Vergessen als Regularisierung in einem energieerhaltenden Kontextgraphen

Wo begrenztes Gedächtnis die Nächstes-Zeichen-Vorhersage über Daten und Kapazität hinweg verbessert — und wo lokale Statistik an ihre Decke stößt

Marcel Langjahr

Pantareon GmbH
https://pantareon.io/
langjahr@pantareon.io

Juni 2026

Zusammenfassung

Wir halten Substrat, Daten und Evaluation fest und variieren nur das Gedächtnis-Regime. Das Substrat ist ein Kontextgraph variabler Ordnung mit PPM-C-Backoff: ohne Backprop, online, zählbasiert, mit einer aus dem Strom gewachsenen Topologie und ohne gelernte Gewichte. Das Regime ist entweder unbegrenzte Akkumulation (*plain*) oder Energieerhaltung (*conserve*), in der jedes Deposit aus einem globalen Zerfall finanziert wird, Kanten unter einem Floor geprunt werden und die Knotenzahl an einem Deckel gehalten wird, indem die energieärmsten Kontexte verdrängt werden, sodass die Gesamtenergie einen an den physischen Speicher gekoppelten Flussgleichgewichts-Fixpunkt erreicht. Wir stellen eine einzige Frage: Hilft begrenztes Vergessen der Nächstes-Zeichen-Vorhersage, und wo?

Über zwei Achsen und drei Korpora ist die Antwort konsistent. Auf der *Datengrößen*-Achse senkt Vergessen die held-out-Bits-pro-Zeichen bei jeder Korpusgröße, und die Marge wächst, je knapper die Daten werden—von einem Gleichstand bei 408k Zeichen zu -0.077 bei 50k. Auf der *Kapazitäts*-Achse verschiebt Vergessen das Bias–Varianz-Optimum zu tieferen Kontexten und verflacht die Kurve: Wo das plain-Zählen bei Ordnung 4 sein Optimum erreicht und dann in Varianz ertrinkt, nutzt der erhaltende Graph Ordnung 6 produktiv. Der Effekt ist robust über einen Faktor 25 in der Zerfallsrate (ein sauberes U mit Optimum) sowie über drei Korpora und zwei Autoren (acht Fenster, richtungsgleich, -0.07 bis -0.12 Bits/Zeichen an der besten Ordnung). Eine zweite Operation auf demselben Substrat—backoff-lokale Zustandsverschmelzung (state merging) nach Jensen–Shannon-Divergenz—halbiert die Knotenzahl grob bei gehaltener Qualität und schärft die Generierung, ohne ein Zeichen des vorhergesagten Repräsentationskollapses: Die Suffix-Hierarchie absorbiert die geteilte Struktur. Aber keines der Regimes rührt die Decke an, die wir direkt messen: Bei 11.3 Millionen Zeichen erreicht das Modell 1.83 Bits/Zeichen held-out und emittiert orthographisch perfekten, lokal flüssigen Text ohne weitreichende Kohärenz. Lokale Statistik, perfektioniert, ist nicht Bedeutung. Die Befunde konvergieren auf eine Lesart—Vergessen ist ein Regularisierer, der genau dann Sample-Effizienz erkauft, wenn die Kapazität die Daten übersteigt, und die Bedeutungs-Decke ist eine Eigenschaft des lokalen, suffix-gebundenen Substrats, die kein Gedächtnis-Regime heben kann.

Geltungsbereich dieses Papers: Die Experimente unten sind bewusst klein und Einzelmaschine: ein Zero-Dependency-Zähl-Modell über 10^5–10^7 Zeichen, ein einziger Online-Pass, Minuten auf einer CPU. Sie sind keine Behauptungen über Verhalten auf Frontier-Skala. Ihr Zweck ist vergleichend und mechanistisch—den Beitrag des *Gedächtnis-Regimes* zu isolieren, indem Substrat, Daten und Evaluation festgehalten werden. Absolute Bits-pro-Zeichen-Zahlen sollten nur relativ zueinander und zu den genannten Baselines (uniform, Unigramm, Ordnung 1) gelesen werden. Die Korpora sind deutsch—zwei vom Verfasser selbst, einer die Übersetzung eines anderen Autors—, Sprach-Generalität ist also nur insoweit indiziert, als der Effekt einen Wechsel von Autor und Register überlebt; ein nicht-deutscher Korpus und formale Konfidenzintervalle wären nötig, um sie als etabliert zu bezeichnen.

1 Einleitung

Das hier untersuchte Substrat entstammt derselben Grundüberzeugung wie die Emergente Graphentheorie: dass Funktion von einem *aus Daten gewachsenen diskreten Graphen* unter einer Erhaltungsgröße getragen wird, nicht von einem festen parametrischen Modell, das durch globale Optimierung gefittet wird. Im Kontextgraphen ist das wörtlich zu nehmen. Ein Knoten ist ein Kontext—ein Suffix des Stroms bis zu einer Maximalordnung D. Eine Kante trägt die *Energie* eines Folgesymbols in diesem Kontext, und das ist schlicht dessen akkumulierte Zählung. Vorhersage ist eine Relaxation die Suffixkette hinab (PPM-C-Backoff): Der längste passende Kontext wird mit seinen kürzeren Vorfahren gemischt, wobei die Escape-Masse zur Wurzel fließt. Es gibt keine gelernten Gewichte; die Topologie *ist* das Modell.

