

Memory load amplifies quantization noise in transformers

Stefano Fazzino

sfazzino@gmail.com

University of Catania <https://orcid.org/0000-0002-3015-4938>

Article

Keywords: quantization, memory capacity, transformers, statistical physics of learning, language models

Posted Date: October 13th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11265298/v1>

License:  This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: There is **NO** Competing Interest.

Memory load amplifies quantization noise in transformers

Abstract

Language models trained on more data degrade more when their weights are rounded to a few bits, yet the reason is not known. We address the question in a transformer that memorizes synthetic biographies, where the information stored per parameter is known exactly. A fact survives rounding as long as its margin, the lead of the correct answer over its competitors, exceeds the shift that rounding induces. As the model fills up, margins shrink, as Gardner’s theory predicts, but the shift grows, and this growth accounts for half of the extra bits needed. Tracing the rounding error through the network, we find that rounding acts as independent noise and that its growth comes from the network: the gain of downstream layers explains three quarters of it, the grid one quarter. Fed into a survival law over competing answers, the same description predicts the fraction of lost facts without fitted parameters.

Keywords: quantization, memory capacity, transformers, statistical physics of learning, language models

Training a language model for longer usually makes it better. Once its weights are rounded to a few bits, however, the opposite can happen. Scaling studies have found that the damage done by post-training quantization grows with the number of training tokens, to the point that, beyond a certain threshold, further training makes the quantized model worse^{1,2}. Since quantization is the way many large models are actually deployed^{3–5}, this is more than a curiosity. Yet the scaling laws are empirical. They track the average loss on text and leave open both what is lost and why a model that has learned more should be more fragile.

A natural place to look for an answer is the statistical physics of memory. In Gardner’s analysis of the perceptron, the margin with which a set of random patterns can be stored shrinks to zero as the number of patterns per weight approaches capacity^{6,7}. Read as a statement about quantization, this means that a fuller network holds each pattern with less room to spare and therefore tolerates less rounding. The argument, however, takes the perturbation as given. In a single-layer network this is legitimate, because the error at the output is set by the grid and by inputs that do not depend on what has been learned. In a deep network it is not: an error introduced in one weight matrix must cross every subsequent layer before it reaches the output, and

nothing guarantees that the layers transmit it in the same way whatever the network has learned.

Here we show that this second channel weighs as much as the first. We work in the controlled setting introduced by Allen-Zhu and Li⁸, in which a transformer memorizes biographies of fictitious persons and the information stored per parameter is known exactly. As the model fills up with facts its margins shrink, but the shift that rounding induces on them grows by a factor of two and a half, and this growth accounts for about half of the additional bits needed. Following the rounding error linearly through the network, we find that rounding behaves, to within a few per cent, as independent noise, and that the variance it induces on each margin factorizes into three terms: the size of the inputs to each matrix, the step of its grid and the gain of the layers downstream. The gain explains three quarters of the growth. A fuller network is thus more fragile not only because it holds its knowledge more tightly, but because it amplifies any disturbance of its weights along the very directions from which that knowledge is read. Combined with the geometry of the readout and a survival law over competing answers, the same description closes the chain that leads from the quantization grid to the facts that are lost.

Results

A fact survives while its margin outlasts the noise

We trained a GPT-type transformer with four layers, hidden size 128 and tied input and output embeddings, for a total of 1.1 million parameters, on biographies of persons with seven attributes drawn uniformly at random, such as birth date, city and employer (Methods). Each attribute value is a single token, and each person carries 45.6 bits. By varying the number of persons, we supplied the model with between 0.25 and 2 bits per parameter. The information actually stored, estimated from the cross-entropy on the attribute tokens, saturates at about 1.6 bits per parameter, in line with the capacity found by Allen-Zhu and Li⁸, and we denote it by α_s . After training, all block weights were rounded to b bits on a symmetric grid whose step is set, row by row, by the largest absolute weight, and every fact was tested individually.

