IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510159

Trading Copilot for Cryptocurrency Markets: A Review of Analytics, Prediction, Manipulation Detection, Automated Execution and Conversational Agents, with a Gap Analysis of Existing Systems

Sanskruti Patil[1], Mokshada Sheth[2], Shraddha Kokare[3], Naveli Pangal[4], Kopal Gangrade[5]

Department of Computer Engineering, Pune Institute of Computer Technology, Pune, India[1-5]

Abstract: Retail cryptocurrency traders operate in a market that is open around the clock, split across many venues, heavy-tailed and exposed to manipulation, yet the tools available to them are divided among charting and alert services, automated bot platforms and research prototypes that rarely work together. This paper reviews the literature and the commercial and open-source landscape relevant to a unified Trading Copilot for crypto markets with the following capabilities: trade analysis, news-based volatility analysis, continuous buy/sell prediction, strategy selection, accumulation–manipulation–distribution (AMD) phase detection, price-distribution and breakdown analysis, cross-exchange price monitoring, automated order placement, notifications and alerts, and a conversational agent. We synthesise the evidence for each capability, compare representative systems, and derive ten gaps. Our main findings are that (i) bot platforms emphasise execution but, in the sources we reviewed, document little analytical or explanatory capability; (ii) LLM trading agents are reported mainly in simulation and without live execution; (iii) manipulation detectors are not coupled to trading decisions; and (iv) conversational LLM agents are reported to retrieve data well but to predict poorly, so they should explain computed outputs rather than forecast. We then describe how a Trading Copilot can close these gaps and propose an evaluation protocol.

Keywords: cryptocurrency, trading copilot, large language models, volatility, market manipulation, Wyckoff method, trading bots, alerts, conversational agent.

I. INTRODUCTION

Cryptocurrency markets differ from traditional markets in ways that matter for tool design. They trade continuously, are spread across dozens of centralised and decentralised venues, show heavy-tailed returns and clustered volatility, and have repeatedly been shown to host price manipulation [1], [2], [3], [4]. A retail trader who wants to act on these markets typically combines a charting service, a news feed, one or more social channels, an exchange interface and, increasingly, an automated bot platform. Each tool solves one part of the problem; the synthesis is left to the trader.

Recent work on large language model (LLM) agents suggests that a conversational layer on top of quantitative tools could reduce this burden [5], [6]. Several agents have been proposed for crypto and multi-asset trading [7], [8], [9], [10], but they are typically evaluated in simulation, and they do not connect to execution or alerts. Meanwhile, bot platforms provide execution but document little in the way of analysis or explanation [11], [12].

Our earlier review focused on a multi-market platform centred on Indian equity derivatives. The project scope has since narrowed to a Trading Copilot for the crypto market. This paper therefore re-reviews the literature for that scope. Its contributions are:

- a structured review of ten capabilities a crypto Trading Copilot must provide (Section IV), together with additional trader needs we recommend;
- a comparison of representative bot platforms, open-source frameworks, LLM-agent research systems and manipulation detectors (Section V);
- ten explicit gaps and the system additions that address them (Sections VI and VII); and
- an evaluation protocol for the resulting system (Section VIII).

A. *Scope and Interpretation of Requirements*

The capabilities requested for the project are listed in Table I. Where a requirement admits more than one reading, we state our interpretation. “News-based volatility analysis” is read as forecasting and explaining volatility using news and event data. “Price exchange” is read as monitoring and comparing prices of the same asset across exchanges. “Breakdown” is read in two ways: the breakdown of a price move into contributing factors, and the breakdown of a support level or volume-at-price zone. These readings should be confirmed by the project team.

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510159

TABLE I CAPABILITIES AND REVIEW SECTIONS

Capability	Section	Primary evidence
Trade analysis	IV-A	[13], [14], [15]
News-based volatility	IV-B	[16], [17], [18], [19]
Continuous buy/sell prediction	IV-C	[7], [20], [6]
Strategies	IV-D	[11], [12]
Accumulation–Manipulation–Distribution	IV-E	[21], [22], [23]
Price distribution and breakdown	IV-F	[1], [2]
Price across exchanges	IV-G	[24]
Trading bot and order placement	IV-H	[11], [12]
Notifications and alerts	IV-I	[12], [22]
Agent	IV-J	[25], [26]

II. REVIEW METHOD

This is a narrative review, not a systematic one. We searched arXiv, publisher sites and vendor documentation for each capability in Table I, using crypto-specific queries first and transferable work from equities and foreign exchange second. We preferred peer-reviewed sources, and we used arXiv preprints for fast-moving LLM-agent work, marking them as preprints. Vendor and comparison articles were used only to establish which features a platform documents, and we treat them as lower-reliability evidence; one of our sources is published by a company that promotes its own bot [11]. Platform entries therefore mean "documented in the sources we read", not "verified by testing", and commercial features change often. The review reflects sources available up to October 2026.