Diesem Substrat lässt sich ein Energieerhaltungs-Regime hinzufügen. Jedes neue Deposit wird aus einem globalen Zerfall finanziert—einem Vergessen mit Rate δ pro Symbol—, Kanten, die unter einen Floor fallen, werden geprunt und geben ihre Energie an den Pool zurück, und die Knotenzahl wird an einem Deckel gehalten, indem die energieärmsten Kontexte verdrängt werden. Die Gesamtenergie erreicht dann einen Flussgleichgewichts-Fixpunkt (Deposit-Zufluss = Zerfalls-Abfluss), und wird der Deckel im physischen Speicher gesetzt, hält sich der Graph selbst an dieser Wand: Seine verfügbare Energie *ist* seine Speicherzuteilung. Das ist das diskrete, erhaltende Analogon eines Flussgleichgewichts, und es ist die Eigenschaft, die der Rest des Papers befragt.

Daraus folgt eine natürliche Hoffnung: dass Vergessen, indem es verwirft, was nicht mehr reaktiviert wird, sich als Regularisierer verhalten und das Modell aus weniger Text generalisieren lassen würde. Dieses Paper prüft diese Hoffnung direkt und trennt zwei Dinge, die leicht vermengt werden. Das eine ist das *Substrat*: der Zähl-Graph und sein PPM-C-Backoff. Das andere ist das *Gedächtnis-Regime*: unbegrenzte Akkumulation versus Erhaltung. Indem wir dasselbe Substrat unter beiden Regimes auf denselben Daten fahren und das bessere Regime an die Grenzen von Korpusgröße und Modellkapazität treiben, können wir fragen, wo begrenztes Vergessen hilft—und, wo nicht, was uns das über das Substrat selbst sagt.

2 Experimenteller Aufbau

Gemeinsames Harness. Text wird auf Zeichenebene tokenisiert, mit einem aus dem Korpus gelesenen Vokabular; kein externer Tokenizer wird verwendet. Zwei Metriken werden berichtet. Die *held-out*-Bits-pro-Zeichen (bpc) sind die Kreuzentropie der Nächstes-Zeichen-Vorhersage auf einem Validierungs-Endstück, auf dem das Modell nie trainiert; der Train/Validierungs-Split macht den Unterschied zwischen Memorisierung und Generalisierung sichtbar. Die *prequentielle* bpc entstammt einem einzigen Online-Pass—jedes Zeichen erst vorhersagen, dann deponieren—; ihr kumulatives Mittel entspricht der Codelänge, die ein adaptiver Kompressor berichtet, und ist daher direkt mit gzip/bzip2/xz vergleichbar. Baselines sind die uniforme Schranke $\log_2 V$, die Unigramm-Entropie (Buchstabenhäufigkeiten) und die bedingte Entropie der Ordnung 1.

Substrat. Ein Kontextgraph variabler Ordnung. Deposits sind lokal und online: An jeder Position erhält jeder der durchlaufenen Kontexte (Ordnungen 0 bis D) eine Einheit Energie auf dem beobachteten Folgesymbol. Die Vorhersage mischt die Ordnungen per PPM-C-Escape,

$$p \leftarrow \frac{c_s + q\, p_{\text{lower}}}{T + q}, \tag{1}$$

von der Wurzel aufwärts gefaltet, wobei c_s der Zählstand von Symbol s im Knoten ist, T dessen Summe und q die Zahl der gesehenen distinkten Symbole (der Escape-Zähler); der Term der Ordnung -1 ist die uniforme $1/V$. Das Substrat ist in beiden Regimes identisch.

Plain-Regime. Zählungen wachsen nur; der Graph ist unbegrenzt. Das ist ein sauberes adaptives PPM-C und dient als Kontrolle.

Conserve-Regime. Deposits werden aus einem Zerfall δ pro Symbol finanziert (lazy: angewandt bei Berührung und in einem periodischen Sweep); Kanten unter einem Floor ϕ werden geprunt; und die Knotenzahl ist gedeckelt—durch ein hartes Knoten-Limit oder ein Budget an residentem Speicher—, indem die energieärmsten Kontexte bis auf eine Wassermarke verdrängt werden. Zerfall, Floor und Deckel lassen sich unabhängig schalten: Mit $\delta = \phi = 0$ und einem Speicherbudget wächst der Graph wie *plain* bis zur Wand und verdrängt dann nur unter Druck; mit $\delta, \phi > 0$ und ohne Budget vergisst er kontinuierlich, unabhängig vom

Speicher. Der Verdrängungs-*Score* ist wählbar: rohe Evidenzmasse (TOTAL); Energiedichte $\rho = \text{total}/2^{H(p)}$ (Masse pro effektivem Folgesymbol, sodass Konzentration vor Häufigkeit rangiert); oder nutzungsgewichtete Divergenz vom Backoff, usage $\times$ $D_{\mathrm{KL}}(p(\cdot \mid \text{context}) \,\|\, p(\cdot \mid \text{backoff}))$, die einen Kontext für das belohnt, was er über seinen kürzeren hinaus beisteuert, von der Masse abgesehen.

