The Economics of Large Language Models: Token Allocation, Fine-Tuning, Optimal Pricing

Dirk Bergemann[1] Alessandro Bonatti[2] Alex Smolin[3]

[1]Yale University and CEPR

[2]MIT Sloan and CEPR

[3]Toulouse School of Economics and CEPR

Motivation: Technological Context

LLMs are a general purpose technology:

- Chatbots, copy-editing, translation;
- Software development;
- Information search and summarization, etc.

$\Rightarrow$ need for abstraction and aggregation.

Motivation: Technological Context

LLMs are a general purpose technology:

- Chatbots, copy-editing, translation;
- Software development;
- Information search and summarization, etc.

$\Rightarrow$ need for abstraction and aggregation.

Economic value on a given task can be improved via:

- **Input tokens**: context and details, RAG (retrieval-augmented generation).
- **Output tokens**: detailed answers.
- **Fine-tuning tokens**: model adjustment (e.g., BloombergGPT).

$\Rightarrow$ need for multidimensionality of token characteristics and uses.

Economic Context

Generative AI market: $71B in 2025, projected $890B by 2032 (43% CAGR).

Global private investment in generative AI: $33.9B in 2024 (up 18.7%).

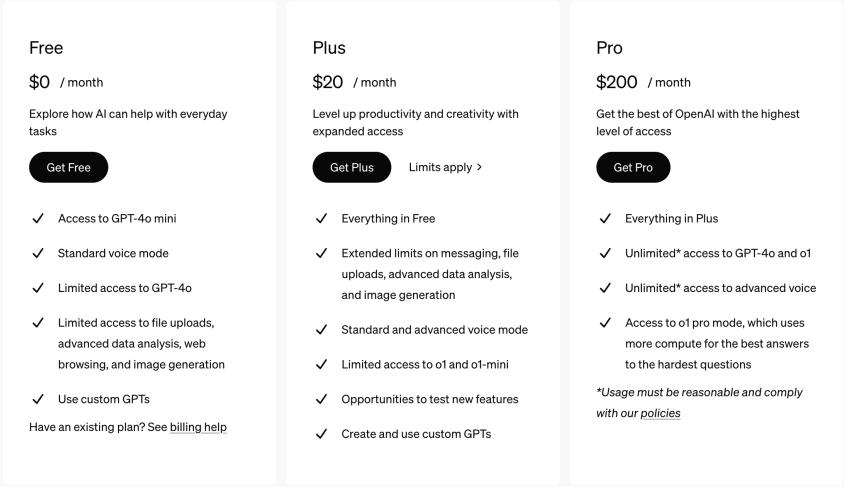
OpenAI: $12B annualized revenue mid-2025; $500B valuation Oct 2025.

Anthropic: $5B ARR July 2025 (up from $1B Dec 2024); $61.5B valuation.

92% of Fortune 500 use LLM products; 78% of organizations use AI (2024).

Pricing via **API** access, enterprise **licenses**, and consumer **subscriptions**.

Motivation: LLM Pricing



Free

$0 / month

Explore how AI can help with everyday tasks

Get Free

- ✓ Access to GPT-4o mini
- ✓ Standard voice mode
- ✓ Limited access to GPT-4o
- ✓ Limited access to file uploads, advanced data analysis, web browsing, and image generation
- ✓ Use custom GPTs

Have an existing plan? See billing help

Plus

$20 / month

Level up productivity and creativity with expanded access

Get Plus Limits apply ›

- ✓ Everything in Free
- ✓ Extended limits on messaging, file uploads, advanced data analysis, and image generation
- ✓ Standard and advanced voice mode
- ✓ Limited access to o1 and o1-mini
- ✓ Opportunities to test new features
- ✓ Create and use custom GPTs

Pro

$200 / month

Get the best of OpenAI with the highest level of access

Get Pro

- ✓ Everything in Plus
- ✓ Unlimited* access to GPT-4o and o1
- ✓ Unlimited* access to advanced voice
- ✓ Access to o1 pro mode, which uses more compute for the best answers to the hardest questions

Usage must be reasonable and comply with our policies

ChatGPT subscription pricing differs by precision, customization, and request limits.

Motivation: LLM Pricing

Model	Pricing
gpt-4o-2024-08-06	$3.750 / 1M input tokens $1.875 / 1M cached** input tokens $15.000 / 1M output tokens $25.000 / 1M training tokens
gpt-4o-2024-08-06	$2.50 / 1M input tokens $1.25 / 1M cached** input tokens $10.00 / 1M output tokens

OpenAI API pricing of input, output, and fine-tuning tokens.
Bottom: Pricing of a baseline model. Top: Pricing of a fine-tuned model.

