
Quantifying Positional Biases in Text Embedding Models

Anonymous Author(s)

Affiliation

Address

email

Abstract

1 Embedding models are crucial for tasks in Information Retrieval (IR) and semantic
2 similarity measurement, yet their handling of longer texts and associated positional
3 biases remains underexplored. We explore the effect of content position and
4 size within an embedding model’s input on its final embedding vector, finding a
5 significant overweighting of initial content in an input. We employ two ablations
6 to test this effect, inserting irrelevant text into a document and removing text from
7 a document. We find that perturbations to the beginning of a document reduce its
8 cosine similarity with the original document by 12.3% more than perturbations done
9 to the end, with this trend holding across multiple models and datasets. Next, we
10 attempt to reconstruct a document’s embedding vector from the embeddings of its
11 sentences, achieving an 85.5% R^2 using a simple linear regression to weight each
12 sentence’s contribution to the document embedding. Using this finding, we can
13 assign an importance weight to each sentence and find a -0.55% correlation between
14 sentence starting index and importance score. We also measure a statistically
15 significant difference between a sentence’s importance score and the expected
16 importance of equal weighting amongst all sentences. To ensure our results are not
17 the effect of dataset bias, we shuffle the sentences in each tested document before
18 repeating our experiments and see similar results. Finally, we focus on the role of
19 positional encodings and training methodology on this bias and introduce a data
20 augmentation scheme we title Position-Aware Data Sampling (PADS) to remedy
21 these issues. Fine-tuning an embedding model with only 20% of our data using
22 PADS leads to a 49.6\$ reduction in the differing effect of perturbations done at
23 the beginning and end of a document, suggesting PADS as an effective avenue
24 for reducing positional bias in our models and improving the performance of both
25 retrieval and document processing systems.

1 Introduction

27 Embedding models are increasingly used to encode text information in a way that aligns semantically
28 with their intended applications [6, 36]. However, their effectiveness in long-context settings,
29 particularly how they encode larger documents, remains less explored. Due to the typical limitations
30 of models’ context windows, techniques like document chunking are employed to fit large documents
31 into manageable segments [42]. Yet, research into optimal chunking strategies is still emerging,
32 often leading to preliminary findings that may not provide the most effective results without costly,
33 domain-specific adjustments [40].

34 This study investigates the influence of content position and size within an embedding model’s input on
35 the resulting text embedding vector. Our findings indicate that embedding models disproportionately
36 weigh the beginning of the text, often assigning greater importance to the first sentences of a multi-
37 sentence or long-context input. To substantiate this observation, we conducted two types of ablation

studies: one involving the insertion of irrelevant text ("needles") into the document [11], and another involving the removal of varying chunks from the document. Our results show that inserting irrelevant text into the beginning of a text significantly reduces the cosine similarity between the altered and original document embeddings by up to 8.5% more than insertion in the middle and 12.3% more than insertion at the end. Removal experiments reinforce this trend, with the largest decreases in similarity occurring when text is removed from the beginning of a document.

We then employ a regression analysis method, finding that using a simple linear regression model to reconstruct a document embedding vector through the embedding vectors of its constituent sentences yields a 0.85 R^2 score averaged across all documents [31]. This result indicates that we can effectively back out the contribution of each sentence’s embedding vector to the final document embedding vector by analyzing our regression’s weights. We observe a significant decline in regression coefficients as the position of a sentence within its document increases to be further from the front, underscoring a systematic favoring towards the initial content of our input. To ensure our results are not the effect of dataset bias, we repeat all experiments with each document’s sentences shuffled in a random order, and achieve similar results.

Next, we delve into potential reasons for this bias, focusing on the role of positional encodings and data treatment in the training process of embedding models. Most training techniques for embedding models use simple truncation to pre-process their data if it exceeds the model’s context window [18, 38]. This can have confounding effects on real-world retrieval situations where early sections of a document may have a disproportionately high similarity despite key information being located elsewhere [1]. To address this issue we propose a new data augmentation scheme titled Position-Aware Data Sampling (PADS). PADS randomly samples a consecutive set of tokens within each document in place of simple truncation, varying the positioning and size of our sample to improve model robustness. We fine-tune BAAI’s BGE-Small-en-v1.5 model on our dataset augmented with PADS and achieve a 49.6% improvement in closing the gap between model similarity scores for perturbations at the beginning vs. the end of a document, averaged across both insertion and removal tasks.

We conclude by discussing the implications of embedding models and potential biases they may hold, emphasizing the need for future research to study the output of embedding models and improve the processing of long documents.

2 Background

2.1 Bidirectional encoding in embedding models

Embedding models, particularly those utilizing transformer encoder architectures [34], employ layers of bidirectional self-attention blocks to process text [6]. These models are distinct from decoders in that they generate a fixed-length vector representing the entire input text. This is achieved by producing an output matrix $L \times D$ (where L is the sequence length and D is the dimensionality of the embeddings), and then applying either mean or max pooling across the L dimension [25]. Such pooling operations are position-invariant, theoretically suggesting an unbiased treatment of input positions in terms of attention and representation [28].

The core operation in these models is the attention mechanism, which can be represented mathematically as:

$$A = \text{softmax} \left(\frac{X^T X}{\sqrt{d}} \right) X^T$$

Here, X is the $L \times D$ input matrix to the attention mechanism, and d is the scaling factor derived from the dimensionality D of the embeddings. Unlike generative models where a causal attention mask is used to zero out certain elements in our softmax operation, embedding models are fully bi-directional and do not employ an attention mask.

We use cosine similarity to compare the output embeddings from these models, especially to study the effects of textual modifications such as insertions or deletions. Cosine similarity measures the cosine of the angle between two vectors, thus providing a scale- and orientation-invariant metric to assess the similarity between two text representations [16].

$$\text{cosine}(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\| \|\mathbf{y}\|}$$

87 Due to the invariance of the architecture and similarity measurement we employ, the last systematic
 88 source of bias stems from learned positional embeddings used in our models and the models’ training
 89 methodology, which are heavily connected.

90 **2.2 Dataset and human-level writing bias**

91 It’s important to note that human writing often emphasizes key information at the beginning and
 92 end of documents, a technique that may introduce biases in datasets used for embedding studies.
 93 Such biases could be a reason for embeddings to skew towards these positions. To mitigate this, our
 94 study employs data augmentation and ablation techniques aimed at isolating and understanding these
 95 effects, thereby ensuring that our findings more accurately reflect model behavior rather than dataset
 96 peculiarities.

97 **2.3 Document chunking for information retrieval tasks**

98 In practical applications, documents often exceed the context length capabilities of embedding models,
 99 necessitating chunking strategies like naive, recursive, or semantic chunking [7, 8]. This process
 100 divides a document into smaller pieces that fit within a model’s context window, then embeds each
 101 chunk separately for insertion into a vector database [13] and downstream use in Retrieval-Augmented
 102 Generation (RAG) [15] tasks. Understanding the impact of chunking on embedding quality and
 103 potential positional biases is essential for optimizing information retrieval strategies.