Zustandsverschmelzung. Eine separate Operation verschmilzt Kontexte, die einen direkten Backoff teilen (dasselbe Suffix ohne das älteste Symbol) und nahezu identisch vorhersagen. Ähnlichkeit ist die Jensen–Shannon-Divergenz [5] zwischen den Folgesymbol-Verteilungen der Geschwister (symmetrisch, ihre Wurzel eine echte Metrik). Die Suffix-Hierarchie dient als *gratis Nachbarschaftsindex*: Nur Geschwister unter einem Backoff sind Kandidaten, ein Pass ist also $O(N \times \text{group size})$, nie All-Pairs. Eine Verschmelzung summiert die Evidenz der Mitglieder in einen Repräsentanten und aliasiert die übrigen darauf, sodass sich die Vorhersage für jedes Mitglied zur gepoolten Verteilung auflöst. Der Pass ist auf einen Kollaps-Fehlermodus instrumentiert (Abschnitt 5).

Korpora. *Codex*: ein Kunstsprachen-/Philosophie-Text, 107k Zeichen, Vokabular 104 (lokal irregulär, beschwörend). *Prose*: der deutsche Prosaroman des Verfassers, 299k Zeichen. *Mixed*: eine 408k-Konkatenation, verwendet für den Kapazitäts-Sweep. *External*: die deutsche Übersetzung einer englischen Fantasy-Serie eines anderen Autors, $\approx$11.3M Zeichen, Vokabular 104. Die ersten beiden sind eigene Texte des Verfassers; der dritte kontrolliert für Autor und Register.

3 Vergessen als Regularisierung

Die motivierende Beobachtung ist ein einzelner Punkt, der der naiven Erwartung widerspricht. Auf *Codex* bei $D = 5$ erreicht conserve ($\delta = 8\times10^{-5}$, $\phi = 0.01$) 2.367 bpc prequentiell gegen 2.502 für plain—bessere Vorhersage mit *weniger* Knoten (42.6k vs. 56.8k). Der Grund ist Bias–Varianz: Bei Ordnung 5 auf 10^5 Zeichen werden die meisten Tiefe-5-Kontexte nur einmal gesehen, und ihre Zählungen sind Rauschen. Plain behält diese Singletons und vertraut ihnen beim nächsten Auftreten; conserve lässt sie zerfallen und prunt sie über den Floor, sodass das Modell auf den kürzeren, gut geschätzten Kontext zurückfällt. Vergessen wirkt als Regularisierung. Der Rest dieses Abschnitts fragt, ob dieser einzelne Punkt eine Kurve ist, und entlang welcher Achse.

Datengrößen-Achse. Hält man $D = 5$ fest und variiert die gesehene Menge an *Prose* (conserve als reiner Regularisierer: Zerfall und Floor an, kein bindendes Budget), zeigt Tabelle 1 conserve bei jeder Größe unter plain, mit der größten Marge in der Mitte des Bereichs.

Tabelle 1: Prequentielle bpc auf wachsenden Präfixen von *Prose*, $D = 5$ (niedriger ist besser). Conserve: $\delta = 8 \times 10^{-5}$, $\phi = 0.01$.

Zeichen	plain	conserve	Δ
10k	3.438	3.381	-0.057
25k	2.903	2.789	-0.114
50k	2.607	2.466	-0.141
100k	2.417	2.289	-0.128
200k	2.303	2.225	-0.078

Das Δ ist ein Buckel, keine Gerade: Es steigt bis 50k, plateaut, zieht sich dann zusammen. Der sich zusammenziehende rechte Arm ist das erwartete Verhalten—mit mehr Daten braucht plain die Regularisierung weniger. Der steigende linke Arm ist subtiler: Unter $\sim$50k gibt es zu wenig wiederkehrende tiefe Struktur, um sie vom Rauschen zu trennen, sodass selbst der Regularisierer wenig hat, worauf er wirken kann. Sample-Effizienz braucht ein Minimum an Daten, bevor sie greift.

Kapazitäts-Achse. Hält man den Korpus fest (*Mixed*, 408k) und sweept die Ordnung D, treten zwei Bias–Varianz-U-Kurven zutage (Tabelle 2).

Tabelle 2: Prequentielle bpc über der Kontext-Ordnung auf *Mixed* (408k). Plains Optimum liegt flach; conserves liegt tief, mit flacherer Kurve.

D	plain	conserve
2	2.729	2.725
3	2.313	2.346
4	**2.175**	2.202
5	2.208	**2.177**
6	2.298	**2.191**

Die Kurven kreuzen zwischen $D = 4$ und $D = 5$. Plains Optimum ist flach ($D = 4$); darüber dominiert die Varianz ($D = 6$: 2.298). Conserves Optimum ist tief ($D = 6$), und sein U ist viel flacher—plains steiler Anstieg bei $D = 6$ wird zu sanften 2.191. *Vergessen verschiebt das Kapazitäts-Optimum nach oben*, und genau das ist es, was einen

Regularisierer definiert: Er lässt mehr Modellkapazität zu, bevor Overfitting einsetzt. Auf diesem großen Korpus allerdings liegt plains getuntes Bestes (2.175) hauchdünn vor conserves getuntem Besten (2.177)—ein Gleichstand, kein Sieg. Der Wert des Vergessens ist hier nicht „besser als optimal getuntes plain“, sondern „robust gegen ein zu hohes D“: Bei $D = 6$ ist conserve um 0.107 bpc besser. Es verzeiht die Kapazitätswahl.