Anthropic API Pricing Structure (October 2025)

On-Demand pricing

Region: US East (N. Virginia) and US West (Oregon)

Anthropic models	Price per 1,000 input tokens	Price per 1,000 output tokens
Claude 3.5 Sonnet	$0.003	$0.015
Claude 3 Opus*	$0.015	$0.075
Claude 3 Haiku	$0.00025	$0.00125
Claude 3 Sonnet	$0.003	$0.015
Claude 2.1	$0.008	$0.024
Claude 2.0	$0.008	$0.024
Claude Instant	$0.0008	$0.0024

**Claude 3 Opus is currently available in the US West (Oregon) Region*

Anthropic's tiered pricing across Claude 4.1, Sonnet 4.5, and Haiku 3.5 models.
Differentiation in input/output token pricing and prompt caching costs.
Sonnet 4.5 features context-dependent pricing for prompts above/below 200K tokens.

LLM Revenue Mix (latest public figures, mid-2025)

Provider (model line)	Consumers	API / Enterprise	Source & date
OpenAI (ChatGPT / GPT-4o)	$\approx 75\,\%$	$\approx 25\,\%$	CFO Sarah Friar interview, 28 Oct 2024
Anthropic (Claude 3)	$10 - 15\,\%$	$70 - 75\,\%$	Sacra market report, May 2025
Cohere (Command)	$\approx 15\,\%$	$\approx 85\,\%$	Sacra market report, May 2025
Google Gemini	immaterial	$\approx 100\,\%$	WSJ Pichai interview, Jan 16, 2025
Microsoft Copilot stack	small	majority	MSFT Q2 FY 2025 earnings call

$\star$ Older data on Perplexity suggest a more even revenue split than other providers

Our Framework

Monopoly pricing of LLMs.

Multidimensional screening problem with the following features:

- Buyers solve a variety of tasks.
- A variety of inputs are combined to create economic value across tasks.
- Some inputs are task specific, while others improve value on all tasks.
- Buyers differ in the weights attached to different tasks.

A seller can design a menu of items that specify various inputs and prices.

Related Literature

Multidimensional screening and nonlinear pricing: Rochet and Choné (1998), Armstrong (1996), Manelli and Vincent (2007), Daskalakis et al. (2017), Haghpanah and Siegel (2025).

Mechanism design with production complementarities: Castro and Jiménez (2024), Fiat et al. (2016), Devanur et al. (2020) ("1.5-dimensional").

Selling information and data: Babaioff et al. (2012), Bergemann, Bonatti, and Smolin (2018), Yang (2022).

Emerging work on AI pricing: Mahmood et al. (2024), Fish et al. (2024), Duetting et al. (2024), Demirer and Fradkin (2025).

Tasks and Technology (example)

- A continuum of tasks, $i \in [0, 1]$.
- To execute these tasks, the buyer uses:

  J classes of per-task tokens $x_i = (x_{i1}, \ldots, x_{iJ})$;

  K classes of fine-tuning tokens $z = (z_1, \ldots, z_K)$.
- Precision on each task is given by a CES "gain function:"

$$g(x_i, z) = \Psi(x_i)\,\Phi(z) = \left(\sum\nolimits_{j=1}^{J} \alpha_j x_{ij}^{\rho}\right)^{\sigma/\rho} \Phi(z),$$

  with Φ strictly concave, $0 < \sigma < 1$, $\rho \leq 1$, $\alpha_j > 0$, and $\sum_{j=1}^{J} \alpha_j = 1$.
- Tractable functional form and in line with observed "scaling laws."

Cobb-Douglas gain function:

$$g(x_{i1}, x_{i2}, z) = x_{i1}^{\alpha} x_{i2}^{\beta} (b + z)^{\gamma},$$

where α, β, γ are sensitivity parameters, $\alpha + \beta + \gamma \leq 1$, and $b > 0$ is a baseline productivity parameter (so that fine tuning is not necessary for production).

Buyer and Seller Payoff

- Buyer type, a collection of weights $\boldsymbol{w} = (w_i)_{i=0}^1$.
- Buyer $\boldsymbol{w}$'s payoff from tokens $((x_i)_{i=0}^1, z)$ and payment t:

$$\int_0^1 w_i \, g(x_i, z) di - t.$$

- Marginal costs of task-specific and fine-tuning tokens: $c_j, \hat{c}_k > 0$.
- Seller payoff from $((x_i)_{i=0}^1, z)$ and t:

$$t - \sum_{j=1}^J c_j \int_0^1 x_{ij} \, di - \sum_{k=1}^K \hat{c}_k z_k.$$

- Buyer type is distributed according to $F_{\boldsymbol{w}}$.
- Seller offers a direct menu of items and transfers:

$$\mathcal{M} = (I(\boldsymbol{w}), t(\boldsymbol{w}))_{\boldsymbol{w}}.$$

- Items are part of the design. In the paper, two regimes:
  1. Token budgets: $I = (X, Z) \in \mathbb{R}_+^J \times \mathbb{R}_+^K$; buyer allocates across tasks.
  2. Token distributions: $I = ((x_i)_{i=0}^1, z)$, full contracting (e.g., task caps).
- We characterize profit-maximizing menus of each kind.
- Today, focus on token budgets (1.).