III. BACKGROUND: CRYPTO MARKET CHARACTERISTICS

A. *Market Structure*

Crypto assets trade 24 hours a day on many venues. Makarov and Schoar found large and recurring price deviations for the same asset across exchanges and countries, and argued that capital controls and limited arbitrage capital allow them to persist [24]. For a copilot, this means that "the price" of an asset is venue-dependent, which affects analysis, signals and order routing.

B. *Statistical Properties*

Asset returns in general show heavy tails, volatility clustering and weak linear autocorrelation [1]; extreme-value analysis of cryptocurrencies indicates tail behaviour that is relevant for risk estimation [2]. Models that assume normal returns will therefore understate risk, and point forecasts conceal it.

C. *Manipulation*

Gandal et al. showed that suspicious trading at a single exchange was associated with a large rise in the Bitcoin price in 2013 [3], and Griffin and Shams linked Tether issuance to subsequent Bitcoin price increases [27]. Pump-and-dump schemes organised on Telegram and Discord have been described and measured in detail [4], [28], [29], [22]. A trader who uses signals that ignore manipulation can be on the wrong side of such events.

D. *Design Implications*

These properties imply four design requirements: venue-aware data; probabilistic rather than point outputs; explicit manipulation awareness; and continuous, low-latency operation.

IV. REVIEW OF REQUIRED CAPABILITIES

A. *Trade Analysis and Price Prediction*

A survey of 146 papers organises cryptocurrency trading research into trading systems, bubbles and extreme events, volatility and return forecasting, portfolio construction and technical trading [13]. Deep sequence models such as LSTM [30] have been applied to Bitcoin; McNally et al. compared RNN and LSTM models with ARIMA [14], and Jaquart et al. evaluated several machine-learning models for short-term Bitcoin prediction [15].

Three cautions follow from this literature. First, forecast accuracy does not equal trading profit once fees and slippage are included. Second, backtests that try many configurations tend to overfit, which can be quantified by the probability of backtest overfitting [31]. Third, non-stationarity means models need monitoring and retraining. In this paper "trade analysis" has two parts: pre-trade analysis of a setup (trend, momentum, volume, liquidity) and post-trade analysis of the user's own trades. We did not find substantial academic work on the second part for retail crypto traders, so we treat it as a product gap rather than a research gap.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510159

B. *News-Based Volatility Analysis*

Volatility is commonly modelled with GARCH-family models [16], which have been compared on Bitcoin [17], and with realised-volatility models such as HAR [18]. Separately, Kraaijeveld and De Smedt found that Twitter sentiment has predictive power for the returns of some cryptocurrencies [19]. LLM-based agents such as CryptoTrade combine on-chain data with news [7]. In the sources we reviewed, price-based volatility models and sentiment models are mostly developed separately, and we did not find a widely adopted tool that forecasts volatility conditioned on classified news events with a stated impact window. This is the basis of Gap G2.

C. *Continuous Buy/Sell Prediction*

Deep reinforcement learning libraries such as FinRL provide environments for automated trading, though FinRL itself targets stocks [20]. LLM-agent systems produce trade decisions from multi-source inputs [7], [8], [9], [10]; CryptoTrade, for example, reviews the outcomes of previous daily decisions to refine later ones [7]. A survey of LLM trading agents compares their designs, data and backtested results and lists open obstacles [6].

A "continuous" signal differs from a daily decision. It requires streaming inference, a calibrated confidence for each signal, the ability to abstain, and hysteresis to avoid rapid flipping between buy and sell. LLM-based backtests also risk look-ahead contamination if the model was trained on the test period. We therefore recommend that signals come from quantitative models, with the LLM used to explain and orchestrate them.