Die zwei Achsen sind eine Fläche. Ein einziges Prinzip vereint sie: Die relevante Größe ist das Verhältnis von Kapazität zu Daten. Auf demselben *Mixed*-Korpus kippt das Vorzeichen von Δ zwischen $D = 3$ (plain gewinnt) und $D = 5$–6 (conserve gewinnt); und wenn der Korpus schrumpft, wandert plains Optimum zu kleinerem D, während conserves Vorteil in der Über-D-Region wächst. Der Best-vs.-Best-Vergleich über Datengrößen (jede Seite bei ihrem eigenen optimalen D) macht das konkret (Tabelle 3): Der Vorteil des vergessenden Modells wächst monoton, je knapper die Daten werden, und verschwindet in einen Gleichstand, wenn Daten reichlich sind.

Tabelle 3: Beste held-out-/prequentielle bpc beim jeweils optimalen D (niedriger ist besser). Der Vorteil des Vergessens wächst, wenn die Daten schrumpfen.

Korpus	bestes plain	bestes conserve	Δ
Prose 50k	2.513	2.436	-0.077
Prose 100k	2.339	2.289	-0.050
Mixed 408k	**2.175**	2.177	$+0.002$

Robustheit. Drei Prüfungen verhindern, dass dies die Eigenheit eines einzelnen Korpus ist. *Erstens*, ein zweiter Autor: Derselbe 50k-D-Sweep auf *External* (ein anderer Autor, ein anderes Register) ergibt ein conserve-Best von 2.766 gegen ein plain-Best von 2.886 bei $D = 4$, $\Delta = -0.120$—größer als auf der eigenen Prosa des Verfassers, nicht kleiner. *Zweitens*, mehrere Fenster: Über drei Fenster von *Prose* beträgt die Best-D-Marge -0.066 bis -0.077, über zwei Fenster von *Codex* -0.108 bis -0.121 und über drei Fenster von *External* -0.105 bis -0.120. Die absolute bpc wandert mit dem Fenster (mancher Text ist leichter), aber Δ ist stabil, weil die Fenster-Varianz beide Regimes gleichermaßen trifft und sich im Vergleich heraushebt (Tabelle 4). *Drittens*, die Zerfallsrate: Auf *Codex* 50k bei $D = 5$ (plain 2.727) schlägt conserve plain über einen Faktor 25 in δ, mit einem sauberen U und einem Optimum bei 8×10^{-5} (Tabelle 5)—die Bias–Varianz-Signatur einer echten Regularisierungsstärke, kein getunter Zufall.

Tabelle 4: Best-D-Marge Δ (conserve − plain) über Korpora, Autoren und Fenster. Durchweg richtungsgleich.

Korpus	Autor	Fenster	Δ (bestes D)
Prose	Autor A	3	$-0.066 \ldots -0.077$
Codex	Autor A	2	$-0.108 \ldots -0.121$
External	Autor B	3	$-0.105 \ldots -0.120$

Tabelle 5: Zerfalls-Sweep auf *Codex* 50k, $D = 5$ (plain-Referenz 2.727). Ein U mit Optimum; robust über einen Faktor 25.

Zerfall δ	conserve	vs. plain
2×10^{-5}	2.655	-0.072
5×10^{-5}	2.590	-0.137
8×10^{-5}	**2.559**	-0.168
2×10^{-4}	2.584	-0.143
5×10^{-4}	2.721	-0.006

Zusammengenommen—zwei Achsen, drei Korpora, zwei Autoren, acht Fenster und ein Zerfalls-U—ist die Lesart, dass *begrenztes Vergessen im überparametrisierten Regime sample-effizient ist*: Es hilft genau dann, wenn die Kapazität übersteigt, was die Daten tragen, und spielt sonst unentschieden. Wir behaupten nichts über den gemessenen Trend hinaus. Die Korpora sind sämtlich deutsche Prosa (zwei von einem Autor), es gibt keine formalen Konfidenzintervalle, und der absolute Gewinn ist bescheiden (0.05–0.12 bpc); ein nicht-deutscher Korpus und gematchte Statistik wären nötig, um von etabliert statt indiziert zu sprechen.

4 Skalierung und die Decke

Um zu sehen, wo das Substrat steht und wo es endet, trainieren wir es auf *External* in voller Größe (11.3M Zeichen, $D = 6$). Ein einzelner Online-Pass dauert Minuten auf einer CPU und liefert einen Graphen mit 1.33M Knoten und 3.0M Kanten, der 75 MB auf der Platte belegt und $\sim$389 MiB resident—kein Speicherbudget nötig auf dieser Skala. Drei verschiedene Zahlen fallen aus demselben Lauf, und sie müssen auseinandergehalten werden (Tabelle 6).

Tabelle 6: Drei Messungen auf *External* (11.3M, $D = 6$). Sie beantworten verschiedene Fragen und dürfen nicht vermengt werden. Baselines: uniform 6.700, Bigramm 3.489.