Efficient Allocation

- In general, the entire profile of valuations $\boldsymbol{w} = (w_i)_{i=0}^1$ determines the optimal level of fine-tuning and the resulting surplus.
- However, the CES gain function simplifies the analysis drastically.
- Define the aggregate type:

$$\theta = \left(\int_0^1 w_i^{\frac{1}{1-\sigma}} di \right)^{1-\sigma}.$$

Proposition 1 (Efficient Allocation)

In the socially efficient allocation, all types with the same aggregate type θ consume the same total number of task-specific tokens, of fine-tuning tokens, and generate the same surplus.

Proposition 1 (Efficient Allocation)

In the socially efficient allocation, all types with the same aggregate type θ consume the same total number of task-specific tokens, of fine-tuning tokens, and generate the same surplus.

- Total number of tokens and surplus depend only on the aggregate type θ.
- The number of task-specific tokens x_i is proportional to $w_i^{\frac{1}{1-\sigma}}$.
- Fine-tuning parameters (i.e., the function $\Phi(z)$) do not affect the equivalence classes, i.e., the aggregate type θ.

Menus of Token Budgets

Token Budgets

- Assume that the seller can contract only on the *total* number of tokens:

$$\mathcal{M} = (X(\boldsymbol{w}), Z(\boldsymbol{w}), t(\boldsymbol{w}))_{\boldsymbol{w}}.$$

- $X \in \mathbb{R}_+^J$ are token budgets that buyer can freely allocate across tasks.
- A mechanism design problem with an infinitely-dimensional type, a multidimensional allocation, and moral hazard.

- Assume that the seller can contract only on the *total* number of tokens:

$$\mathcal{M} = (X(\boldsymbol{w}), Z(\boldsymbol{w}), t(\boldsymbol{w}))_{\boldsymbol{w}}.$$

- $X \in \mathbb{R}_+^J$ are token budgets that buyer can freely allocate across tasks.
- A mechanism design problem with an infinitely-dimensional type, a multidimensional allocation, and moral hazard.
- Again, the CES-style gain function simplifies the analysis significantly.

Buyer-Optimal Allocation

A buyer with type $\boldsymbol{w} = (w_i)_{i=0}^1$ and budgets $(X, Z) \geq 0$ solves:

$$U(X, Z, \boldsymbol{w}) = \max_{(x_i)_{i\in[0,1]}} \Phi(Z) \int_0^1 w_i \underbrace{\left(\sum\nolimits_{j=1}^J \alpha_j x_{ij}^\rho\right)^{\sigma/\rho}}_{=\Psi(x_i)} di,$$

$$\text{s.t.} \int_0^1 x_{ij} di = X_j, \quad \forall j = 1, ..., J.$$

Buyer-Optimal Allocation

A buyer with type $\boldsymbol{w} = (w_i)_{i=0}^1$ and budgets $(X, Z) \geq 0$ solves:

$$U(X, Z, \boldsymbol{w}) = \max_{(x_i)_{i\in[0,1]}} \Phi(Z) \int_0^1 w_i \underbrace{\left(\sum\nolimits_{j=1}^J \alpha_j x_{ij}^\rho\right)^{\sigma/\rho}}_{=\Psi(x_i)} di,$$

$$\text{s.t.} \int_0^1 x_{ij} di = X_j, \quad \forall j = 1, ..., J.$$

Lemma 1 (Buyer-Optimal Payoff)

For any $\boldsymbol{w}$ *and* $(X, Z) \geq 0$, $U(X, Z, \boldsymbol{w}) = \theta(\boldsymbol{w})\Psi(X)\Phi(Z) \triangleq \theta\, Q$.

- Buyer's payoff depends on $\theta(\boldsymbol{w}) := (\int_0^1 w_i^{\frac{1}{1-\sigma}} di)^{1-\sigma}$ and aggregate quality Q.
- Reduces to a single-dimensional problem with type θ and allocation Q.

Cost Minimization

- The minimal costs of delivering aggregate quality Q is:

$$C(Q) \triangleq \min_{X,Z\geq 0} \sum_{j=1}^{J} c_j X_j + \sum_{k=1}^{K} \hat{c}_k Z_k$$
$$\text{s.t. } \Psi(X)\Phi(Z) = Q.$$

- $C(Q)$ is strictly increasing and strictly convex with $C'(0) = 0$
  $\Rightarrow$ the analysis of Mussa and Rosen (1978) applies.

- Denote the distribution of θ by F_θ. The relevant virtual (aggregate) type:

$$h(\theta) \triangleq \theta - \frac{1 - F_\theta(\theta)}{f_\theta(\theta)}.$$

- Assume that $h(\theta)$ is increasing.

Proposition 2 (Optimal Menu of Token Budgets)

The optimal menu of token budgets is given by

$$(X(\theta), Z(\theta), t(\theta))_\theta,$$

where $(X(\theta), Z(\theta))$ *are cost-minimizing tokens that deliver quality* $Q(\theta)$*, which satisfies* $h(\theta) = C'(Q(\theta))$*, and* $t(\theta) = \theta\, Q(\theta) - \int_0^\theta Q(s)ds$.

