

Gamma Mixture Modeling for Cosine Similarity in Small Language Models

Kevin Player* 

Software Engineering Institute
Carnegie Mellon University

October 8, 2025

Abstract

We study the cosine similarity of sentence transformer embeddings and observe that they are well modeled by gamma mixtures. From a fixed corpus, we measure similarities between all document embeddings and a reference query embedding. Empirically we find that these distributions are often well captured by a gamma distribution shifted and truncated to $[-1, 1]$, and in many cases, by a gamma mixture. We propose a heuristic model in which a hierarchical clustering of topics naturally leads to a gamma-mixture structure in the similarity scores. Finally, we outline an expectation-maximization algorithm for fitting shifted gamma mixtures, which provides a practical tool for modeling similarity distributions.

1 Introduction

Cosine similarity is a widely used measure in semantic search [1] and the advent of sentence transformers has driven widespread adoption of text-matching systems [2, 3], where similarity scores are often used directly. In many semantic search applications, only the ranking of documents matters. In this paper, we instead focus on the statistical significance of a match. This perspective enables applications such as identifying the most surprising assignment of sentence fragments in a summary to fragments in a document, see an example in Figure 1. More generally, it can be used to compare the

Summary: Two tourist buses have been destroyed by fire in a suspected arson attack in Belfast city centre .

Document: A fire alarm went off at the Holiday Inn in Hope Street at about 04:20 BST on Saturday and guests were asked to leave the hotel. As they gathered outside they saw the two buses , parked side-by-side in the car park , engulfed by flames. One of the tour groups is from Germany , the other from China and Taiwan. It was their first night in Northern Ireland. The driver of one of the buses said many of the passengers had left personal belongings on board and these had been destroyed. Both groups have organised replacement coaches and will begin their tour of the north coast later than they had planned. Police have appealed for information about the attack. Insp David Gibson said : `` It appears as though the fire started under one of the buses before spreading to the second. `` While the exact cause is still under investigation , it is thought that the fire was started deliberately .

Match score 0 = 1142.68 bits: Two tourist buses have been destroyed by fire in a suspected arson ---- two buses , parked side-by-side in the car park , engulfed by flames. One

Match score 1 = 72.44 bits: attack ---- attack.

Match score 2 = 37.50 bits: in Belfast city ---- in Northern Ireland.

Figure 1: The most significant matching of three sentence fragments in a summary(queries) with fragments in the document. Example from xsum dataset [4] using pvalues modeled from all-MiniLM-L6-v2 [13].

combined significance of multiple matches from one search result against those from another.

A common approach to computing a p-value is the permutation test [5], which uses an empirical distribution as the null. In the context of cosine similarity, there are many examples of this approach [6, 7, 8], and specifically for sentence embeddings, see [9, 10]. While this technique is effective, it requires a sufficiently large dataset to accurately model the tail of the null distribution and offers limited insight into its overall shape. In this paper, we propose an alternative: an accurate modeling approach that requires far less data while providing a better representation of the tail behavior.

There are several natural distributions to consider when modeling cosine similarities. Smith et al. [11] study biological data with multivariate normals. The beta distribution (rescaled to $[-1, 1]$) is another candidate, given the $[-1, 1]$ support and the algebraic form of the dot product. The von Mises–Fisher (vMF) distribution is particularly appealing, as it models a Gaussian conditioned on the unit sphere $|x| = 1$, matching the normalization inherent to cosine similarity.

However, an example histogram in Figure 2 shows a typical empirical distribution. It is asymmetric, with a long right tail and a nonzero mean. The normal distribution fails to capture the asymmetry, while both the beta and vMF distributions produce a heavy left tail rather than the observed right tail (see also Section 2.1). Surprisingly, a simple gamma distribution, truncated and shifted to $[-1, 1]$, provides an excellent fit.

Gopal and Yang [12] introduced a hierarchical vMF mixture model for clustering. Building on this idea, we present a simplified hierarchical sampling argument that heuristically explains why cosine similarities may follow a gamma mixture distribution, see Section 4. Empirically, we find that gamma mixtures not only arise naturally from this perspective but also fit the observed data remarkably well.