A fact is recalled if its correct value y is the most probable token over the whole vocabulary. It is held, therefore, by a set of margins D_j , the differences between the logit of y and those of its competitors j . Because the readout matrix is tied to the embeddings and is not quantized, rounding reaches the output only through the shift $\delta\mathbf{h}$ of the final hidden representation, and each margin moves by

$$\delta D_j = (\mathbf{w}_y - \mathbf{w}_j) \cdot \delta\mathbf{h}, \quad (1)$$

where $\mathbf{w}_i$ is the readout row of token i . All the shifts of a fact share the term $\mathbf{w}_y \cdot \delta\mathbf{h}$, and are therefore correlated: when the logit of the correct answer drops, all competitors draw closer at once. If the shifts are Gaussian, with mean μ , a common part of variance σ_a^2 and a part specific to each competitor of variance σ_e^2 , the probability that a fact

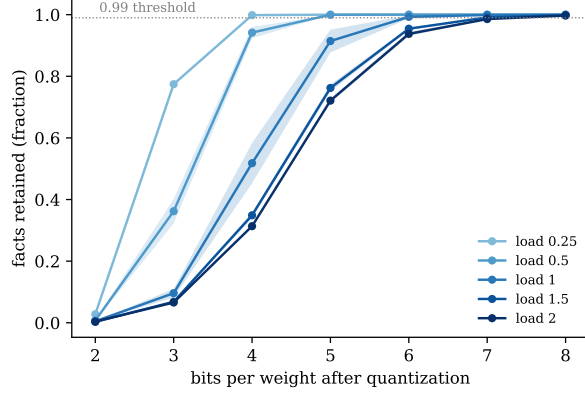


Fig. 1 Bits needed to preserve memorized facts. Fraction of the facts recalled in full precision that are still recalled after rounding the block weights to b bits, for the five loads supplied to the model (in bits per parameter). Lines are the mean over two seeds and bands the range between them. The dotted line marks the 99% threshold.

survives is

$$P = \int \frac{e^{-x^2/2}}{\sqrt{2\pi}} \prod_j \Phi\left(\frac{D_j + \mu + \sigma_a x}{\sigma_e}\right) dx, \quad (2)$$

where Φ is the standard normal distribution function: conditioned on the common term x , the competitors are independent, and a fact is lost as soon as any of them overtakes the correct answer. Equation (2) makes plain what decides the fate of a fact. When the mean shift is small, only the ratio between the margins and the noise matters, so that shrinking the margins and amplifying the noise are interchangeable; and since every additional bit roughly halves the grid step, the precision required grows as the logarithm of that ratio.

Figure 1 shows the fraction of recalled facts that survive rounding. All curves share the same step-like shape, but the step moves towards finer grids as the load grows. With 0.25 bits per parameter, four-bit rounding preserves 99.9% of the facts; with one bit per parameter it preserves 52%, and six or seven bits are needed to keep 99% of them. At the highest loads eight bits are required.

Both sides of the ratio change with the load (Fig. 2). The median margin falls from 9.3 to 2.7 logits, and the lower tail falls faster, its fifth percentile dropping from 7.9 to 0.3 logits, as Gardner’s analysis leads one to expect. The noise, which in a perceptron would depend on the grid alone, grows instead: the standard deviation of the margin shift rises from 2.3 to 5.2 logits at four bits between $\alpha_s = 0.25$ and $\alpha_s = 1$, and then stays nearly constant. Since halving the noise costs one bit, the two effects can be counted in the same currency. Between these two loads the fifth percentile of the margins shrinks by a factor of 2.6, from 7.9 to 3.0 logits, which is worth about 1.4 bits, and the noise grows by a factor of 2.3 at four bits and 2.5 at six, worth 1.2 to 1.3 bits. Together they come close to the two and a half bits that b^* , the precision needed to keep 99% of the facts, actually gains, from four to six or seven. Half of