Proposition 2 (Optimal Menu of Token Budgets)

The optimal menu of token budgets is given by

$$(X(\theta), Z(\theta), t(\theta))_\theta,$$

where $(X(\theta), Z(\theta))$ *are cost-minimizing tokens that deliver quality* $Q(\theta)$*, which satisfies* $h(\theta) = C'(Q(\theta))$*, and* $t(\theta) = \theta\, Q(\theta) - \int_0^\theta Q(s)ds$.

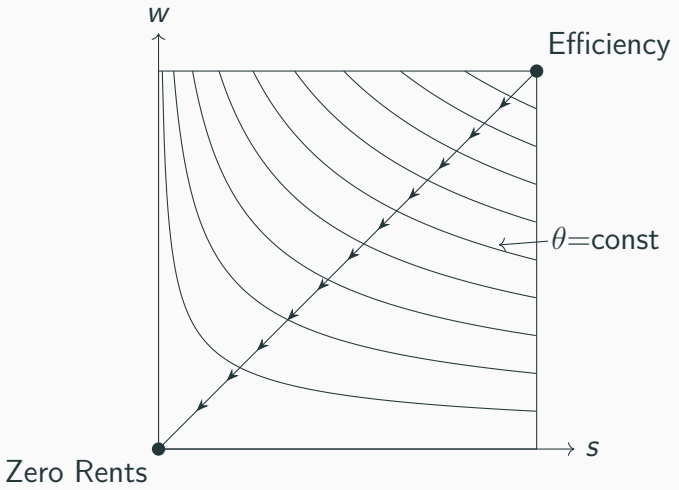
- All types $\boldsymbol{w} = (w_i)_{i=0}^1$ corresponding to the same θ pick the same item.
- Downward distortions in aggregate quality.
- Constrained efficient token allocation and production (cf. Atkinson-Stiglitz, Diamond-Mirlees, and incentive separability in Doligalski et al., 2023).

- We focus on a special case, $\boldsymbol{w} \sim (w, s)$:

$$w_i = \begin{cases} w, & \text{if } i \leq s, \\ 0, & \text{if } i > s, \end{cases}$$

- Tasks are homogeneous, but buyers differ in scale and (per-task) value.
- E.g., powering a customer support chatbot: scale and value per customer.

Value-Scale Heterogeneity: Binding ICs



- "Indifference curve" = equivalence class with the same aggregate type θ.

Global Optimality

Assumption 1 (Distribution Separability)

There exist f_1 and f_2 such that for all $\boldsymbol{w}$, the density $f(\boldsymbol{w}) = f_1(\theta(\boldsymbol{w}))f_2(\boldsymbol{w})$, and f_2 is homogeneous of degree zero.

(Knowing $\theta(\boldsymbol{w})$ provides no information about relative value of different tasks.)

Proposition 3 (Optimality of Token Budgets)

Under Assumption 1, an optimal mechanism is a menu of token budgets.

(1) The optimal menu of token budgets is implementable via a cost-based tariff (i.e., a menu $(B_k, T_k)_k$, such that the buyer pays T_k for access to a budget B_k that she can use to buy tokens at their marginal costs).
(2) Extend Armstrong (1996)'s proof to multiple goods per task + fine tuning.

Summary So Far

The talk:

- Developed an economic framework for pricing of LLMs.
- Characterized optimal monopolistic menu design.
- Analyzed the use of token budgets as a screening device.

The paper:

- Solves optimal menu of token distributions in two special cases.
- Establishes conditions for two-part-tariff implementation (both settings), i.e., quantity discounts for tokens as in Maskin and Riley (1984).

Multiple Models

Competing Models

LLM pricing with multiple, competing models.

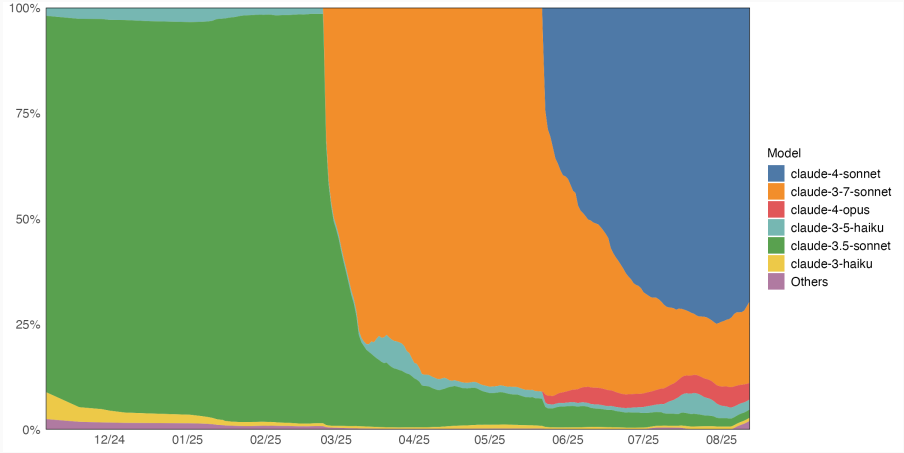
Contrast three cases:

1. Efficient allocation.
2. Multimodel monopolist.
3. Leader vs. competitive fringe.

Focus on (3.) to capture proprietary vs. open source.