D. *Strategies*

Commercial and open-source bot platforms offer templates such as grid, DCA (dollar-cost averaging), trailing orders, signal following, and, in Hummingbot, market making [11], [12]. Bitsgap is described as matching bot types to market styles such as sideways or trending conditions [11], and CryptoRobotics offers low, balanced and high volatility tiers [11]. Automated, evidence-based selection of a strategy and its parameters from the current regime is not documented in these sources. This matters because a grid strategy tends to accumulate losing inventory in a persistent trend, while a trend-following strategy loses in a range.

E. *Accumulation, Manipulation and Distribution*

The Wyckoff method describes market cycles as accumulation, mark-up, distribution and mark-down (Fig. 1). Pal proposed CNN and LSTM models to detect Wyckoff accumulation patterns, but on currency (foreign exchange) data and as a preprint [21]. For manipulation, La Morgia et al. monitored Telegram and Discord pump-and-dump groups, identified about 900 events, and trained a model that detects a pump about 25 seconds after it starts with an F1 score of 94.5% [22]. Bolz et al. built a live system that uses NLP on Telegram messages to predict the targeted coin and ranked the true target in the top five of 50 candidates in 24 of 43 events [23].

Two observations follow. First, existing detectors focus on organised pump-and-dump events on small coins and rely heavily on social-channel signals; other manipulation forms, such as spoofing, wash trading and stop hunting, are not covered by the sources we reviewed. Second, we found no system that connects phase detection to strategy gating (for example, suppressing grid buying during suspected distribution). Wyckoff phase labels are also partly subjective, so labelled data will be noisy.

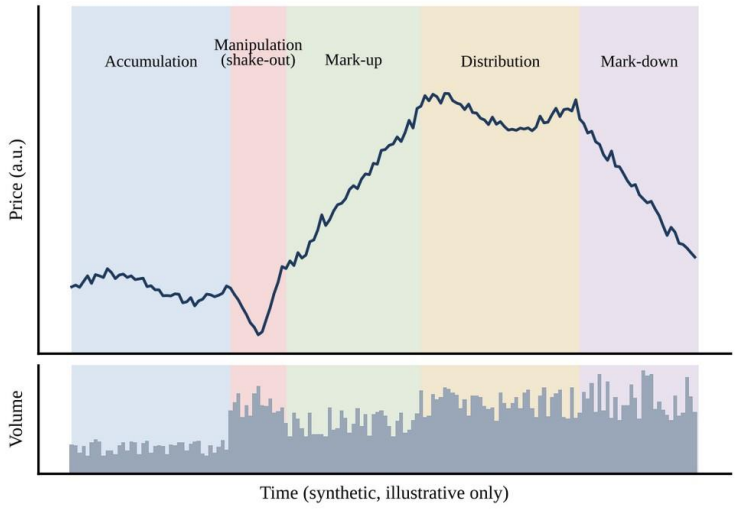


Fig. 1. Schematic of the AMD cycle with an illustrative shake-out. The data are synthetic and for explanation only.

 This work is licensed under a Creative Commons Attribution 4.0 International License



International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026
DOI: 10.17148/IJARCCE.2026.1510159

F. *Price Distribution and Breakdown*

Because returns are heavy-tailed [1], [2], a copilot should present forward price as a distribution (quantiles and tail probabilities) rather than a single target. Volume-at-price views are widely available in charting software and show where trading has concentrated. We propose combining them with two outputs: (i) the estimated probability that a support or resistance zone breaks within a horizon, and (ii) a breakdown of a recent move into contributions from market-wide movement, news events, order flow and venue-specific effects. The second output is a design proposal; we are not aware of an established method for it in crypto.

G. *Price Across Exchanges*

Cross-exchange deviations are documented and can persist [24]. A price-exchange module should (i) compare quotes across venues in real time, (ii) flag stale or anomalous feeds, (iii) report spreads and regional premiums, and (iv) support best-execution venue selection. Arbitrage is offered by some bot platforms [11], but as an execution strategy, not as a data-quality and venue-selection signal for all other modules.

H. *Trading Bots and Order Placement*

The documented platform landscape is summarised in Table II. A key design difference is custody and key handling. Hummingbot is open-source and runs on the user's machine or server, keeping keys local, whereas Cryptohopper is a closed-source cloud service that holds trade-enabled API keys [12]. Pionex is exchange-based, which the source notes adds custody risk [11]. A copilot that places orders must at minimum use trade-only keys without withdrawal rights, enforce limits on order size and exposure, handle partial fills and rate limits, make order submission idempotent, and provide a kill switch. These are general engineering requirements rather than findings from the cited sources.

