ML TRAINING ENVIRONMENT

Complete Implementation Guide

AIAlgoTradeHits.com Trading Intelligence Platform

Document Version	2.0
Generated	January 08, 2026
Project	aialgotradehits
Current Accuracy	67% (Saleem Model)
Target Accuracy	85%
Primary Model	XGBoost + Gemini 2.5 Pro Ensemble

Owners: Saleem Ahmad (ML Model Development) | Irfan Qazi (Platform Architecture)

TABLE OF CONTENTS

1. Executive Summary

2. Current ML Training Environments

2.1 Saleem's ML Training Lab (Local Streamlit)

2.2 GCP BigQuery ML Environment

2.3 Hybrid XGBoost + Gemini Model

3. Tools and Libraries Stack

4. Data Architecture

4.1 Data Sources

4.2 Feature Engineering (24+ Features)

4.3 Data Splitting Strategy

5. Model Architecture

5.1 XGBoost Primary Model

5.2 Gemini 2.5 Pro Integration

5.3 Ensemble Strategy

6. Fully Automated GCP Cloud Process

6.1 Cloud Storage Buckets

6.2 Vertex AI Pipeline

6.3 Automated Training Schedulers

7. Roadmap: 67% to 85% Accuracy

7.1 Phase 1: Data Quality

7.2 Phase 2: Feature Engineering

7.3 Phase 3: Model Optimization

7.4 Phase 4: Ensemble Refinement

8. Implementation Timeline

9. Appendix: Code References

1. EXECUTIVE SUMMARY

This document provides a comprehensive guide to the ML Training Environment for the AIAlgoTradeHits.com trading intelligence platform. It covers two primary training environments: **Saleem's Local ML Training Lab** (Streamlit-based) achieving 67% accuracy, and the **GCP Cloud-based environment** using BigQuery ML and Vertex AI.

The goal is to merge these environments into a **fully automated GCP Cloud process** that leverages Google's Gemini 2.5 Pro for enhanced accuracy, targeting **85% directional accuracy** for trading predictions.

KEY METRICS AT A GLANCE

Metric	Current	Target	Improvement
Overall Accuracy	67%	85%	+18%
UP Direction Accuracy	68.5%	87%	+18.5%
DOWN Direction Accuracy	65%	83%	+18%
High-Confidence Win Rate	70%	90%	+20%
Model Latency	500ms	<100ms	5x faster

2. CURRENT ML TRAINING ENVIRONMENTS

2.1 Saleem's ML Training Lab (Local Streamlit)

Saleem Ahmad has built a local ML Model Training Dashboard that provides an interactive environment for data analysis, feature selection, model training, and result visualization.

Component	Technology	Purpose
Package Manager	Homebrew	System package management
Python Version	pyenv + Python 3.12.8	Stable ML-compatible runtime
Virtual Environment	ml_env	Isolated dependencies
UI Framework	Streamlit	Interactive web dashboard
Data Processing	pandas, numpy	Data manipulation
ML Framework	XGBoost	Core gradient boosting model
Preprocessing	scikit-learn	Feature scaling, metrics
Visualization	matplotlib	Charts and plots
OpenMP Runtime	libomp	XGBoost parallelization (macOS)

Saleem's 16-Feature Model (68.5% UP Accuracy)

#	Feature	Weight	Category
1	pivot_low_flag	25%	KEY - Reversal Detection
2	pivot_high_flag	25%	KEY - Reversal Detection
3	rsi	10%	Momentum
4	rsi_slope	8%	Momentum Derivative
5	macd_cross	8%	Trend Signal
6	macd_histogram	6%	Momentum Strength
7	cci	5%	Cyclical Momentum
8	momentum	5%	Price Momentum
9	mfi	4%	Money Flow
10	awesome_osc	4%	Market Momentum
11	vwap_daily	-	Volume-Weighted Price
12	rsi_overbought	-	RSI Signal
13	rsi_oversold	-	RSI Signal
14	rsi_zscore	-	RSI Normalized
15	macd	-	MACD Line
16	macd_signal	-	MACD Signal Line

XGBoost Parameters (Saleem's Validated Settings)

Parameter	Value	Rationale
max_depth	8	Deep enough for complex patterns
learning_rate	0.3	Fast convergence
n_estimators	100	Balanced training time
objective	binary:logistic	Binary classification
eval_metric	logloss	Probability calibration
random_state	42	Reproducibility

2.2 GCP BigQuery ML Environment

The cloud-based ML environment uses Google BigQuery ML for scalable model training directly on stored data, eliminating data movement overhead and enabling SQL-based ML workflows.