Messung	bpc	was sie ist
in-sample (2. Pass)	1.507	Wiedererkennung
online / prequentiell	1.937	Kompression
held-out (volles Training)	**1.827**	Generalisierung

Die Ein-Pass-Eigenschaft ist real; der Kaltstart ist ein Messartefakt. Ein naheliegender Einwand ist, dass das Zähl-Modell, wie ein Gradienten-Lerner, mehr als einen Pass über die Daten brauchen könnte. Braucht es nicht—und das ist beweisbar, nicht bloß erhofft. Ein zweiter Pass über denselben Text verdoppelt jede Zählung, aber die Verteilung $p(s) = c_s/T$ ist unverändert, weil Zähler und Nenner identisch skalieren. *Die Verteilungen sind nach einem Pass final*; Zählen sättigt sofort, wo Gradientenabstieg die Repräsentation in jeder Epoche umformt und deshalb viele braucht. Die in-sample-1.507 ist kein Lernen; sie ist Wiedererkennung—der Graph sagt Text vorher, den er bereits deponiert hat. Was die prequentielle Messung sehr wohl enthält, ist ein Kaltstart (die ersten $\sim$100k Zeichen werden auf einem sich noch füllenden Graphen vorhergesagt), aber auf 11.3M Zeichen hebt das den Mittelwert nur um $\sim$0.02 bpc. Die ehrliche Generalisierungszahl ist die held-out-1.827, gemessen auf einem Endstück, auf dem das Modell nie trainiert hat, und die Validierungskurve fällt am Ende des Trainings noch—das Modell ist daten-, nicht kapazitätslimitiert.

Die Decke. Generierung aus dem vollen Graphen macht die Grenze hörbar. Eigennamen werden korrekt und in Menge reproduziert; die Orthographie ist sauber; lokale Fragmente von drei bis acht Wörtern lesen sich wie die Quelle. Aber grammatische Kongruenz bindet nicht über das Kontextfenster hinaus: Innerhalb von ~ 6 Zeichen stimmt alles, während Subjekt–Verb–Objekt über einen ganzen Satz hinweg zerfließt. Der Output ist ein nahezu perfekter lokaler Statistiker und nichts weiter: *Wörter richtig, Sätze beinahe, Bedeutung abwesend.* Entscheidend: Das verbesserte sich nicht von 10^5 auf 10^7 Zeichen—zwei Größenordnungen—, weil die Grenze nicht die Daten sind, sondern das Fehlen weitreichender Bindung. Ein Zeichen-n-Gramm, perfektioniert, wird kein Modell der Bedeutung. Das ist die saubere empirische Demonstration, wo die Methode ihre Decke hat, und die Decke ist kategorisch, keine Tuning-Lücke.

Externe Referenz. Bei der Kompression sitzt das Substrat, wo ein treues PPM sitzen sollte. Zur Orientierung: Publizierte Zeichen-Level-Zahlen auf dem Standard-Benchmark enwik8 [11] liegen grob bei: gzip ~ 3.1, bzip2 ~ 2.3, gute PPM-Varianten ~ 1.9, kleine/mittlere Zeichen-Transformer ~ 1.1–1.4. Die held-out-1.83 des Substrats ist PPM-Niveau, wie konstruiert. Ein Transformer [10] würde besser komprimieren ($\sim$0.4–0.7 bpc) und, wichtiger, die Kohärenz-Achse erklimmen, die dem n-Gramm verschlossen ist—aber er erkauft das mit GPU-Stunden über viele Epochen, und aus nur 11M Zeichen von Grund auf trainiert würde er sein enwik8-Bestes nicht erreichen und Overfitting riskieren. Die vertretbare Positionierung ist deshalb nicht „besser als ein Transformer“ (das ist es nicht), sondern: Das Substrat erreicht Vorhersage auf PPM-Niveau in *einem einzigen CPU-Pass*, und sein Hebel ist Sample-Effizienz im datenarmen Regime, nicht Kompression. Wir haben keinen gematchten Transformer laufen lassen; ihn auf demselben Korpus und derselben Hardware zu fahren ist der sauberste Weg, diese Orientierung in eine Messung zu verwandeln.

5 Zustandsverschmelzung

Wenn die Decke das Fehlen von Bindung zwischen Kontexten ist, ist der naheliegende Zug, funktional äquivalente Kontexte Evidenz teilen zu lassen. Backoff-lokales Merging (Abschnitt 2) tut das billig und verschmilzt nur Geschwister unter einem gemeinsamen Backoff. Zwei Fragen entscheiden, ob es sicher ist und ob es hilft: Kollabiert es, und verbessert es Vorhersage oder Generierung?

Kein Kollaps—der Backoff ist das Vakuum. Merging nach Ähnlichkeit hat einen eingebauten Runaway: Ein verschmolzener Knoten hat eine geglättete Verteilung, eine geglättete Verteilung ist *mehr* anderen Knoten ähnlich, ein verschmolzener Knoten könnte also zum Attraktor werden, der den Graphen in wenige diffuse Mega-Knoten schluckt (ein gravothermisches Analogon). Der Pass ist auf genau das instrumentiert: Knotenzahl gegen mittlere prädiktive Entropie. Die Messung ist eindeutig (Tabelle 7): Aggressiveres Merging *senkt* die mittlere Entropie, statt sie davonlaufen zu lassen, und senkt die held-out-bpc am Über-Kapazitäts-Modell. Wäre es kollabiert, würde die Entropie steigen und die bpc brechen; beide bewegen sich in die sichere Richtung. Die Suffix-Hierarchie ist das implizite „Vakuum“—geteilte Struktur fließt zum Backoff, statt sich in verschmolzenen Knoten zu konzentrieren—, eine explizite Barriere ist also nicht nötig. Das war ein

realer, testbarer Fehlermodus, ausgeschlossen durch Messung, nicht wegdefiniert.

Tabelle 7: Merge-Schwellen-Sweep, held-out, *External* 50k bei $D = 6$ (plain-Validierung 2.972). Aggressive Verschmelzung senkt Knoten, bpc *und* mittlere Entropie—kein Kollaps.

