Measuring What the Crawler Sees: Discovery Curves, Core Persistence, and Shell Dynamics in Longitudinal Web Crawls

Michael Paris[0000−0003−2077−6984], Hande Çelikkanat[0000−0003−2858−5867], and Luca Foppiano[0000−0002−6114−6164]

Common Crawl Foundation, Beverly Hills, USA
{micha,hande,luca}@commoncrawl.org

Abstract. A longitudinal web crawl is a sequence of partial samples of an evolving URL population. Pairwise containment between two crawls is the standard probe; under a simple *urn* model of the crawl — each round samples a fraction of the URLs and replaces a fraction — it recovers two interpretable rates, per-round survival α and coverage c, but treats the population as uniform and consumes one pair at a time. In this work, we define a formal language for talking about a crawl. We extend this analysis with the *discovery curve* $U(s, T)$, the cumulative URL footprint over a sliding window of T crawls starting at s, which under the same urn model is also a closed-form function of (α, c). Containment and the discovery curve are then two projections of one process: independent fits agree on (α, c) when the urn is homogeneous, so any disagreement is itself a measurement. Applied to Common Crawl (2020–2025, domain granularity) and to the German Academic Web (GAW, URL granularity), the two projections disagree on both archives, and a two-component urn with a persistent core fraction κ alongside shell parameters $(\alpha_\partial, c_\partial)$ reconciles the disagreement. A residual on c_∂ remains, signaling that the shell itself is not homogeneous; κ is recorded as the scalar entry point to a rank-resolved generalization, which is left to follow-up work.

Keywords: web archive · crawl coverage · discovery curve · urn model · two-component model · URL lifetime

1 Introduction

Common Crawl — monthly snapshots of the open web, 2–3 billion URLs per crawl, over 90 archives since 2013 — is the foundational corpus for most open and commercial pretraining datasets [15,25], including C4 [27], The Pile [10], RefinedWeb [25], and RedPajama [34], and underlies the web portions of GPT, Llama, Falcon, and PaLM. The quality of every downstream corpus is bounded by which URLs Common Crawl chose to fetch and how persistently it tracked them. Those choices are driven by a small set of static operator knobs — host budgets, harmonic-centrality (hc-rank) thresholds, deletion cutoffs — applied to a 200-billion-edge webgraph that is itself constantly evolving. Their downstream

effect on coverage, persistence, and discovery cost is not directly read off the data.

The per-crawl URL counts and rolling-window unique-URL statistics that Common Crawl already publishes are sufficient observables to recover the parameters of an urn model of the crawl, at the host and domain granularity, where the operator's policy levers act, and without external ground truth. This paper derives the closed-form mapping from those public observables to the urn parameters and applies it to Common Crawl at domain granularity and to the German Academic Web (GAW) at URL granularity, two longitudinal archives produced by crawl architectures with opposing design choices.

Prior work [23] established the pairwise containment $g(\Delta t) = c\,\alpha^{\Delta t}$ as a closed-form recovery of (α, c) under the homogeneous urn. This estimator uses one crawl pair at a time and offers no diagnostic when the homogeneous assumption fails. The discovery curve $U(s, T)$, defined as the cumulative number of distinct URLs observed over T consecutive crawls starting at s, uses the full sequence, and under the same urn model is also a closed-form function of (α, c). Containment and the discovery curve are therefore two projections of one process: independent fits agree on (α, c) when the urn is homogeneous, and the magnitude of any disagreement is itself a measurement. The pairwise fit is recurrence-weighted and reads the persistent *core*; the discovery curve is fresh-discovery-weighted and reads the ephemeral *shell*. A two-component urn that introduces a persistent core fraction κ alongside shell parameters $(\alpha_\partial, c_\partial)$ reconciles the cross-source disagreement on both archives.

The two archives differ structurally. GAW is a Heritrix-based focused crawl seeded from approximately 150 German academic homepages, BFS-driven and is re-harvesting from a fixed URL via hops at each round. Common Crawl is Nutch-based and recursive, with each monthly fetch list derived from the previous `CrawlDB` and ranked by harmonic centrality. Both admit the same urn model: the two projections agree within-dataset on a single (κ, α, c) triple, while the fitted values differ across archives, reflecting the difference in the underlying URL populations.

Section 2 reviews related work. Section 3 develops the urn model, the discovery formula, the two-component decomposition, and the operator-language translation. Section 4 presents empirical results on Common Crawl and GAW. Section 5 covers crawl-policy implications, and Section 6 concludes.

2 Related Work

The contribution sits at the intersection of three threads. First, *urn modeling* for coverage — the statistical machinery that descends from capture–recapture and occupancy theory and that we use to turn crawl observables into (α, c, κ). Second, *measurement of Common Crawl*: how the operator publishes what the crawler does, and how the literature has audited that output. Third, the *impact of Common Crawl*: the role it plays as the raw substrate of large-language-

model pretraining, which is what makes its coverage properties matter beyond the crawler community.

2.1 Urn modeling

Treating the web as a finite urn from which a crawler draws repeated samples is the same problem ecologists faced with animal populations. The two-sample base case is Petersen–Lincoln–Chapman [26,20,5]; its extension to a sequence of samples is Schnabel's multi-occasion census [28], and its extension to heterogeneous catchability is Chao's nonparametric lower bound [4]. The underlying urn machinery goes back to Eggenberger and Pólya [9] and is summarised for the occupancy regime in Kolchin–Sevast'yanov–Chistyakov [17]. Cumulative coverage — how much of the population has been seen after n samples — is the Good–Turing question [13] and, in its text-vocabulary form, Heaps' law for vocabulary growth [14]. The same logic was first applied to the web to bound search-engine size and overlap by Bharat and Broder [2] and Lawrence and Giles [18]. Closest to the present setting, Paris [23] shows that pairwise containment between two crawls under a homogeneous urn recovers (α, c). We extend that line: instead of one crawl pair we use the full sequence as a discovery curve $U(s, T)$ from the same urn, and we promote the homogeneous fit to a two-component core/shell model [4,36] when the single-population assumption fails.

