1 Context and Problem Statement

TikTok creators increasingly earn through *donations/gifts*—coins purchased by viewers and sent via comments, stickers, and rewards. However, today's donation flow faces three issues:

- **Opaque splits:** Platform vs. creator shares are fixed or unclear, not responsive to content quality or ecosystem goals.
- **Bot/gaming risk:** Unverified activity can inflate gifts/engagement without reflecting genuine viewer value.
- **Uneven incentives:** Smaller or niche creators struggle to capture meaningful donation share despite high-quality content.

Goal. Design a *fair, quality-aware, and sustainable* split of *donation coins* between TikTok (platform) and creators, per video.

2 Business Interpretation (Donations/Gifting)

Viewers purchase coins and gift them on specific videos via comments/rewards. We distinguish:

- **Normal coins** (from unverified wallets/users).
- **Premium coins (ByteCoin)** (from verified wallets/users), which we treat as higher-trust contributions.

For each video v, the platform sets creator fractions x_n (Normal) and x_p (Premium). Our *per-video optimization* (implemented in `donate_opti.py`) chooses (x_n, x_p) to balance:

1. **Platform sustainability:** TikTok keeps enough revenue to fund safety, infra, and growth.
2. **Creator utility & quality:** Higher-quality content (measured via Q) earns more at the margin.
3. **Premium adoption:** Verified/premium gifts are rewarded more than normal gifts.

Key levers:

- *Policy bounds* $(x_n^{\max}, x_p^{\max})$ and a *minimum premium gap* δ to ensure premium gifts reward creators more.
- *Normalized, weighted objective* so revenue, utility, and premium-uplift trade-offs are comparable.
- *Quality score* Q (watch completion, engagement rate/diversity, rewatch, NLP safety) scales creator utility for each video.

3 Quality Score Q (Summary)

Q is a geometric aggregation (clamped to $[0, 1]$) of trust-weighted signals:

$$Q = C \cdot \min\Big(1,\ W^{\phi_W} E^{\phi_E} D^{\phi_D} R^{\phi_R} S^{\phi_S}\Big),$$

where W (watch completion), E (engagement rate with weights on like/comment/share/follow), D (engagement diversity), R (rewatch), S (NLP quality/safety), and $C \in \{0, 1\}$ is a compliance gate. Verified users have higher trust weight in all components.

4 Donation Coin Split (Single-Video Optimization)

This version is implemented in `donate_opti.py`.

Setup

For one video v, let:

- N_n: Normal coins given to this video.
- N_p: Premium coins given to this video.
- $Q \in [0,1]$: quality score for this video.
- $f(Q) \geq 0$: quality multiplier (e.g., $f(Q) = 1 + \theta Q$ with $\theta \geq 0$).

Decision variables (creator fractions for this video):

$$x_n \in [0, x_n^{\max}], \qquad x_p \in [0, x_p^{\max}].$$

Revenue and Payout

$$\text{TikTok revenue (kept)} : \quad R = (1 - x_n)\, N_n + (1 - x_p)\, N_p.$$

$$\text{Creator payout (this video)} : \quad \text{pay} = x_n\, N_n + x_p\, N_p.$$

Objective Before Normalization

Choose nonnegative weights $\lambda_{\text{rev}}, \lambda_{\text{util}}, \lambda_{\text{inc}}$. We maximize:

$$\max_{x_n, x_p} \quad \underbrace{\lambda_{\text{rev}}\, R}_{\text{TikTok revenue}} \quad + \quad \underbrace{\lambda_{\text{util}}\ f(Q)\ u(\text{pay})}_{\text{creator utility (quality-scaled)}} \quad + \quad \underbrace{\lambda_{\text{inc}}\,(x_p - x_n)}_{\text{premium adoption incentive}} \quad .$$

Utility choice $u(\cdot)$:

$$u(\text{pay}) = \begin{cases} \log(\varepsilon + \text{pay}) & \text{(diminishing returns, concave)} \\ \text{pay} & \text{(linear, simpler)} \end{cases} \qquad \text{with } \varepsilon > 0 \text{ small if using } \log.$$

Why Normalization?

- Without scaling, the revenue term R (in absolute coins) may dominate the entire objective, since payouts can be very large compared to $x_p - x_n$ or log-utilities.
- To ensure a balanced trade-off, each term is normalized into $[0,1]$, so that the weights $\lambda_{\text{rev}}, \lambda_{\text{util}}, \lambda_{\text{inc}}$ truly reflect their relative importance.

Normalized Terms

$$R_{\text{norm}} = \frac{R - R_{\min}}{R_{\max} - R_{\min}}, \qquad U_{\text{norm}} = \frac{f(Q)\, u(\text{pay})}{U_{\max}}, \qquad I_{\text{norm}} = \frac{(x_p - x_n) - \delta}{\Delta - \delta}.$$

where:

- $R_{\min}$: platform revenue floor.
- $R_{\max}$: maximum possible revenue if no coins are shared.
- $U_{\max}$: an upper bound on $f(Q)\, u(\text{pay})$, e.g. evaluated at $x_n^{\max}, x_p^{\max}$.
- δ: required minimum premium gap, Δ: maximum allowed premium gap.