Component	Configuration	Purpose
Project	aialgotradehits	GCP Project ID
Dataset	crypto_trading_data	Primary data storage
ML Dataset	ml_models	Model artifacts storage
Region	us-central1	Compute location
Storage	BigQuery	Petabyte-scale data warehouse

BigQuery ML Models Deployed

Model Name	Type	Accuracy	Features
xgboost_daily_direction	BOOSTED_TREE_CLASSIFIER	52.8%	10 core indicators
xgboost_v2_improved	BOOSTED_TREE_CLASSIFIER	58-63%	24 + interactions
xgboost_v2_significant_moves	BOOSTED_TREE_CLASSIFIER	60-65%	>1% moves only
xgboost_hourly_direction	BOOSTED_TREE_CLASSIFIER	55-58%	12 hourly indicators
xgboost_5min_direction	BOOSTED_TREE_CLASSIFIER	52-55%	8 execution indicators

2.3 Hybrid XGBoost + Gemini Ensemble Model

The production system uses a hybrid ensemble combining XGBoost's quantitative analysis with Google Gemini's qualitative reasoning for enhanced prediction accuracy.

Component	Weight	Role
XGBoost Quantitative	60%	Technical indicator analysis, probability scores
Gemini 2.5-flash	40%	Sentiment analysis, market context, reasoning
Ensemble Output	100%	Weighted direction + confidence level

Ensemble Decision Logic

```
ensemble_score = (xgb_weight * xgb_score * xgb_confidence) + (gemini_weight * gemini_score * gemini_confidence) if ensemble_score > 0.2: direction = 'UP' elif ensemble_score < -0.2: direction = 'DOWN' else: direction = 'NEUTRAL'
```

3. TOOLS AND LIBRARIES STACK

Core ML Frameworks

Library	Version	Use Case
XGBoost	1.x+	Primary gradient boosting classifier
scikit-learn	1.x+	Preprocessing, metrics, cross-validation
TensorFlow/Keras	2.x+	LSTM models (optional)
pandas	2.0+	Data manipulation and analysis
numpy	1.24+	Numerical computations
scipy	1.x+	Statistical tests, KL divergence
matplotlib	3.x+	Visualization and plots
seaborn	0.12+	Statistical data visualization
joblib	-	Model serialization

Google Cloud Platform Services

Service	Purpose	Cost Tier
BigQuery	Data warehouse, ML training	$5-10/month storage
BigQuery ML	SQL-based model training	Pay per query
Vertex AI	Managed ML, Gemini API	$0.0005/1K chars
Cloud Functions Gen2	Serverless compute	$15-20/month
Cloud Run	Container deployment	$5-10/month
Cloud Scheduler	Automated job triggers	$0.30/month
Cloud Storage	Model artifacts, exports	<$1/month
Secret Manager	API key storage	<$1/month

AI/LLM Integration

Component	Model	Integration
Primary LLM	Gemini 2.5 Pro	Qualitative analysis, Text-to-SQL
Fallback LLM	Gemini 1.5 Pro	Backup for API limits
ADK Framework	Google ADK 0.3.0+	Agent Development Kit
Text-to-SQL	NL2SQL Engine	Natural language queries
MCP Toolbox	v0.3.0	BigQuery tool integration