Anthropic Claude Model Market Shares

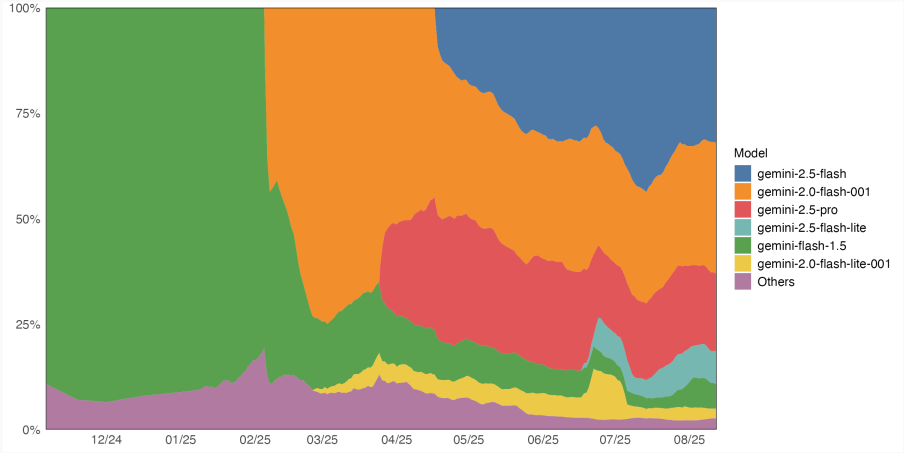
(Demirer and Fradkin, 2025)



Market share of Claude models over time (Nov 2024 - Aug 2025).
Transition from Claude 3.5 Sonnet dominance to Claude 4 Sonnet by mid-2025.
Claude 3.7 Sonnet serving as an intermediate release.

Google Gemini Model Market Shares

(Demirer and Fradkin, 2025)



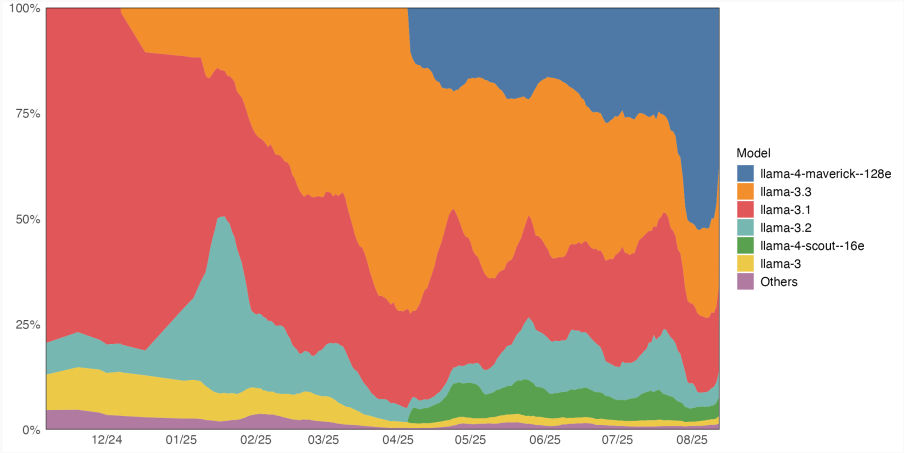
Market share of Gemini models over time (Nov 2024 - Aug 2025).

Gemini Flash 1.5 gradually replaced by Gemini 2.0 Flash and Gemini 2.5 Flash variants.

Gemini 2.5 Pro gaining significant share in later months.

Meta Llama Model Market Shares

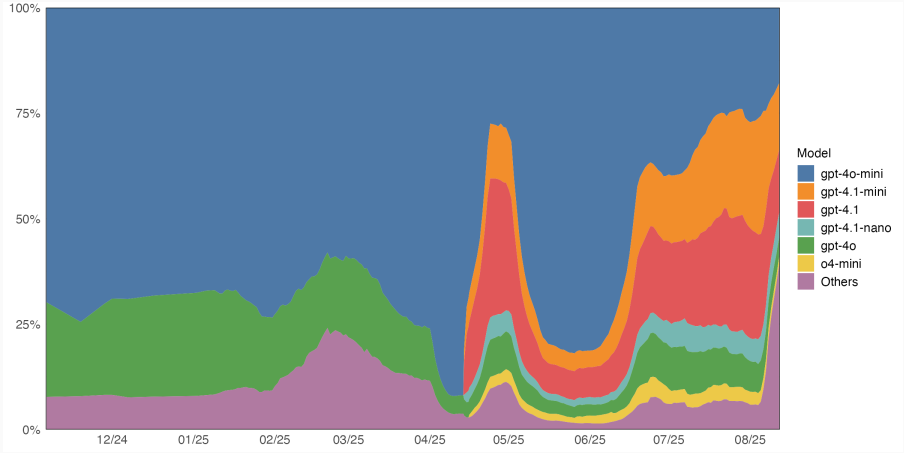
(Demirer and Fradkin, 2025)



Market share of Llama models over time (Nov 2024 - Aug 2025).
Llama 3.1 and 3.3 dominate early, Llama 4 Maverick gains from mid-2025.