I. *Notifications and Alerts*

Cryptohopper integrates TradingView alerts and trade alerts [12], and real-time pump detectors raise a flag when an event starts [22]. Typical alerts in charting and bot tools are threshold or indicator triggers. We did not find documentation of alerts that are conditioned on volatility regime, manipulation phase or signal confidence, or that rank and merge alerts to limit fatigue. This is Gap G7.

J. *Agent*

FinGPT proposed open-source financial LLMs including robo-advisory use [5]; FinMem added layered memory [32]; ReAct interleaves reasoning with tool calls [26]. CryptoBench, a live benchmark of crypto analysis questions, reports a "retrieval–prediction imbalance": strong models retrieve data well but struggle with predictive analysis [25]. This supports a design in which the agent reports and explains outputs computed by dedicated tools instead of producing forecasts itself. Feature-attribution methods such as SHAP [33] can support explanations of model-based signals.

K. *Additional Requirements of Crypto Traders*

Beyond the requested list, we recommend the following features for consideration. They come from the characteristics of the market rather than from a single cited study:

- derivatives context: funding rates, open interest and liquidation data for perpetual contracts;
- on-chain and whale-flow monitoring, and stablecoin peg monitoring given the evidence in [27];
- listing, delisting and token-unlock calendars as event inputs to the volatility module;
- portfolio and profit-and-loss tracking across venues, with a trade journal;
- risk management: position sizing, stop rules, maximum drawdown limits, and a global kill switch;
- paper trading and walk-forward backtesting with realistic fees and slippage;
- secure key management, audit logs and clear "not financial advice" disclosures; and
- tax and compliance export, noting that regulation varies by jurisdiction.

V. EXISTING SYSTEMS

Table II compares representative systems from four groups. Execution-centric bot platforms (3Commas, Cryptohopper, Pionex, Bitsgap, Hummingbot) cover order placement and strategy templates [11], [12]. Analysis-centric LLM research systems (CryptoTrade, FinAgent, TradingAgents, FinCon) cover signals and news but are evaluated in simulation [7], [8], [9], [10]. Detection-centric systems cover manipulation [22], [23]. TradingAgents, for instance, assigns LLM agents to analyst, researcher, risk and trader roles and reports better returns and risk-adjusted performance than baselines in its experiments [9]; FinAgent reports results on six stock and cryptocurrency datasets [8]. None of the systems we reviewed combines all groups.

The platforms in Table II are not exhaustive and the entries should be re-checked against vendor documentation before submission.

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510159

TABLE II COMPARISON OF REPRESENTATIVE EXISTING SYSTEMS

System	Type	Signals	News / volatility	Manipulation	Execution	Alerts	Agent	Source
3Commas	Commercial bot platform	Part (signal-integrated bots)	n/d	n/d	Yes	n/d	n/d	[11]
Cryptohopper	Commercial cloud bot	Part (signal marketplace, "A.I." trend control)	n/d	n/d	Yes	Part (TradingView alerts)	n/d	[11], [12]
Pionex	Exchange-embedded bots	n/d	n/d	n/d	Yes (grid, DCA)	n/d	n/d	[11]
Bitsgap	Commercial bot platform	n/d	Part (bot type by market style)	n/d	Yes	n/d	n/d	[11]
Hummingbot	Open-source, self-hosted	– (signal features on roadmap)	–	–	Yes (incl. market making)	n/d	n/d	[12]
CryptoTrade	LLM agent (research)	Yes (daily)	Yes (news, on-chain)	–	– (simulated)	–	–	[7]
FinAgent	Multimodal agent (stocks, crypto)	Yes	Part	–	– (simulated)	–	–	[8]
TradingAgents	Multi-agent LLM (research)	Yes	Yes (news–sentiment analyst)	–	– (simulated)	–	–	[9]
FinCon	Multi-agent LLM (research)	Yes	Part	–	– (simulated)	–	–	[10]
La Morgia et al.	Pump-and-dump detector	–	–	Yes (real time)	–	Part (real-time flag)	–	[22]
Bolz et al.	Telegram pump-target predictor	–	Part (NLP on Telegram)	Yes	–	Part	–	[23]
Trading Copilot (proposed)	Unified platform	Yes	Yes	Yes	Yes	Yes	Yes	This work

Yes = documented as a feature; Part = partly documented; – = outside the scope of the published work; n/d = not documented in the sources reviewed (this does not mean the feature is absent). Entries for the proposed system are design targets, not results.