2.2 Common Crawl metrics

Common Crawl publishes monthly archives, approximately 2–3 billion URLs each, with operator-side statistics — per-crawl URL counts, host counts, language distribution, and rolling unique-URL footprints over the last N crawls.[1] These descriptive metrics document the crawl as it ships; they do not themselves model what is being sampled. Critical readings have audited what the corpus excludes by construction: Stolz and Hepp [30] on structured-data pitfalls, Baack [1] on the operational filters, and Dodge et al. [8] on demographic and source skew once the data reaches downstream corpora. Closer to longitudinal measurement, Thompson [33] proposes an improved methodology for treating Common Crawl as a longitudinal source — explicitly addressing the cross-crawl comparability that any time-series analysis on top of CC must confront. The longitudinal-measurement literature on web crawls more broadly — per-page change-rate modelling [7,3], freshness-aware refresh policies [6,21], identification of genuinely new pages [35], and URL persistence over multi-year windows [16,12] — frames the crawler itself as the system to model rather than the crawl output. Most recently, Garg et al. [11] sample 27 million URLs across 26 years of the Internet Archive's Wayback Machine expressly to revisit *how long a web page lasts*. Our work is closer in spirit to that thread, but inverted: we read the parameters of the underlying sampling process *from* the already-published crawl observables, without instrumenting the crawler or re-fetching pages — and the per-URL

[1] `https://commoncrawl.org/statistics`

life expectancy ℓ we recover (§3.5) is the urn-model analogue of that empirical longevity question.

2.3 Impact: Common Crawl as LLM substrate

Kaplan et al. [15] showed that pretraining performance scales with corpus size, and Common Crawl — the largest open web source — became the standard substrate of open pretraining pipelines. Direct CC-derived corpora include C4 [27], The Pile [10], OSCAR [22], RefinedWeb [25], RedPajama [34,37], Dolma [29], FineWeb [24], DCLM [19], and most recently Nemotron-CC [31]; each treats Common Crawl as ground truth and competes on filtering — language ID (CommonLID [32] and downstream variants), deduplication, quality classification, perplexity scoring. The web portions of GPT, Llama, Falcon, and PaLM all sit on top of this stack. Filter quality, however, is bounded by fetch coverage: if a URL is never fetched, no downstream filter can recover it. The coverage of the *upstream* fetch policy — which URLs Common Crawl chose, how persistently it tracked them, what fraction of the live web that produces — is the target measurement of the present paper, read from the crawler's own output at the (host, domain) granularity where operator policy levers act.

3 Method

This section turns a crawl's raw URL counts into a small dictionary of interpretable quantities: a per-round *coverage* c (what share of the live population each crawl samples), a *survival* α (what share of the URL set persists to the next crawl), a *core fraction* κ (the share the operator's policy keeps permanently visible), and — recombined from these — a per-URL *life expectancy* ℓ (how many crawls a URL is expected to last). A reader who wants the operational payoff before the derivations can take those four glosses and skip to the results (§4); the rest of this section derives them from public crawl statistics.

3.1 Revisiting the Urn Model

We model a crawl as repeated sampling from a dynamic population. An *urn* contains N unique elements. Each round $t = 0, 1, 2, \ldots$ consists of two steps:

1. **Sampling:** Draw a fraction c of the urn uniformly (the *coverage* per round).
2. **Churning:** Retain a fraction α of the urn; replace the remaining $\bar{\alpha} \equiv 1 - \alpha$ with $N\bar{\alpha}$ brand-new elements never seen before.

We use the bar shorthand $\bar{c} \equiv 1 - c$ for the per-round miss rate and $\bar{\alpha} \equiv 1 - \alpha$ for the per-round churn rate throughout. In every round, each element in the urn meets exactly one of three fates (Fig. 1): it is *replaced* (churned out, probability $\bar{\alpha}$); or it survives and is *sampled* into the crawl (probability αc); or it survives but is *missed* in the sample, persisting unseen into the next round (probability $\alpha\bar{c}$). The three probabilities sum to one.

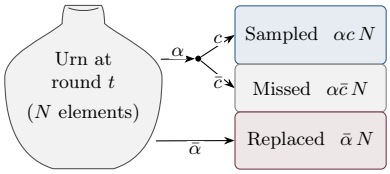


Fig. 1: Per-round area decomposition of the urn. The left box is the urn at round t, of size N; each element falls into one of three outcomes whose rectangle heights are proportional to the joint probabilities. *Sampled* elements ($\alpha c\, N$) appear in the round-t crawl; *missed* elements ($\alpha\bar{c}\, N$) persist unseen and remain eligible for later rounds; *replaced* elements ($\bar{\alpha}\, N$) leave the urn and are overwritten by fresh, never-seen elements. The top two bands together form the survived count αN, and the three areas sum to N.

The persists-unseen rate $\alpha\bar{c} = \alpha(1-c)$ controls the system's memory; its complement $1-\alpha(1-c) = \bar{\alpha}+\alpha c$ is the *resolved rate* — the per-round probability that an element *does not persist unseen*, either churning out or being observed.