4. DATA ARCHITECTURE

4.1 Data Sources

Source	Data Type	Refresh Rate	Cost
TwelveData	OHLCV + Indicators	Hourly/Daily	$229/month
Kraken	Buy/Sell Volume	Real-time	Free API
FRED	Economic Indicators	Daily	Free
Finnhub	News, Sentiment	Real-time	Free tier
CoinMarketCap	Crypto Metadata	Daily	Basic tier

4.2 Feature Engineering (24+ Features)

Daily Features (24 Indicators)

Category	Features	Count
Momentum	RSI_14, MACD, MACD_Histogram, ROC, Stoch_K, Stoch_D	6
Trend	SMA_20/50/200, EMA_12/20/26/50/200, Ichimoku_Tenkan/Kijun	10
Volatility	ATR_14, BB_Upper, BB_Middle, BB_Lower	4
Strength	ADX, Plus_DI, Minus_DI	3
Flow	MFI, CMF	2

Derived Features (Feature Interactions)

Feature	Formula	Purpose
RSI_Volume_Interaction	RSI_14 * Volume_Ratio	Momentum confirmed by volume
MACD_ATR_Interaction	MACD_Histogram * (ATR/Close * 100)	Momentum vs volatility
ADX_Trend_Interaction	ADX * Trend_Direction	Trend strength + direction
RSI_ADX_Interaction	RSI_14 * (ADX / 100)	Momentum in trending market
Stoch_Volume_Interaction	Stoch_K * Volume_Ratio	Reversal + volume confirmation
Momentum_5d/10d/20d	(Close - Close_t-n) / Close_t-n * 100	Multi-period momentum

4.3 Data Splitting Strategy (CRITICAL)

IMPORTANT: The data splitting strategy is crucial for accurate model evaluation. We use a time-based split to prevent look-ahead bias and ensure the model is tested on truly unseen future data.

Dataset	Date Range	Purpose	Size Estimate
TRAINING	Beginning of data to Dec 31, 2022	Model learning	~70% of data
TESTING	Jan 1, 2023 to Dec 31, 2023	Hyperparameter tuning	~15% of data
VALIDATION	Jan 1, 2024 to Present (2025+)	Final evaluation	~15% of data

Time-Series Cross-Validation (Training Phase)

During training, we use TimeSeriesSplit with 5 folds to ensure proper temporal ordering. This prevents data leakage and provides robust accuracy estimates.

Parameter	Value	Rationale
n_splits	5	Balance between bias and variance
Shuffling	DISABLED	Preserve temporal order
Gap	1 day	Prevent same-day leakage
Expanding Window	Yes	Train on growing history

5. MODEL ARCHITECTURE

5.1 XGBoost Primary Model

XGBoost serves as the primary quantitative model, providing fast and accurate predictions based on technical indicators. The model outputs probability scores for UP/DOWN directions.

Layer	Configuration	Purpose
Input	24-40 features	Technical indicators + derived
Preprocessing	RobustScaler	Handle outliers, normalize
Estimators	100-200 trees	Ensemble complexity
Max Depth	6-8 levels	Pattern complexity
Learning Rate	0.05-0.1	Convergence speed
Subsample	0.8	Regularization
ColSample	0.8	Feature randomization
Output	Binary probability	UP/DOWN with confidence

5.2 Gemini 2.5 Pro Integration

Gemini 2.5 Pro provides qualitative analysis by interpreting market context, sentiment, and complex patterns that pure quantitative models might miss.

Parameter	Value	Purpose
Model	gemini-2.5-pro	Latest reasoning model
Temperature	0.1	Consistent predictions
Max Tokens	8192	Detailed analysis
Response Format	JSON	Structured output
Fallback	gemini-1.5-pro	API rate limit backup

Gemini Prompt Structure

```
Input: {symbol, price, RSI, MACD, ADX, trend_regime, buy_pressure, sell_pressure, crosses} Output JSON: { "direction": "UP" | "DOWN" | "NEUTRAL", "confidence": "HIGH" | "MEDIUM" | "LOW", "reasoning": "Brief explanation of analysis", "risk_level": "HIGH" | "MEDIUM" | "LOW", "key_factors": ["factor1", "factor2", "factor3"] }
```

5.3 Ensemble Strategy

The ensemble combines XGBoost's technical analysis (60% weight) with Gemini's qualitative insights (40% weight) to produce more robust trading signals.