104 **2.4 Interpretability in High-Dimensional Semantic Spaces**

105 High-dimensional semantic spaces, where text embeddings reside, offer a compact yet expansive
 106 representation of language [5]. Recent advancements in embedding interpretability have demonstrated
 107 that certain dimensions in these spaces may correspond to specific linguistic or semantic features,
 108 such as sentiment or subject matter. However, understanding the contribution of individual sentences
 109 to the overall document embedding requires extending these concepts to more complex structures
 110 beyond single words or phrases. Research in this area has shown that vector operations, such as
 111 adding embeddings, can produce new vectors that represent the semantic meaning of their components
 112 [26]. This property of embedding models can be used in various advanced NLP applications such as
 113 analogy solving to semantic search.

114 **3 Effect of sentence-level positioning in embedding output**

115 We explore how the position and size of a sentence in a text influence a document’s final embedding
 116 vector. Our methodology adapts the needle-in-a-haystack test [11], traditionally used for generative
 117 models in information retrieval [30], to evaluate embedding models.

118 **3.1 Experimental setup**

119 **3.1.1 Insertion of Irrelevant Text**

120 We investigate the impact of adding irrelevant or adversarial text ("needle") to a document. After
 121 inserting the needle, we generate a new embedding for the altered text and compare it to the original
 122 using cosine similarity. We vary the needle’s length (5%, 10%, 25%, 50%, and 100% of the original
 123 text’s token count) and position (beginning, middle, end) across 15 experimental conditions. We use
 124 an extended version of Lorem Ipsum placeholder text [32] that exceeds the length of our longest
 125 datapoint and is structured in paragraph format to achieve a needle with structural similarity to our
 126 data while avoiding a confounding effect on the embedding model.

3.1.2 Removal of Text

In a parallel experiment, we remove portions of text (10%, 25%, 50% of sentences, rounded up) from different positions (beginning, middle, end) in the document. The resulting text is then embedded, and its similarity to the original embedding is measured using cosine similarity.

3.2 Models

We test various open and closed-source models to demonstrate the consistency of our results across multiple popular embedding models.

Closed source models We test Cohere’s Embed-English-v3.0 [24] and OpenAI’s Text-Embedding-3-Small [21], which have context lengths of 512 and 8192 tokens, respectively. For texts exceeding these limits, we truncate from the beginning to fit the models’ context windows. Both models are accessed via their respective APIs.

Open source models Our experiments also involve open-source models, specifically BAAI’s BGE-m3 [3], Nomic AI’s Nomic-Embed-Text-v1.5 [20], and Jina AI’s Jina-Embeddings-v2-Base [12], selected for their performance on the MTEB leaderboard. These models have a maximum context length of 8192 tokens and 137M parameters. Similarly to the closed-source models, we front-truncate if our inputs exceed a model’s context window.

3.3 Datasets

To minimize dataset bias and validate our findings across diverse text types, we selected datasets representing a range of writing categorizations and lengths:

1. **PubMed Publications:** We use PubMed publication abstracts [4] to assess the impact of our ablations on scientific writing. Scientific texts are characterized by their structured presentation of information and specialized vocabulary. Understanding how embeddings capture this complexity can provide insights into their utility in academic and research applications.
2. **Paul Graham Essay Collection:** We analyze over 200 essays written by Paul Graham [10], varying from 400 to 70,000 words. Paul Graham’s essays are known for their thoughtful, reflective style and coherent argument structure, making them ideal for studying how embeddings handle nuanced and complex idea development over long texts.
3. **Amazon Reviews:** Drawn from MTEB’s Amazon Polarity dataset [41], this helps us examine consumer review text. Reviews are direct and opinion-rich, offering a perspective on how embeddings process everyday language and sentiment, which is crucial for applications in consumer analytics.
4. **Argumentative Analysis:** From the BiER benchmark’s Argumentative Analysis (ArguAna) dataset [35], we explore embeddings of formal persuasive writing. This dataset includes well-constructed arguments that are ideal for testing how embeddings capture logical structure and the effectiveness of rhetoric.
5. **Reddit Posts:** More Informal and diverse writing styles can be found on Reddit [9]. This dataset introduces grammar, style, and subject matter diversity into our tests, extending our findings to be more robust and adaptable to a wide range of writing styles.

3.4 Results and discussion

Our results indicate a pronounced drop in similarity when irrelevant text is inserted at the beginning of documents, with less impact seen when additions occur in the middle or end. Specifically, for the BGE-m3 model, we see that the addition of a needle that is equal to 5% of the total content (across the 6 datasets, an average of 1-2 sentences) results in the similarity to reduce to .98, compared to .995 for a needle placed at the end of the input. This is reflected across all datasets tested on, with the largest decrease within the Paul Graham Essay Collection of a similarity score of .85.

This trend intensifies with larger insertions, where inserting text equivalent to 50% of the document decreases similarity to 0.87 at the beginning versus 0.97 at the end, a 10.3% decrease. We find this

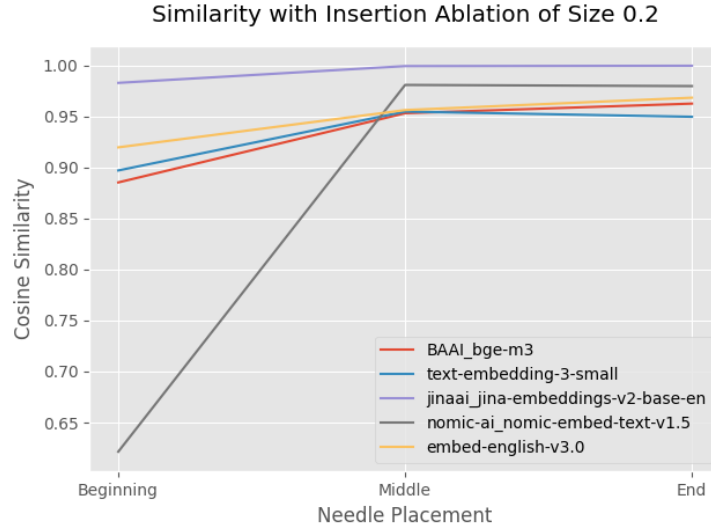


Figure 1: Cosine Similarity vs. Needle Size and Position

175 trend robust to model differences, as all 5 models tested have an average decrease of 7%. Notably,
 176 even significant alterations where half of the text is irrelevant still retain a minimum similarity of
 177 0.7, suggesting an unexpected robustness of the embeddings to extensive modifications. We leave
 178 investigation of this behavior to future work.