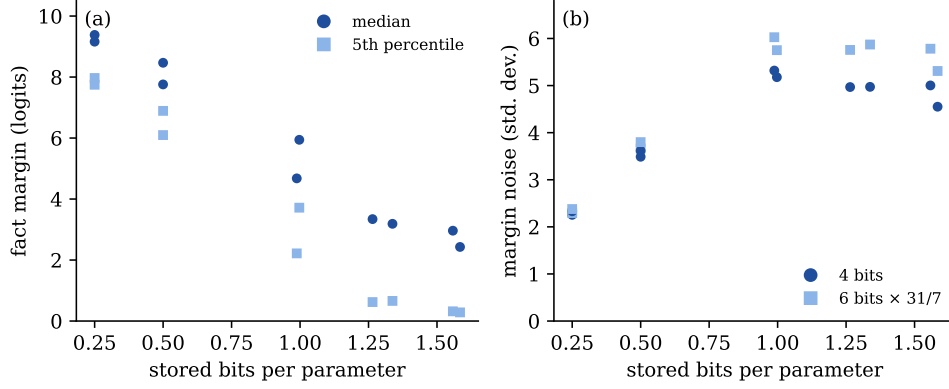


Fig. 2 Margins shrink and noise grows with the stored information. **a**, Median and fifth percentile of the margin of the facts recalled in full precision, as a function of the stored bits per parameter α_s . **b**, Standard deviation of the margin shift induced by rounding to four bits (circles) and to six bits multiplied by $31/7$ (squares), the ratio of the numbers of grid levels. Each point is a single run.

the extra precision that a fuller model needs is spent, in other words, on noise that a perceptron would not produce.

Rounding is noise that can be traced through the network

Where does this noise come from? To first order the shift of the hidden representation is

$$\delta \mathbf{h} \simeq \sum_{\ell} J_{\ell} \delta W_{\ell}, \quad (3)$$

where δW_{ℓ} is the rounding error of the ℓ -th weight matrix and J_{ℓ} is the derivative of $\mathbf{h}$ with respect to its weights, which we computed exactly by forward-mode automatic differentiation. Suppose now that the errors of the individual weights behave as independent random variables, uniformly distributed over one grid step, as in the classical theory of quantization noise⁹: the error of a weight in row i of matrix ℓ then has variance $\Delta_{\ell,i}^2/12$, where $\Delta_{\ell,i}$ is the step of that row. Under this assumption the variance of the shift of the margin $m = D_r$ towards the runner-up r , $\delta m = \mathbf{u} \cdot \delta \mathbf{h}$ with $\mathbf{u} = \mathbf{w}_y - \mathbf{w}_r$, splits into a sum of contributions from the 24 matrices of the blocks, each of which we write as the product of three factors,

$$\text{Var}(\delta m) \simeq \sum_{\ell} G_{\ell} S_{\ell} X_{\ell}, \quad S_{\ell} = \frac{1}{12} \sum_i \Delta_{\ell,i}^2, \quad X_{\ell} = \langle \|\mathbf{x}_{\ell}\|^2 \rangle. \quad (4)$$

Each term has a clear meaning. X_{ℓ} is the mean squared norm of the inputs $\mathbf{x}_{\ell}$ of the matrix, S_{ℓ} summarizes its grid steps, and their product is the total variance of the perturbation that the matrix injects. The gain G_{ℓ} is the variance of the margin produced by a perturbation of unit variance at that point, averaged over facts: it measures how strongly the rest of the network carries an error introduced in matrix ℓ to the direction from which the fact is read.

The assumption of independent noise is far from obvious, since rounding is a deterministic function of weights that training has correlated in many ways. We therefore tested it before relying on it, comparing, for 800 persons in each of ten models (five loads and two seeds), the measured shift of the margins with the linear response to the actual rounding error and with the prediction of Eq. (4). At six and eight bits the three coincide to within 3% in every model (Fig. 3a), and the correlation between the elements of the predicted and measured covariance matrices of $\delta\mathbf{h}$ lies between 0.92 and 0.98. As far as its effect on the output is concerned, the rounding of a trained network is indistinguishable from random noise with the same step. At four bits the linear approximation begins to fail, overestimating the shift by up to 19% at the highest loads, so that the analysis below is carried out at six bits, where the noise grows by a factor of 2.5 between $\alpha_s = 0.25$ and $\alpha_s = 1$ and changes by only 4% beyond, as at four bits.