Step	Component	Output
1	XGBoost Prediction	UP/DOWN probability (0.0 - 1.0)
2	Gemini Analysis	Direction + Confidence + Reasoning
3	Weight Application	XGB: 60%, Gemini: 40%
4	Score Calculation	ensemble_score = weighted sum
5	Direction Decision	UP if score > 0.2, DOWN if < -0.2
6	Confidence Level	HIGH/MEDIUM/LOW based on total confidence

6. FULLY AUTOMATED GCP CLOUD PROCESS

This section outlines the complete automation of ML training in GCP, eliminating manual intervention and enabling continuous model improvement.

6.1 Cloud Storage Buckets

Bucket	Purpose	Retention
gs://aialgotradehits-ml-models/	Trained model artifacts	90 days
gs://aialgotradehits-training-data/	Exported training datasets	30 days
gs://aialgotradehits-predictions/	Daily predictions archive	365 days
gs://aialgotradehits-metrics/	Performance metrics logs	180 days
gs://aialgotradehits-function-source/	Cloud Function deployments	30 days

6.2 Vertex AI Pipeline

Step	Component	Trigger	Duration
1	Data Extraction	Daily 1:00 AM ET	5 min
2	Feature Engineering	After extraction	10 min
3	Model Training (XGBoost)	Weekly Sunday	30 min
4	Model Evaluation	After training	5 min
5	Model Deployment	If accuracy improved	2 min
6	Prediction Generation	Daily 4:30 AM ET	10 min
7	Performance Monitoring	Continuous	Ongoing

6.3 Automated Training Schedulers

Scheduler	Schedule	Function	Purpose
bulletproof-hourly-all	0 * * * *	bulletproof-fetcher	Hourly data collection
bulletproof-daily-all	0 1 * * *	bulletproof-fetcher	Daily data with indicators
ml-daily-predictions	30 4 * * *	ml-analysis	Generate daily predictions
ml-weekly-retrain	0 2 * * 0	ml-training	Weekly model retraining
drift-detector	0 */6 * * *	drift-detection	Data/concept drift check
gap-detector-hourly	30 * * * *	gap-detector	Data quality validation

7. ROADMAP: 67% TO 85% ACCURACY

This section outlines the strategic phases to improve model accuracy from the current 67% (Saleem's model) to the target 85% through systematic enhancements.

7.1 Phase 1: Data Quality Enhancement

Action	Expected Improvement	Priority
Deduplicate data (one row per date)	+2-3%	HIGH - CRITICAL
Handle missing values with forward fill	+1-2%	HIGH
Remove outliers (>4 std)	+1%	MEDIUM
Add more historical data (2015-2025)	+2%	HIGH
Cross-validate with multiple symbols	+1%	MEDIUM

Phase 1 Target: 67% → 73%

7.2 Phase 2: Advanced Feature Engineering

Action	Expected Improvement	Priority
Add feature interactions (RSI*Volume)	+2%	HIGH
Add lagged features (t-1, t-5, t-10)	+2%	HIGH
Multi-timeframe features (daily + hourly)	+3%	HIGH
Pivot high/low detection (Saleem feature)	+2%	HIGH - KEY
Add VWAP, Volume Profile	+1%	MEDIUM
Fibonacci retracement levels	+1%	MEDIUM
Candlestick pattern detection	+1%	LOW

Phase 2 Target: 73% → 80%

7.3 Phase 3: Model Optimization

Action	Expected Improvement	Priority
Hyperparameter tuning (Optuna/Grid)	+2%	HIGH
Class balancing (SMOTE, weighting)	+1%	MEDIUM
Focus on significant moves (>1%)	+1%	MEDIUM
Ensemble with Random Forest	+1%	MEDIUM
Early stopping optimization	+0.5%	LOW