*kplayer@andrew.cmu.edu

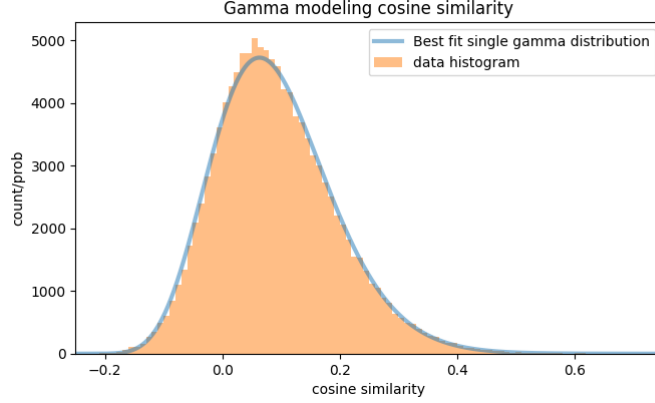


Figure 2: Example distribution D_q for the abstract ‘Using Genetic Algorithms for Texts Classification Problems’ (arXiv dataset). A shifted gamma distribution provides a good fit.

In Section 2, we present empirical distributions and demonstrate that they are often well modeled by gamma distributions and gamma mixtures. Section 3 presents a formal formulation of our gamma mixture models along with the corresponding Expectation-Maximization (EM) algorithm. Section 4 presents a heuristic model based on the hierarchical clustering of topics. In Section 5, we demonstrate that additional datasets and smaller language models are likewise well modeled by gamma distributions. Finally we finish in Section 6 with a warm start technique for our EM and some benchmarks.

2 Modeling Cosine Similarity

Let $\mathcal{S}$ denote the set of all possible sentences, and let $E : \mathcal{S} \rightarrow \mathbb{R}^n$ be an embedding. In this paper, we mainly use the default `BERTopic` sentence transformer `all-MiniLM-L6-v2`, $n = 384$, based on MiniLM[13]. We focus on a *topical* subset $\mathcal{S}_0 \subseteq \mathcal{S}$, which consists of sentences drawn from a particular domain-specific corpus, we mainly consider the arXiv abstracts dataset [14]. Given a fixed query $q \in \mathcal{S}_0$, we study the distribution $D_q(\mathcal{S}_0)$ of

$$x = \text{cosine_sim}(E(q), E(d)) \text{ for } d \in \mathcal{S}_0. \quad (1)$$

Crucially, this restriction to $\mathcal{S}_0$ significantly affects the shape of D_q . In particular, the distribution tends to have a positive mean, reflecting topical coherence.

Empirically, we observe that a (c -shifted) gamma distribution truncated

to $[-1, 1]$

$$G(\alpha, c, \lambda)(x) = \frac{(x - c)^{\alpha-1} e^{-\lambda(x-c)} \lambda^\alpha}{\Gamma(\alpha)} \quad (2)$$

often provides a good fit to the distribution of cosine similarities (see Figure 2):

α	c	λ
13.3	-0.28	35.5

The need for both a shift and truncation is surprising: cosine similarity is supported on $[-1, 1]$, whereas the gamma distribution is supported on $[0, \infty]$. In practice this means we are modeling with the tail-truncated portion of a shifted gamma, with most of the omitted mass lying in a highly improbable region.

In cases where a single gamma distribution does not adequately capture the observed distribution, a mixture of gamma distributions often yields a better approximation. Again, the precise theoretical justification remains open, but empirical results show strong agreement with this model (see Figure 3):

i	τ_i	α_i	c_i	λ_i
1	0.10	67.1	-0.20	109.0
2	0.90	19.2	-0.25	45.8

where τ_i is the mixture parameter over states i .

2.1 Von Mises-Fisher Modeling

Gopal and Yang [12] model semantic similarity using von Mises-Fisher vMF distribution in d dimensions. The vMF distribution has density on the sphere $|x| = 1$

$$f(x) = C_d(\kappa) e^{\kappa \mu^T x} \quad (3)$$

where $C_d(\kappa)$ does not depend on x , and $|\mu| = 1$. Without loss of generality, we can pick μ to be a basis vector along the first dimension, and then cosine similarity, $t = \mu^T x \in [-1, 1]$, is just the first coordinate of x . We integrate along the other dimensions of x to find a pdf for the cosine similarity t

$$g(t) \propto (1 - t^2)^{\frac{d-3}{2}} e^{\kappa t} \quad (4)$$

upto a constant that depends on d and κ . This distribution is centered at positive t but exhibits a heavy left tail, see Figure 4, which conflicts with the right-tailed empirical behavior.

