"Bittensor : A Peer-to-Peer Intelligence Market"

Problem with Traditional AI → intelligence needs to be a commodity that is expensively mined, monetarily valueable, transferable and generally useful.
Bittensor's Sol^n → suggests a framework → machine intelligence is measured by other intelligence systems. Market operates as a network of computers sharing representations P2P using a digital ledger to record ranks and provide decentralized incentives.

Model

Intelligence defined as → $y = f(x)$, trained over dataset → $D = [X, Y]$ to minimize loss f^n → $\mathcal{L} = E_D[Q(y, f(x))]$

network have $\underline{\underline{n}}$ such f's → peers $(F = f_0, f_1, \cdots f_n)$
→ Each peer holds stakes → $S = [s_i]$, represented on digital ledger.

Objective → primary goal is → distribution of stake I as an ~~intat~~ incentive to peers who contribute to minimizing a stake-weighted ML objective : $\sum_i^n \mathcal{L}_i * s_i$. Distribution must be resistant to ~~collision~~ collusion.

→ Peer Ranking Algorithm → peers use the outputs of other functions as input to themselves → $f(F(x))$, & learn a set of weights $W = [w_{i,j}]$
→ Initial Ranking Calculation → Idealized scoring $\underline{R = W^T \cdot S}$ is achieved by setting weights using a Fisher's Information pruning score. ∴ each peer's incentive is equivalent to its pruning score.

→ Collusion Vulnerability → naive approach → not resistant to collusion → Peers could vote for themselves. The digital ledger cannot audit the internal parameters of each model, only the inter model weights W, making this attack trivial

Incentive Mechanism: Avoiding Collusion

Incentive Function → $I(W, S)$ → limits rewards to peers who have not reached network consensus. core assumption → no single group of peers holds 50% of network's stakes

Trust vulnerability → from weights W, trust matrix T is inferred → where $t_{i,j} = 1$ if there is a connection between peer i & j.

consensus C → peers are defined as having reached "consensus" if they have connections from more than 50% of the stake in the network. computed using continuous sigmoid f^n → $C = \sigma(\rho(T^T S - \kappa))$

sigmoid f^n

temperature

shift term

Scaled Incentive → final incentive I is calculated by scaling the original rankings R with the consensus term C: $I = R \cdot C$. → larger of two competing sub graphs will exponentially increase its proportion of the network through inflation over time

Bonds: Incentivizing Correct weight Selection

Speculation Based Reward → $b_{i,j} \in B$ represents proportion of bonds owned by peer i in peer j.

Bonds Accumulation → Bonds accumulate at each step similar to token inflation : $\Delta B = W \cdot S$ $(B_{t+1} = B_t + W \cdot S)$

Redistribution of Incentive → chain then redistributes the normal incentive scores $\Delta S = B^T \cdot I$

Final Stake update → $\Delta S = 0.5\, B^T I + 0.5 I$.

ΔS then determines network incentives for n peers,

updating total stake → $S_{t+1} = S_t + \tau \Delta S$

Reaching Consensus

→ Ensuring Honest Node Dominance.

→ Loss Term (L): → $L = -R \cdot (c - 0.5)$

→ If $\mathcal{L} < 0$ [redacted], → majority of inflation is being distributed towards peers with more than 0.5 consensus.

→ so chain increases no. of weights