JS-Schwelle	Knoten	val-bpc	mittlere Entropie
ohne Merging	68337	2.972	0.277
< 0.1	56657	3.029	0.313
< 0.3	53897	2.957	0.284
< 0.6	49352	2.869	0.253
< 1.0	47443	2.812	0.236

Kompression, und eine Divergenz zwischen der Zahl und dem Leser. Auf dem vollen 11.3M-Graphen halbiert Merging die Knotenzahl grob bei gehaltener Qualität: JS<0.3 verschmilzt 568k Kontexte (1.33M$\rightarrow$ 761k Knoten), während sich die mittlere Entropie kaum bewegt (0.514 $\rightarrow$ 0.532); JS<0.6 verschmilzt 686k ($\rightarrow$643k) und drückt die Entropie auf 0.476. Aber held-out-bpc und Lesequalität laufen nicht parallel. Im 50k-held-out-Sweep verbessert aggressiveres Merging die bpc immer weiter; in der Vollkorpus-Generierung (kontrolliert mit einem Seed über die Schwellen hinweg) liest sich JS$\approx$0.3 am besten—die Glossarlisten-Artefakte und Singleton-Ausrutscher des unverschmolzenen Modells sind weggeglättet, Wörter bleiben ganz—, während bei JS$<$ 0.6 Wortbrüche und seltsame Komposita zunehmen, selbst während die Entropie weiter fällt. Die mittlere Entropie ist ihrem unverschmolzenen Wert genau bei JS$\approx$0.3 am nächsten, d. h. Konzentration ohne Verschmieren. Die Lektion ist, dass bpc *mittlere* Vorhersage misst, nicht Ganzwort-Integrität; für die Eigenschaft, auf die es hier ankommt—Kohärenz—, ist das Leserurteil das trennschärfere Instrument, und die beiden lassen sich durch Über-Verschmelzung auseinandertreiben.

Was Merging ist und was nicht. Es verbessert das Über-Kapazitäts-Regime und erzeugt einen sichtbar saubereren lokalen Sampler bei grob der Hälfte der Knoten; in held-out-bpc bei 50k stößt es plains getuntes flaches Optimum *nicht* vom Thron (2.812 bei $D = 6$ vs. plain 2.723 bei $D = 4$), genau wie conserve bei großen Daten unentschieden spielt statt zu gewinnen. Und es hebt die Decke nicht: Es verschmilzt suffix-teilende Geschwister, nicht suffix-disjunkte semantische Verwandte, Bedeutung bleibt also abwesend. Es ist ein schärferer Statistiker, kein Schritt über die kategorische Grenze.

6 Transfer und selektives Vergessen

Das Erhaltungs-Regime ist per Konstruktion ein kleines Modell aufmerksamkeitsgetriebenen Gedächtnisses: Was reaktiviert wird, persistiert; was nicht, verblasst; und endliche Kapazität zwingt das Schwächste hinaus. Wir sondieren es mit korpusübergreifender Akkumulation. Lädt man den *Codex*-Graphen und streamt dann *Prose*, verschmelzen die Vokabulare sauber (Codex' 104 Symbole erweitern sich auf 117, als Proses Latin-1-Zeichen angehängt werden, alte IDs stabil), und echte prädiktive Struktur wird transferiert: *Prose* wird mit vorgeladenem Codex bei 2.090 bpc prequentiell vorhergesagt gegen ~ 2.25 allein—rund 0.16 bpc geteilter deutscher Kurzkontext-Struktur hinübergetragen.

Vergessen und Akkumulation ziehen gegeneinander—korrekterweise. Unter Erhaltung überlebt der Transfer, verblasst aber: Der Wiederanlauf startet ~ 0.6 bpc niedriger als ein Kaltstart, aber da *Prose* Codex' distinktive Kontexte nie reaktiviert, zerfallen sie und werden geprunt, und der finale Graph ($\sim$35k Knoten) ist in der Größe von *Prose* allein nicht zu unterscheiden. Die Generierung bestätigt es fürs Ohr: Eine Anfrage, die aus dem frischen Codex-Graphen Codex' beschwörendes Register zog, zieht aus dem verblassten Proses narratives Register. Das ist kein Defekt, sobald das Ziel ausgesprochen ist. Ein System, das den ersten Korpus für immer perfekt hielte, gleichgültig, wie viel vom zweiten eintrifft, wäre eine Datenbank, kein Gedächtnis; das Verblassen unreaktivierten Materials unter neuer Aufmerksamkeit ist die Funktion. Und das Verblassen ist *selektiv*, nicht uniform: Ein Name, den beide Korpora teilen, wird reaktiviert und überlebt, während nur das nicht überlappende Material weginterferiert—Überlappung konsolidiert, das Einzigartige wird überschrieben, und das ist die Struktur retroaktiver Interferenz [9], nicht passiven Zerfalls.