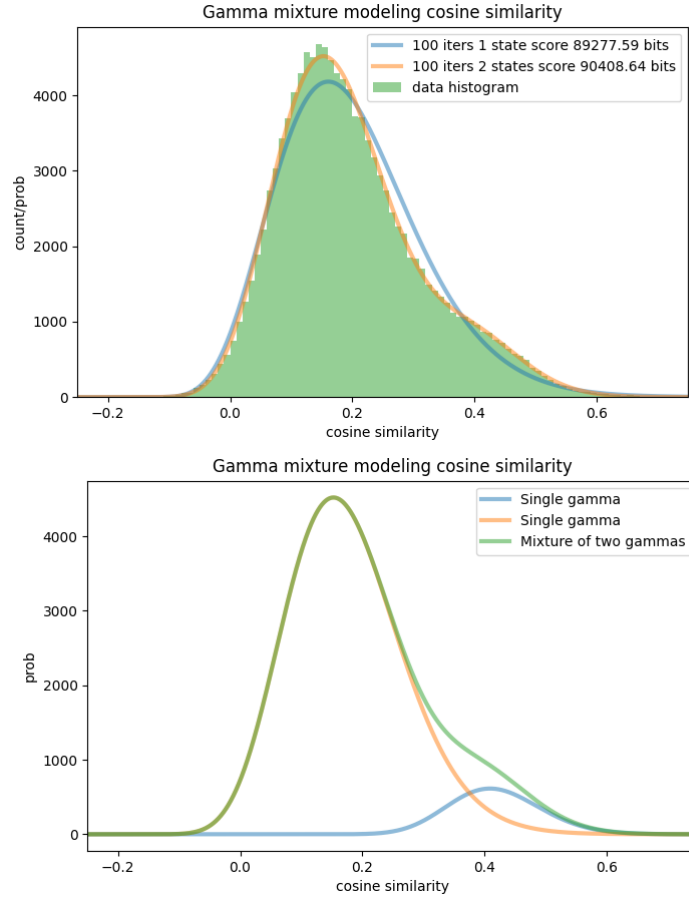


Figure 3: An example of a D_q , q in this case is the arXiv abstract for ‘‘Why Global Performance is a Poor Metric for Verifying Convergence of Multi-agent Learning’’ in the arXiv dataset. D_q is fit well by a mixture of two gamma distributions.

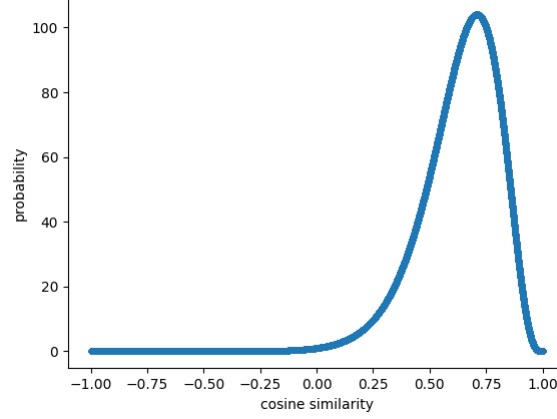


Figure 4: Typical vMF distribution of cosine similarity ($d = 10$ and $\kappa = 10$).

3 (Shifted) Gamma Mixture Math

3.1 Mixture Formulation

Suppose we are given a dataset x_t , and we wish to model it as a mixture of s shifted¹ gamma distributions $G_{\alpha_i, c_i, \lambda_i}$:

$$P_{\alpha_i, c_i, \lambda_i}(x) = \sum_{i=1}^s \tau_i G_{\alpha_i, c_i, \lambda_i}(x) = \sum_{i=1}^s \tau_i \frac{(x - c_i)^{\alpha_i - 1} e^{-\lambda_i(x - c_i)} \lambda_i^{\alpha_i}}{\Gamma(\alpha_i)} \quad (5)$$

where τ_i are the mixture weights with $\sum \tau_i = 1$. We can use expectation maximization (EM) [15] to do this.

3.2 Expectation Maximization Setup

It is well known how to fit Gamma mixture models using Expectation Conditional² Maximization ECM [16]. We present an version for our shifted GMMs. The expectation (E) step is to compute

$$\gamma_{t,i} = \text{Prob} \left(\begin{array}{c} \text{state } i \\ \text{and} \\ \text{sample } t \end{array} \right) = \tau_i G_{\alpha_i, c_i, \lambda_i}(x_t) / P_{\alpha_i, c_i, \lambda_i}(x_t) \quad (6)$$

¹The model assumes that $x_t > c_i$

²Conditional just means that we split the maximization step up into parts.