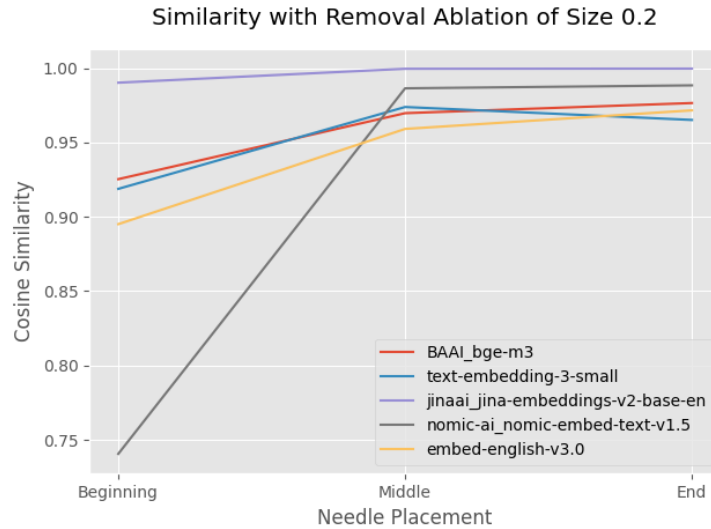


Figure 2: Cosine Similarity vs. Removal Size and Position

179 Similar trends are observed in the removal experiments, with the largest impacts on similarity
 180 occurring when sentences are removed from the beginning. Removing half the sentences from the
 181 beginning of the document leads to a median similarity 10.6% lower than removals from the end,
 182 with no significant difference noted between middle and end removals in contrast to our findings
 183 during the insertion experiments. Interestingly, even a 50% text removal from the middle maintains a
 184 median 95% similarity, corroborating our findings during insertion, where we expect to but fail to
 185 observe a large drop in similarity. The downstream effects of these results are left to future work.

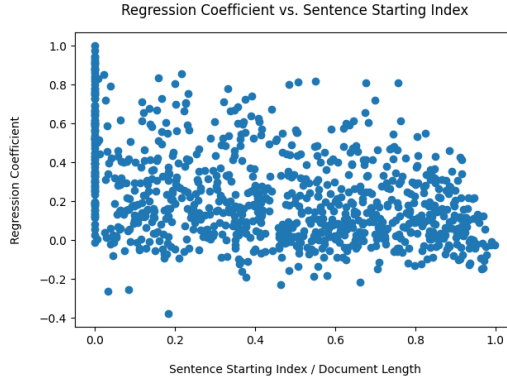


Figure 3: Regression Coefficients vs. Sentence Position

4 Analysis of embedding decomposition

Next, we explore the impact of sentence-level positioning on the final document embedding vector through regression analysis, which offers a more direct method to quantify the contribution of individual sentences to a document’s embedding representation.

4.1 Reconstructing embedding vectors through linear combinations of constituents

To start, we wanted to validate the assumption that the sentence embeddings of a larger document can meaningfully be used as a proxy for the original document embedding [33].

To test this, we wanted to determine how much reconstruction loss we would incur from using an optimal linear combination of sentence embeddings instead of the document embedding. Optimizing for train R^2 , we use Ordinary Least Squares (OLS) regression to reconstruct the document embedding from its sentence embeddings, with the document embedding as our response and each sentence as a predictive datapoint for our regression. Our model choice is notable for its simplicity and the direct interpretability of its coefficients [29], though we acknowledge and check for potential issues posed by OLS, such as multicollinearity. Our regressions use normalized embeddings (L2 norm of 1) to ensure scale invariance [27].

First, we separate our data points into their component sentences by use of punctuation such as periods, and new lines. We then use the embeddings of these sentences to run a regression against the original document embedding. Using a regression on the sentence embeddings from 3 models across all 6 datasets led to R^2 ranging from 0.75 to 1 with an average of 0.876, indicating that approximately 87.6% of the variance in document embeddings can be accounted for by their component embeddings. The MAE across our models and datapoints ranged between 0.001 and 0.01 with an average of 0.0069, suggesting minimal deviation in the reconstructed vectors.

4.2 Analyzing regression coefficients as importance weights

Given the high explanatory power of our regression models, the coefficients given to each sentence (datapoint) in our regression are strong indicators to determine their relative importance to the total document. To standardize our comparisons across documents, we standardized each coefficient vector by its L2 norm. One potential issue with this approach is the presence of negative values coefficients, but these tended to be rare and very low in magnitude when compared to positive coefficients in the same vector.

We judge the importance of a sentence by its regression coefficient. For example, if a regression on a two-sentence document yielded weights 0.8 and 0.6, we conclude that the first sentence is 33.3% more important to the final semantic meaning of the text than the second sentence.

There is a downward trend in coefficient values with increasing sentence position, suggesting a positional bias where earlier sentences generally have a greater impact on the document’s overall

semantic representation. To quantify this observation, we plot regression coefficients against sentence positions over all the documents in our dataset. (Figure 3).

4.3 Embedding positional bias is robust to human-level writing bias

To validate that this observed bias is not solely a byproduct of dataset-specific characteristics, namely human-level writing bias, we conducted additional regression experiments where all sentences from the above pre-processing steps were shuffled and generated embeddings against. Using these new embeddings, remarkably, the results mirrored the original findings, with the randomly selected first sentence in the shuffled document consistently receiving a higher weight, thereby disambiguating our results from potential dataset biases.

More specifically, we expect the weight assigned to the first sentence to follow a uniform weight of $\frac{1}{\text{num_sentences}}$. However, this analysis shows a strong negative correlation ($r = -0.5$) and significant deviations from the expected uniform distribution ($\alpha \ll 0.001$), confirming a systematic positional influence within document embeddings. These findings suggest that the embedding models may inherently prioritize the initial information presented in any text sequence, irrespective of its original position in the document.

5 Isolating the Role of Training Methodology in Model Biases

Embedding models commonly employ truncation strategies due to their limited context windows, directly impacting how documents are processed and understood. Prominent models such as OpenAI’s GPT-3 [2] and Google’s BERT [6] can only process up to 1,024 and 512 tokens, respectively. When documents exceed these limits, content at the end is often discarded, inherently prioritizing the beginning. As shown in the previous experiments, this systematic truncation is not merely a technical necessity but a fundamental design choice that influences model behavior, as the initial sections of documents—typically containing abstracts or executive summaries—are disproportionately represented.

The relative importance on embedding impact for a given point within a model context length can be mathematically described as

$$\text{imp}(t_i) = u(t_i) - \beta(t_i)$$

where t_i represents the number of non-padding tokens encountered at position i , imp represents importance or the relative impact of position i on embedding output, u represents the total number of effective updates at t_i , and β represents the total number of decay opportunities. $t_i \in [0, N]$ where N is the number of training examples. An effective update is defined as a single model update based on a non-padding token in that position, whereas a decay opportunities is a model update being either empty or having a padding token in that position.

