Ad Click Prediction Using Cloud-Based Machine Learning

Industry: Media & Entertainment

Class: Cloud Computing (IMT 589) / Fawad Khan

Executive Summary

In the highly dynamic media and entertainment industry, digital advertisers face constant pressure to maximize engagement and return on investment (ROI). Our project proposes a real-time ad click prediction system powered by Azure-based cloud services. Leveraging scalable machine learning infrastructure, our system enables precise ad targeting, rapid experimentation, and actionable business insights. Our cloud strategy emphasizes a PaaS-first model, integrated analytics, and future extensibility. This document outlines the full architecture, strategy, and operational considerations for our solution.

1. Business Context & Cloud Strategy

1.1 Problem Statement and Industry Background (Kimmy)

Our team worked on building a real-time prediction system aimed at improving advertising performance in the media and entertainment industry. Since click-through rates (CTR) are so directly tied to revenue, we knew we had to design something responsive, scalable, and flexible enough to handle large volumes of user data.

1.2 Cloud Strategy Overview (Kimmy)

Right from the start, we chose to avoid using on-premise or hybrid models, mainly because they didn't offer the speed and elasticity we needed. So instead, we built our architecture entirely in the cloud.

1.3 Alignment with Company Goals (Kimmy)

But this isn't just about infrastructure or tools. It's about helping the company move toward broader goals like better customer experience, smarter ad targeting, and faster decision-making. With this system, we can tailor ad content more precisely to user behavior. And when people see content that feels relevant, they're more likely to click or engage. Live dashboards and built-in A/B testing features have also made it easier for teams to test ideas and adjust campaigns on the go. It's not always perfect, but being able to respond quickly has already made a noticeable difference. Over time, we think this approach will improve ROI and help the company stay competitive in a fast-changing market.

2. Financial, Operational, and Migration Considerations

2.1 Partnering with the CFO (Kimmy)

Because of the financial changes cloud adoption brings, we also had to loop in the CFO early. When you're shifting from upfront capital investments (CapEx) to usage-based operational costs (OpEx), it’s not just a technical move, it’s a budgeting shift too. The finance team helped us set realistic usage forecasts and cost expectations. Without their support, we might have underestimated expenses or run into approval issues later.

2.2 Cloud Migration Phases (Kimmy + Wynter)

That partnership also helped us start thinking about FinOps, getting finance and engineering to work together to monitor and manage cloud spending. While we are fully committed to Azure at this stage, we’ve also discussed the potential of adopting a multi-cloud approach in the future, especially as a way to reduce vendor lock-in and increase operational flexibility if our workloads grow or diversify further. Here is a timeline:

Short-Term (0-6 months)

-Monitoring & Alerting:

- Deploy Azure Monitor with custom metrics and Application Insights to detect anomalies.
- Set up Azure Alerts for critical system thresholds.

-Escalation Procedures:

- Implement incident workflows in Azure DevOps with integration to Microsoft Teams for real-time collaboration.

-Fallback Mechanisms:

- Use Azure Traffic Manager for failover routing.
- Implement Azure Function-based fallback logic for degraded model performance.

Medium-Term (6-18 months)

-Model Monitoring & Validation:

- Use Azure Machine Learning's built-in drift detection.
- Automate validation pipelines in Azure ML Pipelines.

-Staging & Testing:

- Build isolated test environments using Azure DevTest Labs.

- Simulate traffic loads with Azure Load Testing.

Long-Term (18+ months)

-Latency Reduction:

- Explore Azure Stack Edge for edge computing.

-Privacy-Preserving Learning:

- Integrate Azure Confidential Computing and investigate federated learning architectures.

-Reduce Vendor Lock-in:

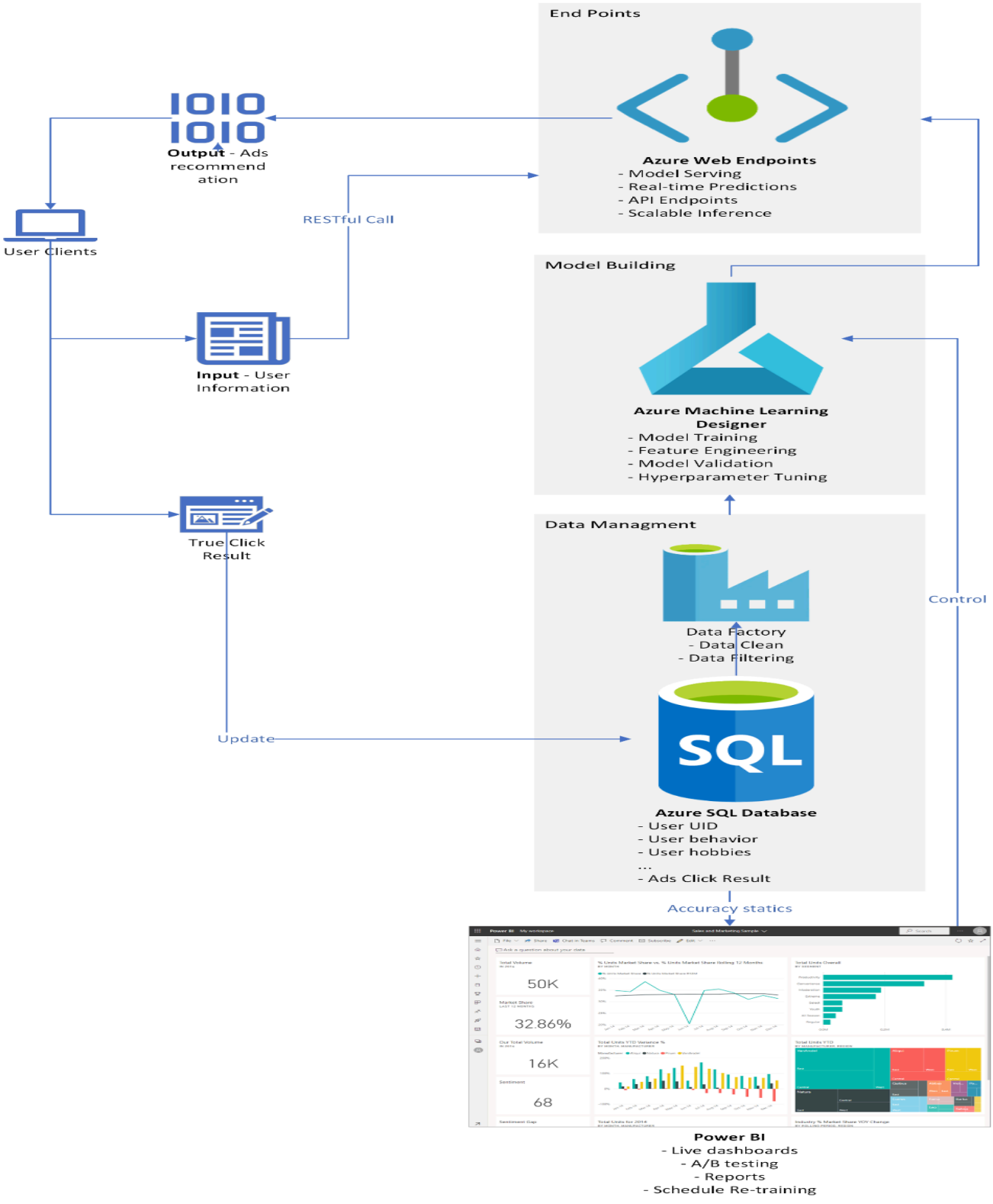
- Develop containerized services with Azure Kubernetes Service (AKS) for portability.
- Standardize APIs with Azure API Management to enable abstraction from backend services.

2.3 Organizational Impact (Kimmy)

Our IT department is starting to look different too. We're moving away from traditional siloed roles and forming smaller, more focused groups that work alongside people in marketing, finance, and data. These new teams are thinking more like product builders than system maintainers. We're not just introducing new tools, we're gradually creating a workflow where people can test, adjust, and learn in the process. Things aren't perfect yet, but this approach gives us the space to improve steadily and respond more effectively as needs evolve.

3. Cloud Architecture and Service Models

3.1 Proposed Architecture Diagram (Paige + Winter)



3.2 Cloud Model Selection (Winter)

Initial Cloud Strategy Suitability: Platform as a Service (PaaS)

-Rapid prototyping & development using Azure ML Designer

-Low setup overhead with managed services like SQL Database and Web Endpoints

-Ideal for small teams with limited DevOps resources

-Integrated tools like Data Factory and Power BI

-Low-code/no-code access for fast iteration

Long-Term Strategy: PaaS + SaaS

-Scalable, resilient, and secure with auto-scaling and RBAC

-Monitoring with Azure Monitor

-Cost optimization via tagging, spot pricing, and reserved instances

-SaaS layer via Power BI for accessible, collaborative insights

3.3 Cloud Service Models and Tools (Winter)

-PaaS: Primary Model

- Azure ML Designer
- Azure SQL Database
- Azure Web Endpoints
- Azure Data Factory

-SaaS: Supporting Model

- Power BI

-Why?

- Managed infrastructure

- Fast development with low-code tools
- Built-in high availability and integration
- Friendly UX for business stakeholders

3.4 Cloud Provider Justification (Paige)

We plan to adopt Microsoft Azure as our primary cloud provider. The reasons are:

-PaaS-first design: Azure services like ML Designer, SQL Database, Web Endpoints, and Data Factory offer a full pipeline for machine learning with minimal infrastructure maintenance.

-Strong integration with Microsoft ecosystem: Azure integrates seamlessly with tools like Power BI, which we use for client-facing reporting. This ensures data security, single sign-on, and easy connectivity.

-Enterprise compliance readiness: Azure meets regulatory standards like SOC 2, ISO 27001, and PCI DSS, and provides built-in tools for GDPR/CCPA enforcement.

