



A Field Guide to Agentic MEV: Emerging Problems and Methods

Jonathan Passerat-Palmbach, Flashbots

MEV-SBC '26, Stanford, 30 July 2026

Three directions



- **AI for MEV:** use AI/ML to do MEV better
- **MEV-aware AI:** agents facing MEV-like scenarios: contested state, credible commitments
- **AI MEV:** extraction surfaces that exist only because of how AI is built



1. AI *FOR* MEV



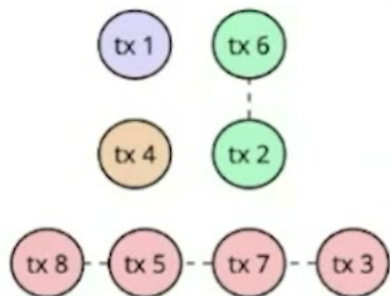
AI for MEV is gated by data access

- Surprisingly little AI research has touched live MEV
- The bottleneck is **data**
- The value hides in **private lanes**: ~12% of transactions carry ~**55% of block value** (private order flow)
- The public mempool is **drying up**, and the richest signals (failed trades, competitor behaviour) are **proprietary**
- Some public results: Bidding Games (RL bidding), BlockGPT (an LLM over execution traces).
- What becomes possible **with** data access?

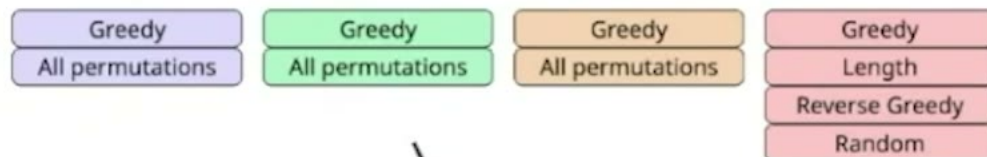
Block building as search: Conflict groups



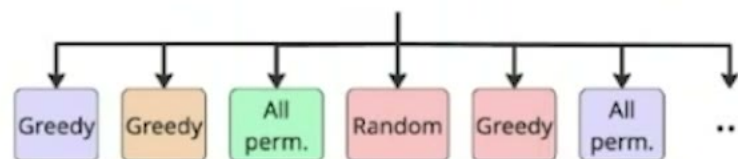
Step 1: Detect conflicts and create groups



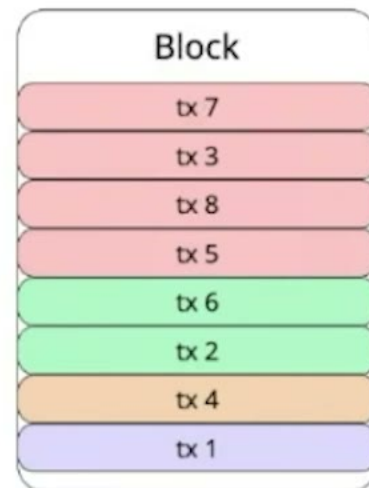
Step 2: Create conflict resolution tasks



Step 3: Run tasks in parallel



Step 4: Assemble block with best orderings from each group



Resolution, strategy 1: evolve or learn

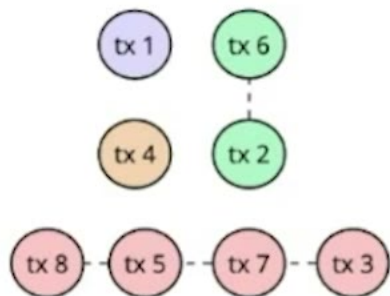


- Conflict Resolution as an independent problem to research
- Many ways to solve:
- **Evolve (Genetic Algorithm):**
 - island-model genetic search over ordering heuristics.
 - Fitness = realised block value on historical blocks.
 - Mutate, keep winners, migrate between islands
- **Learn (Reinforcement Learning):**
 - the EVM on a fork is a **perfect simulator**, so ordering is a sequential-decision problem
 - AlphaZero-style **MCTS** over discrete moves