OpenAI GPT Model Market Shares

(Demirer and Fradkin, 2025)

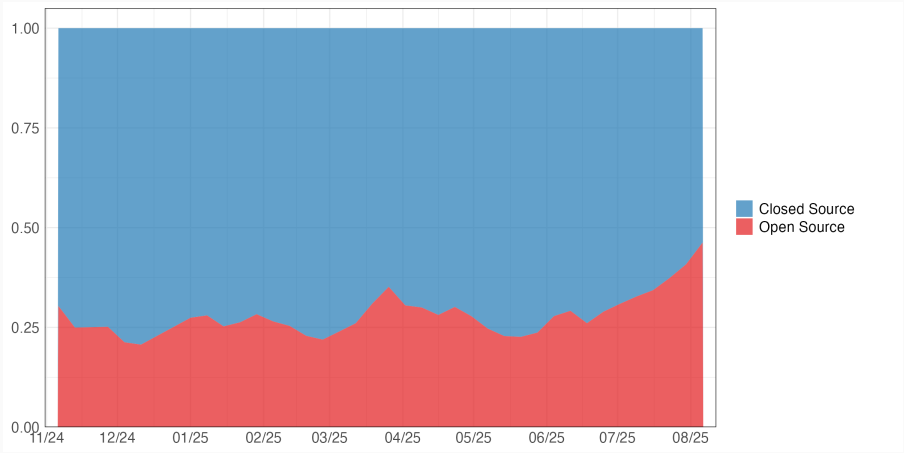


Market share of OpenAI models over time (Nov 2024 - Aug 2025).

GPT-4o Mini dominates throughout.

Significant disruption in April-May 2025 from GPT-4.1 releases.

Open vs. Closed Source Market Shares (Demirer and Fradkin, 2025)



Proportion of open source vs. closed source LLM usage (Nov 2024 - Aug 2025).

Open source models maintain 20-30% market share initially.

Open source grows to nearly 50% by August 2025.

Technology: Two-Model Setup

Consider two models $\ell = 1, 2$, with model-specific gain function:

$$g_\ell(x_{i1}, \ldots, x_{iJ}, z_1, \ldots, z_K) = \left(\sum_{j=1}^{J} \alpha_{\ell j} x_{ij}^{\rho_\ell}\right)^{\sigma/\rho_\ell} \left(\sum_{k=1}^{K} \hat{\alpha}_{\ell k} z_k^{\hat{\rho}_\ell}\right)^{\gamma_\ell/\hat{\rho}_\ell}.$$

Common returns to per-task intensity: σ.

Model-specific returns to fine-tuning: γ_ℓ.

Free variation in $\alpha_{\ell j}, \hat{\alpha}_{\ell k}, \rho_\ell, \hat{\rho}_\ell$, and costs $(c_{\ell 1}, \ldots, c_{\ell J}, \hat{c}_{\ell 1}, \ldots, \hat{c}_{\ell K})$.

Buyer can use both models, but only one model per task $i \in [0, 1]$.

If buyer $\boldsymbol{w}$ uses model ℓ for tasks $i \in I_\ell$, total payoff:

$$\sum_{\ell=1}^{L} \int_{i \in I_\ell} w_i \, g_\ell(x_{\ell i}, z_\ell) \, di.$$

Indirect Utility and Cost Functions

Consider token budgets $(B_\ell)_{\ell=1,2}$, where $B_\ell = (X_{\ell 1}, \ldots, X_{\ell J}, Z_{\ell 1}, \ldots, Z_{\ell K})$.

Define aggregate quality as

$$Q_\ell = g_\ell(X_{\ell 1}, \ldots, X_{\ell J}, Z_{\ell 1}, \ldots, Z_{\ell K}).$$

Agent's optimization of models and tokens across tasks $\Rightarrow$ indirect utility function

$$U(\theta, Q_1, Q_2) = \theta\Big(Q_1^{1/\sigma} + Q_2^{1/\sigma}\Big)^{\sigma},$$

where the aggregate type is (as before)

$$\theta = \Big(\int_0^1 w_i^{1/(1-\sigma)} di\Big)^{1-\sigma}.$$

Cost of providing aggregate quality Q through model ℓ:

$$C_\ell(Q) = c_\ell Q^{\frac{1}{\sigma+\gamma_\ell}}.$$

Leader vs. Fringe

A leader vs. a competitive fringe that prices tokens at marginal cost.

Leader's model has aggregate cost parameter $c_L > 0$, returns to intensity $\sigma > 0$, and returns to fine-tuning $\gamma_L > 0$, such that $\sigma + \gamma_L < 1$.

