Map of Encoders – Mapping Sentence Encoders using Quantum Relative Entropy

Gaifan Zhang[†] **Danushka Bollegala**[†,◇*]
[†]University of Liverpool [◇]Amazon
{sggzhan8,danushka}@liverpool.ac.uk

Abstract

We propose a method to compare and visualise sentence encoders at scale by creating a **map of encoders** where each sentence encoder is represented in relation to the other sentence encoders. Specifically, we first represent a sentence encoder using an embedding matrix of a sentence set, where each row corresponds to the embedding of a sentence. Next, we compute the Pairwise Inner Product (PIP) matrix for a sentence encoder using its embedding matrix. Finally, we create a feature vector for each sentence encoder that reflects its Quantum Relative Entropy (QRE) with respect to a unit base encoder. We construct a map of encoders covering 1101 publicly available sentence encoders, providing a new perspective of the landscape of the pre-trained sentence encoders. Our map accurately reflects various relationships between encoders, where encoders with similar attributes are proximally located on the map. Moreover, our encoder feature vectors can be used to accurately infer downstream task performance of the encoders, such as in retrieval and clustering tasks, demonstrating the correctness of our map.[1]

Figure 1: Map of Encoders. Top: Map for 1101 sentence encoders, visualised by t-SNE and coloured by the encoder type. Bottom: A 30% zoomed-in view of the dotted area showing the top-7 nearest neighbours for bge-base and bge-large. Nearest neighbours belong to the same primary architecture and are closely located.

1 Introduction

A large number of sentence encoders have been developed following different training algorithms (Gao et al., 2021; Chen et al., 2023; Xu et al., 2023; Li et al., 2020) and fine-tuned on diverse tasks and datasets. As at December 2025, 17,515 sentence encoders are published in Hugging Face Hub, and their global landscape remains unclear. This is a significant blocker when improving or selecting sentence encoders for NLP applications. Moreover, there is no universally agreed-upon metric for comparing sentence encoders. For example, models are often grouped by attributes such as their parameter size, developer, pre-trained data source, fine-tuned tasks, etc. in the Hugging Face Hub, while leaderboards such as MTEB group models by their task performance (Muennighoff et al., 2023).

To address this need, we propose a scalable and systematic method for comparing sentence encoders. Unlike decoder-only Large Language Models (LLMs) that generate texts, which can be compared using likelihood scores (Oyama et al., 2025b), mapping sentence encoders is a signifi-

*Corresponding author.
[1]https://github.com/LivNLP/Map-of-Encoders

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 14160–14208

cantly challenging task due to three main reasons: (1) **multi-variateness**: unlike the scalar likelihood scores that can be directly obtained from decoders, encoders return multivariate embeddings, which require special care when comparing. (2) **misalignment**: the vector spaces spanned by independently trained sentence encoders are often misaligned and incomparable. (3) **dimensionality mismatch**: the dimensionalities of the encoders vary significantly from one another (i.e. in [384, 4096]) and are usually high. Therefore, previously proposed visualisation methods for text-generating decoder LLMs cannot be applied directly to sentence encoders.

To fill this gap, we propose a method to map a given set of sentence encoders onto a shared two-dimensional space. First, we represent each sentence encoder using an *embedding matrix* computed from a fixed set of sentences. Second, we convert the embedding matrix into a PIP matrix, which acts as an encoder dimensionality-independent representation. We further normalise the PIP matrix to create a density matrix. Third, we use QRE to define a divergence measure between two encoders that simultaneously considers the eigenspaces of their density matrices, providing a scale, rotation and translation invariant information-theoretic comparator. Using the spectral components in the QRE computation, we represent all encoders by feature vectors in a shared feature space. Finally, we use t-SNE (van der Maaten and Hinton, 2008) and create a map that contains 1101 sentence encoders as shown in Figure 1.

We see that encoders of varying parameter sizes, trained on different datasets and fine-tuned on diverse tasks, belonging to multiple primary architectures, are intuitively grouped in our map (§ 5), highlighting their attributes. Moreover, we find a strong correlation (Spearman > 0.75) between the actual performance of a downstream task of an encoder and that predicted using its feature vector for retrieval, reranking and clustering tasks (§ 5.3). This result implies that our map of encoders faithfully represents model performance and is practically useful for inferring attributes or performance of a novel encoder.

2 Related Work

Numerous architectures have been proposed in prior work for creating sentence embeddings such as bi-directional MLM-based ones (Devlin et al., 2019a; Liu et al., 2019; Song et al., 2020) and autoregressive LLM-based ones (BehnamGhader et al., 2024; Wang et al., 2024; Zhang et al., 2025). Moreover, embedding learning algorithms significantly affect encoder performance (Gao et al., 2021; Raffel et al., 2020). Despite the growing complexity of the sentence encoder space, there is no systematic method to empirically quantify the relationships among different sentence encoders.