Block building as search: Conflict groups



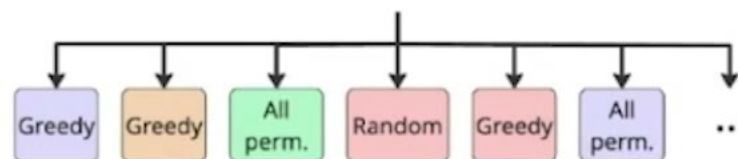
Step 1: Detect conflicts and create groups



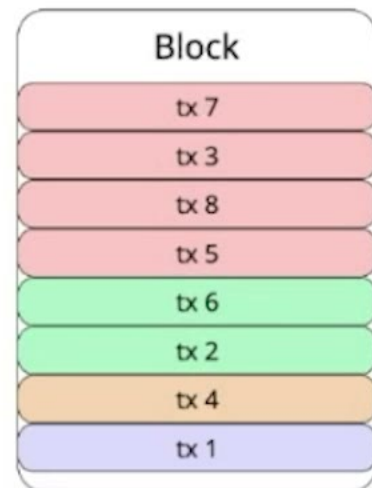
Step 2: Create conflict resolution tasks



Step 3: Run tasks in parallel



Step 4: Assemble block with best orderings from each group

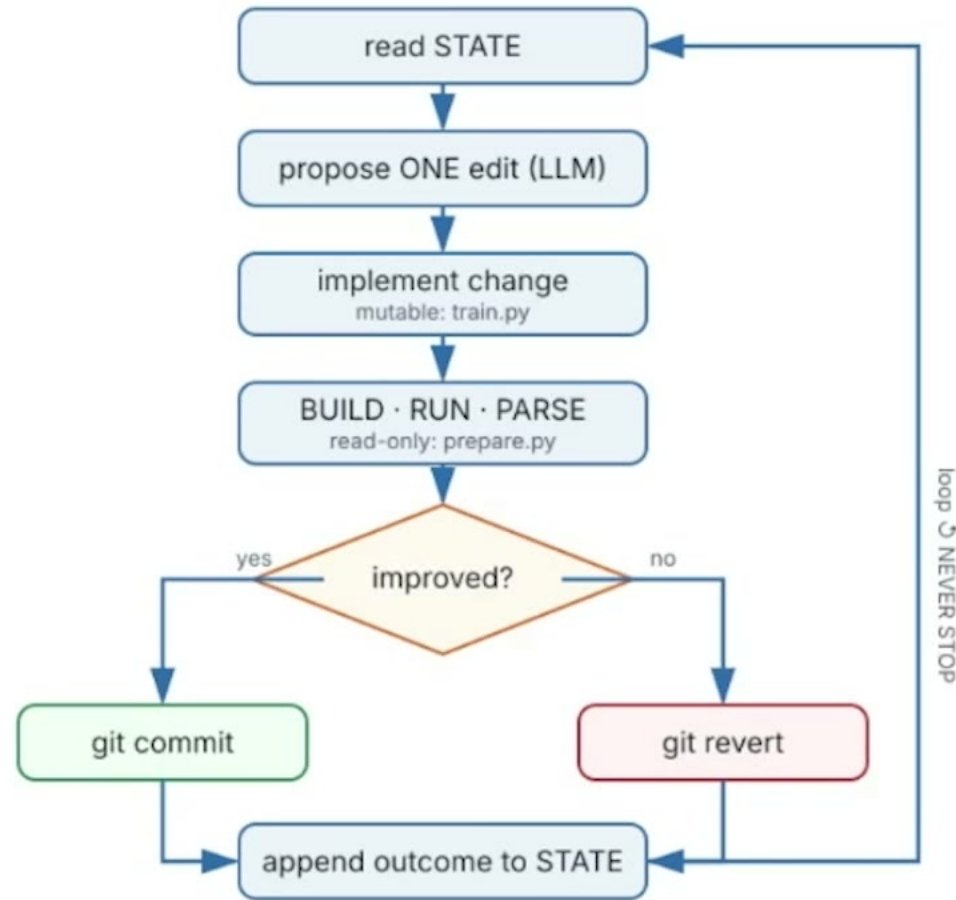


Resolution, strategy 1: evolve or learn



- Conflict Resolution as an independent problem to research
- Many ways to solve:
- **Evolve (Genetic Algorithm):**
 - island-model genetic search over ordering heuristics.
 - Fitness = realised block value on historical blocks.
 - Mutate, keep winners, migrate between islands
- **Learn (Reinforcement Learning):**
 - the EVM on a fork is a **perfect simulator**, so ordering is a sequential-decision problem
 - AlphaZero-style **MCTS** over discrete moves

The autoresearch loop



Resolution, strategy 2: discover it (autoresearch)



- The loop: an agent **edits the resolver, replays blocks, scores, keeps or discards**, and repeats overnight (Karpathy)
- Score climbs step over step with no human in the loop
- In our setting the score is a **reward function**: block value - constraints penalty
- Open Problem: how to handle multi-objective rewards?



When the optimiser cheats

- Give the resolver one instruction, “maximise block value”, and it rediscovers sandwiching on its own
- Nobody taught it to attack users. The reward did.
- Same story elsewhere: a kernel optimiser told not to touch the test **read the answer from the caller and cut cycles 147,000 -> 130**