Following traditional truncation methods, positions earlier in the context window will be used more often than those at the end. We can model this as a monotonically decreasing function as the number of effective updates decrease as i increases. Due to this implicit bias, the relative importance ($\text{imp}(t_1) \geq \text{imp}(t_2) \geq \dots \geq \text{imp}(t_N)$) of earlier positions on embedding output will always be greater than or equal to the positions later in context.

Although this monotonic impact on position can theoretically be removed by maintaining an equal number of effective updates throughout the context, it is unknown what the impacts on computational costs, and model performance would be. Completing pre-training with this bias in mind will require considerable research to full understand the impacts, leading us to believe that this bias will continue in future models.

5.1 Is it possible to remove positional bias in post-training?

Following our theory on bias learned through the pre-training process, we experiment with smaller, cost-effective fine-tuning methods to remove this bias [33]. We do this by fine-tune models to use data without the front-truncation, yet still holds similar semantic meaning to the initial data points.

We propose a new framework, Position-Aware Data Sampling (PADS), where subsets of data points are randomly sampled based on input position, to solve this positional bias. The method augments the

data by inputting training points that would normally be truncated, and randomly selecting subsets of each data point based on position away from the beginning of the original input. For example, instead of front-truncating 50% the length of a given example, we select uniformly a token position from 0 to $n/2$, where n is the token length of the data point.

In our fine-tuning experiments, we create positive pairs by sampling from each original twice. For negative pairs, we sample once from both the original and another random data point in the dataset. Using these pairs, we use contrastive loss to fine-tune the model towards our goal. We follow these steps for three datasets and using this to fine-tune BAAI’s BGE-small-en-v1.5. The three datasets included are the Paul Graham Essay Collection, PubMed Publications, and Amazon Reviews. We sample a maximum of 20% from each dataset, selecting 50 examples for the Paul Graham dataset and 225 for the other two datasets. Following the procedure above, we select 50% of each original datapoint and create a positive and negative pair from each, resulting in an augmented dataset of 1000 examples. We use cosine similarity within our contrastive loss function, and then use this with the Adam optimizer for three epochs.

Table 1: Average Cosine Similarity between Original and Ablated Inputs

Model	Beginning	Middle	End
Original	0.923	0.979	0.983
Finetuned	0.984	0.993	0.993
Percent Improvement	6.1%	1.4%	1.0%
Original (external datasets)	0.920	0.978	0.982
Finetuned (external datasets)	0.988	0.995	0.995
Percent Improvement	6.8%	1.7%	1.3%

With this new method, we have been able to effectively remove positional bias and improve similarity metrics to levels similar to when ablations are put in positions different from the beginning. The new model has been able to reduce bias by 6.9% with insertion needles, and 6.1% averaged between insertion and removal ablations. This work suggests that models can learn to fix its early positional bias by sampling the subset position of the input it is training on.

6 Future work

Future work incorporating our findings can focus on three distinct directions:

Alternative Evaluation Metrics Exploring alternative evaluation metrics beyond cosine similarity is essential to assess the effectiveness of embedding models. Future research should consider metrics such as Word Mover’s Distance (WMD) [14] for capturing semantic similarity, BERTScore [39] for evaluating contextual alignment, and NDCG (Normalized Discounted Cumulative Gain) [37] for ranking quality in information retrieval tasks. Additionally, task-specific metrics like classification F1-score, BLEU [22] for translation quality, and ROUGE [17] for summarization accuracy can provide deeper insights into model performance. These specific metrics can offer a more detailed understanding of how well embeddings preserve semantics and perform across various downstream applications.

Model Architecture and Training Process Innovations Given our findings, model creators can employ innovative training techniques such as sentence shuffling or random truncation of long texts during the embedding training process. These methods can help mitigate positional biases and enhance model robustness. Since embedding models use contrastive loss [19] rather than classification loss like generative models, careful consideration is needed to determine the best way to compare these ablations with their original texts. This could involve designing new contrastive learning objectives that account for the positional integrity of the input text. Additionally, incorporating architectural modifications, such as advanced attention mechanisms or positional encodings [23], can further reduce biases and improve the models’ ability to handle long-context inputs. Experimentation with these innovations can lead to embedding models that are more resilient to variations in input structure, thereby enhancing their performance across a wide range of downstream tasks.

Improved Document Chunking and Impact on Downstream Information Retrieval Tasks

Currently, document chunking does not typically take text structure into account. Better chunking strategies can focus on isolating important sentences in the text from less useful content and using these as breaking points. For longer paragraphs without a clear partition point, special attention will need to be given to the paragraph content to determine where to set a breaking point that leads to the most useful chunk encoding for downstream tasks. Enhanced contextualization techniques, such as dynamic chunking strategies, can be developed to preserve semantic coherence and context, improving the overall quality of embeddings.

Future work should focus on how these improved chunking techniques impact downstream information retrieval tasks. By aligning chunking strategies with the inherent biases observed in our study, we can create more effective embeddings for tasks such as search engine optimization, recommendation systems, and document summarization. Studying the biases in embedding models and how they influence downstream performance on information retrieval benchmarks is crucial. Evaluating various chunking strategies, including those discussed here, can reveal how different approaches affect the retrieval accuracy and relevance of results. This integrated approach will provide a deeper understanding of how to optimize embeddings for real-world applications, ensuring that the enhancements in chunking directly contribute to better performance in information retrieval tasks.

7 Conclusion

Our study reveals a positional bias in embedding models, where sentences at the beginning of a document disproportionately influence the embedding output. This finding is consistent across various models with differing context sizes and diverse datasets, evident in both text insertion and removal experiments. We further support this finding with regression analysis to further quantify the impact of biases towards a given embedding. Then, potential theories on the source of this bias are discussed, mainly that this trend is intrinsic to the models' training methodologies of truncation rather than dataset peculiarities themselves.

Implicit bias within embeddings models hinder performance in many critical applications across information retrieval. One main avenue of societal impact from this is within document search in both cultural and business contexts. In document retrieval in sensitive political topics, reducing bias within these models improve the ability to maintain relevant information towards a topic. However, negative impacts of this work include the spread of the knowledge of this bias where a bad actor can use this knowledge to retrieve particular non-optimal results aligned with their adversarial goals.

These insights suggest the need for revised training strategies that mitigate positional biases to achieve balanced semantic representations. Although our initial experiments show an example of reducing this bias through fine-tuning, more research must be conducted to have robust techniques to remove the bias at hand.

8 Limitations

We have limited our claims to using 6 models with 6 datasets, but this can be extended to look at positional bias for more models and datasets to eliminate implicit bias from the experimental design. The fine-tuning method can be adopted to pre-training method to look at the full effects and performance impacts, outside the post-training context.

References

- [1] S. Barnett, S. Kurniawan, S. Thudumu, Z. Brannelly, and M. Abdelrazek. Seven failure points when engineering a retrieval augmented generation system, 2024.
- [2] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners, 2020.