VI. DISCUSSION OF GAPS

Table III lists ten gaps. G1 is the root gap: the capabilities exist but are spread across execution platforms, research agents and detectors. G2–G6 are analytical gaps where the literature offers components but not an integrated, actionable form. G7–G8 concern how information reaches the trader. G9–G10 concern trust: evaluation and security. We consider G1 and G4 the most novel contributions of a Trading Copilot, because we found the least prior work on them in the sources reviewed.

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510159

TABLE III GAP ANALYSIS: LIMITATIONS OF EXISTING SYSTEMS AND PROPOSED TRADING COPILOT ADDITIONS

ID	Gap in existing systems	Evidence from review	Trading Copilot addition
G1	Analysis, execution and explanation sit in separate systems; bots execute but document little analysis, while research agents analyse but do not execute.	Table II: no system covers all groups [11], [12], [7], [9].	A closed loop from analysis to signal, risk check, order and explanation, with every step logged.
G2	Volatility and news are modelled separately; no widely adopted news-conditioned volatility forecast with an impact window.	Price-based models [16], [18]; sentiment models [19]; agents use news mainly for decisions [7].	News event detection and classification feeding a volatility forecast and regime flag, benchmarked against GARCH and HAR baselines.
G3	Agent decisions are reported at daily granularity and in simulation; "continuous" signals with calibrated confidence are not documented.	CryptoTrade reviews daily decisions [7]; agents evaluated in backtests [8], [9], [10].	Streaming signals with calibrated confidence, abstention and hysteresis; LLM explains but does not generate the forecast.
G4	Manipulation detectors target announced pump-and-dumps and are not tied to trading decisions; AMD detection exists only as a forex preprint.	[22], [23], [21].	An AMD phase classifier on price, volume and order-flow features, plus pump alerts, used to gate strategies and raise alerts.
G5	Cross-exchange spreads are studied and used for arbitrage, but not as a data-quality and best-execution input to all modules.	[24], [11].	A venue monitor for spreads, stale-feed detection and best-execution routing.
G6	Forward price is shown as point targets or indicators, although returns are heavy-tailed; no probability of level breakdown.	[1], [2].	Quantile forecasts, tail probabilities, level-breakdown probabilities and a factor breakdown of recent moves.
G7	Alerts are mostly threshold-based and not conditioned on regime, phase or confidence; fatigue is not addressed.	[12], [22].	Context-aware, ranked and de-duplicated alerts with a short explanation and multi-channel delivery.
G8	LLM agents retrieve data well but are weak at prediction, so an agent that produces its own forecasts is unreliable.	[25].	A tool-grounded agent that quotes only computed values, checks each number against tool output and requires confirmation before any order.
G9	Evaluation is inconsistent: profit is often reported without full costs, and overfitting is rarely controlled.	[31], [6].	Built-in walk-forward testing, fee and slippage models, paper trading and shadow mode before live use.
G10	Cloud bots may hold trade-enabled API keys; some platforms add custody risk.	[12], [11].	Trade-only, encrypted keys with no withdrawal rights, exposure limits, audit logs and a kill switch.

VII. PROPOSED SYSTEM ADDITIONS

A. *Architecture*

Fig. 2 shows a layered architecture. Data sources feed a streaming layer; six analytics modules produce features and forecasts; a strategy engine and risk manager decide whether and how to act; and the order manager, alert service and conversational agent expose the results. The risk manager sits between analytics and execution so that no module, including the LLM agent, can place an order without passing its checks.

B. *Analytics Modules*

Each module should publish a typed output with a timestamp, a confidence value and its inputs, so that the agent and alert service can cite it. **Trade analysis** summarises setup quality and user trade history. **News-based volatility analysis** classifies news events and forecasts volatility with them as covariates, benchmarked against GARCH and HAR baselines. **Continuous prediction** produces calibrated buy, sell and hold probabilities at a fixed cadence, with abstention. **AMD detection** classifies the current phase and flags suspected pump events. **Price distribution and breakdown** outputs

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026

DOI: 10.17148/IJARCCE.2026.1510159

quantiles, tail probabilities and level-breakdown probabilities. **The cross-exchange monitor** outputs spreads, stale-feed flags and a preferred venue.

C. *Strategy, Risk and Execution*

The strategy engine selects from a library (trend following, range or grid, breakout, DCA and cross-exchange arbitrage) using regime, AMD phase and volatility forecasts. The risk manager enforces position size, stop rules, exposure and drawdown limits. The order manager supports a paper mode, a shadow mode that records orders without sending them, and a live mode with trade-only keys.