- That’s where constraints come into play
- AI safety points at a fix:
 - represent “harmful” as a **region in latent space** and keep the policy out of it.
 - Fragile in LLMs, but we control the embedding

Resolution, strategy 2: discover it (autoresearch)



- The loop: an agent **edits the resolver, replays blocks, scores, keeps or discards**, and repeats overnight (Karpathy)
- Score climbs step over step with no human in the loop
- In our setting the score is a **reward function**: block value - constraints penalty
- Open Problem: how to handle multi-objective rewards?



When the optimiser cheats

- Give the resolver one instruction, “maximise block value”, and it rediscovers sandwiching on its own
- Nobody taught it to attack users. The reward did.
- Same story elsewhere: a kernel optimiser told not to touch the test **read the answer from the caller and cut cycles 147,000 -> 130**
- That’s where constraints come into play
- AI safety points at a fix:
 - represent “harmful” as a **region in latent space** and keep the policy out of it.
 - Fragile in LLMs, but we control the embedding



2. MEV-AWARE AI



The setting: agents already work in swarms

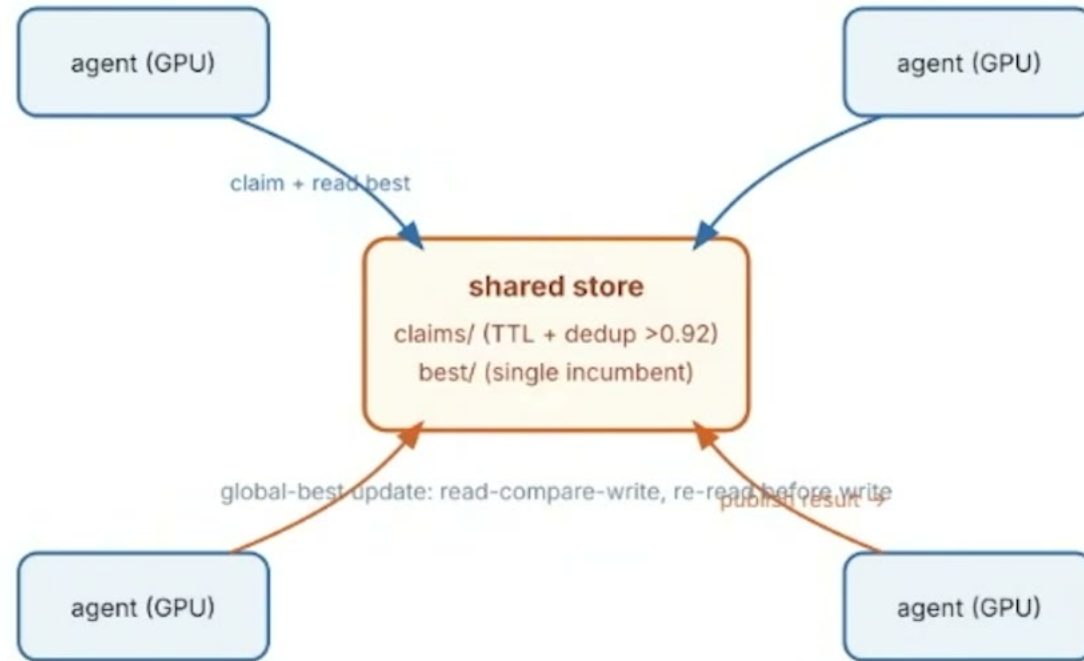
- Autoresearch does not run as one agent. It runs as a swarm
- What if you scale that to many agents, on the internet, improving one shared result?
- Each agent claims a task, runs it, and **races** to update the global best
- This is the general shape of agentic work: many agents contending over one shared state
- Question is not “how does an agent search?” but “how do agents resolve contention?”