The maximization (M) step is more involved and will fill out the remainder of this section. We first write out the γ weighted log likelihood function

$$Q(\alpha_i, c_i, \lambda_i) = \log(P_{\alpha_i, c_i, \lambda_i}) = \sum_{t,i} \gamma_{t,i} \left(\begin{array}{l} \log \tau_i + (\alpha_i - 1) \log(x_t - c_i) \\ -\lambda_i(x_t - c_i) + \alpha_i \log \lambda_i - \log \Gamma(\alpha_i) \end{array} \right) \quad (7)$$

and then take some derivatives.

3.3 Reestimating τ_i

We compute

$$\frac{\partial Q}{\partial \tau_i} = \sum_t \gamma_{t,i} \frac{1}{\tau_i} \quad (8)$$

and use a Lagrange multiplier ν on $\sum \tau_i = 1$, $\frac{\partial Q}{\partial \tau_i} = \nu$, to update τ_i

$$\hat{\tau}_i = \frac{\sum_t \gamma_{t,i}}{\sum_{t,j} \gamma_{t,j}} \quad (9)$$

3.4 Elimination of λ_i

To simplify the maximization of α_i and λ_i , we will eliminate λ_i . Compute

$$0 = \frac{\partial Q}{\partial \lambda_i} = \sum_t \gamma_{t,i} \left(-(x_t - c) + \frac{\alpha_i}{\lambda_i} \right) \quad (10)$$

and write λ_i in terms of α_i

$$\lambda_i = \alpha_i \kappa_i \quad (11)$$

where

$$\kappa_i = \frac{\sum_t \gamma_{t,i}}{\sum_t \gamma_{t,i}(x_t - c_i)} \quad (12)$$

is the inverse of the γ -weighted mean³ of $x_t - c_i$. We now plug this expression back into the Q-function to obtain a reduced form:

$$Q_0(\alpha_i, c_i) := Q(\alpha_i, c_i, \lambda_i = \alpha_i \kappa_i), \quad (13)$$

which we will maximize over α_i in the next step.

³This mirrors the structure of the rate parameter in maximum likelihood fitting of the standard gamma distribution.

3.5 Reestimating α_i and λ_i

We differentiate Q_0 to find an equation in terms of α_i having eliminated λ_i

$$\frac{\partial Q_0}{\partial \alpha_i} = \sum_t \gamma_{t,i} (\log(x_t - c_i) - \kappa_i(x_t - c_i) + \log \alpha_i + 1 + \log \kappa_i - \psi(\alpha_i)) \quad (14)$$

where ψ is the digamma function. We can next compute

$$\frac{\partial^2 Q_0}{\partial \alpha_i^2} = \sum_t \gamma_{t,i} \left(\frac{1}{\alpha_i} - \psi^{(1)}(\alpha_i) \right) < 0 \quad (15)$$

where the positivity comes from a known trigamma inequality⁴ [17]. So equation (14) is monotone increasing and we can find a root $\hat{\alpha}_i$ by bisecting it. Then we use equation (11) to update λ_i as $\hat{\lambda}_i = \hat{\alpha}_i \kappa_i$.

3.6 Reestimating c_i

Next, we focus on c_i

$$\frac{\partial Q}{\partial c_i} = \sum_t \gamma_{t,i} \left(\frac{1 - \alpha_i}{x_t - c_i} + \lambda_i \right) \quad (16)$$

and the second derivative is

$$\frac{\partial^2 Q}{\partial c_i^2} = \sum_t \gamma_{t,i} \frac{1 - \alpha_i}{(x_t - c_i)^2}. \quad (17)$$

Since $\gamma_{t,i} \geq 0$ and the denominator is always positive, the sign of the second derivative is determined entirely by $1 - \alpha_i$, which is fixed. Therefore, Q is either strictly convex, linear, or strictly concave in c_i , depending on the sign of $1 - \alpha_i$. In particular, generically⁵, the equation (16) has at most one solution, and we can find the root $\hat{c}_i$ efficiently using a bisection method.

3.7 Non-convexity of Q_0

Although it might be desirable to jointly reestimate $Q_0(\alpha_i, c_i)$, the Hessian for the t -th summand

$$\begin{aligned} H &= \begin{bmatrix} \frac{\partial^2 Q}{\partial \alpha_i^2} & \frac{\partial^2 Q}{\partial \alpha_i \partial c_i} \\ \frac{\partial^2 Q}{\partial \alpha_i \partial c_i} & \frac{\partial^2 Q}{\partial c_i^2} \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{\alpha_i} - \psi^{(1)}(\alpha_i) & -\frac{1}{x_t - c_i} + \kappa_i \\ -\frac{1}{x_t - c_i} + \kappa_i & \frac{1 - \alpha_i}{(x_t - c_i)^2} \end{bmatrix} \end{aligned} \quad (18)$$

⁴This is consistent with the standard gamma distribution, where the log-likelihood is also strictly concave in α after eliminating λ for the same reason.