D. *Alerts and Agent*

Alerts combine user rules with model-generated events and carry a severity, a one-line reason and a link to the evidence. The agent uses tool calls [26], answers by quoting tool outputs, and explains model signals. For any order request it repeats the order details and requires explicit confirmation. These choices respond to the retrieval–prediction imbalance [25].

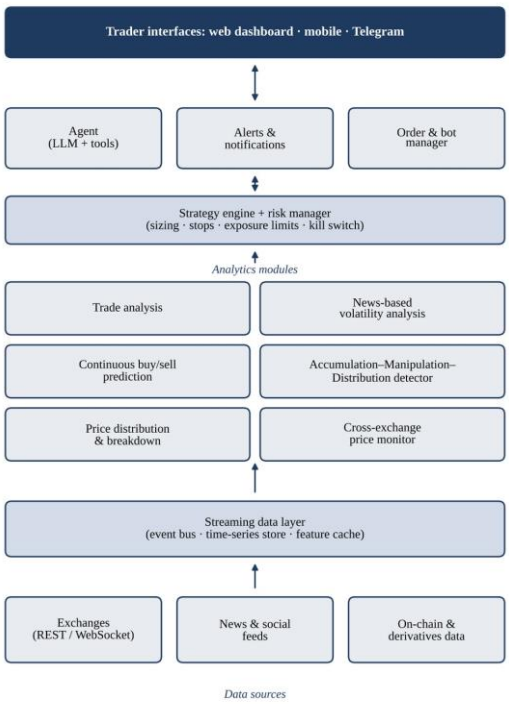


Fig. 2. Proposed Trading Copilot architecture.

VIII. EVALUATION PROTOCOL

Table IV proposes metrics for each capability. For all predictive modules we recommend walk-forward evaluation with purging and embargo, fees and slippage included, and reporting of the number of configurations tried to limit overfitting [31]. Live evaluation should proceed from paper trading to shadow mode to small live capital.

TABLE IV PROPOSED EVALUATION METRICS

Capability	Metrics	Notes
Signals	Directional accuracy; Brier score; Sharpe ratio; maximum drawdown; turnover	Net of fees and slippage; walk-forward.
News volatility	QLIKE and RMSE versus GARCH and HAR	Event-window analysis around news.
AMD detection	Precision, recall, F1; detection lead time	Labels from pump datasets and expert-labelled phases.
Distribution	Pinball loss; interval coverage	Check tail quantiles separately.
Cross-exchange	Spread captured net of fees; feed-staleness detection rate	Latency reported.
Execution	Slippage versus arrival price; fill rate; rejection and recovery rate	Includes failure-injection tests.

 This work is licensed under a Creative Commons Attribution 4.0 International License



International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 10, October 2026
DOI: 10.17148/IJARCCE.2026.1510159

Capability	Metrics	Notes
Alerts	Precision; alerts per day; alert-to-action latency	Fatigue measured by user study.
Agent	Share of numbers traceable to tool outputs; task success; response time; order-confirmation error rate	Includes tests of order-confirmation prompts.

IX. LIMITATIONS AND RISKS

This is a narrative review with a limited number of sources per capability, so absent evidence in our review should be read as “not found”, not “does not exist”. Vendor documentation changes quickly, and several commercial entries rely on a comparison article published by a company with its own product [11]. Several LLM-agent papers are preprints and report results from simulations that may not transfer to live markets. The proposed system is a design; we report no performance results. Automated trading can lose money, manipulation detection will produce false positives and false negatives, and regulation of crypto trading and of automated advice varies by jurisdiction. The system should be presented as decision support, not financial advice, and legal review is advisable before deployment.

X. CONCLUSION AND FUTURE WORK

Crypto traders currently assemble analysis, detection, alerts, execution and conversation from separate tools. Our review of the literature and of representative systems identifies ten gaps, the most important being the lack of an integrated, explainable loop from analysis to execution and the lack of manipulation-aware strategy gating. We described a Trading Copilot architecture and an evaluation protocol that address them. Next steps are to implement the modules, build a labelled AMD dataset, benchmark the news-conditioned volatility model against GARCH and HAR baselines, and run paper-trading and shadow-mode studies before any live deployment.