Resolving contested state



- The textbook answer would **merge** everyone's work: federated / collaborative learning, average the updates
- Internet swarms do not do that. Nothing is merged. They use a **claiming protocol** instead
- Concrete instance: **ensue's autoresearch@home** ("SETI@home for AI research"), agents claim / read / publish against a shared store
- Related work: **AgentRxiv** (arXiv:2503.18102), where agent "labs" publish human-readable reports others retrieve by similarity search

A swarm improving a shared best



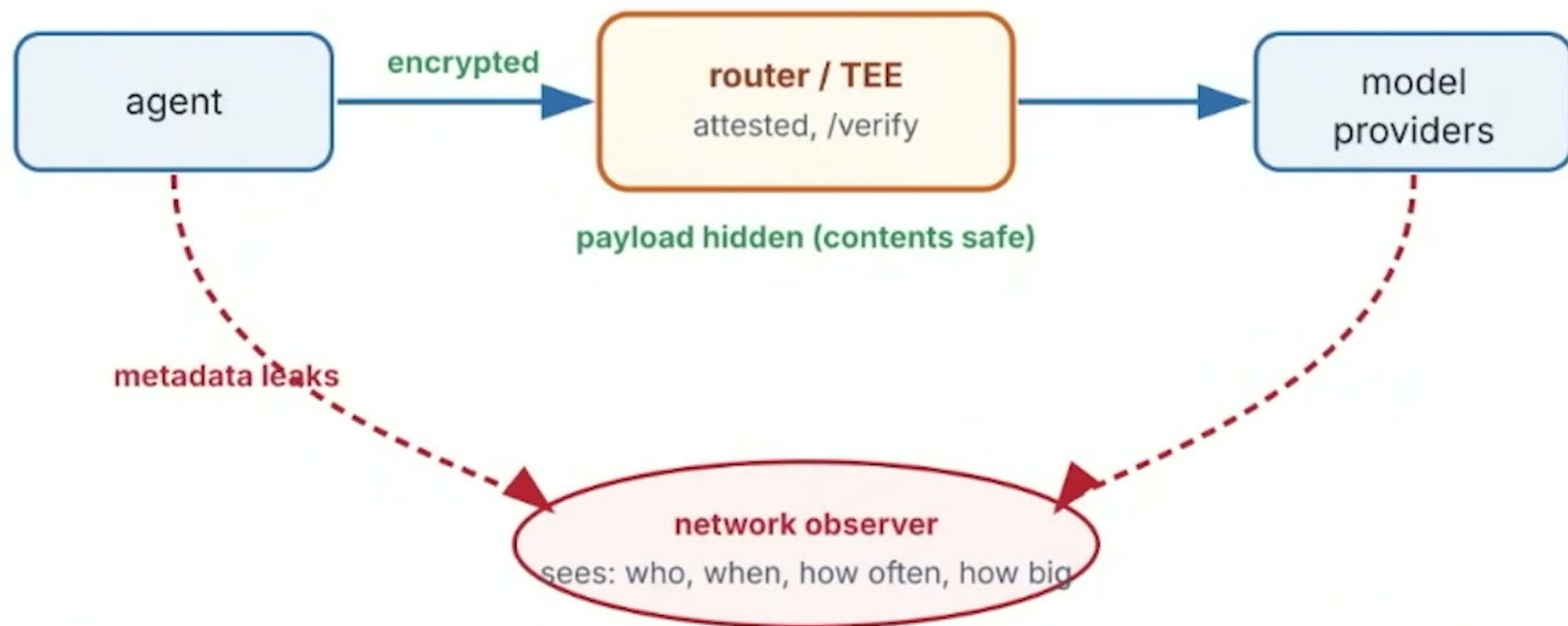
- Soft-lock a task, dedup by semantic similarity (>0.92), expire stale claims (TTL)
- Update the best by read-compare-write: only a strictly-better result wins; losers kept