3.2 Containment: Pairwise crawl intersections

Previous work [23] showed that pairwise containment – the fraction of one crawl's URLs that reappear in another crawl Δt rounds later – has a closed form under the urn model:

$$g(\Delta t) \;=\; \frac{|u_i \cap u_j|}{|u_i|} \;=\; c \cdot \alpha^{\Delta t}\,, \tag{1}$$

where $|u_i| = M = Nc$ is the per-crawl unique element size. A log-linear regression of g against Δt recovers the two parameters (α, c) from pairwise data alone. This is the starting point; the question is what happens when we move from pairs to *sequences*.

3.3 Discovery Curve: Growth of unique elements

Let $\rho(t)$ denote the fraction of the urn that has been seen in at least one of the rounds $0, \ldots, t-1$, with $\rho(0) = 0$. Two streams of revisits enter round t, both filtered by the per-element survival factor α — prior revisits $\rho(t-1)$ and the prior round's new samples $c\,\nu(t-1)$ that just got marked seen:

$$\rho(t) \;=\; \alpha\big(\rho(t-1) + c\,\nu(t-1)\big) \;=\; \alpha c \;+\; \alpha\bar{c}\,\rho(t-1)\,. \tag{2}$$

This is a linear recurrence with driving term αc and ratio $\alpha\bar{c}$ (the persists-unseen rate). Using the identity $\bar{\alpha} + \alpha c = 1 - \alpha\bar{c}$, its solution is

$$\rho(t) \;=\; \frac{\alpha c\,\big(1 - (\alpha\bar{c})^t\big)}{1 - \alpha\bar{c}}\,. \tag{3}$$

The *discovery curve* is the cumulative sum of distinct elements observed through round t:

$$\hat{U}(t) \;=\; \sum_{\tau=0}^{t} \Delta\hat{U}(\tau)\,, \tag{4}$$

where the per-round new fraction $\nu(t) \equiv \Delta\hat{U}(t)/M \;=\; 1-\rho(t)$ counts new discoveries in round t. Summing the geometric series gives the closed form

$$\boxed{\frac{\hat{U}(t)}{M} \;=\; \frac{(t+1)\,\bar{\alpha}}{\bar{\alpha}+\alpha c} \;+\; \frac{\alpha c\left(1-(\alpha\bar{c})^{t+1}\right)}{(\bar{\alpha}+\alpha c)^2}} \tag{5}$$

This is the *discovery formula*: a constant intercept, a linear trend with slope $\nu_\infty = \bar{\alpha}/(1-\alpha\bar{c})$ (the asymptotic new-discovery rate), and a geometric transient that decays with ratio $\alpha\bar{c}$ and vanishes as $t \to \infty$. The persists-unseen rate $\alpha\bar{c}$ is the only memory parameter that appears: as the transient's ratio, and via its complement $1-\alpha\bar{c}$ in every denominator. The same (α, c) that govern the pairwise containment (1) now predict the entire growth curve of the discovery curve — a substantially richer observable derived from the same two parameters.

Discrete derivative and κ-identifiability. The per-round increment $\Delta\hat{U}(t)/M$ is the discrete derivative of $\hat{U}(t)/M$ and carries no information independent of the discovery curve. Fitting the two-component model against the increment alone collapses to the homogeneous case ($\kappa \to 0$ for $t \geq 1$), because the core contributes no new mass after the first crawl. We use the increment as a consistency check only: any well-resolved κ must be anchored from the containment floor (§3.6), not from the accumulation slope. The cross-source diagnostic in §4 therefore reports fits on (containment, discovery curve) only.

3.4 Asymptotic Regime

For large t, $(\alpha\bar{c})^{t+1} \to 0$ and the progressive union becomes linear:

$$\frac{\hat{U}(t)}{M} \;\approx\; \nu_\infty\, t \;+\; I\,, \tag{6}$$

with slope $\nu_\infty = \bar{\alpha}/(1-\alpha\bar{c})$ and intercept

$$I \;=\; 1 \;+\; \rho_\infty\,\tau_h\,, \qquad \tau_h \;\equiv\; \frac{\alpha\bar{c}}{1-\alpha\bar{c}}\,, \tag{7}$$

where τ_h is the *hiding time per resolution* (the average length of a single hidden epoch). The intercept I is the asymptotic line *extrapolated back* to $t = 0$ — $I = \hat{U}_{\text{asy}}(0)/M$ where $\hat{U}_{\text{asy}}(t) \equiv (\nu_\infty t + I)\,M$. The actual cumulative ratio at $t = 0$ is $\hat{U}(0)/M = 1$ (the first crawl is all-new); the difference $I - 1 = \rho_\infty\,\tau_h$ is the integrated transient — revisit rate times hiding time.

Inverting from the observable pair (ν_∞, τ_h) to model parameters (α, c), with $\rho_\infty = 1 - \nu_\infty$:

$$\alpha = \frac{\rho_\infty + \tau_h}{1 + \tau_h}, \tag{8}$$

$$c = \frac{\rho_\infty}{\rho_\infty + \tau_h}. \tag{9}$$

The discovery curve thus admits *two* independent homogeneous fits from its own data: an analytic regression of the full discovery formula (5), and an asymptotic inversion of the linear tail. When the two agree, the population is well-described by a single (α, c); when they diverge, the transient and the tail are sampling different sub-populations — the first sign that a two-component model is needed.