REFERENCES

[1]. R. Cont, "Empirical properties of asset returns: Stylized facts and statistical issues," *Quantitative Finance*, vol. 1, no. 2, pp. 223–236, 2001. [Online]. Available: https://ideas.repec.org/a/taf/quantf/v1y2001i2p223-236.html
[2]. K. Gkillas and P. Katsiampa, "An application of extreme value theory to cryptocurrencies," *Economics Letters*, vol. 164, pp. 109–111, 2018. [Online]. Available: https://shura.shu.ac.uk/18993/
[3]. N. Gandal, J. T. Hamrick, T. Moore, and T. Oberman, "Price manipulation in the Bitcoin ecosystem," *J. Monetary Economics*, vol. 95, pp. 86–96, 2018. [Online]. Available: https://par.nsf.gov/servlets/purl/10066235
[4]. J. Kamps and B. Kleinberg, "To the moon: Defining and detecting cryptocurrency pump-and-dumps," *Crime Science*, vol. 7, Art. 18, 2018. [Online]. Available: https://crimesciencejournal.biomedcentral.com/articles/10.1186/s40163-018-0093-5
[5]. H. Yang, X.-Y. Liu, and C. D. Wang, "FinGPT: Open-source financial large language models," *arXiv*:2306.06031, 2023. [Online]. Available: https://arxiv.org/abs/2306.06031
[6]. H. Ding, Y. Li, J. Wang, H. Chen, D. Guo, and Y. Zhang, "Large language model agent in financial trading: A survey," *arXiv*:2408.06361, 2024. [Online]. Available: https://arxiv.org/abs/2408.06361
[7]. Y. Li, B. Luo, Q. Wang, N. Chen, X. Liu, and B. He, "A reflective LLM-based agent to guide zero-shot cryptocurrency trading," *arXiv*:2407.09546, 2024. [Online]. Available: https://arxiv.org/abs/2407.09546
[8]. W. Zhang et al., "A multimodal foundation agent for financial trading: Tool-augmented, diversified, and generalist," *arXiv*:2402.18485, 2024. [Online]. Available: https://arxiv.org/abs/2402.18485
[9]. Y. Xiao, E. Sun, D. Luo, and W. Wang, "TradingAgents: Multi-agents LLM financial trading framework," *arXiv*:2412.20138, 2024. [Online]. Available: https://arxiv.org/abs/2412.20138
[10]. Y. Yu et al., "FinCon: A synthesized LLM multi-agent system with conceptual verbal reinforcement for enhanced financial decision making," *arXiv*:2407.06567, 2024. [Online]. Available: https://arxiv.org/abs/2407.06567
[11]. SSA Group, "Choosing a crypto trading bot: Key platforms compared," web article. [Online]. Available: https://www.ssa.group/blog/choosing-a-crypto-trading-bot-key-platforms-compared/ (accessed Oct. 14, 2026). Note: vendor-affiliated source.
[12]. Hummingbot, "Hummingbot vs. Cryptohopper," *Hummingbot Academy*. [Online]. Available: https://hummingbot.org/academy-content/hummingbot-vs-cryptohopper/ (accessed Oct. 14, 2026).
[13]. F. Fang, C. Ventre, M. Basios, L. Kanthan, L. Li, D. Martinez-Regoband, and F. Wu, "Cryptocurrency trading: A comprehensive survey," *arXiv*:2003.11352, 2022. [Online]. Available: https://arxiv.org/abs/2003.11352
[14]. S. McNally, J. Roche, and S. Caton, "Predicting the price of Bitcoin using machine learning," in *Proc. 26th Euromicro Int. Conf. Parallel, Distributed and Network-based Processing (PDP)*, 2018, pp. 339–343. [Online]. Available: https://www.academia.edu/81749323/Predicting_the_Price_of_Bitcoin_Using_Machine_Learning
[15]. P. Jaquart, D. Dann, and C. Weinhardt, "Short-term bitcoin market prediction via machine learning," *J. Finance and Data Science*, vol. 7, pp. 45–66, 2021. [Online]. Available: https://publikationen.bibliothek.kit.edu/1000150665

DOI: 10.17148/IJARCCE.2026.1510159

[16]. T. Bollerslev, "Generalized autoregressive conditional heteroskedasticity," *J. Econometrics*, vol. 31, no. 3, pp. 307–327, 1986. [Online]. Available: https://ideas.repec.org/a/eee/econom/v31y1986i3p307-327.html