Final Normalized Objective

The optimization problem becomes:

$$\max_{x_n, x_p} \quad \lambda_{\text{rev}}\, R_{\text{norm}} \; + \; \lambda_{\text{util}}\, U_{\text{norm}} \; + \; \lambda_{\text{inc}}\, I_{\text{norm}}.$$

Constraints

$$
\begin{aligned}
&x_p \;\geq\; x_n + \delta && \text{(premium must reward at least } \delta \text{ more)}\\
&x_p - x_n \;\leq\; \Delta && \text{(cap the premium gap)}\\
&0 \leq x_n \leq x_n^{\max}, \quad 0 \leq x_p \leq x_p^{\max} && \text{(policy bounds)}\\
&R \;\geq\; R_{\min} && \text{(per-video revenue floor)}
\end{aligned}
$$

Notes

- **Normalization:** Ensures each term contributes comparably; λ's now directly encode TikTok's trade-off priorities.
- **Quality sensitivity:** $f(Q) = 1 + \theta Q$ increases creator utility weight for higher Q.
- **Log vs linear:** $\log(\varepsilon + \text{pay})$ prevents corner solutions (always maxing x's) and keeps payouts reasonable even when N_n, N_p are large.
- **Convexity:** With log utility, the objective is concave in (x_n, x_p) and constraints are linear, so this is a well-behaved convex program per video.

How to read it

- If Q is higher, the utility term pulls x_n, x_p upward *for this video*, raising creator take from the same donated coins.
- Premium coins are structurally favored ($x_p \geq x_n + \delta$), encouraging verified, bot-resistant gifting.
- Normalization prevents any one term (e.g., raw revenue) from dominating the objective, so λ's reflect true business priorities.

5 Platform-Wide Per-Video Coin Split Optimization

This section presents the proposed optimization model applied at the *platform-wide* level.

Setup

Let:

- V: set of eligible videos (policy-compliant, $C_v = 1$).
- $N_n(v)$, $N_p(v)$: Normal and Premium coins given to video $v \in V$.
- $Q_v \in [0,1]$: quality score for video v.
- $f(Q_v) \geq 0$: quality multiplier (e.g. $f(Q_v) = 1 + \theta Q_v$).
- $w_v = \left(\dfrac{N_n(v) + N_p(v)}{\sum_{u \in V} N_n(u) + N_p(u)}\right)^{\alpha}$, or quality-centric $w_v \propto Q_v^{\alpha}$ with $\alpha \in (0,1]$.
- Policy constants: $x_n^{\max} \leq 1$, $x_p^{\max} \leq 1$, $\delta \geq 0$ (min premium gap), $\Delta \geq \delta$ (max gap), platform revenue floor $R_{\min}$.
- Small $\varepsilon > 0$ for log utility.

Decision Variables

For each video $v \in V$:

$$x_n(v) \in [0, x_n^{\max}], \qquad x_p(v) \in [0, x_p^{\max}].$$

Revenue and Payouts

Creator payout on video v:

$$\mathrm{pay}_v \;=\; x_n(v)\,N_n(v) \;+\; x_p(v)\,N_p(v).$$

Total TikTok revenue kept:

$$R \;=\; \sum_{v\in V} \Big((1 - x_n(v))\,N_n(v) \;+\; (1 - x_p(v))\,N_p(v) \Big).$$

Objective (Fairness–Efficiency–Incentive)

We maximize a convex combination of fairness, efficiency, and premium adoption terms:

$$\max_{\{x_n(v),x_p(v)\}_{v\in V}} \qquad \lambda_{\mathrm{fair}} \sum_{v\in V} w_v\, f(Q_v)\, \log\!\big(\varepsilon + \mathrm{pay}_v\big) + \lambda_{\mathrm{eff}} \sum_{v\in V} s_v \,\frac{\mathrm{pay}_v}{\sum_{u\in V} \mathrm{pay}_u} + \lambda_{\mathrm{inc}} \sum_{v\in V} \omega_v \,\big(x_p(v) - x_n(v)\big).$$

where:

- s_v: baseline share (e.g. $s_v \propto N_n(v)Q_v + N_p(v)Q_v$).
- ω_v: weight on premium incentive (e.g. $\omega_v \propto N_p(v)$).

Constraints

$$\begin{cases} x_p(v) \;\geq\; x_n(v) + \delta \cdot g(Q_v), & \forall v \in V \quad \text{premium must reward at least } \delta \text{ (optionally quality-aware)} \\ x_p(v) - x_n(v) \;\leq\; \Delta, & \forall v \in V \quad \text{cap premium gap} \\ 0 \leq x_n(v) \leq x_n^{\max},\;\; 0 \leq x_p(v) \leq x_p^{\max}, & \forall v \in V \quad \text{policy bounds} \\ R \;\geq\; R_{\min}, & \text{global platform revenue floor.} \end{cases}$$

Optional additions:

$$\begin{aligned} \sum_{v\in V} \mathrm{pay}_v \;&\leq\; P_{\max}, && \text{(global payout cap)} \\ \underline{p}_v \;\leq\; \mathrm{pay}_v \;&\leq\; \overline{p}_v, && \forall v \in V \quad \text{(per-video bounds).} \end{aligned}$$

Why This Works

- **Fairness:** The log term ensures diminishing returns, preventing one video from taking all payouts.
- **Quality-weighting:** $f(Q_v)$ and w_v amplify higher-quality or under-served videos.
- **Premium incentive:** The per-video gap $x_p(v) - x_n(v)$ rewards quality with stronger premium uplift.
- **Sustainability:** The revenue floor keeps TikTok's overall revenue viable.