The network amplifies, the grid does not

Equation (4) turns the question of why the noise grows into a question of accounting. We recomputed the variance of the model at load 1 replacing, in every matrix, one factor at a time with its value in the model at load 0.25 with the same seed. If V_f is the variance so obtained for factor f , the share of that factor is $q_f = \ln(V_1/V_f)/\ln(V_1/V_{0.25})$, which equals one if the factor alone accounts for the whole growth and zero if it accounts for none of it. Averaged over the two seeds, the gain takes a share $q_G = 0.76$, the grid step $q_S = 0.28$ and the inputs $q_X = -0.15$. The grid becomes somewhat coarser, because training produces rows with larger extreme weights, and the inputs become slightly smaller, damping the growth rather than producing it, but the decisive change is in how the network transmits the error. Averaged over the matrices, the gain increases by a factor of 5.6 between the two loads, the grid step by a factor of 1.65 (Fig. 3b). A fuller network, in other words, does not merely hold its facts more tightly: it amplifies rounding errors along the very directions from which it reads them.

Part of this amplification sits in a single place, the final layer normalization, whose scale parameters multiply the representation just before it is read. Their mean square nearly triples between the two loads and then stops growing, while the norm of $\mathbf{u}$ decreases slowly, so that the final normalization accounts for about a factor of two out of five. The rest is distributed over the blocks, and its centre of gravity moves with the load: at the lowest load the first block provides more than half of the margin variance, at the higher loads the intermediate blocks take a growing part. The same accounting explains why the noise stops growing. Beyond $\alpha_s = 1$ the gain still rises by about 20%, but the grid step and the input norms fall by about 13% each, and the changes compensate, producing the plateau of Fig. 2b.

From the grid to the lost facts

The pieces now fit into a single chain. The statistical description, independent uniform errors propagated through Eq. (3), gives not only the variance of one margin but the whole covariance Σ of $\delta\mathbf{h}$, from the grid steps and the response of the network.

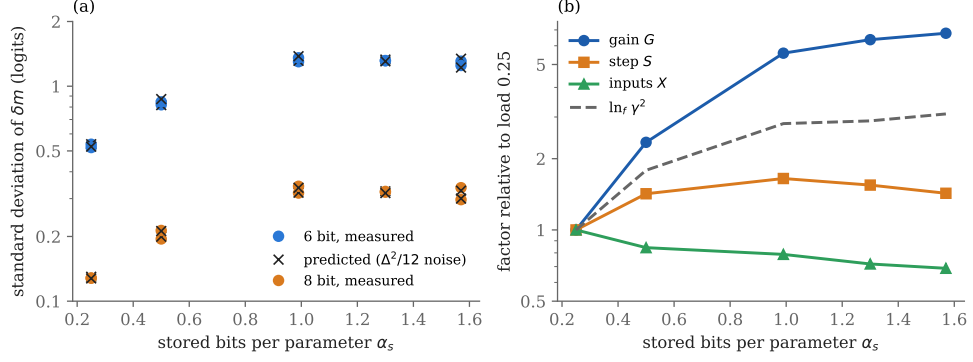


Fig. 3 Anatomy of the quantization noise. **a**, Standard deviation of the margin shift at six and eight bits, measured (circles) and predicted by Eq. (4) with independent uniform errors (crosses), for each of the ten models. **b**, Factors of Eq. (4) at six bits, as a geometric mean over the 24 block matrices and a mean over the two seeds, relative to their value at the lowest load. The dashed line is the mean square γ^2 of the scale parameters of the final layer normalization.

Equation (1) turns Σ into the covariance of the margin shifts, $(\mathbf{w}_y - \mathbf{w}_j)^\top \Sigma (\mathbf{w}_y - \mathbf{w}_k)$, whose average off-diagonal and diagonal elements fix the common and specific parts σ_a and σ_e . Equation (2) finally turns these into a probability of loss for every fact. No quantity along the chain is measured on the quantized model, and none is adjusted to the lost facts.

The chain holds. At six bits, where between 1.4% and 9.4% of the facts are lost in the models that lose at least 20, the predicted losses fall within a factor of 1.5 of the observed ones in four models out of five; in the fifth the prediction is 11% against 6.8% observed, and at the highest loads the predictions tend to exceed the observed losses. Read backwards, the chain also attributes the extra precision required by a fuller model to its causes. Of the two and a half bits gained by b^* between $\alpha_s = 0.25$ and $\alpha_s = 1$, roughly one and a half come from the narrowing of the margins and about one from the growth of the gain, while the coarser grid contributes a third of a bit, partly offset by the smaller inputs.