3.5 Operator language: lifetime, sampled time, hidden time

The urn parameters (α, c) are abstract per-round or per-time unit probabilities. The operator measures crawls. Three derived quantities translate (α, c) into operator-facing units of crawls per URL life:

$$\ell \equiv \frac{\alpha}{\bar{\alpha}}, \tag{10}$$

$$\eta \equiv c\,\ell = \frac{\alpha c}{\bar{\alpha}}, \tag{11}$$

$$\bar{\eta} \equiv \bar{c}\,\ell = \frac{\alpha \bar{c}}{\bar{\alpha}}, \tag{12}$$

respectively the *life expectancy* (expected crawls a URL persists in the urn), *sampled time* (expected crawls a URL is re-fetched over its life), and *hidden time* (expected crawls a URL is alive but unsampled). By construction, the lifetime decomposes additively into time spent sampled and time spent hidden, and the sampling-to-hiding ratio reduces to the per-round coverage odds:

$$\ell = \eta + \bar{\eta}, \tag{13}$$

$$\frac{\eta}{\bar{\eta}} = \frac{c}{\bar{c}}. \tag{14}$$

A complementary timescale, the *hiding time per resolution* τ_h from (7), measures the average length of a single hidden epoch (rounds a hidden URL stays hidden before being sampled or churned out). The two are related by

$$\bar{\eta} = (1 + \eta)\,\tau_h, \tag{15}$$

because a URL's total hidden time equals the per-epoch hiding time times the number of hidden epochs — one before each sample plus a final one ending in churn. τ_h governs the asymptotic intercept (7); $\bar{\eta}$ measures the per-URL hidden cost over the entire life.

This is the bridge between the model and the operator. The web supplies the lifetime ℓ (URL persistence in the operator's urn); the policy levers — host

budgets, hc-score thresholds, recrawl rules, deletion behaviour — control how that lifetime splits between sampled rounds and hidden rounds via c. The cross-source diagnostic loop introduced in §1 returns its readings in (α, c); equations (10)–(14) convert those readings into the language an operator uses to plan a crawl.

3.6 Two-Component Decomposition: Core K and Shell ∂K

When the homogeneous fits from the two projections — (1) on containment and (5) on the discovery curve — disagree on (α, c), the disagreement is the signature of a heterogeneous urn. The minimal extension that closes the gap is a two-component partition: a persistent *core* K of size κN with $(\alpha_K, c_K) = (1, 1)$ — immortal and fully sampled — and an ephemeral *shell* ∂K of size $(1-\kappa)N$ with parameters $(\alpha_\partial, c_\partial)$. The core is graph-driven (the well-linked URLs that any reasonable policy keeps visible); the shell carries the policy-shaped ephemeral mass.

Containment. The core contributes a persistent floor at κ to pairwise containment:

$$\boxed{g(\Delta t) \;=\; \kappa \;+\; (1-\kappa)\, c_\partial\, \alpha_\partial^{\Delta t}} \tag{16}$$

On a log scale this curve is concave; a single-exponential fit cannot capture it, and a joint 3-parameter regression of (16) recovers $(\kappa, \alpha_\partial, c_\partial)$.

Per-crawl core share. Each crawl draws $N\kappa$ core URLs (always sampled) and $N(1-\kappa)c_\partial$ shell URLs, giving a crawl size $M = N\kappa + N(1-\kappa)\, c_\partial$. The fraction of each crawl composed of core mass is

$$\mu \;\equiv\; \frac{N\kappa}{M} \;=\; \frac{\kappa}{\kappa + (1-\kappa)\, c_\partial}\,. \tag{17}$$

Because $c_\partial < 1$ while $c_K = 1$, $\mu > \kappa$ — the core is *over-represented* in any single crawl relative to its urn share.

Discovery curve. The cumulative-unique count splits additively. The core saturates with $N\kappa$ URLs at $t = 0$ and contributes nothing thereafter; the shell follows the homogeneous discovery formula (5) with its own parameters. Letting $B(t;\, \alpha, c)$ denote the right-hand side of (5):

$$\boxed{\frac{\hat{U}_{2c}(t)}{M} \;=\; \mu \;+\; (1-\mu)\, B\big(t;\, \alpha_\partial, c_\partial\big)} \tag{18}$$

The per-round new fraction is $(1-\mu)\, \Delta B(t; \alpha_\partial, c_\partial)$ for $t > 0$ — every newly-discovered URL after the first crawl is shell mass.

Identifiability and the two triples. κ is most cleanly anchored from the containment floor in (16), where the core contributes statically at every Δt. The discovery curve admits a joint 3-parameter regression on (18), but the core plateau and the shell discovery formula trade off along a shallow ridge — so the discovery-curve κ is the looser of the two anchors. Figs. 4 and 5 report the two triples on Common Crawl and the German Academic Web respectively; agreement on κ across the two projections is the diagnostic, and the residual disagreement on c_∂ at short timescales is read as evidence of a structurally non-uniform shell that a single scalar cannot capture.

Operator language at the shell. The lifetime decomposition of §3.5 applies componentwise. The core has $\ell_K \to \infty$ and is fully resolved each round ($\eta_K = \ell_K$, $\bar{\eta}_K = 0$); the shell carries finite $\ell_\partial = \alpha_\partial/(1-\alpha_\partial)$, $\eta_\partial = c_\partial\,\ell_\partial$, $\bar{\eta}_\partial = (1-c_\partial)\,\ell_\partial$. Policy levers act primarily on the shell triple while leaving κ relatively stable.