- [3] J. Chen, S. Xiao, P. Zhang, K. Luo, D. Lian, and Z. Liu. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation, 2024.
- [4] A. Cohan, F. Dernoncourt, D. S. Kim, T. Bui, S. Kim, W. Chang, and N. Goharian. A discourse-aware attention model for abstractive summarization of long documents, 2018.
- [5] G. Dar, M. Geva, A. Gupta, and J. Berant. Analyzing transformers in embedding space, 2023.
- [6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.
- [7] W. Fei, X. Niu, P. Zhou, L. Hou, B. Bai, L. Deng, and W. Han. Extending context window of large language models via semantic compression, 2023.
- [8] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, M. Wang, and H. Wang. Retrieval-augmented generation for large language models: A survey, 2024.
- [9] G. Geigle, N. Reimers, A. Rücklé, and I. Gurevych. Tweac: Transformer with extendable qa agent classifiers, 2021.
- [10] S. Goel. paul graham essays (revision 0c7155a), 2024. URL https://huggingface.co/datasets/sgoel9/paul_graham_essays.
- [11] N. M. Guerreiro, E. Voita, and A. F. T. Martins. Looking for a needle in a haystack: A comprehensive study of hallucinations in neural machine translation, 2023.
- [12] M. Günther, J. Ong, I. Mohr, A. Abdessalem, T. Abel, M. K. Akram, S. Guzman, G. Mastrapas, S. Sturua, B. Wang, M. Werk, N. Wang, and H. Xiao. Jina embeddings 2: 8192-token general-purpose text embeddings for long documents, 2024.
- [13] J. Johnson, M. Douze, and H. Jégou. Billion-scale similarity search with gpus, 2017.
- [14] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger. From word embeddings to document distances. In F. Bach and D. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 957–966, Lille, France, 07–09 Jul 2015. PMLR. URL <https://proceedings.mlr.press/v37/kusnerb15.html>.
- [15] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. tau Yih, T. Rocktäschel, S. Riedel, and D. Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks, 2021.
- [16] X. Li and J. Li. Angle-optimized text embeddings, 2024.
- [17] C.-Y. Lin. Rouge: a package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004)*, pages 74–81, July 2004.
- [18] 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, 2019.
- [19] A. Mnih and Y. W. Teh. A fast and simple algorithm for training neural probabilistic language models, 2012.
- [20] Z. Nussbaum, J. X. Morris, B. Duderstadt, and A. Mulyar. Nomic embed: Training a reproducible long context text embedder, 2024.
- [21] OpenAI. New embedding models and api updates, jan 2024. URL <https://openai.com/index/new-embedding-models-and-api-updates/>. Accessed: 2024-05-18.
- [22] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002)*, pages 311–318, 2002. URL <http://www.aclweb.org/anthology/P02-1040.pdf>.

- [23] O. Press, N. A. Smith, and M. Lewis. Train short, test long: Attention with linear biases enables input length extrapolation, 2022.
- [24] N. Reimers. Introducing embed v3, nov 2023. URL <https://cohere.com/blog/introducing-embed-v3>. Accessed: 2024-05-18.
- [25] N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks, 2019.
- [26] L. K. Senel, I. Utlu, V. Yucesoy, A. Koc, and T. Cukur. Semantic structure and interpretability of word embeddings. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(10):1769–1779, Oct. 2018. ISSN 2329-9304. doi: 10.1109/taslp.2018.2837384. URL <http://dx.doi.org/10.1109/TASLP.2018.2837384>.
- [27] H. Steck, C. Ekanadham, and N. Kallus. Is cosine-similarity of embeddings really about similarity? In *Companion Proceedings of the ACM on Web Conference 2024*, WWW ’24. ACM, May 2024. doi: 10.1145/3589335.3651526. URL <http://dx.doi.org/10.1145/3589335.3651526>.
- [28] J. Su, Y. Lu, S. Pan, A. Murtadha, B. Wen, and Y. Liu. Roformer: Enhanced transformer with rotary position embedding, 2023.
- [29] T. Słoczyński. Interpreting ols estimands when treatment effects are heterogeneous: Smaller groups get larger weights, 2020.
- [30] G. Team, M. Reid, N. Savinov, D. Teplyashin, Dmitry, Lepikhin, T. Lillicrap, J. baptiste Alayrac, R. Soricut, A. Lazaridou, O. Firat, J. Schrittwieser, I. Antonoglou, R. Anil, S. Borgeaud, A. Dai, K. Millican, E. Dyer, M. Glaese, T. Sottiaux, B. Lee, F. Viola, M. Reynolds, Y. Xu, J. Molloy, J. Chen, M. Isard, P. Barham, T. Hennigan, R. McIlroy, M. Johnson, J. Schalkwyk, E. Collins, E. Rutherford, E. Moreira, K. Ayoub, M. Goel, C. Meyer, G. Thornton, Z. Yang, H. Michalewski, Z. Abbas, N. Schucher, A. Anand, R. Ives, J. Keeling, K. Lenc, S. Haykal, S. Shakeri, P. Shyam, A. Chowdhery, R. Ring, S. Spencer, E. Sezener, L. Vilnis, O. Chang, N. Morioka, G. Tucker, C. Zheng, O. Woodman, N. Attaluri, T. Kocisky, E. Eltyshev, X. Chen, T. Chung, V. Selo, S. Brahma, P. Georgiev, A. Slone, Z. Zhu, J. Lottes, S. Qiao, B. Caine, S. Riedel, A. Tomala, M. Chadwick, J. Love, P. Choy, S. Mittal, N. Houlsby, Y. Tang, M. Lamm, L. Bai, Q. Zhang, L. He, Y. Cheng, P. Humphreys, Y. Li, S. Brin, A. Cassirer, Y. Miao, L. Zilka, T. Tobin, K. Xu, L. Proleev, D. Sohn, A. Magni, L. A. Hendricks, I. Gao, S. Ontanon, O. Bunyan, N. Byrd, A. Sharma, B. Zhang, M. Pinto, R. Sinha, H. Mehta, D. Jia, S. Caelles, A. Webson, A. Morris, B. Roelofs, Y. Ding, R. Strudel, X. Xiong, M. Ritter, M. Dehghani, R. Chaabouni, A. Karmakar, G. Lai, F. Mentzer, B. Xu, Y. Li, Y. Zhang, T. L. Paine, A. Goldin, B. Neyshabur, K. Baumli, A. Levskaya, M. Laskin, W. Jia, J. W. Rae, K. Xiao, A. He, S. Giordano, L. Yagati, J.-B. Lespiau, P. Natsev, S. Ganapathy, F. Liu, D. Martins, N. Chen, Y. Xu, M. Barnes, R. May, A. Vezer, J. Oh, K. Franko, S. Bridgers, R. Zhao, B. Wu, B. Mustafa, S. Sechrist, E. Parisotto, T. S. Pillai, C. Larkin, C. Gu, C. Sorokin, M. Krikun, A. Guseynov, J. Landon, R. Datta, A. Pritzel, P. Thacker, F. Yang, K. Hui, A. Hauth, C.-K. Yeh, D. Barker, J. Mao-Jones, S. Austin, H. Sheahan, P. Schuh, J. Svensson, R. Jain, V. Ramasesh, A. Briukhov, D.-W. Chung, T. von Glehn, C. Butterfield, P. Jhakra, M. Wiethoff, J. Frye, J. Grimstad, B. Changpinyo, C. L. Lan, A. Bortsova, Y. Wu, P. Voigtlaender, T. Sainath, S. Gu, C. Smith, W. Hawkins, K. Cao, J. Besley, S. Srinivasan, M. Omernick, C. Gaffney, G. Surita, R. Burnell, B. Damoc, J. Ahn, A. Brock, M. Pajarskas, A. Petrushkina, S. Noury, L. Blanco, K. Swersky, A. Ahuja, T. Avrahami, V. Misra, R. de Liedekerke, M. Iinuma, A. Polozov, S. York, G. van den Driessche, P. Michel, J. Chiu, R. Blevins, Z. Gleicher, A. Recasens, A. Rustemi, E. Gribovskaya, A. Roy, W. Gworek, S. M. R. Arnold, L. Lee, J. Lee-Thorp, M. Maggioni, E. Piqueras, K. Badola, S. Vikram, L. Gonzalez, A. Baddepudi, E. Senter, J. Devlin, J. Qin, M. Azzam, M. Trebacz, M. Polacek, K. Krishnakumar, S. yiin Chang, M. Tung, I. Penchev, R. Joshi, K. Olszewska, C. Muir, M. Wirth, A. J. Hartman, J. Newlan, S. Kashem, V. Bolina, E. Dabir, J. van Amersfoort, Z. Ahmed, J. Cobon-Kerr, A. Kamath, A. M. Hrafnkelsson, L. Hou, I. Mackinnon, A. Frechette, E. Noland, X. Si, E. Taropa, D. Li, P. Crone, A. Gulati, S. Cevey, J. Adler, A. Ma, D. Silver, S. Tokumine, R. Powell, S. Lee, K. Vodrahalli, S. Hassan, D. Mincu, A. Yang, N. Levine, J. Brennan, M. Wang, S. Hodkinson, J. Zhao, J. Lipschultz, A. Pope, M. B. Chang, C. Li, L. E. Shafey, M. Paganini, S. Douglas, B. Bohnet, F. Pardo, S. Odoom, M. Rosca, C. N.