Leader cost function:

$$C_L(Q) = c_L Q^{\frac{1}{\sigma+\gamma_L}}.$$

Fringe's model has aggregate cost parameter $c_F \in [0, c_L)$, returns to intensity $\sigma > 0$, and returns to fine-tuning $\gamma_F \in [0, \gamma_L)$.

Fringe cost function:

$$C_F(Q) = c_F Q^{\frac{1}{\sigma+\gamma_F}}.$$

Buyer's Problem

The buyer has a private aggregate type θ.

θ is distributed according to F with $f > 0$ everywhere on $[0, 1]$.

Define quality variables $q_L \triangleq Q^{\frac{1}{\sigma+\gamma_L}}$ and $q_F \triangleq Q_F^{\frac{1}{\sigma+\gamma_F}}$.

The buyer's utility (when buying from the leader) can be written as

$$u(\theta, q_L) = \max_{q_F \geq 0} \theta(q_L^{(\sigma+\gamma_L)/\sigma} + q_F^{(\sigma+\gamma_F)/\sigma})^\sigma - c_F q_F, \tag{1}$$

with an outside option $u_0(\theta) = u(\theta, 0)$.

Leader's Problem

Define the limit quality as

$$\hat{q}(\theta) \triangleq \left(\frac{\theta\sigma}{c_F}\right)^{\sigma/((1-\sigma)(\sigma+\gamma_L))}. \tag{2}$$

If purchased leader quantity $q_L < \hat{q}(\theta)$, buyer buys fringe tokens to achieve the optimal total budget $\hat{q}(\theta)$; if $q_L \geq \hat{q}(\theta)$, buyer is exclusive to the leader.

With $h(\theta)$ the virtual value, define leader-optimal interior quantity as

$$q_L^{int}(\theta) = \left(\frac{(\sigma+\gamma_L)h(\theta)}{c_L}\right)^{1/(1-\sigma-\gamma_L)}. \tag{3}$$

Main Result

Proposition 4 (Leader-Fringe Equilibrium)

Let $\gamma_F = 0$ and assume F satisfies the monotone hazard rate condition.

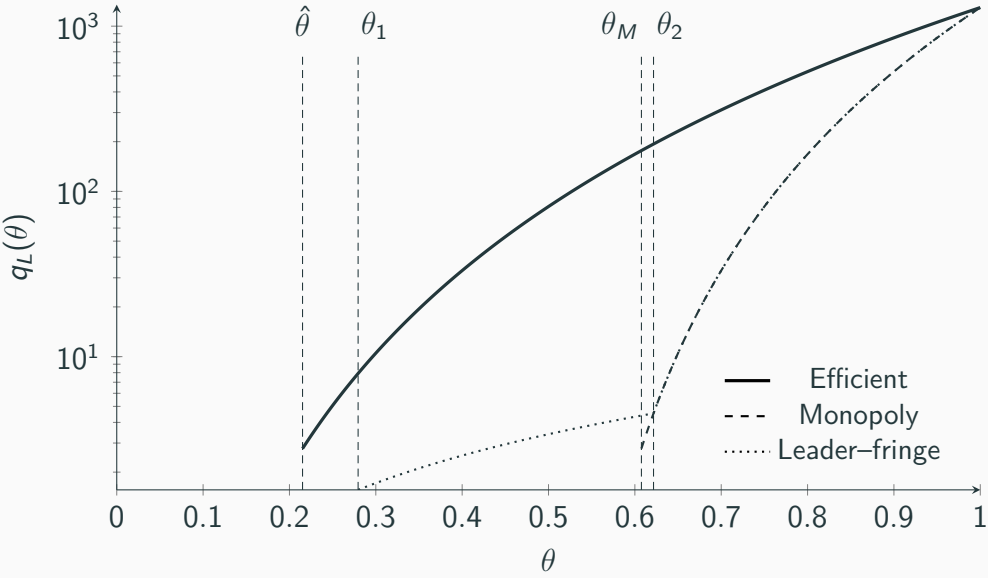
There exist $\underline{\theta} \leq \overline{\theta}$, such that:

1. *For $\theta \leq \underline{\theta}$, $q_L(\theta) = 0$ and $q_F(\theta) > 0$.*
2. *For $\theta \in (\underline{\theta}, \overline{\theta})$, $q_L(\theta) = \hat{q}(\theta)$ and $q_F(\theta) = 0$.*
3. *For $\theta \geq \overline{\theta}$, $q_L(\theta) = q^{int}(\theta)$ and $q_F(\theta) = 0$.*