⁵The linear $\alpha = 1$ or $\sum_t \gamma_{t,i} = 0$ case requires no c_i update.

reveals that the objective is generally non-convex since the scaled determinant

$$(x_t - c_i)^2 \det(H) = \left(\frac{1}{\alpha_i} - \psi^{(1)}(\alpha_i) \right) (1 - \alpha_i) - \left(\frac{1}{x_t - c_i} + \kappa_i \right)^2 \quad (19)$$

is negative in the range of typical x_t that we encounter. It is an open problem to see if nonlinear fitting Q_0 in one step is faster than the ECM coordinate zig-zag⁶.

4 Hierarchical Modeling

We motivate the use of gamma mixture models to characterize cosine similarity. One promising perspective arises from viewing the embedding space E through the lens of topic modeling [18], where documents and queries are associated with a hierarchy of latent topics, and embeddings are organized around their respective topic centers. Under this view, cosine similarities are naturally right-skewed, as a given query tends to be close to a cluster center in the embedding space.

To make this intuition precise, we turn to Algorithm 1, which constructs a hierarchical tree of cluster centers. At each iteration, a node in the binary tree splits into **degree** child nodes, centered near their parent. The centers are perturbed according to a correlation strength parameter η , creating a structured dependency across levels. This generative process induces a heavy right tail in the distribution of similarities: the cosine similarity between the query q and a sampled embedding d depends on their relative positions in the tree, i.e., how recently they share a common ancestor. When d is drawn from a node closely related to q , the similarity is high; otherwise, it decays due to increasing separation in the latent space.

With a binary tree, **degree** = 2, and sufficient **depth** (e.g., 20), we obtain enough samples to meaningfully study the distribution. For example, with $\eta = 0.95$, the distribution is well-approximated by a single gamma (see Figure 5). Increasing to $\eta = 0.995$, a mixture of two gammas fits better, see Figure 6.

4.1 Heuristic Argument – Mixture of Multiple Hierarchies

We base our heuristic argument on viewing the distribution in Figure 5 as a mixture of level-wise contributions from the tree generated by Algorithm 1. Each successive level contains half as many nodes and thus contributes

⁶We currently only do one update of α_i and c_i per EM-step, but updates could be done in any order or even repeatedly per step.

Algorithm 1 Simulation of Hierarchical Clustering Distribution (thanks to Chat-GPT for explaining my code in L^AT_EX)

Require: Depth m , Ratio η , Degree k , Dimension $n = 384$, Seed $s = 1$

```

1: Set random seed to  $s$ 
2: Initialize  $X \leftarrow$  one vector with entries from  $\text{uniform}(-1, 1)$  in  $\mathbb{R}^n$ 
3: for  $i = 1$  to  $m$  do
4:   Initialize empty list  $Y$ 
5:   for each vector  $x$  in  $X$  do
6:     Sample  $k$  new vectors uniformly from  $(-1, 1)^n$ 
7:     For each, compute  $y \leftarrow \eta \cdot x + \text{noise}$ 
8:     Append all  $y$  to  $Y$ 
9:   end for
10:  Set  $X \leftarrow$  concatenate all vectors in  $Y$ 
11: end for
12: for each vector  $x$  in  $X$  do
13:   Normalize  $x$  to unit length
14: end for
15: Let  $q \leftarrow X_0$  (the original vector)
16: for each  $x$  in  $X$  do
17:   Compute  $C \leftarrow q \cdot x$  (cosine similarity)
18: end for
19: return  $C$ 

```

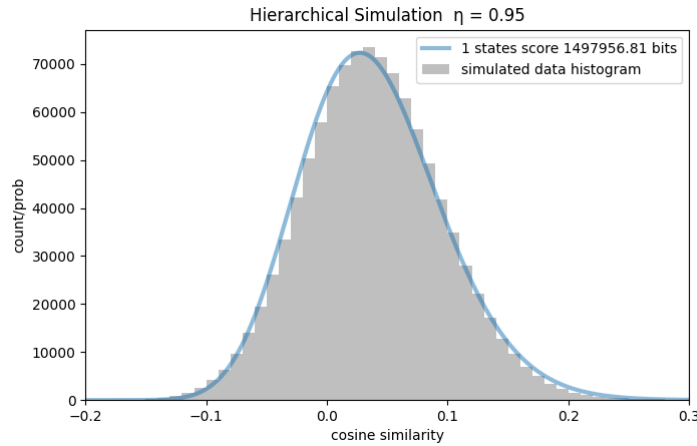


Figure 5: Histogram and fitted gamma for $\eta = 0.95$ in Algorithm 1. It is fit well by a single gamma.

