AI Solution Architecture

This section details the technical AI architecture for the Lead Qualification & Deal Intelligence Hub, including the predictive ML scoring engine, LLM-powered personalization layer, lightweight RAG for property context, and real-time inference pipeline.

Architecture Overview

The system implements a **Predictive ML + LLM Hybrid Architecture**. Traditional gradient boosting models handle lead scoring and deal prediction (where historical data patterns are strong), while an LLM generates personalized outreach content and synthesizes lead insights. A lightweight RAG layer provides property context for message personalization.

Core Components:

1. **Lead Scoring Engine:** Gradient boosting classifier predicting transaction likelihood
2. **Deal Prediction Model:** Time-series aware model forecasting conversion probability over 30/60/90 day windows
3. **LLM Personalization Layer:** Foundation model for generating contextual outreach messages
4. **Property Context RAG:** Retrieval over MLS listings for property-specific personalization
5. **Real-Time Inference Pipeline:** Event-driven scoring updates as lead behavior changes

Lead Scoring Model Architecture

Model Selection

Component	Choice	Rationale
Primary Model	XGBoost Classifier	Handles mixed feature types, missing data, and imbalanced classes; interpretable feature importance
Alternative	LightGBM	Faster training for larger datasets; leaf-wise growth handles high-cardinality categoricals
Output	Probability score (0-100) + category	Hot (80-100), Warm (50-79), Cold (20-49), Long-term Nurture (<20)

Feature Engineering

The scoring model ingests features across four categories:

- **Behavioral Signals:** MLS listing views (count, recency, frequency), property saves, search filter patterns, time-on-listing, return visit rate
- **Engagement Metrics:** Email open rate, SMS response rate, call answer rate, response latency, appointment show rate
- **Demographic Indicators:** Inferred price range, location preferences, property type interest, buyer vs. seller signals
- **Financial Readiness:** Pre-approval status (if shared), income range indicators, first-time buyer flags, down payment readiness signals

Training & Evaluation

- **Training Data:** Historical leads with known outcomes (closed, lost, still nurturing); minimum 10,000 labeled examples for initial model
- **Target Variable:** Binary classification (converted to transaction within 90 days vs. not)
- **Evaluation Metrics:** AUC-ROC > 0.75, Precision@k for top 20% of leads, calibration (predicted probability matches actual conversion rate)
- **Retraining Cadence:** Monthly automated retraining with performance drift detection; alert if AUC drops >5%

LLM-Powered Personalization

The LLM layer generates personalized outreach content based on lead behavior, property interests, and agent communication style.

Model Configuration

- **Model:** GPT-4o-mini or Claude 3 Haiku (cost-optimized for high-volume message generation)
- **Temperature:** 0.6 for email drafts (balanced creativity), 0.4 for SMS (more concise/predictable)
- **Context Window:** Inject lead profile summary, recent activity, property interests, and agent voice guidelines

Prompt Template: Personalized Follow-Up

```
[SYSTEM]
You are a friendly, professional real estate agent assistant. Write concise,
personalized messages that reference specific properties or behaviors. Never
be pushy. Match the communication style provided.
[CONTEXT]
Lead: {name} | Score: {score} | Recent Activity: {activity_summary}
Properties Viewed: {property_list} | Price Range: {price_range}
Agent Style: {style_guidelines}
[TASK]
Write a {message_type} follow-up referencing their interest in
{specific_property_or_area}. Keep under {word_limit} words.
```

Property Context RAG

A lightweight RAG layer retrieves relevant property details for message personalization and lead insight generation.

Component	Specification
Data Source	MLS listings via RESO Web API; includes property descriptions, features, photos metadata, days-on-market, price history
Embedding Model	text-embedding-3-small (cost-efficient for high listing volume); 1536 dimensions
Vector Database	Pinecone Serverless (scales with listing count) or Supabase pgvector (cost-effective for smaller teams)
Update Frequency	Nightly sync with MLS; real-time updates for status changes (active → pending → sold)
Retrieval Use Cases	(1) Inject property highlights into outreach messages, (2) Find similar listings to recommend, (3) Generate market comparison context

Real-Time Inference Pipeline

Lead scores update dynamically as new behavioral signals arrive, enabling timely agent notifications.

- **Event Triggers:** CRM webhook on new lead, email open/click events, MLS listing view (via pixel tracking), SMS reply, form submission
- **Processing:** Events flow through Kafka/SQS queue → feature computation service → model inference → score update → notification dispatch
- **Latency Target:** < 5 seconds from event to updated score; < 30 seconds to agent notification for Hot lead threshold crossing
- **Batch Fallback:** Daily batch re-scoring for all leads ensures no lead falls through cracks if events are missed

Infrastructure & Cost Optimization

Component	Specification
Deployment	AWS (Lambda for inference, ECS for batch) or GCP (Cloud Run); multi-tenant SaaS architecture
Model Serving	XGBoost model serialized via joblib; served via SageMaker Serverless or self-hosted FastAPI
LLM Cost Control	Use GPT-4o-mini/Haiku for bulk generation; cache common message templates; rate limiting per team
Monitoring	Model performance dashboard (AUC over time, feature drift); LLM usage tracking; conversion rate vs. score correlation

[End of AI Solution Architecture Section]