identity of users before granting them access to resources.
- Fraud Detection Engine: This machine learning model is trained on a dataset of user behavior data that is labeled with whether or not the user engaged in fraudulent activity. The model learns to identify patterns in the data that are indicative of fraud.
- Data Lake: The data lake stores all of the user behavior data, including the preprocessed data and the output of the user identity verification model. The data lake can be used to train new machine learning models and to analyze user behavior data over time.
- Model Repository: The model repository stores the user identity verification model and the fraud detection model. The model repository makes it easy to deploy and manage the models.
- API Management Platform: The API management platform exposes the user identity verification service to other applications. The API management platform also provides features such as authentication, authorization, and rate limiting.

Architecture Overview

The technical architecture diagram shows a three-tier architecture:

- Presentation tier: The presentation tier is responsible for interacting with the user. It may consist of a web application, a mobile app, or a chatbot.
- Application tier: The application tier contains the business logic of the system. It includes the user behavior data collector, the data preprocessor, the user identity verification model, the user identity verification service, and the fraud detection engine.
- Data tier: The data tier stores the user behavior data and the machine learning models. It includes the data lake and the model repository.

Data Flow

The following is a high-level overview of the data flow in the system:

- The user behavior data collector collects user behavior data from various sources.
- The data preprocessor preprocesses the user behavior data.
- The user identity verification model is used to verify the identity of the user.

Table 1: Components and Technologies

S.No	Component	Description	Technology
1	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Js etc.
2	Application Logic-1	Logic for a process in the application	Java / Python
3	Application Logic-2	Logic for a process in the application	IBM Watson STT service
4	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
5	Database	Data Type, Configurations etc.	MySQL, NoSQL, etc.
6	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
7	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local Filesystem
8	External API-1	Purpose of External API used in the application	IBM Weather API, etc.
9	External API-2	Purpose of External API used in the application	Aadhar API, etc.
10	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model, etc.
11	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud	Local Server Configuration: Cloud Server Configuration : Local, Cloud Foundry, Kubernetes, etc.

Table 2: Application Characteristics

S.No	Characteristics	Description	Technology
1	Open-Source Frameworks	List the open-source frameworks used	Technology of Opensource framework
2	Security Implementations	List all the security / access controls implemented, use of firewalls etc. e.g. SHA-256, Encryptions, IAM Controls, OWASP etc.	SSL/TLS, HTTPS, OAuth 2.0, JWT, Firewalls, IAM Controls, OWASP
3	Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Micro-services architecture
4	Availability	Justify the availability of application (e.g. use of load balancers, distributed servers etc.)	Load balancers, distributed servers, multi-availability zone deployment
5	Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc.	Caching, CDN, load balancing

Security Considerations

The system is designed with security in mind. The user behavior data is encrypted at rest and in transit. The user identity verification model is trained on a secure dataset. The system is also designed to resist common attacks, such as SQL injection and cross-site scripting.

Scalability and Performance

The system is scalable to handle a large number of users. The data lake and machine learning platform can be scaled to handle the increasing volume of user behavior data. The user identity verification service is designed to process user requests quickly and efficiently.

Conclusion

This technical architecture diagram provides a reference for how to build an AI system that verifies user identities based on their online behavior patterns. The system is designed with security, scalability, and performance in mind.