[Distribution Statement A] Approved for public release and unlimited distribution.

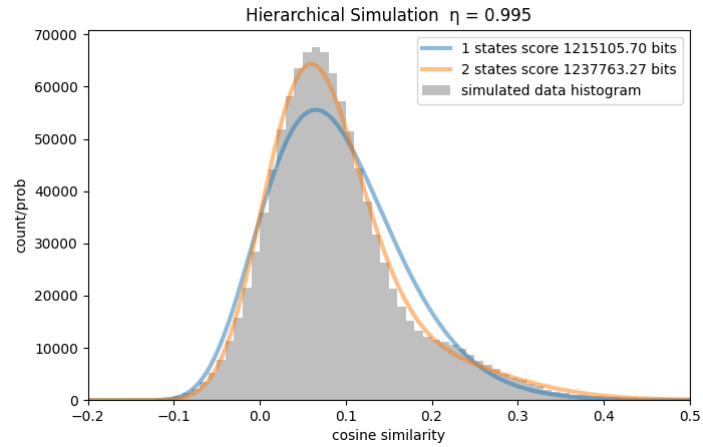


Figure 6: Histogram and fitted gamma mixture for $\eta = 0.995$ in Algorithm 1. It is fit well by a mixture of two gammas.

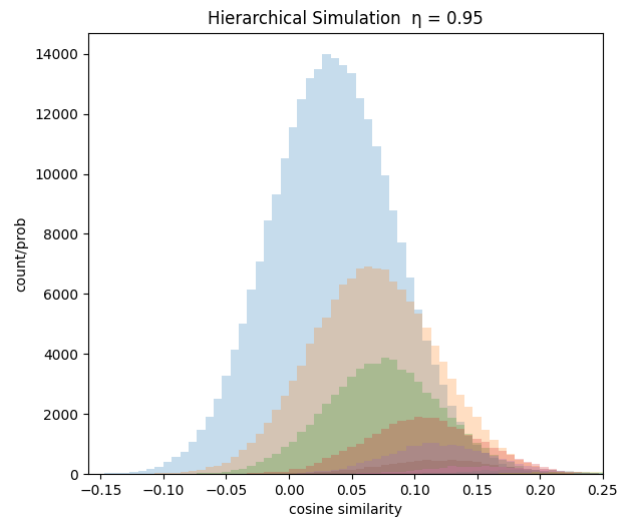


Figure 7: Histograms for $\eta = 0.95$ in Algorithm 1. Colors indicate the hierarchy level whose center is closest to the query. The overlapping contributions from different levels blend to form a heavy right tail; Compare with Figure 5.

[Distribution Statement A] Approved for public release and unlimited distribution.

half as many samples, producing a geometric decay clearly visible in the color-coded breakdown of Figure 7. Empirically, each level’s component distribution is similarly shaped, shifts rightward in similarity space, and maintains comparable variance. This approximates, in the continuum, the convolution of an exponential decay with a more symmetric kernel⁷.

Averaging over latent factors – such as depth, correlation strength, and topic context – across multiple instances of Algorithm 1, and their level-wise components produces a distribution shaped by overlapping exponential effects (with similar rates) and hidden variability. This blend naturally aligns with the behavior of a gamma mixture model and motivates its use as a flexible and interpretable fit.

5 Other Models and Data

In addition to the `all-MiniLM-L6-v2` sentence embedding based on MiniLM[13], we consider two other small language models based on MPNet[19] and RoBERTa[20]:

	model	speed	dimension	layers	context
	<code>all-MiniLM-L6-v2</code>	1000	384	6	256
	<code>all-mpnet-base-v2</code>	200	768	12	384
	<code>all-roberta-large-v1</code>	80	1024	24	128

Speed is in sentences per second on a `Tesla V100-PCIE-16GB` GPU. The dimension is embedding dimension, and the context window is measured in tokens.