Das Maß ist Häufigkeit, noch nicht Salienz. Unter einem Speicherdeckel testeten wir alle drei Verdrängungs-Scores auf dem *Codex*$\rightarrow$*Prose*-Strom bei einem bindenden 18 MiB-Budget. Sie ergeben im Wesentlichen dieselbe held-out-bpc (2.117–2.118) und dasselbe verblasste Register in der Generierung und unterscheiden sich nur in der Gleichgewichts-Topologie—Masse behält wenige fette Hubs, Dichte behält mehr, dünnere, schärfere Knoten (ρ =

total/2^H verschiebt sehr wohl, *welche* Kontexte überleben), und Wert (KL vom Backoff) vertraut Singletons. Keiner bewahrt Codex' distinktives Material, weil dieses Material im kombinierten Strom niederfrequent ist (einmal gesehen in Codex, nie aufgefrischt durch Prose); seine rohe Masse ist also klein, und es sitzt unter der Verdrängungskante bei jedem Score, der Masse im Zähler behält. Der ehrliche Schluss ist, dass das gegenwärtige Vergessen *frequenzgetrieben* ist, wo ein biologisches Gedächtnis nach *Salienz* konsolidiert—Überraschung, Vorhersagefehler, Belohnung. Die richtige offene Frage ist deshalb nicht, wie sich das Verblassen verhindern lässt (das war eine Fehlrahmung), sondern nach welchem Salienz-Maß das System vergessen sollte; KL vom Backoff—„behalte, was dich relativ zum kürzeren Kontext überrascht hat"—ist der prinzipiengeleitete nächste Kandidat, aber auf diesen Daten änderte er die Topologie, ohne schon die bpc zu ändern, was selbst die Warnung ist, dass das Maß zu ändern nicht dasselbe ist wie besser vorherzusagen.

7 Diskussion

Zwei Befunde verstärken einander. Begrenztes Vergessen erkauft Sample-Effizienz genau dort, wo die Modellkapazität die Daten übersteigt: Es senkt die held-out-bpc über Datengrößen hinweg (die Marge wächst, wenn die Daten schrumpfen), verschiebt und verflacht das Kapazitäts-Optimum und ist robust über Korpora, Autoren und Zerfallsrate. Backoff-lokales Merging schärft dasselbe Substrat—halbiert die Knotenzahl grob bei gehaltener Qualität, ohne Kollaps, weil die Suffix-Hierarchie geteilte Struktur absorbiert. Beides sind Operationen *auf* dem Zähl-Graphen, keine Substratwechsel; der Energieerhaltungs-Prior verdient seinen Platz, indem er das Verhalten messbar ändert (welche Kontexte überleben, wie das Bias–Varianz-Optimum wandert), und das ist die einzige Lizenz, die ein struktureller Prior haben sollte.

Aber dieselben Experimente ziehen eine harte Linie. Die Bedeutungs-Decke—Wörter richtig, Sätze beinahe, Bedeutung abwesend—bewegte sich von 10^5 bis 10^7 Zeichen nicht, und kein Gedächtnis-Regime rührte sie an. Sie ist eine Eigenschaft des lokalen, suffix-gebundenen Substrats: Vorhersage relaxiert eine Suffixkette hinab, und Kontexte, die kein Suffix teilen, teilen keinen Pfad. Das spiegelt, auf anderem Mechanismus, die Lektion der Begleitstudie dieser Substrat-Familie, wo das Festhalten der Architektur bei variierter Lernregel die Sprachfähigkeit im gradientenbasierten Credit Assignment verortete und nicht in der Dynamik. Hier verortet das Festhalten des Substrats bei variiertem Gedächtnis-Regime einen realen, aber begrenzten Nutzen—Sample-Effizienz unter Über-Kapazität—im Vergessen, und verortet Bedeutung in *keinem* der Regimes: Sie verlangt Repräsentationsmaschinerie, die der Zähl-Graph nicht hat. Beide Lesarten sind eine lokale Instanz des allgemeinen Musters, dass Suche und Lernen über generischer Berechnung handentworfene strukturelle Priors übertreffen [12]; der Erhaltungs-Prior hilft präzise als Regularisierer und nicht als Weg zum Verstehen.

Die Energie-/EGT-Lesart übersteht das ehrlich. Das RAM-gekoppelte Regime ist ein echtes Flussgleichgewicht—der Graph atmet an seiner Speicherwand, statt unbegrenzt zu wachsen—, und eine aus dem kosmologischen Setting importierte Sorge (dass fusionsartiges Merging zu einem singulären Attraktor kollabieren würde, wenn nicht ein Vakuum-Freiheitsgrad die Trennung erzwingt) erwies sich als Name für einen realen, testbaren Fehlermodus, den die Messung ausgeschlossen hat: Die Backoff-Hierarchie spielt die Rolle des Vakuums bereits. Aber die Analogie ist *generativ*, nicht tragend. Dichte änderte die Topologie, ohne die bpc zu ändern; der Kollaps trat nicht ein. Die Disziplin, die das Projekt verlangt, ist, die Physik als Quelle von Fragen zu halten, nicht von Antworten—„Erhaltung", „Dichte", „Vakuum" auf Mechanismen zeigen zu lassen, die sich ihren Platz dann verdienen müssen, indem sie das gemessene held-out-Verhalten messbar ändern, und jene zu verwerfen, die nur das Bild rund machen.