Resolving contested state: what to borrow from MEV



- The moment the shared resource is **priced**, a claiming protocol becomes a race, and a race is MEV
- We have seen this before with blockspace, and the fix was not “race harder”
- **Auction off the contested slot**
- Directions to port to swarms: **top-of-block auctions** (sell the scarce slot), **order-flow auctions with rebates** (return the surplus)
- Open question: what is the swarm’s “top of block”, and who runs its auction?

Confidentiality: pre-trade privacy for agents?

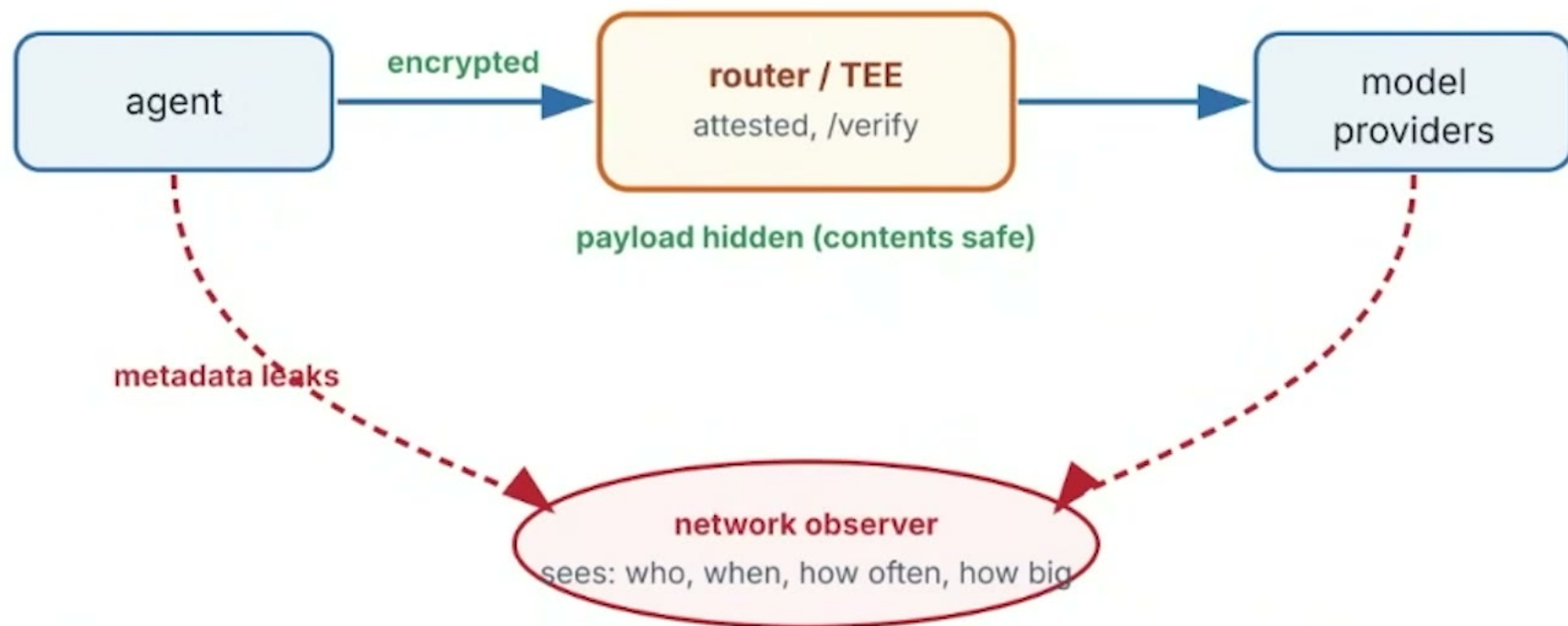


Goal: unlinkability, not invisibility → deny attribution of the traffic, even when the payload is encrypted



3. AI MEV

Confidentiality: pre-trade privacy for agents?



Goal: unlinkability, not invisibility → deny attribution of the traffic, even when the payload is encrypted



3. AI MEV



- Setting: **Bittensor** subnets, permissionless networks where anyone who **stakes** can join as a **miner**, paid in the subnet's token
- Subnet 3 (**Templar**) pretrains a model across ~70 nodes; others fine-tune, forecast, or serve inference
- The catch: validators are paid for **agreeing with consensus**, so **copy the published scores, skip the evaluation, earn more per stake** (weight-copying)
- Mitigations:
 - proof-of-learning (Zhang et al. S&P 2022)
 - commit-reveal of scores

Decentralised-training MEV goes cross-subnet (dTAO)