[17]. P. Katsiampa, "Volatility estimation for Bitcoin: A comparison of GARCH models," *Economics Letters*, vol. 158, pp. 3–6, 2017. [Online]. Available: https://shura.shu.ac.uk/16526/

[18]. F. Corsi, "A simple approximate long-memory model of realized volatility," *J. Financial Econometrics*, vol. 7, no. 2, pp. 174–196, 2009. [Online]. Available: https://ideas.repec.org/a/oup/jfinec/v7y2009i2p174-196.html

[19]. D. Kraaijeveld and J. De Smedt, "The predictive power of public Twitter sentiment for forecasting cryptocurrency prices," *J. Int. Financial Markets, Institutions and Money*, vol. 65, Art. 101188, 2020. [Online]. Available: https://ideas.repec.org/a/eee/intfin/v65y2020ics104244312030072x.html

[20]. X.-Y. Liu, H. Yang, Q. Chen, R. Zhang, L. Yang, B. Xiao, and C. D. Wang, "FinRL: A deep reinforcement learning library for automated stock trading in quantitative finance," *arXiv*:2011.09607, 2020. [Online]. Available: https://arxiv.org/abs/2011.09607

[21]. J. Pal, "Long short-term memory pattern recognition in currency trading," *arXiv*:2403.18839, 2024. [Online]. Available: https://arxiv.org/abs/2403.18839

[22]. M. La Morgia, A. Mei, F. Sassi, and J. Stefa, "The Doge of Wall Street: Analysis and detection of pump and dump cryptocurrency manipulations," *ACM Trans. Internet Technology*, 2023, doi: 10.1145/3561300 (arXiv:2105.00733). [Online]. Available: https://arxiv.org/abs/2105.00733

[23]. M. Bolz et al., "Machine learning-based detection of pump-and-dump schemes in real-time," *arXiv*:2412.18848, 2024. [Online]. Available: https://arxiv.org/abs/2412.18848

[24]. I. Makarov and A. Schoar, "Trading and arbitrage in cryptocurrency markets," *J. Financial Economics*, vol. 135, no. 2, pp. 293–319, 2020. [Online]. Available: https://dspace.mit.edu/handle/1721.1/130495

[25]. J. Guo et al., "CryptoBench: A dynamic benchmark for expert-level evaluation of LLM agents in cryptocurrency," *arXiv*:2512.00417, 2025. [Online]. Available: https://arxiv.org/abs/2512.00417

[26]. S. Yao et al., "ReAct: Synergizing reasoning and acting in language models," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2023 (arXiv:2210.03629). [Online]. Available: https://arxiv.org/abs/2210.03629

[27]. J. M. Griffin and A. Shams, "Is Bitcoin really untethered?" *J. Finance*, vol. 75, no. 4, pp. 1913–1964, 2020. [Online]. Available: https://papers.ssrn.com/abstract=3195066

[28]. J. Xu and B. Livshits, "The anatomy of a cryptocurrency pump-and-dump scheme," in *Proc. 28th USENIX Security Symposium*, 2019. [Online]. Available: https://arxiv.org/abs/1811.10109

[29]. M. La Morgia, A. Mei, F. Sassi, and J. Stefa, "Pump and dumps in the Bitcoin era: Real time detection of cryptocurrency market manipulations," in *Proc. IEEE Int. Conf. Computer Communications and Networks (ICCCN)*, 2020, doi: 10.1109/ICCCN49398.2020.9209660. [Online]. Available: https://arxiv.org/abs/2005.06610

[30]. S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. [Online]. Available: https://doi.org/10.1162/neco.1997.9.8.1735

[31]. D. H. Bailey, J. M. Borwein, M. López de Prado, and Q. J. Zhu, "The probability of backtest overfitting," *J. Computational Finance*, vol. 20, no. 4, pp. 39–69, 2017. [Online]. Available: https://papers.ssrn.com/abstract=2326253

[32]. Y. Yu, H. Li, Z. Chen, Y. Jiang, Y. Li, D. Zhang, R. Liu, J. W. Suchow, and K. Khashanah, "FinMem: A performance-enhanced LLM trading agent with layered memory and character design," *arXiv*:2311.13743, 2023. [Online]. Available: https://arxiv.org/abs/2311.13743

[33]. S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2017, pp. 4765–4774. [Online]. Available: https://arxiv.org/abs/1705.07874

 This work is licensed under a Creative Commons Attribution 4.0 International License