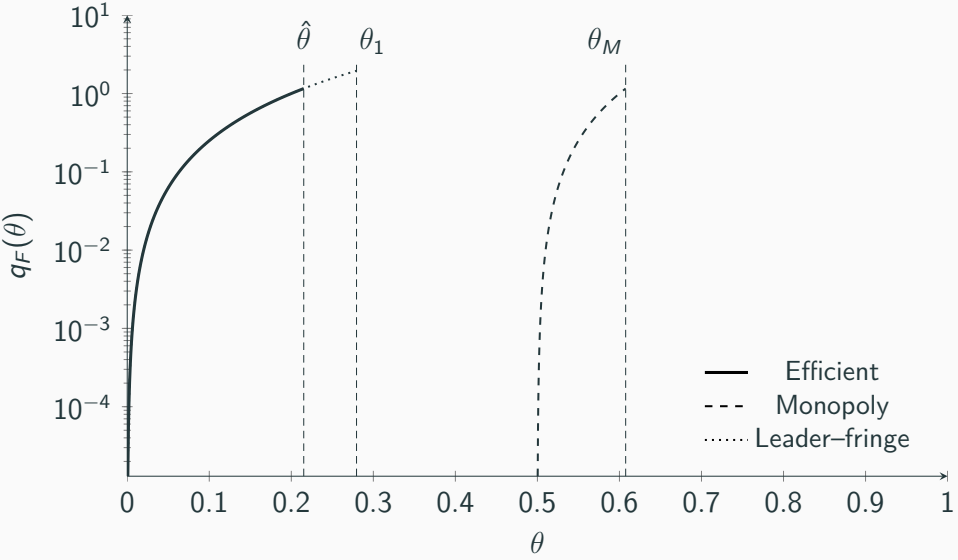
(Depending on parameters, any region may be empty.)

Next: compare leader-fringe with socially efficient allocation and with multi-model monopolist (uniform distribution example).

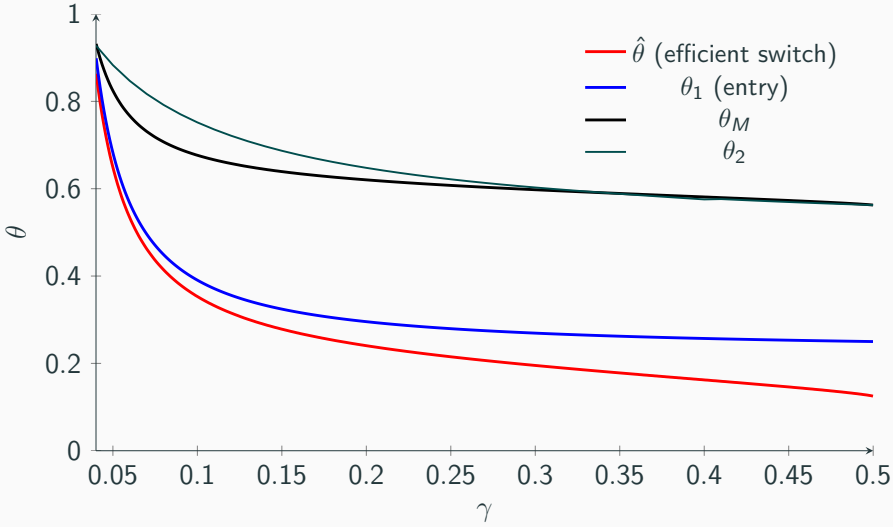
Leader Quantity



Fringe Quantity



Thresholds


1
0.8
0.6
0.4
0.2
0
θ
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45
0.5
γ
$\hat{\theta}$ (efficient switch)
θ_1 (entry)
θ_M
θ_2

Future Directions

- Competition of differentiated LLMs.
- Investment incentives.
- Regulation and antitrust.
- Bundling and integration.

Thank You

Menus of Token Distributions

Token Distributions

- Assume that $\boldsymbol{w}$ is distributed according to $F_{\boldsymbol{w}}$.
- The seller can contract on token distribution across tasks:

$$\mathcal{M} = ((x_i(\boldsymbol{w}))_{i\in[0,1]}, z(\boldsymbol{w}), t(\boldsymbol{w}))_{\boldsymbol{w}}.$$

- E.g., limits on # of requests, task-specific model variations or servers.
- Mechanism design with infinitely-dimensional types and allocations.

Token Distributions

- Assume that $\boldsymbol{w}$ is distributed according to $F_{\boldsymbol{w}}$.
- The seller can contract on token distribution across tasks:

$$\mathcal{M} = \left((x_i(\boldsymbol{w}))_{i\in[0,1]}, z(\boldsymbol{w}), t(\boldsymbol{w})\right)_{\boldsymbol{w}}.$$

- E.g., limits on # of requests, task-specific model variations or servers.
- Mechanism design with infinitely-dimensional types and allocations.

We make progress under **two type structures**:

- Two types.
- Value-scale heterogeneity.

- With many arbitrary types $\boldsymbol{w}$ the binding incentive structure is unclear.
- We focus on a special case, $\boldsymbol{w} \sim (w, s)$:

$$w_i = \begin{cases} w, & \text{if } i \leq s, \\ 0, & \text{if } i > s, \end{cases}$$

  in which w and s are independently distributed according to F_w and F_s.
- Tasks are homogeneous, but buyers differ in scale and (per-task) value.
- E.g., powering a customer support chatbot: scale and value per customer.