4 Results

4.1 Common Crawl: The Discovery-Curve Object

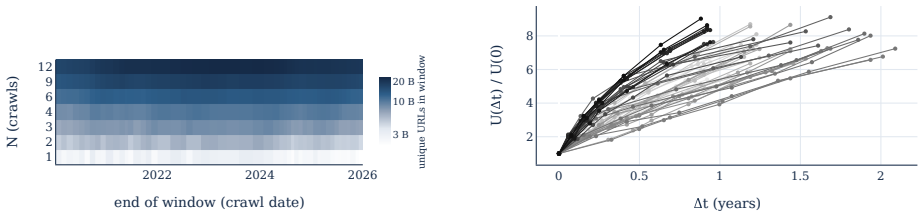
Common Crawl publishes a per-crawl URL count and a family of rolling-window unique-URL counts `url_last_N` for $N \in \{1, 2, 3, 4, 6, 9, 12\}$ on the operator's crawl-statistics page.[2] Pivoted across all valid (start, window-length) pairs in the 2020–2025 monthly archive, these counts trace the discovery curve $U(s, T)$ as a function of the starting crawl s and window length T (Fig. 2a). Two features come straight off the figure. First, U grows monotonically in T but with a clearly decelerating slope — the hallmark of a finite urn with revisits. Second, dividing each curve by its single-crawl cardinality collapses the family across starts (Fig. 2b), confirming that on this window the per-start trajectories are well-described by a common normalised shape. That common shape is the regression target for the cross-source fits below.

The normalised discovery curve is a single projection of the underlying sampling process; pairwise containment $g(\Delta t)$ is the second. Under a homogeneous urn both are governed by the same (α, c), so fitting each independently and overlaying both fits on both observables turns cross-source agreement into a falsifiability test — where the two projections disagree, the homogeneous urn is the wrong model, and the size and shape of the gap dictate the minimal extension. The remainder of this section runs that test at *domain* granularity, the unit at which CC's host-budget and harmonic-centrality levers actually act, and the result shapes §4.3 (closed-archive validation).

4.2 Common Crawl: Cross-Source Fits at Domain Granularity

Both projections are reduced to the same 2020–2025 monthly window used in Fig. 2 and aggregated to the domain population. Each figure below is a single split-axis panel: the discovery curve fills the upper half above the $y = 1$

[2] `https://commoncrawl.github.io/cc-crawl-statistics/plots/crawlsize`



(a) Heatmap $U(s, T)$ (lag T vs. crawl date)

(b) Normalised $U(\Delta t)/U(0)$ vs. window length

Fig. 2: Discovery curve on Common Crawl, 2020–2025. **(a)** Heatmap $U(s, T)$ over the monthly archive: lag T on the y-axis, starting crawl date s on the x-axis, cell colour = unique URLs in the rolling window $[s, s + T]$. The public `url_last_N` statistic is the row at fixed T. **(b)** Normalised $U(\Delta t)/U(0)$ against the actual elapsed time Δt in years (rather than the crawl count N, since CC cadence ranges from biweekly to occasional 1–2 month gaps in 2021–2022, so a $N = 12$ window can span anywhere from 0.9 to 2.1 years), one line per ending crawl (grey = older, black = newer): dividing by the single-crawl cardinality collapses the cross-start differences and isolates the discovery-formula shape under the urn model — the regression target for Figs. 3–4. The 2020–2025 window is the same interval used in every subsequent CC panel.

line, pairwise containment the lower half below it. On each, both fit sources are overlaid — burgundy fit on containment, navy fit on the discovery curve — so the cross-source gap is read off directly: agreement looks like the two colours coinciding in both halves, disagreement is the gap between them.

The two homogeneous fits diverge visibly in both halves of Fig. 3: containment pulls α high and c moderate (it is dominated by the persistent recurring mass), the discovery curve pulls them lower (it is dominated by churn-driven fresh discoveries). Neither single (α, c) explains both projections at once. The minimal extension that closes this gap is the two-component urn of §3.6: a persistent core of fraction κ (immortal, fully sampled) plus a shell with its own $(\alpha_\partial, c_\partial)$. Refitting both projections under this model gives Fig. 4.

The two-component fits on Fig. 4 land on a common $\kappa^{(d)} \approx 0.4$ from both projections — the mass split that the homogeneous urn could not see. The shell parameters $(\alpha_\partial^{(d)}, c_\partial^{(d)})$ also tighten substantially, though a residual containment-vs-discovery gap on $c_\partial^{(d)}$ remains; that gap is the empirical entry point to a rank-resolved treatment of the shell, which we return to briefly in §5. Before doing so we ask whether the same two-projection procedure — and the same structural prediction (cross-source convergence on a single (κ, α, c) triple) — transfers to a crawler built on opposing design philosophies.

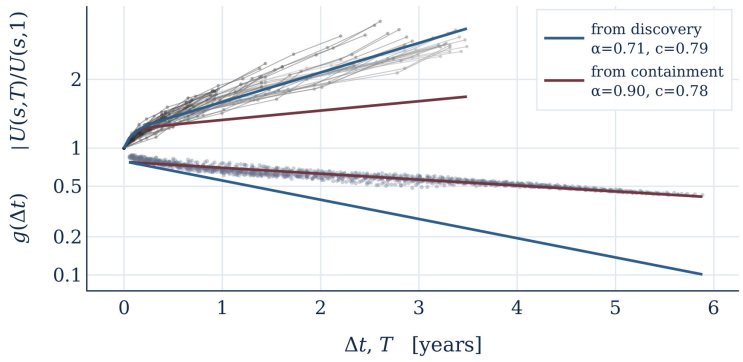


Fig. 3: **Common Crawl — homogeneous cross-source view at domain granularity, 2020–2025.** A single split-axis panel: the discovery curve $U^{(d)}(s,T)/U^{(d)}(s,1)$ fills the upper (linear) half above the dashed $y = 1$ line, pairwise containment $g^{(d)}(\Delta t)$ the lower (logarithmic) half, both against elapsed time in years. Each homogeneous fit is drawn as one curve in *both* halves: burgundy uses (α_g, c_g) fit on containment, navy uses (α_U, c_U) fit on the discovery curve. If the domain population were homogeneous the two colours would coincide in both halves; instead they split apart above *and* below $y = 1$ — the homogeneous-fit failure that motivates the two-component extension in Fig. 4.