Phase 3 Target: 80% → 83%

7.4 Phase 4: Gemini 2.5 Pro Ensemble Refinement

Action	Expected Improvement	Priority
Fine-tune Gemini prompts for market context	+1%	HIGH
Add sentiment analysis from news	+0.5%	MEDIUM
Optimize ensemble weights dynamically	+0.5%	HIGH
Add market regime detection	+0.5%	MEDIUM
Implement confidence-based filtering	+0.5%	HIGH

Phase 4 Target: 83% → 85%+

7.5 SALEEM'S ENVIRONMENT vs PROPOSED 90%+ ACCURACY SYSTEM

This section provides a detailed side-by-side comparison highlighting the key differences between Saleem's current local ML Training Lab and the proposed fully automated GCP-based environment designed to achieve 90%+ accuracy.

INFRASTRUCTURE COMPARISON

Aspect	SALEEM'S ENVIRONMENT (67%)	PROPOSED ENVIRONMENT (90%+)
Platform	Local macOS machine	Google Cloud Platform (fully managed)
Compute	Single machine (limited CPU)	Auto-scaling Cloud Functions + Vertex AI
Storage	Local files/CSV	BigQuery (petabyte-scale) + Cloud Storage
UI	Streamlit dashboard	React Web App + API + Vertex AI Agent
Scheduling	Manual runs	Cloud Scheduler (automated 24/7)
Monitoring	Manual inspection	Automated drift detection + alerts
Scalability	Single user	Multi-user, enterprise-ready
Cost	$0 (local compute)	~$300/month (fully managed)

MODEL ARCHITECTURE COMPARISON

Aspect	SALEEM'S MODEL (67%)	PROPOSED MODEL (90%+)
Primary Model	XGBoost only	XGBoost + Gemini 2.5 Pro Ensemble
Features	16 features	40+ features (24 base + interactions + lags)
Feature Types	Technical indicators only	Technical + Sentiment + Multi-timeframe
Key Features	Pivot High/Low (manual)	Pivot Detection + AI Pattern Recognition
Ensemble	None	XGBoost (60%) + Gemini (40%) + Optional RF/LSTM
AI Integration	None	Gemini 2.5 Pro for qualitative analysis
Confidence Levels	Basic	HIGH/MEDIUM/LOW with probability calibration
Explainability	Feature importance only	SHAP values + Gemini reasoning

DATA PIPELINE COMPARISON

Aspect	SALEEM'S APPROACH (67%)	PROPOSED APPROACH (90%+)
Data Source	CSV upload (manual)	TwelveData + Kraken + FRED APIs (automated)
Data Freshness	Depends on manual upload	Real-time (hourly/daily automated)
Historical Depth	Limited by local storage	10+ years (2015-2025) in BigQuery
Data Quality	Manual inspection	Automated deduplication + gap detection

Feature Calc	Manual in dashboard	Pre-calculated in BigQuery (SQL-based)
Multi-Timeframe	Single timeframe	4 timeframes (Daily/Hourly/5min/1min)
Symbols	One at a time	200+ stocks, 50+ crypto, 40+ ETFs (batch)
Train/Test Split	User-defined in UI	Standardized (Train: <2022, Test: 2023, Val: 2024+)

KEY INNOVATIONS TO ACHIEVE 90%+ ACCURACY

The following innovations differentiate the proposed system and enable the jump from 67% to 90%+ accuracy:

Innovation	Description	Accuracy Impact
1. Gemini 2.5 Pro Integration	AI-powered qualitative analysis of market context, sentiment, and complex patterns that pure technical anal	+8-10%
2. Multi-Timeframe Confluence	Combine signals from Daily (strategy), Hourly (timing), 5min (execution) for confirmation. Only trade when a	+5-7%
3. Advanced Feature Interactions	RSI*Volume, MACD*ATR, ADX*Trend - capture relationships between indicators that single features miss	+2-5%
4. Pivot Detection (Saleem Key Feature)	Automated pivot high/low detection using local extrema - proven 25% feature importance in Saleem model	+3-4%
5. Dynamic Confidence Filtering	Only act on HIGH confidence predictions (>70% probability). Low confidence = smaller position or no trad	+5-8%
6. Market Regime Detection	Identify STRONG_UPTREND, CONSOLIDATION, etc. and use regime-specific models for each condition	+3-5%
7. Automated Drift Detection	Detect when market behavior changes and trigger model retraining before accuracy degrades	+2-3%
8. Ensemble Voting	XGBoost + Random Forest + Gemini all vote. Only trade when majority agrees. Reduces false signals	+4-6%