Discussion

A network filled with information is penalized twice, and only one of the two penalties was anticipated by the theory of memory capacity. Its margins shrink, as in Gardner’s perceptron. But the network also learns to amplify rounding errors along the directions from which it reads its knowledge, and this amplification, worth about one bit in our model, is a property of the trained network rather than of the grid on which its weights are rounded. Because rounding acts as independent noise, the amplification can be measured directly, matrix by matrix, without retraining and without quantizing the model, and together with the margins it predicts, within a factor of 1.5, which fraction of the knowledge will be lost.

This picture suggests a mechanism for the scaling laws of quantization^{1,2}: a model of fixed size trained on more data stores more information per parameter and, if

our findings carry over, both narrows its margins and amplifies rounding errors more strongly. It also points to a gap in current practice. Most of the effort in post-training quantization goes into the grid, through finer groups, the treatment of outlier weights and activations, or rounding schemes that account for second-order effects^{3–5,10}. Our decomposition indicates that, as far as they act on the grid, these methods address the smaller of the two contributions to the growth of the noise with the load, while the larger one is set during training. Methods that allocate precision according to the curvature of the loss¹¹ capture the gain implicitly; the decomposition makes it explicit, shows how it grows with the information stored and locates where in the network it resides.

These conclusions rest on a small model trained on arbitrary associations, which share no structure across examples, on a single architecture, round-to-nearest quantization and two seeds, and the dependence of the gain on depth and width remains to be established. The decisive test is whether the gain of large pretrained models grows with the number of training tokens, as our results predict. The method needed to carry it out is the one used here, and it applies unchanged to models of any size. If the test succeeds, the gain would become a diagnostic in its own right: a quantity computed from the full-precision weights alone that tells, before any compression is applied, how much precision a model can afford to lose.

Methods

Data and model

Each person is identified by a first and last name, drawn from 400 and 1,000 options so that every pair is unique, and has seven attributes (birth day, month and year, city, field of study, university and employer) with 28, 12, 100, 200, 100, 300 and 263 possible values, drawn uniformly and independently. A biography is a sequence of 16 tokens of the form *first last born D M Y in city studies field at university works_for company*, and carries $\sum_j \log_2 K_j \simeq 45.59$ bits. The model is a pre-normalization GPT-type transformer with four layers, hidden size 128, four attention heads and tied embeddings (1,099,136 parameters). The supplied load, the number of persons times their bits divided by the number of parameters, takes the values 0.25, 0.5, 1, 1.5 and 2 (6,027 to 48,217 persons), with two seeds per load that change both the initialization and the biographies. Training uses 200 presentations of each biography, AdamW with peak learning rate 2×10^{-3} , $\beta = (0.9, 0.98)$, weight decay 0.1 on matrices, batches of 256 sequences, gradient clipping at 1, linear warm-up over the first 2% of steps and cosine decay to 10% of the peak, with cross-entropy on all tokens. The stored load α_s is the number of persons times the sum over attributes of $\log_2 K_j$ minus the mean cross-entropy (in bits) on that attribute’s tokens, divided by the number of parameters⁸.

Quantization and margins

All weights of the linear layers of the blocks (786,432 parameters) are rounded to the nearest level of a symmetric grid with $2^{b-1} - 1$ positive levels, whose step $\Delta_{\ell,i}$ is the

largest absolute weight of row i divided by $2^{b-1} - 1$. Embeddings and normalization parameters are kept in full precision. For each recalled fact the margin is the difference between the logit of the correct value and the largest of all other logits; b^* is the smallest b at which at least 99% of the recalled facts survive.

Linear propagation and decomposition

For each model we sampled 800 persons with the seed of the model and kept the facts recalled in full precision (between 4,000 and 5,600). The linear response of Eq. (3) was computed with forward-mode automatic differentiation of the full forward pass, using the actual rounding errors as tangent. For the statistical description we drew, for each matrix separately, four realizations of independent errors uniformly distributed in $[-\Delta_{\ell,i}/2, \Delta_{\ell,i}/2]$, propagated each of them linearly and estimated $\text{Var}_{\ell}(\delta m)$, the margin variance due to matrix ℓ ; the gain is $G_{\ell} = \text{Var}_{\ell}(\delta m)/(S_{\ell}X_{\ell})$, and X_{ℓ} is averaged over all positions of the sampled sequences. Since errors in different matrices are independent, the variances add, which gives Eq. (4). The geometric means in Fig. 3b are taken over the 24 matrices after averaging each factor over the two seeds.