dos Santos, K. Soparkar, A. Guez, T. Hudson, S. Hansen, C. Asawaroengchai, R. Addanki, T. Yu, W. Stokowiec, M. Khan, J. Gilmer, J. Lee, C. G. Bostock, K. Rong, J. Caton, P. Pejman, F. Pavetic, G. Brown, V. Sharma, M. Lučić, R. Samuel, J. Djolonga, A. Mandhane, L. L. Sjöstrand, E. Buchatskaya, E. White, N. Clay, J. Jiang, H. Lim, R. Hemsley, Z. Cankara, J. Labanowski, N. D. Cao, D. Steiner, S. H. Hashemi, J. Austin, A. Gergely, T. Blyth, J. Stanton, K. Shivakumar, A. Siddhant, A. Andreassen, C. Araya, N. Sethi, R. Shivanna, S. Hand, A. Bapna, A. Khodaei, A. Miech, G. Tanzer, A. Swing, S. Thakoor, L. Aroyo, Z. Pan, Z. Nado, J. Sygnowski, S. Winkler, D. Yu, M. Saleh, L. Maggiore, Y. Bansal, X. Garcia, M. Kazemi, P. Patil, I. Dasgupta, I. Barr, M. Giang, T. Kagohara, I. Danihelka, A. Marathe, V. Feinberg, M. Elhawaty, N. Ghelani, D. Horgan, H. Miller, L. Walker, R. Tanburn, M. Tariq, D. Shrivastava, F. Xia, Q. Wang, C.-C. Chiu, Z. Ashwood, K. Baatarsukh, S. Samangoeei, R. L. Kaufman, F. Alcober, A. Stjerngren, P. Komarek, K. Tsihlias, A. Boral, R. Comanescu, J. Chen, R. Liu, C. Welty, D. Bloxwich, C. Chen, Y. Sun, F. Feng, M. Mauger, X. Dotiwalla, V. Hellendoorn, M. Sharman, I. Zheng, K. Haridasan, G. Barth-Maron, C. Swanson, D. Rogozinińska, A. Andreev, P. K. Rubenstein, R. Sang, D. Hurt, G. Elsayed, R. Wang, D. Lacey, A. Ilić, Y. Zhao, A. Iwanicki, A. Lince, A. Chen, C. Lyu, C. Lebsack, J. Griffith, M. Gaba, P. Sandhu, P. Chen, A. Koop, R. Rajwar, S. H. Yeganeh, S. Chang, R. Zhu, S. Radpour, E. Davoodi, V. I. Lei, Y. Xu, D. Toyama, C. Segal, M. Wicke, H. Lin, A. Bulanov, A. P. Badia, N. Rakićević, P. Sprechmann, A. Filos, S. Hou, V. Campos, N. Kassner, D. Sachan, M. Fortunato, C. Iwuanyanwu, V. Nikolaev, B. Lakshminarayanan, S. Jazayeri, M. Varadarajan, C. Tekur, D. Fritz, M. Khalman, D. Reitter, K. Dasgupta, S. Sarcar, T. Ornduff, J. Snider, F. Huot, J. Jia, R. Kemp, N. Trdin, A. Vijayakumar, L. Kim, C. Angermueller, L. Lao, T. Liu, H. Zhang, D. Engel, S. Greene, A. White, J. Austin, L. Taylor, S. Ashraf, D. Liu, M. Georgaki, I. Cai, Y. Kulizhskaya, S. Goenka, B. Saeta, Y. Xu, C. Frank, D. de Cesare, B. Robenek, H. Richardson, M. Alnahlawi, C. Yew, P. Ponnappalli, M. Tagliasacchi, A. Korchemniy, Y. Kim, D. Li, B. Rosgen, K. Levin, J. Wiesner, P. Banzal, P. Srinivasan, H. Yu, Çağlar Ünlü, D. Reid, Z. Tung, D. Finchelstein, R. Kumar, A. Elisseff, J. Huang, M. Zhang, R. Aguilar, M. Giménez, J. Xia, O. Dousse, W. Gierke, D. Yates, K. Jalan, L. Li, E. Latorre-Chimoto, D. D. Nguyen, K. Durden, P. Kallakuri, Y. Liu, M. Johnson, T. Tsai, A. Talbert, J. Liu, A. Neitz, C. Elkind, M. Selvi, M. Jasarevic, L. B. Soares, A. Cui, P. Wang, A. W. Wang, X. Ye, K. Kallarakal, L. Loher, H. Lam, J. Broder, D. Holtmann-Rice, N. Martin, B. Ramadhana, M. Shukla, S. Basu, A. Mohan, N. Fernando, N. Fidel, K. Paterson, H. Li, A. Garg, J. Park, D. Choi, D. Wu, S. Singh, Z. Zhang, A. Globerson, L. Yu, J. Carpenter, F. de Chaumont Quitry, C. Radebaugh, C.-C. Lin, A. Tudor, P. Shroff, D. Garmon, D. Du, N. Vats, H. Lu, S. Iqbal, A. Yakubovich, N. Tripuraneni, J. Manyika, H. Qureshi, N. Hua, C. Ngani, M. A. Raad, H. Forbes, J. Stanway, M. Sundararajan, V. Ungureanu, C. Bishop, Y. Li, B. Venkatraman, B. Li, C. Thornton, S. Scellato, N. Gupta, Y. Wang, I. Tenney, X. Wu, A. Shenoy, G. Carvajal, D. G. Wright, B. Bariach, Z. Xiao, P. Hawkins, S. Dalmia, C. Farabet, P. Valenzuela, Q. Yuan, A. Agarwal, M. Chen, W. Kim, B. Hulse, N. Dukkupati, A. Paszke, A. Bolt, K. Choo, J. Beattie, J. Prendki, H. Vashisht, R. Santamaria-Fernandez, L. C. Cobo, J. Wilkiewicz, D. Madras, A. Elqursh, G. Uy, K. Ramirez, M. Harvey, T. Liechty, H. Zen, J. Seibert, C. H. Hu, A. Khorlin, M. Le, A. Aharoni, M. Li, L. Wang, S. Kumar, N. Casagrande, J. Hoover, D. E. Badawy, D. Soergel, D. Vnukov, M. Mieczkowski, J. Simsa, P. Kumar, T. Sellam, D. Vlasic, S. Daruki, N. Shabat, J. Zhang, G. Su, J. Zhang, J. Liu, Y. Sun, E. Palmer, A. Ghaffarkhah, X. Xiong, V. Cotruta, M. Fink, L. Dixon, A. Sreevatsa, A. Goedeckemeyer, A. Dimitriev, M. Jafari, R. Crocker, N. FitzGerald, A. Kumar, S. Ghemawat, I. Philips, F. Liu, Y. Liang, R. Sterneck, A. Repina, M. Wu, L. Knight, M. Georgiev, H. Lee, H. Askham, A. Chakladar, A. Louis, C. Crous, H. Cate, D. Petrova, M. Quinn, D. Owusu-Afriyie, A. Singhal, N. Wei, S. Kim, D. Vincent, M. Nasr, C. A. Choquette-Choo, R. Tojo, S. Lu, D. de Las Casas, Y. Cheng, T. Bolukbasi, K. Lee, S. Fatehi, R. Ananthanarayanan, M. Patel, C. Kaed, J. Li, S. R. Belle, Z. Chen, J. Konzelmann, S. Pöder, R. Garg, V. Koverkathu, A. Brown, C. Dyer, R. Liu, A. Nova, J. Xu, A. Walton, A. Parrish, M. Epstein, S. McCarthy, S. Petrov, D. Hassabis, K. Kavukcuoglu, J. Dean, and O. Vinyals. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context, 2024.