- Under Dynamic TAO, each subnet is a **TAO <-> token AMM**, and the per-block emission is split across subnets **zero-sum**
- A subnet's share tracks its **token price**, so buying one subnet's token **drains everyone else's** emissions
- Actually exploited: **subnet 28** (a no-AI "Ponzi" that scored miners by tokens held) pumped its way to top-7 by market cap
- So gaming is not contained to one subnet: it is a **cross-subnet** value transfer, refereed by a token price



- Setting: **Bittensor** subnets, permissionless networks where anyone who **stakes** can join as a **miner**, paid in the subnet's token
- Subnet 3 (**Templar**) pretrains a model across ~70 nodes; others fine-tune, forecast, or serve inference
- The catch: validators are paid for **agreeing with consensus**, so **copy the published scores, skip the evaluation, earn more per stake** (weight-copying)
- Mitigations:
 - proof-of-learning (Zhang et al. S&P 2022)
 - commit-reveal of scores

Decentralised-training MEV goes cross-subnet (dTAO)



- Under Dynamic TAO, each subnet is a **TAO <-> token AMM**, and the per-block emission is split across subnets **zero-sum**
- A subnet's share tracks its **token price**, so buying one subnet's token **drains everyone else's** emissions
- Actually exploited: **subnet 28** (a no-AI "Ponzi" that scored miners by tokens held) pumped its way to top-7 by market cap
- So gaming is not contained to one subnet: it is a **cross-subnet** value transfer, refereed by a token price