Survival law

For each recalled fact we recorded the 32 competitors with the largest logits in full precision and their margins D_j . Given a covariance Σ of $\delta \mathbf{h}$, the covariance of the shifts follows from Eq. (1); its mean off-diagonal element gives σ_a^2 and its mean diagonal element minus σ_a^2 gives σ_e^2 , fact by fact, and Eq. (2) is evaluated by Gauss–Hermite quadrature. In the prediction of the Results, Σ is the covariance of $\delta \mathbf{h}$ produced by the independent uniform errors of the statistical description, and $\mu = 0$, since in the linear description the noise has zero mean. The common and specific parts are thus computed separately for each fact, so that facts read along directions in which $\delta \mathbf{h}$ varies more are treated as more exposed.

Preregistration

The analysis of the noise was registered before the ten models were examined, with four hypotheses and decision rules: agreement of the statistical description with the measured shift within 15% at six and eight bits in all models; growth of the six-bit noise by at least 1.5 between loads 0.25 and 1 and change below 20% between 1 and 2; a factor declared dominant if its share is at least 0.6; and prediction of the six-bit losses within a factor of 1.5 in at least 75% of the models with at least 20 lost facts. All four were supported. The remaining analyses (the final normalization, the contributions of the blocks and the plateau) are exploratory.

Data availability. All data are generated synthetically by the procedure described in Methods; the generated datasets and trained models will be deposited in a public repository upon publication.

Code availability. Code to generate the data, train the models and reproduce all analyses, together with the preregistration, will be made available in a public repository upon publication.

Author contributions. S.F. conceived the study, carried out the experiments and the analysis, and wrote the manuscript.

Competing interests. The author declares no competing interests.

References

- [1] Kumar, T. *et al.* Scaling laws for precision. *Proc. Int. Conf. Learn. Represent.* (2025).
- [2] Ouyang, X. *et al.* Low-bit quantization favors undertrained LLMs: scaling laws for quantized LLMs with 100T training tokens. *Proc. Annu. Meet. Assoc. Comput. Linguist.* (2025).
- [3] Dettmers, T., Lewis, M., Belkada, Y. & Zettlemoyer, L. LLM.int8(): 8-bit matrix multiplication for transformers at scale. *Adv. Neural Inf. Process. Syst.* **35**, 30318–30332 (2022).
- [4] Frantar, E., Ashkboos, S., Hoefler, T. & Alistarh, D. GPTQ: accurate post-training quantization for generative pre-trained transformers. *Proc. Int. Conf. Learn. Represent.* (2023).
- [5] Lin, J. *et al.* AWQ: activation-aware weight quantization for on-device LLM compression and acceleration. *Proc. Mach. Learn. Syst.* **6** (2024).
- [6] Gardner, E. The space of interactions in neural network models. *J. Phys. A* **21**, 257–270 (1988).
- [7] Gardner, E. & Derrida, B. Optimal storage properties of neural network models. *J. Phys. A* **21**, 271–284 (1988).
- [8] Allen-Zhu, Z. & Li, Y. Physics of language models: part 3.3, knowledge capacity scaling laws. Preprint at <https://arxiv.org/abs/2404.05405> (2024).
- [9] Widrow, B. & Kollár, I. *Quantization Noise* (Cambridge University Press, 2008).
- [10] Nagel, M., Amjad, R. A., van Baalen, M., Louizos, C. & Blankevoort, T. Up or down? Adaptive rounding for post-training quantization. *Proc. Int. Conf. Mach. Learn.* 7197–7206 (2020).
- [11] Dong, Z., Yao, Z., Gholami, A., Mahoney, M. W. & Keutzer, K. HAWQ: Hessian aware quantization of neural networks with mixed-precision. *Proc. IEEE/CVF Int. Conf. Comput. Vis.* 293–302 (2019).