4.3 Validation on the German Academic Web

The German Academic Web (GAW) is a closed Heritrix-based focused crawl seeded from German academic homepages, biannual rather than monthly, with URL-level snapshots over 2013–2019. Its design philosophy is the opposite of CC's: a fixed seed hopper, breadth-first traversal, no harmonic-centrality reranking. Repeating the two-projection two-component check on this archive therefore tests whether the structural prediction — cross-source convergence on a single (κ, α, c) triple — is a property of the framework or an artefact of the CC pipeline. GAW supplies a single longitudinal trajectory rather than a family indexed by window-start s (the closed archive has one fixed start), so the cross-source test here is single-trajectory; its value is in whether the two projections agree at all, not in the per-trajectory variance.

The fitted (κ, α, c) on GAW differ numerically from the CC-domain values — the two crawls sample different web populations and operate at different cadences — but the structural signature holds: both projections converge on the same triple, and the homogeneous reading of either projection alone is biased in the same direction as on Common Crawl. The framework transfers across architectures. What it does not yet resolve is the residual disagreement on c_∂ visible in Fig. 4, nor whether the persistent core is a single rigid block or has internal structure of its own. Both questions live on the rank axis, and we sketch the path to a rank-resolved $\kappa(r)$ profile in the discussion, leaving the full development to follow-up work.

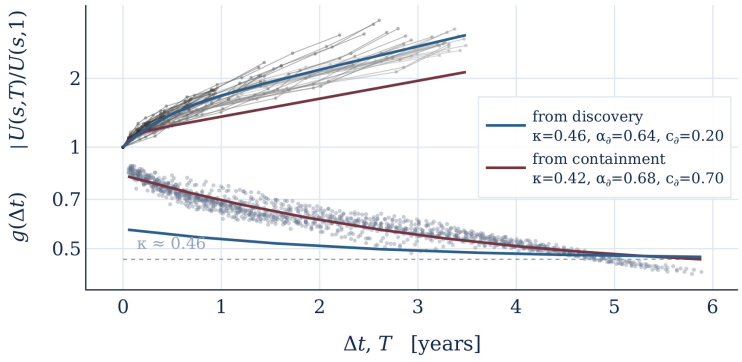


Fig. 4: **Common Crawl — two-component cross-source view at domain granularity, 2020–2025.** Same split-axis panel as Fig. 3 (discovery above $y = 1$, containment below), overlaid now with two-component fits: $(\kappa_g^{(d)}, \alpha_{\partial,g}^{(d)}, c_{\partial,g}^{(d)})$ from containment and $(\kappa_U^{(d)}, \alpha_{\partial,U}^{(d)}, c_{\partial,U}^{(d)})$ from the discovery curve; the dashed floor marks the shared core fraction $\kappa^{(d)}$. The two fits now agree on $\kappa^{(d)} \approx 0.4$ — cross-source convergence on the mass split that the homogeneous urn cannot resolve. The residual disagreement on $c_\partial^{(d)}$, visible where the two curves fail to coincide in the containment half, reflects the asymmetry between containment (recurrence-weighted) and the discovery curve (fresh-discovery-weighted) — a signature of a structurally non-uniform shell that a single scalar cannot capture.

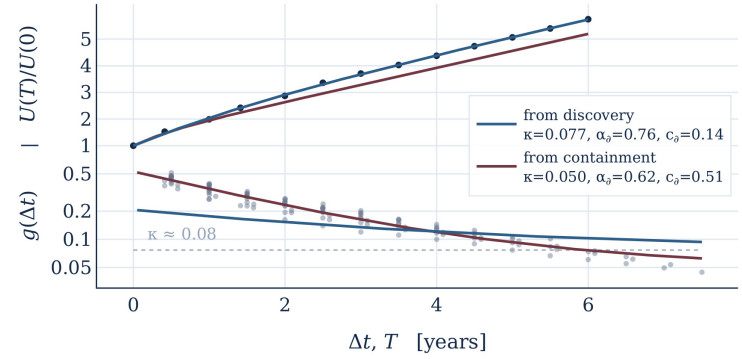


Fig. 5: **German Academic Web — two-component validation.** The same split-axis two-component cross-source check as Fig. 4 (discovery above $y = 1$, containment below), applied to the closed-archive GAW. GAW supplies a single longitudinal trajectory rather than a family indexed by window-start (the closed archive has one fixed start), so this is a single-trajectory cross-source test; cross-source agreement on a single (κ, α, c) triple is the ground-truth check on the framework before transferring it to Common Crawl. Fitted (κ, α, c) values are annotated in the panel legend.

5 Discussion

The question this paper started with was whether the operator can infer the effective sampling fraction c, persistence α, and core/shell decomposition κ from the crawler's own output, at the (host, domain) granularity where policy levers actually act, and without external ground truth. Two cross-source diagnostics on the 51-crawl 2020–2025 Common-Crawl-domain window answer yes: homogeneous fits to containment and to the discovery curve disagree visibly on (α, c); the two-component fit reconciles them on a common $\kappa^{(d)} \approx 0.4$ and reduces the gap on $(\alpha_\partial^{(d)}, c_\partial^{(d)})$ to a single residual on c_∂. The same procedure on the closed-archive German Academic Web converges on its own triple $(\kappa, \alpha, c) \approx (0.06, 0.62, 0.51)$ at URL level, and the cross-source structural prediction — two projections, one triple — transfers across crawl architectures.

