To MRL or not to MRL: Random Vector Truncation is as Effective as Matryoshka Representation Learning, Except With Heavy Truncation

Sotaro Takeshita[1], Yurina Takeshita[3], Simone Paolo Ponzetto[1], Daniel Ruffinelli[1,2]
[1]Data and Web Science Group, University of Mannheim, Germany
[2]NEC Laboratories Europe, Heidelberg, Germany
[3]Independent Researcher
{sotaro.takeshita, ponzetto, druffinelli}@uni-mannheim.de

Text embeddings are widely used in many NLP tasks, from retrieval (Huang et al., 2020) to recommender systems (Zhao et al., 2023) to many others (Zhao et al., 2024). To reduce costs while retaining performance in such tasks, the use of small but well-performing embeddings is desirable. E.g. in text retrieval, bi-encoder models with less than 1B parameters are a common choice for first-stage retrieval over an entire collection of documents (Zhao et al., 2024).

To provide more flexibility in this regard, Matryoshka Representation Learning (MRL) (Kusupati et al., 2022) is an approach that adds additional terms to the training objective so that text encoders are able to provide representations that are simultaneously useful at various embedding sizes, available by simply truncating the resulting vectors at sizes pre-determined during training. The use of MRL is widely adopted, with some of the latest models providing this benefit out-of-the-box, such as Jina-Embeddings-V5 (Akram et al., 2026), Qwen3-Embedding (Zhang et al., 2025), and EmbeddingGemma (Vera et al., 2025). At the same time, recent works have empirically shown that randomly truncating text embeddings is a promising approach to obtaining vectors of reduced size, as the impact in performance is minimal unless vector sizes are reduced by at least 70% (Tsukagoshi and Sasano, 2025; Takeshita et al., 2025; Inkiriwang et al., 2025). However, none of these studies compared random truncation to MRL, the industry standard for truncation, so it is unclear how the two methods compare as effective ways to trade off performance for runtime and memory costs.

In this project, we compare performance at various vector truncation levels using models trained with and without MRL. Our results show that unless heavily truncating embeddings, i.e. reducing their size by about 80%, randomly truncated embeddings of non-MRL models are at least competitive with, and often superior in performance compared to models trained with MRL. We found this across 8 open models (e.g. Fig. 1, top), 10 text encoders we trained from pre-trained language models (e.g. Fig. 1 bottom) and 24 embedding-based downstream tasks for classification and retrieval.

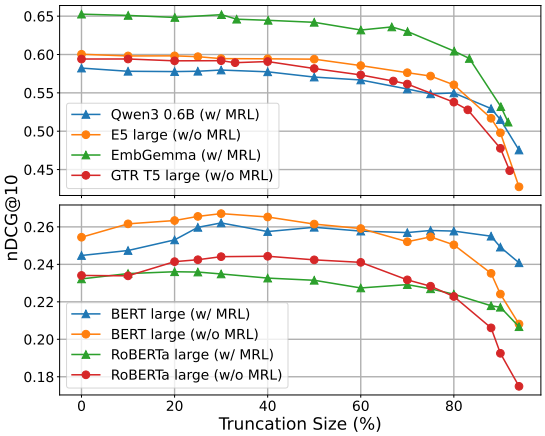

Figure 1: (Top) Robustness of open text encoders as truncation levels increase looks the same whether trained with or without MRL. (Bottom) When models differ only in their use of MRL, truncation on non-MRL models is superior unless heavy truncation is applied.

These results suggest that random truncation is indeed a highly effective method for reducing many of the costs associated with text encoders, and that the additional cost of training with MRL is only beneficial in scenarios where reducing embedding size by about 80% or more is desirable. Further evidence of this is the fact that while MRL requires that truncation dimensions be chosen before training, MRL behaves the same as random truncation when truncating outside of those selected dimensions, suggesting that truncation robustness may largely not come from MRL but be inherent to the learned representations.

References

Mohammad Kalim Akram, Saba Sturua, Nastia Havriushenko, Quentin Herreros, Michael Günther, Maximilian Werk, and Han Xiao. 2026. jina-embeddings-v5-text: Task-targeted embedding distillation. *arXiv preprint arXiv:2602.15547*.

Jui-Ting Huang, Ashish Sharma, Shuying Sun, Li Xia, David Zhang, Philip Pronin, Janani Padmanabhan, Giuseppe Ottaviano, and Linjun Yang. 2020. Embedding-based retrieval in facebook search. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2553–2561.

Nathan Inkiriwang, Necva Bölücü, Garth Tarr, and Maciej Rybinski. 2025. Do we really need all those dimensions? an intrinsic evaluation framework for compressed embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 13305–13323.

Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham Kakade, Prateek Jain, and 1 others. 2022. Matryoshka representation learning. *Advances in Neural Information Processing Systems*, 35:30233–30249.

Sotaro Takeshita, Yurina Takeshita, Daniel Ruffinelli, and Simone Paolo Ponzetto. 2025. Randomly removing 50% of dimensions in text embeddings has minimal impact on retrieval and classification tasks. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 27693–27714.

Hayato Tsukagoshi and Ryohei Sasano. 2025. Redundancy, isotropy, and intrinsic dimensionality of prompt-based text embeddings. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 25915–25930, Vienna, Austria. Association for Computational Linguistics.

Henrique Schechter Vera, Sahil Dua, Biao Zhang, Daniel Salz, Ryan Mullins, Sindhu Raghuram Panyam, Sara Smoot, Iftekhar Naim, Joe Zou, Feiyang Chen, and 1 others. 2025. Embeddinggemma: Powerful and lightweight text representations. *arXiv preprint arXiv:2509.20354*.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *Preprint*, arXiv:2506.05176.

Wayne Xin Zhao, Jing Liu, Ruiyang Ren, and Ji-Rong Wen. 2024. Dense text retrieval based on pretrained language models: A survey. *ACM Transactions on Information Systems*, 42(4):1–60.

Xiangyu Zhao, Maolin Wang, Xinjian Zhao, Jiansheng Li, Shucheng Zhou, Dawei Yin, Qing Li, Jiliang Tang, and Ruocheng Guo. 2023. Embedding in recommender systems: A survey. *arXiv preprint arXiv:2310.18608*.