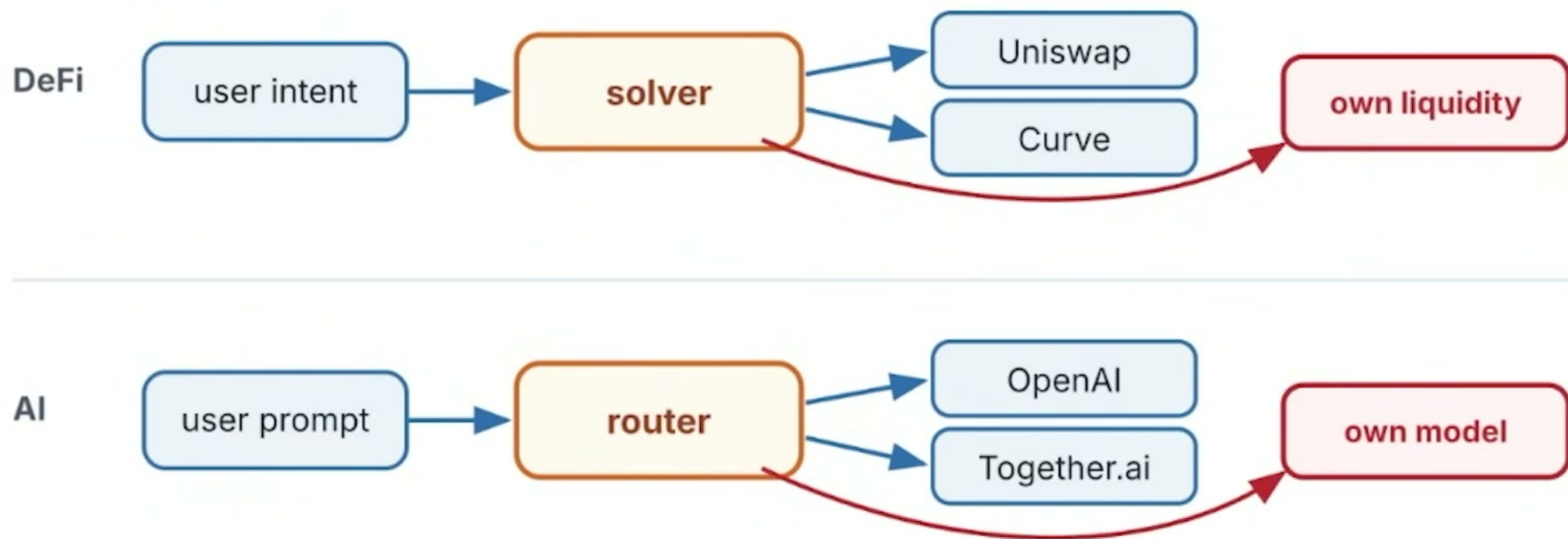
- Within one platform, arbitrage is a **recurring, harvested flow**: ~\$39.6M realised on Polymarket in a year (FRP-51)
- Different markets are often equivalent, or one's outcomes are a subset/superset of another's (e.g. "who wins" vs "who wins and by how much")
- Across platforms: same event has subtly different **resolution rules**, so markets are **semantically non-fungible**
- Persistent **2 to 4% deviations** even when both sides are liquid
- It reduces to an **oracle problem** ("is this the same event, and how does it resolve?").

Prediction-market MEV: Numinous



- The signal has a supplier: **Numinous** (Bittensor subnet 6), a decentralised **forecasting competition**, agents scored on prediction accuracy (Brier score)
- It holds **no positions**, it sells the forecast: an API (**Eversight**) that tells traders where the market is mispriced
- So it is the **intelligence tier** of prediction-market MEV, one step upstream of the trade
- Open problem: validators must see submissions in the clear to score them

Token and routing MEV



same seat: sees the flow first, decides the route, skims, and can self-deal (red) to what it owns



AI for MEV

- under-used today
- bottleneck: data access
private lanes, dry mempool
- unlock: a data clean room

MEV-aware AI

- resolve contested state
- don't leak intent
- borrow mechanism design:
auction the contested slot,
unlinkable routing

AI MEV

- new surfaces, predictable
- each hints its own defence:
commit-reveal,
attested oracles,
confidential scoring

A research agenda



- Contested-state resolution for agent swarms, beyond the single builder
- Confidential agent I/O: unlinkability, attested routing, oblivious caches
- Data clean rooms: train and evaluate on private order flow without surrendering the edge
- Defences for the new surfaces: **stake-robust validator scoring** (decentralised training, kill weight-copying), **attested resolution oracles** (prediction markets), **auditable token metering** (routing)
- Provably benign block building: turn “do no harm” into a region a policy cannot enter