Das verortet die eigentliche nächste Entscheidung, und sie ist eine Substratfrage, keine Tuning-Frage. Die Bedeutungs-Decke zu durchstoßen braucht entweder weitreichende Bindung—Attention, was dieses Substrat verlässt—oder Generalisierung zwischen suffix-disjunkten Kontexten, was eine *Geometrie* auf dem Knotenraum braucht, die ein lokalistischer Suffixbaum nicht besitzt. Die Suffix-Hierarchie ist ein gratis Nachbarschaftsindex für suffix-teilende Kontexte, und Merging nutzt ihn aus; aber zwei Kontexte ohne gemeinsames Suffix und doch verwandter Bedeutung sind unerreichbar, und Brute-Force-Nachbarschaftssuche ist $O(N^2)$, unpraktikabel bei 10^6 Knoten. Die Kandidaten sind ein metrischer Index über dem Verteilungsraum (Locality-Sensitive Hashing [7]) oder eine sparse verteilte Repräsentation, in der Überlappung die Nachbarschaft *ist* [6, 8]—Letzteres ein echter Substratwechsel, per Konstruktion verlustbehaftet. Welchen man nimmt, ist die offene Architekturentscheidung; die vorliegenden Ergebnisse sagen nur, dass das Zähl-Substrat sauber an seine Decke gebracht worden ist und dass die

Decke dort liegt, wo lokale Statistik endet.

8 Fazit

Hält man Substrat, Daten und Evaluation fest und variiert nur das Gedächtnis-Regime, so finden wir, dass begrenztes Vergessen ein Regularisierer ist, der genau dann Sample-Effizienz erkauft, wenn die Modellkapazität die Daten übersteigt. Auf der Datengrößen-Achse wächst sein Vorsprung vor dem unbegrenzten Zählen, wenn die Daten schrumpfen (von einem Gleichstand bei 408k zu -0.077 bei 50k); auf der Kapazitäts-Achse verschiebt es das Bias–Varianz-Optimum zu tieferen Kontexten und verflacht die Kurve; und der Effekt ist robust über drei Korpora, zwei Autoren, acht Fenster und einen Faktor 25 in der Zerfallsrate, bei -0.07 bis -0.12 Bits/Zeichen an der besten Ordnung. Eine zweite Operation auf demselben Substrat, backoff-lokale Zustandsverschmelzung, halbiert die Knotenzahl grob bei gehaltener Qualität, ohne den vorhergesagten Repräsentationskollaps. Keines der Regimes hebt die Decke, die wir direkt messen: Bei 11.3 Millionen Zeichen erreicht das Modell 1.83 Bits/Zeichen held-out und produziert orthographisch perfekten, lokal flüssigen, global bedeutungslosen Text—lokale Statistik, perfektioniert, ist nicht Bedeutung, und die Grenze ist kategorisch. Das Ergebnis ist konstruktiv für einen energieerhaltenden Zähl-Graphen als sample-effizienten, CPU-billigen Ein-Pass-Lokalstatistiker, und negativ für die starke These, dass Bedeutung aus der Eigendynamik des Substrats emergiert. Zu etablieren, ob die Sample-Effizienz-Marge gegen einen Transformer auf demselben Korpus und derselben Hardware hält, und ob eine Geometrie auf dem Knotenraum die Decke durchstoßen kann, ohne das Zähl-Substrat aufzugeben, ist die natürliche nächste Messung.

Verfügbarkeit des Quellcodes. Das Modell ist ein einzelnes Zero-Dependency-Rust-Crate (`ctxgraph`): der Kontextgraph variabler Ordnung, die PPM-C-Mischung aus Gl. (1), das Erhaltungs-Regime mit den drei Verdrängungs-Scores, backoff-lokales Merging und Checkpointing mit Vokabular- und Alias-Persistenz. Alle Zahlen oben werden von seinen held-out- und prequentiellen Ganzdatei-Harnesses erzeugt.

Literatur

[1] von der Malsburg, C., "The Correlation Theory of Brain Function", in *Models of Neural Networks II*, E. Domany, J. L. van Hemmen, K. Schulten (eds.), Springer, New York, pp. 95–119 (1994); orig. MPI für Biophysikalische Chemie, Internal Report 81-2 (1981).

[2] Shannon, C. E., "Prediction and Entropy of Printed English", *Bell System Technical Journal* **30**(1), 50–64 (1951).

[3] Cleary, J. G., Witten, I. H., "Data Compression Using Adaptive Coding and Partial String Matching", *IEEE Transactions on Communications* **32**(4), 396–402 (1984).

[4] Moffat, A., "Implementing the PPM Data Compression Scheme", *IEEE Transactions on Communications* **38**(11), 1917–1921 (1990).

[5] Lin, J., "Divergence Measures Based on the Shannon Entropy", *IEEE Transactions on Information Theory* **37**(1), 145–151 (1991).

[6] Kanerva, P., *Sparse Distributed Memory*, MIT Press, Cambridge, MA (1988).

[7] Charikar, M. S., "Similarity Estimation Techniques from Rounding Algorithms", *Proceedings of the 34th Annual ACM Symposium on Theory of Computing (STOC)*, pp. 380–388 (2002).

[8] Plate, T. A., "Holographic Reduced Representations", *IEEE Transactions on Neural Networks* **6**(3), 623–641 (1995).

[9] McCloskey, M., Cohen, N. J., "Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem", *Psychology of Learning and Motivation* **24**, 109–165 (1989).

[10] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., Polosukhin, I., "Attention Is All You Need", *Advances in Neural Information Processing Systems* **30** (2017).

[11] Mahoney, M., "Large Text Compression Benchmark", `http://mattmahoney.net/dc/text.html` (accessed 2026).

[12] Sutton, R. S., "The Bitter Lesson", *incompleteideas.net/IncIdeas/BitterLesson.html* (2019).