-Robust monitoring and FinOps support: Azure Monitor, Application Insights, and cost management tools help track system health and optimize cloud spending.

-Azure gives us the right balance of scalability, compliance, and developer productivity needed for our ad intelligence platform.

4. Technical Execution & Risk Management

4.1 Experimentation in the Cloud (Paige)

Experimentation in the cloud is essential to our strategy. It enables rapid iteration on models, testing of data pipelines, and experimentation with minimal setup overhead. Azure's platform services like ML Designer and Data Factory, allow for fast, scalable, and low-code experimentation without requiring dedicated infrastructure teams.

The cloud also offers safer experimentation through features like version control, rollback, and sandboxed environments. For ad-click prediction systems, where A/B testing and continuous refinement are critical, cloud-based experimentation dramatically shortens development cycles and increases agility.

4.2 Business-Critical Workloads on Cloud (Paige)

-Real-time CTR prediction across multiple channels

-Feature engineering pipelines

-Model training and retraining for drift correction

-A/B testing orchestration and variant control

-Client-specific dashboards via model APIs + Power BI

4.3 Multi-cloud Considerations (Paige + Kimmy)

We do not plan to adopt a multi-cloud strategy at this stage. Our focus is on building a highly integrated pipeline within Azure to simplify operations, maintain compliance, and minimize architectural complexity. Multi-cloud introduces challenges in:

-Maintaining data consistency and compliance policies across platforms

-Duplicating monitoring and cost governance frameworks

-Managing team expertise and operational overhead

While we recognize that avoiding vendor lock-in has long-term value, the current benefits of a unified Azure environment outweigh the cost and complexity of supporting multiple cloud providers.

4.4 Limitations and Constraints (Winter + Paige)

-Data Integration Challenges

▪ Cross-channel identity resolution: Unifying user behavior across web, mobile, and email requires complex probabilistic matching

▪ Data quality inconsistencies: Different data sources (Google Analytics, Firebase, SendGrid, POS) have varying data formats and quality standards

▪ Real-time data synchronization: 24-hour latency in the Basic tier limits responsiveness to rapidly changing user behavior

▪ Feature engineering complexity: Computing meaningful features from high-dimensional, sparse data requires extensive preprocessing

-Model Performance Constraints

▪ Cold start problem: New users/products lack historical data for accurate predictions

▪ Concept drift: Ad preferences change seasonally, requiring frequent model retraining

▪ Class imbalance: Low click-through rates (typically 0.5-2%) create skewed training datasets

▪ Feature lag: Real-time features (last 5 minutes) may not capture longer-term behavioral patterns

-Azure-Specific Dependencies

▪ Vendor lock-in: Deep integration with Azure services limits multi-cloud flexibility

▪ Regional availability: Some Azure ML features not available in all regions

▪ Service limits: API Management has hard limits on concurrent connections

▪ Cost unpredictability: Auto-scaling can lead to unexpected charges during traffic surges

-Privacy Regulation Challenges

▪ GDPR compliance: Real-time data processing complicates right-to-deletion requests

▪ Consent management: Tracking consent across multiple channels and data sources

▪ Data minimization: Balancing prediction accuracy with privacy requirements

-Industry-Specific Limitations

▪ Cookie deprecation: Reduced web tracking capability affects feature quality

▪ iOS privacy updates: App Tracking Transparency limits mobile data collection

▪ Ad blocker prevalence: Reduced visibility into user behavior

▪ Platform policy changes: Dependency on third-party data sources (Google, Meta) creates risk

-Resource Allocation Issues

- Technical expertise shortage: Limited availability of Azure ML specialists
- Development velocity: Complex architecture slows feature development
- Testing complexity: End-to-end testing requires expensive staging environments
- Maintenance overhead: Multiple services require ongoing operational attention

5. Compliance & Support Strategy

5.1 Compliance Requirements (Winter)

Key Regulations: GDPR, CCPA, COPPA, SOC 2, ISO 27001, PCI DSS

Azure Support:

-Data residency & Encryption

-Data filtering via Data Factory

-Audit trails and policy enforcement

Conclusion: Cloud migration enhances compliance rather than blocks it. The focus is on governance and proper controls using Azure's compliance tools.

5.2 Technical Support Reorganization (Winter)

-Team & Skills:

- Upskill + hire for Azure, ML Ops, endpoint support
- Create pods by function (data, ML, API)

Infrastructure

-Azure Monitor + automated alerting

-Logging and pipeline observability

SLA Management

-Different SLAs for real-time vs batch

Documentation

-Runbooks, API, and dashboard guides

-Escalation and incident workflows

6. Future Work and Extensions (Paige)

-Client-level Power BI dashboards

-Edge computing via Azure Stack Edge

-Reinforcement learning for smarter ad allocation

-Consent-aware pipelines for GDPR/CCPA

-FinOps automation (alerts, tags, optimization)

-Containerization via AKS for portable ML services