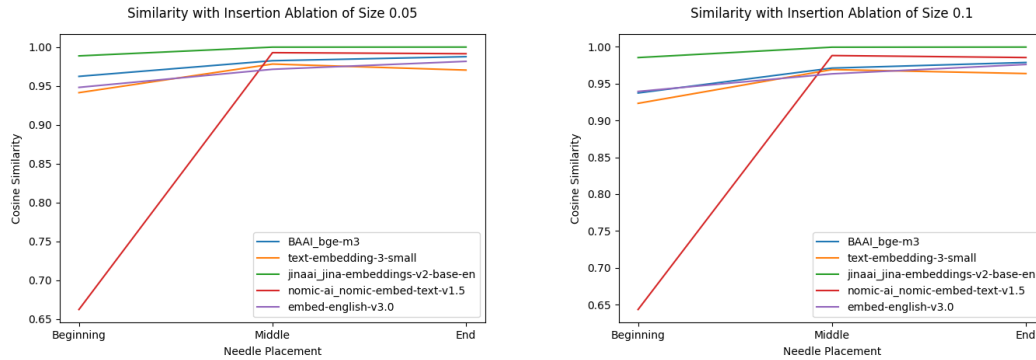
[31] G. Tennenholtz, Y. Chow, C.-W. Hsu, J. Jeong, L. Shani, A. Tulepbergenov, D. Ramachandran, M. Mladenov, and C. Boutilier. Demystifying embedding spaces using large language models, 2024.

[32] R. C. Timmer, D. Liebowitz, S. Nepal, and S. Kanhere. Tsm: Measuring the enticement of honeyfiles with natural language processing, 2022.