What the urn parameters represent. The asymptotic regime of the discovery formula gives the operator two regression-only observables — the asymptotic new-discovery slope ν_∞ and the long-run seen-fraction $\rho_\infty = 1 - \nu_\infty$ — and inversion sends $(\nu_\infty, \rho_\infty)$ to (α, c). Read this way c is the per-round coverage fraction — directly tunable by the operator through host budgets, hc-rank thresholds, and revisit cadence; α is the per-round survival rate of the urn, which the operator does not pick (the web supplies it through the persistence of the eligible URL set); and κ is the joint readout of which URLs the policy keeps eligible (operator) and how stable that eligible set is on the web (web). Empirically the conjecture lines up: c moves visibly between crawlers (CC-domain ≈ 0.78, GAW-URL ≈ 0.51) tracking the operating-point difference, while α moves with what the population actually retains; calling out this split is a conjecture, not a theorem, and tying it down would require holding one side fixed while varying the other. A subtlety on α: the urn has no rebirth mechanism, so any apparent comeback in the data — a URL absent for several crawls and reappearing later — folds in either as an extended run of survived-and-missed events (long ℓ, high $\bar{\eta}$) or as a fresh element happening to share the identifier; the two are observationally indistinguishable and both fold via the same $\alpha^{\Delta t}$ kernel. The recovered α is therefore not a strict web survival rate but the survival rate the urn sees — a crawl-side readout that absorbs short dormancies into the hidden-time term and only registers operator-visible disappearances as churn.

Inhomogeneity, κ, and a rank-resolved outlook. The scalar κ recovered here is the simplest possible summary of population heterogeneity: a mass split between a fully resolved core and a homogeneously churning shell. The residual containment-vs-discovery gap on $c_\partial^{(d)}$ visible in Fig. 4 is the signal that the shell itself is not homogeneous — containment, being recurrence-weighted, anchors on the inner shell, while the discovery curve, being fresh-discovery-weighted, drags toward the outer shell. The natural extra observable is the per-domain rank change between consecutive crawls on the operator's hc-rank axis, which promotes the scalar κ to a continuous rigidity profile $(\mu(r), D(r))$ — the same

axis on which host budgets and harmonic-centrality thresholds act. The full development — drift, diffusion, and a survivor-flux balance that ties together the four rank-resolved profiles — is left to follow-up work; here we record only that the framework's diagnostic chain (discovery curve → cross-source disagreement → κ) terminates in a scalar that has a natural lift to the operator's policy axis, and that the residual on c_∂ is its first measurable consequence.

6 Conclusion

This paper extends the language an operator has for talking about a crawl. From the same per-crawl URL counts that operators already publish, two projections of the urn process — pairwise containment and the discovery curve — pin down a coverage fraction c (what share of the urn each round samples), a churn fraction $\bar{\alpha} = 1 - \alpha$ (what share of the urn each round replaces), an asymptotic unique rate $\nu_\infty = \bar{\alpha}/(\bar{\alpha} + \alpha c)$ (what fraction of each new crawl is genuinely new), and a core fraction κ (what share of the urn the policy keeps fully resolved).

Recombined, the same parameters give the per-URL life expectancy $\ell = \alpha/\bar{\alpha}$ in crawls, the resolve time $\eta = c\,\ell$ (crawls a URL is re-fetched over its life), and the hidden time $\bar{\eta} = \bar{c}\,\ell$ (crawls a URL is alive but unsampled). A crawl that was previously a sequence of raw counts becomes a small dictionary of comparable quantities: coverage, churn, unique rate, life expectancy, resolve time, hidden time, core fraction. The dictionary is the contribution.

Disclosure of Interests. The authors have no competing interests to declare.

References

1. Baack, S.: A critical analysis of the largest source for generative ai training data: Common crawl. In: Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency. pp. 2199–2208 (2024)
2. Bharat, K., Broder, A.: A technique for measuring the relative size and overlap of public web search engines. Computer Networks and ISDN systems **30**(1-7), 379–388 (1998)
3. Brewington, B.E., Cybenko, G.: How dynamic is the Web? Computer Networks **33**(1–6), 257–276 (2000). https://doi.org/10.1016/S1389-1286(00)00045-1
4. Chao, A.: Estimating the population size for capture-recapture data with unequal catchability. Biometrics **43**(4), 783–791 (1987). https://doi.org/10.2307/2531532
5. Chapman, D.G.: Some properties of the hypergeometric distribution with applications to zoological censuses. University of California Publications on Statistics **1**, 131–160 (1951)
6. Cho, J., Garcia-Molina, H.: The evolution of the web and implications for an incremental crawler. In: Proc. VLDB. pp. 200–209 (2000)
7. Cho, J., Garcia-Molina, H.: Synchronizing a database to improve freshness. In: Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data. pp. 117–128. ACM (2000). https://doi.org/10.1145/342009.335391