In addition to the arXiv abstracts dataset [14], we consider Wikipedia [21] and `ag_news` [22]. Consider the distributions in all 9 pairings of the 3 models with the 3 datasets in Figure 8. These are formed by taking the first sentence in each dataset and computing the cosine similarity against the first 100K other sentences. They are all pretty well described by a single gamma distribution.

6 Warm Starting Speed up

We fit on a smaller set of data at first to warm start the convergence and to speed up the algorithm. We typically do this for 95% of the iterations on 1/20 of the data; only spending the last 5% of the iterations on the full data set. The difference in score is negligible, but the speed gain is an order of magnitude. Our C++ code is currently competitive with `scipy.stats.gamma.fit`:

⁷The kernel can be modeled as a gamma with large α , looking approximately Gaussian in this case.

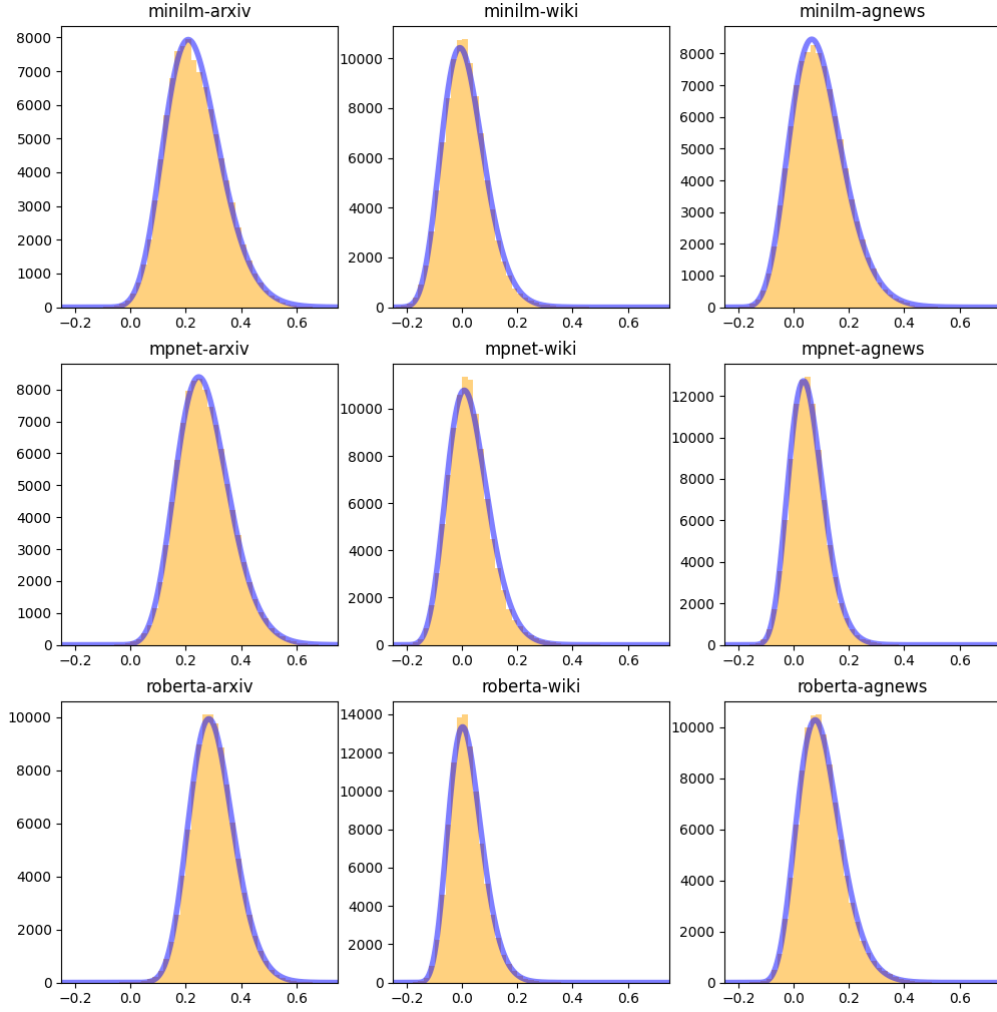


Figure 8: Distribution of cosine similarity per model and dataset, each fit with a single gamma.

[Distribution Statement A] Approved for public release and unlimited distribution.

algorithm	states	time(ms)
scipy.stats.gamma.fit	1	863
ours	1	116
ours	2	236
ours	4	399

7 Acknowledgments

Copyright 2025 Carnegie Mellon University.