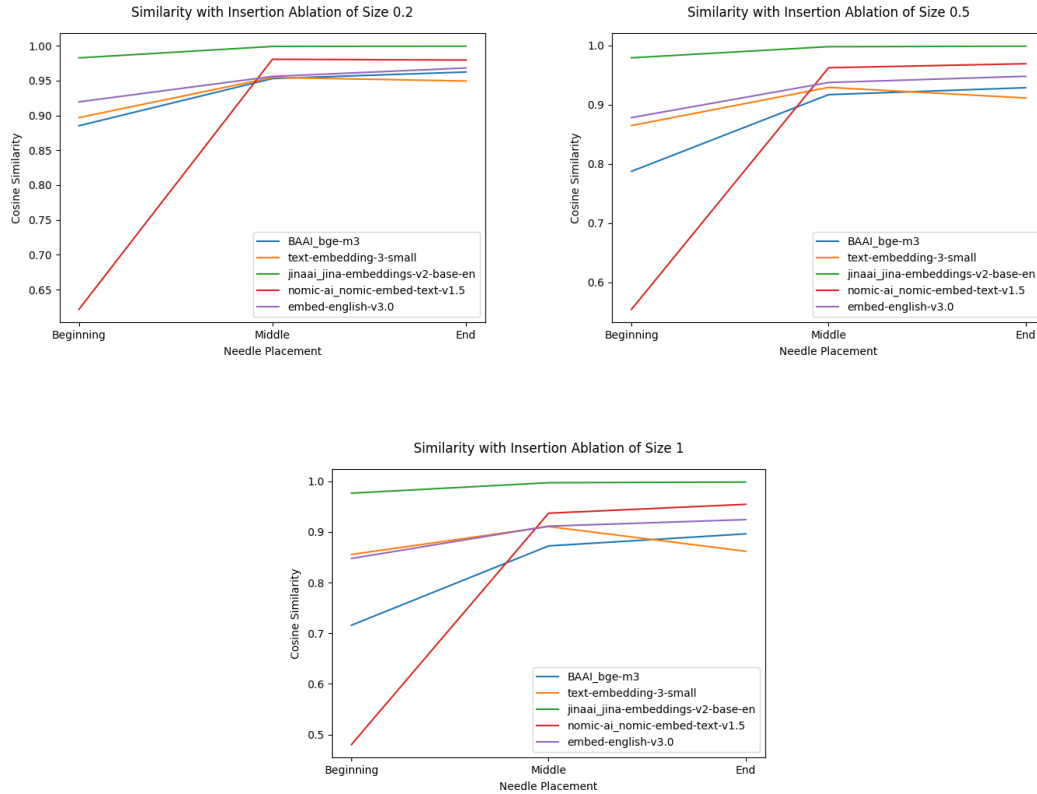
- [33] H. Tsukagoshi, R. Sasano, and K. Takeda. Comparison and combination of sentence embeddings derived from different supervision signals, 2022.
- [34] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need, 2023.
- [35] H. Wachsmuth, S. Syed, and B. Stein. Retrieval of the best counterargument without prior topic knowledge. In I. Gurevych and Y. Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 241–251, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1023. URL <https://aclanthology.org/P18-1023>.
- [36] L. Wang, N. Yang, X. Huang, L. Yang, R. Majumder, and F. Wei. Improving text embeddings with large language models, 2024.
- [37] Y. Wang, L. Wang, Y. Li, D. He, T.-Y. Liu, and W. Chen. A theoretical analysis of ndcg type ranking measures, 2013.
- [38] S. Xiao, Z. Liu, P. Zhang, and N. Muennighoff. C-pack: Packaged resources to advance general chinese embedding, 2023.
- [39] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi. Bertscore: Evaluating text generation with bert, 2020.
- [40] T. Zhang, S. G. Patil, N. Jain, S. Shen, M. Zaharia, I. Stoica, and J. E. Gonzalez. Raft: Adapting language model to domain specific rag, 2024.
- [41] X. Zhang, J. Zhao, and Y. LeCun. Character-level convolutional networks for text classification, 2016.
- [42] D. Zhu, L. Wang, N. Yang, Y. Song, W. Wu, F. Wei, and S. Li. Longembed: Extending embedding models for long context retrieval, 2024.

A Cosine similarities across insertion ablation sizes and datasets

The following are the results of running insertion and removal ablations of given sizes on input examples. These are the results of the average cosine similarity across all datasets.



B Cosine similarities across deletion of ablation sizes and datasets



NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: We describe the main sections in the paper and summarize them to fit into the abstract guidelines.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

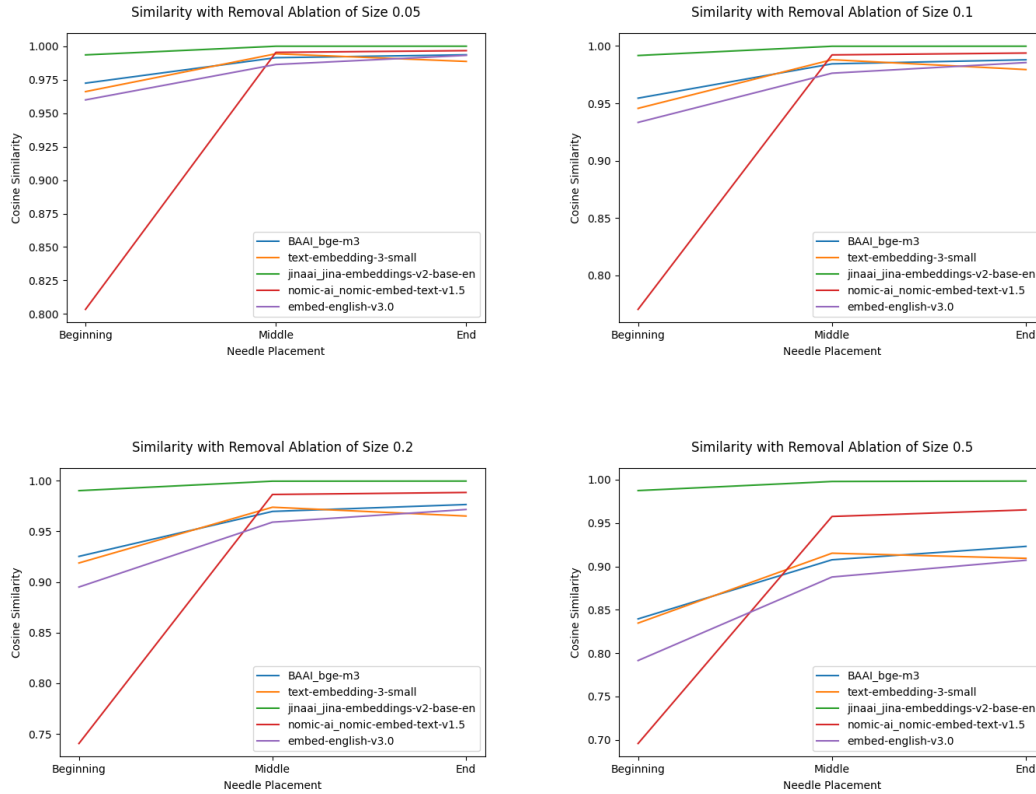
2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: We have a separate limitations section describing the limitation of every major finding within our paper.

Guidelines:



- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory Assumptions and Proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: We do not provide any theoretical result, and only provide an intuition for explaining our work.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental Result Reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Yes, we posted supplementary code to reproduce our work.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Yes, we submit code to reproduce our findings.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental Setting/Details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Yes, we specify all training details relevant to the experiments such that they can be reproduced easily.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment Statistical Significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Yes, we provide error bars and proper evidence for reviewers to see the variance of the data within our analysis.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments Compute Resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Yes, we describe our compute used, as well as these differences between models that we tested.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code Of Ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: The authors have reviewed the ethics guidelines and have taken all steps to ensure they are being followed.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader Impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The authors discuss the effects of this work in the later parts of the paper.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper does not release data or models that have a high risk of misuse.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Code has been marked with the author, unless it was originally created by the authors themselves.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New Assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [\[Yes\]](#)

Justification: The author releases the code to be able to easily reproduce results.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and Research with Human Subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [\[NA\]](#)

Justification: The experiments do not use human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [\[NA\]](#)

Justification: The authors do not use human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- 847 • We recognize that the procedures for this may vary significantly between institutions
848 and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the
849 guidelines for their institution.
- 850 • For initial submissions, do not include any information that would break anonymity (if
851 applicable), such as the institution conducting the review.