ACCURACY PROGRESSION ROADMAP

Stage	Improvements Applied	Expected Accuracy
Baseline (Saleem)	16 features, XGBoost, local training	67%
+ Data Quality	Deduplication, gap filling, more history	72%
+ Feature Engineering	Interactions, lags, multi-timeframe	78%
+ Model Optimization	Hyperparameter tuning, class balancing	82%
+ Gemini Ensemble	AI qualitative analysis, sentiment	86%
+ Confidence Filtering	Only HIGH confidence trades	90%+
+ Regime-Specific Models	Different models per market condition	92-95%

PRESERVING SALEEM'S KEY INNOVATIONS

The proposed system will incorporate and enhance Saleem's proven innovations:

Saleem Innovation	Why It Works	How We Enhance It
Pivot High/Low Detection	25% feature importance - catches reversals	Add AI-powered pattern recognition
16 Core Features	Focused, avoid overfitting	Keep core + add proven interactions
RSI Slope	Momentum direction, not just level	Add multi-period slopes (1d, 5d, 10d)
MACD Cross Signal	Trend change detection	Combine with volume confirmation
Streamlit UI	Interactive analysis	Replicate in React + add AI chat
XGBoost Focus	Fast, interpretable, accurate	Keep as primary + add ensemble

8. IMPLEMENTATION TIMELINE

Phase	Focus Area	Key Deliverables
Phase 1	Data Quality	Deduplicated tables, gap filling, historical backfill
Phase 2	Feature Engineering	Feature interactions, multi-timeframe, pivots
Phase 3	Model Optimization	Tuned hyperparameters, class balancing
Phase 4	Ensemble Refinement	Gemini integration, confidence filtering
Maintenance	Ongoing	Weekly retraining, drift monitoring

Success Criteria: The model is considered production-ready when it achieves 85%+ directional accuracy on the 2024-2025 validation set with consistent performance across multiple symbols and market conditions.

9. APPENDIX: CODE REFERENCES

File	Purpose	Location
hybrid_xgboost_gemini_model.py	Ensemble model training	C:/1AITrading/Trading/
xgboost_model_reports_v3.py	Individual symbol analysis	C:/1AITrading/Trading/
improve_ml_model_accuracy.py	BigQuery ML improvements	C:/1AITrading/Trading/
model_performance_monitor.py	Drift detection	C:/1AITrading/Trading/
create_4timeframe_ml_system.py	Multi-timeframe setup	C:/1AITrading/Trading/
vertex_ai_agent_deployment.py	Vertex AI deployment	C:/1AITrading/Trading/
ML_Training_Quick_Start.ipynb	Interactive notebook	C:/1AITrading/Trading/
README_ML_TRAINING.md	Training documentation	C:/1AITrading/Trading/

BigQuery Tables

Table	Dataset	Description
stocks_daily_clean	crypto_trading_data	Daily stock data with indicators
crypto_daily_clean	crypto_trading_data	Daily crypto data
ml_daily_stocks_24	ml_models	ML feature table (24 indicators)
xgboost_daily_direction	ml_models	Trained XGBoost model
model_predictions_log	crypto_trading_data	Prediction history
model_performance_metrics	crypto_trading_data	Performance tracking

---

Document Generated: January 08, 2026 at 07:51 PM

AIAlgoTradeHits.com - Trading Intelligence Platform

For questions, contact: irfan.qazi@aialgotradehits.com | saleem.ahmad@aialgotradehits.com