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

The view, opinions, and/or findings contained in this material are those of the author(s) and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License. Requests for permission for non-licensed uses should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

This work product was created in part using generative AI.

Carnegie Mellon® is registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

DM25-1213

References

- [1] A. Singhal *et al.*, “Modern information retrieval: A brief overview,” *IEEE Data Eng. Bull.*, vol. 24, no. 4, pp. 35–43, 2001.

[Distribution Statement A] Approved for public release and unlimited distribution.

- [2] A. Alqahtani, H. Alhakami, T. Alsubait, and A. Baz, “A survey of text matching techniques,” *Engineering, Technology & Applied Science Research*, vol. 11, no. 1, pp. 6656–6661, 2021.
- [3] R. Winastwan, “Transforming text: The rise of sentence transformers in nlp,” March 2024. Accessed: 2025-09-04.
- [4] S. Narayan, S. B. Cohen, and M. Lapata, “Don’t give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization,” *ArXiv*, vol. abs/1808.08745, 2018.
- [5] J. H. Moore, “Bootstrapping, permutation testing and the method of surrogatedata,” *Physics in Medicine & Biology*, vol. 44, no. 6, p. L11, 1999.
- [6] I. Bendidi, S. Whitfield, K. Kenyon-Dean, H. B. Yedder, Y. E. Mesbahi, E. Noutahi, and A. K. Denton, “Benchmarking transcriptomics foundation models for perturbation analysis: one pca still rules them all,” *arXiv preprint arXiv:2410.13956*, 2024.
- [7] Y. Zhou, M. Kaneko, and D. Bollegala, “Sense embeddings are also biased—evaluating social biases in static and contextualised sense embeddings,” *arXiv preprint arXiv:2203.07523*, 2022.
- [8] Y. Liu, A. Medlar, and D. Glowacka, “Statistically significant detection of semantic shifts using contextual word embeddings,” *arXiv preprint arXiv:2104.03776*, 2021.
- [9] C. May, A. Wang, S. Bordia, S. R. Bowman, and R. Rudinger, “On measuring social biases in sentence encoders,” *arXiv preprint arXiv:1903.10561*, 2019.
- [10] Z. Wu, W. Merrill, H. Peng, I. Beltagy, and N. A. Smith, “Transparency helps reveal when language models learn meaning,” *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 617–634, 2023.
- [11] I. Smith, J. Ortmann, F. Abbas-Aghababazadeh, P. Smirnov, and B. Haibe-Kains, “On the distribution of cosine similarity with application to biology. arxiv 2023,” *arXiv preprint arXiv:2310.13994*, 2023.
- [12] S. Gopal and Y. Yang, “Von mises-fisher clustering models,” in *International Conference on Machine Learning*, pp. 154–162, PMLR, 2014.
- [13] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. Zhou, “Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers,” *ArXiv*, vol. abs/2002.10957, 2020.
- [14] C. Shorten, “Ml-arxiv-papers.” <https://huggingface.co/datasets/CShorten/ML-ArXiv-Papers>, 2022.
- [15] A. Dempster, “Maximum likelihood from incomplete data via the em algorithm,” *Journal of Royal Statistical Society*, vol. 39, pp. 1–22, 1977.

[Distribution Statement A] Approved for public release and unlimited distribution.

- [16] D. S. Young, X. Chen, D. C. Hewage, and R. Nilo-Poyanco, “Finite mixture-of-gamma distributions: estimation, inference, and model-based clustering,” *Advances in Data Analysis and Classification*, vol. 13, no. 4, pp. 1053–1082, 2019.
- [17] “NIST Digital Library of Mathematical Functions.” <https://dlmf.nist.gov/>, Release 1.2.4 of 2025-03-15. F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller, B. V. Saunders, H. S. Cohl, and M. A. McClain, eds.
- [18] M. Grootendorst, “Bertopic: Neural topic modeling with a class-based tf-idf procedure,” *arXiv preprint arXiv:2203.05794*, 2022.
- [19] K. Song, X. Tan, T. Qin, J. Lu, and T.-Y. Liu, “Mpnet: Masked and permuted pre-training for language understanding,” *ArXiv*, vol. abs/2004.09297, 2020.
- [20] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized bert pretraining approach,” *ArXiv*, vol. abs/1907.11692, 2019.
- [21] W. Foundation, “Wikimedia downloads.”
- [22] X. Zhang, J. Zhao, and Y. LeCun, “Character-level convolutional networks for text classification,” in *Advances in Neural Information Processing Systems*, vol. 28, pp. 649–657, 2015.