8. Dodge, J., Sap, M., Marasović, A., Agnew, W., Ilharco, G., Groeneveld, D., Mitchell, M., Gardner, M.: Documenting large webtext corpora: A case study on the Colossal Clean Crawled Corpus. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1286–1305 (2021). https://doi.org/10.18653/v1/2021.emnlp-main.98
9. Eggenberger, F., Pólya, G.: Über die statistik verketteter vorgänge. Zeitschrift für Angewandte Mathematik und Mechanik **3**(4), 279–289 (1923). https://doi.org/10.1002/zamm.19230030407
10. Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., Presser, S., Leahy, C.: The Pile: An 800GB dataset of diverse text for language modeling. arXiv preprint arXiv:2101.00027 (2020), https://arxiv.org/abs/2101.00027
11. Garg, K., Alam, S., Ayala, D., Weigle, M.C., Nelson, M.L.: A longitudinal dataset of URLs sampled from the Wayback Machine. In: Proceedings of the 2025 ACM/IEEE Joint Conference on Digital Libraries (JCDL) (2025)
12. Gomes, D., Silva, M.J.: Modelling information persistence on the web. In: Proceedings of the 6th international conference on Web engineering. pp. 193–200 (2006)
13. Good, I.J.: The population frequencies of species and the estimation of population parameters. Biometrika **40**(3-4), 237–264 (1953). https://doi.org/10.1093/biomet/40.3-4.237
14. Heaps, H.S.: Information Retrieval: Computational and Theoretical Aspects. Academic Press (1978)
15. Kaplan, J., McCandlish, S., Henighan, T., Brown, T.B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., Amodei, D.: Scaling laws for neural language models. arXiv preprint arXiv:2001.08361 (2020), https://arxiv.org/abs/2001.08361
16. Koehler, W.: Web page change and persistence—a four-year longitudinal study. Journal of the American society for information science and technology **53**(2), 162–171 (2002)
17. Kolchin, V.F., Sevast'yanov, B.A., Chistyakov, V.P.: Random Allocations. V. H. Winston & Sons, Washington, D.C. (1978)
18. Lawrence, S., Giles, C.L.: Searching the World Wide Web. Science **280**(5360), 98–100 (1998). https://doi.org/10.1126/science.280.5360.98
19. Li, J., Fang, A., Smyrnis, G., Ivgi, M., Jordan, M., Gadre, S., Bansal, H., Guha, E., Keh, S., Arora, K., et al.: DataComp-LM: In search of the next generation of training sets for language models. In: Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track (2024)
20. Lincoln, F.C.: Calculating waterfowl abundance on the basis of banding returns. US Department of Agriculture Circular **118**, 1–4 (1930)
21. Olston, C., Najork, M.: Web crawling. Foundations and Trends in Information Retrieval **4**(3), 175–246 (2010). https://doi.org/10.1561/1500000017
22. Ortiz Suárez, P.J., Romary, L., Sagot, B.: A monolingual approach to contextualized word embeddings for mid-resource languages. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL). pp. 1703–1714 (2020). https://doi.org/10.18653/v1/2020.acl-main.156
23. Paris, M., Paris, G., Baumann, F.: Estimating absolute web crawl coverage from longitudinal set intersections. arXiv preprint arXiv:2603.15416 (2026)
24. Penedo, G., Kydlíček, H., Ben Allal, L., Lozhkov, A., Mitchell, M., Raffel, C., Von Werra, L., Wolf, T.: The FineWeb datasets: Decanting the web for the finest text data at scale. In: Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track (2024)

25. Penedo, G., Malartic, Q., Hesslow, D., Cojocaru, R., Cappelli, A., Alobeidli, H., Pannier, B., Almazrouei, E., Launay, J.: The RefinedWeb dataset for Falcon LLM: Outperforming curated corpora with web data, and web data only. In: Advances in Neural Information Processing Systems (NeurIPS) (2023), https://arxiv.org/abs/2306.01116
26. Petersen, C.G.J.: The yearly immigration of young plaice into the Limfjord from the German Sea. Report of the Danish Biological Station to the Home Department **6**, 1–48 (1896)
27. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research **21**(140), 1–67 (2020), https://jmlr.org/papers/v21/20-074.html
28. Schnabel, Z.E.: The estimation of the total fish population of a lake. The American Mathematical Monthly **45**(6), 348–352 (1938). https://doi.org/10.1080/00029890.1938.11990818
29. Soldaini, L., Kinney, R., Bhagia, A., Schwenk, D., Atkinson, D., Authur, R., Bogin, B., Chandu, K., Dumas, J., Elazar, Y., et al.: Dolma: an open corpus of three trillion tokens for language model pretraining research. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL). pp. 15725–15788 (2024), https://aclanthology.org/2024.acl-long.840/
30. Stolz, A., Hepp, M.: Towards Crawling the Web for Structured Data: Pitfalls of Common Crawl for E-Commerce. In: Proceedings of the 6th International Workshop on Consuming Linked Data (COLD'15) (2015)
31. Su, D., et al.: Nemotron-CC: Transforming Common Crawl into a refined long-horizon pretraining dataset (2024)
32. Suarez, P.O., Burchell, L., Arnett, C., Mosquera-Gómez, R., Hincapie-Monsalve, S., Vaughan, T., Stewart, D., Ostendorff, M., Abdulmumin, I., Marivate, V., et al.: CommonLID: Re-evaluating state-of-the-art language identification performance on web data. arXiv preprint arXiv:2601.18026 (2026)
33. Thompson, H.S.: Improved methodology for longitudinal web analytics using common crawl. In: Proceedings of the 16th ACM Web Science Conference. pp. 59–69 (2024)
34. Together Computer: RedPajama: An open source recipe to reproduce LLaMA training dataset. https://github.com/togethercomputer/RedPajama-Data (Apr 2023)
35. Toyoda, M., Kitsuregawa, M.: What's really new on the web?: Identifying new pages from a series of unstable web snapshots. In: Proceedings of the 15th International Conference on World Wide Web. pp. 233–241. WWW '06, ACM, New York, NY, USA (2006). https://doi.org/10.1145/1135777.1135815
36. Vanni, F., Lambert, D.: Urn modeling of random graphs across granularity scales: A framework for origin-destination human mobility networks. arXiv preprint arXiv:2508.18544 (2025)
37. Weber, M., Fu, D., Anthony, Q., Oren, Y., Adams, S., Alexandrov, A., Lyu, X., Nguyen, H., Yao, X., Adams, V., Athiwaratkun, B., Chalamala, R., Chen, K., Ryabinin, M., Dao, T., Liang, P., Ré, C., Rish, I., Zhang, C.: RedPajama: an open dataset for training large language models. In: Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track (2024)