Yin and Shen (2018) studied the relationships between the dimensionality and quality of *word* embeddings. They introduced PIP matrices to represent the similarity between words, which contain geometric information about the word embedding space and are independent of the dimensionality of the embeddings. Bollegala (2022) used PIP matrices to compare source embeddings for learning word-level meta-embeddings that aggregate properties from diverse source embeddings to create high-performance meta-embeddings on downstream tasks. In our case, we use QRE to measure the divergence between two encoders using their corresponding PIP matrices over *sentence* embeddings. Kornblith et al. (2019) used PIP matrices to measure pairwise representations of network layers. They proposed Centered Kernal Alignment (CKA) by taking the trace (equivalent to the sum over all eigenvalues) of the product of two PIP matrices. In contrast, we consider the complete eigenspaces of PIP matrices, including both eigenvalues as well as the eigenvectors, which better capture the geometry of the embedding spaces.

Raghu et al. (2017) proposed Singular Value Canonical Correlation Analysis (SVCCA) to compare the internal representations of two neural networks. SVCCA first represents a neuron by a vector of its activation over a set of inputs. Next, the previously created activation vectors for a set of neurons are arranged in a matrix in which Singular Value Decomposition (SVD) is performed to reduce its dimensionality, while retaining the most significant activation patterns. A similar low-dimensional activation-based representation is obtained for another set of neurones from a different network against which we would like to compare. Finally, Canonical Correlation Analysis (CCA) is used to compute an alignment between the two representations. Morcos et al. (2018) found that not all correlations are stable and proposed a projection weighted method that compares the neuron outputs with the canonical projections to compute a weighted mean. Both CKA and SVCCA were originally proposed to compare the internal repre-

sentations of neural networks, while in this work we are interested in comparing the output embeddings from sentence encoders. We treat all sentence encoders as black boxes and do not compare their internal representations, which simplifies the task of mapping a large number of diverse encoders. Moreover, this enables us to compare sentence encoders beyond neural methods such as weighted averaging of static word embeddings (Arora et al., 2017) or co-occurrence counts (Turney and Pantel, 2010).

Quantum NLP has many applications, such as learning word interactions by modelling words in quantum states and representing sentences as density matrices (Li et al., 2018; Wu et al., 2021), or detecting metaphors using density matrices to model the uncertainty and ambiguity of the literal meanings of words (Qiao et al., 2024). We represent the embedding space of a sentence encoder as a density matrix, extending this concept to compare sentence encoders.

Umegaki (1962) defined a measure of divergence for density matrices by subtracting cross entropy from the negative von Neumann Entropy, which is now known as QRE. By representing encoders as density matrices derived from their PIP matrices, we can use QRE to measure the divergence between two given encoders. De Domenico and Biamonte (2016) extended the definition of QRE to complex networks. They used QRE to measure information loss in network structures and perform multilayer clustering by minimising QRE. In contrast, we represent encoders as feature vectors where the axes correspond to the spectral components in the QRE computation.

The most related to our motivation in this paper is the work of Oyama et al. (2025a), where they created a map of the decoder LLMs. Specifically, they proposed the use of log-likelihood scores computed for a fixed set of texts to represent an LLM. Subsequently, a map is created by projecting these log-likelihood vectors using t-SNE. They proved that the squared Euclidean distance between the centred log-likelihood vectors approximates the Kullback-Leibler (KL) divergence between the corresponding LLMs. Note that their map can be applied only for decoder models because it requires scalar log-likelihood scores, while we focus on sentence encoders producing multi-dimensional embeddings. To the best of our knowledge, we are the first to create a large-scale and comprehensive map for sentence encoders.

3 Mapping Sentence Encoders

Let us denote a set of M sentence encoders by $\{f_m\}_{m=1}^{M}$, where each encoder f_m embeds a given sentence s to a d_m dimensional vector, $\boldsymbol{f}_m(s)$. Our goal is to represent all encoders in a common feature space that reflect their relationships. As discussed in § 1, mapping sentence encoders is a challenging task due to their (1) multi-variate, (2) misaligned, and (3) varied dimensional embedding spaces. To obtain a dimensionality-independent and a shared representation for f_m, we first use a fixed set of N sentences[2] $\mathcal{S}$ ($= \{s_n\}_{n=1}^{N}$), and arrange their embeddings $\{\boldsymbol{f}_m(s_n)\}_{n=1}^{N}$ as rows in an embedding matrix $\boldsymbol{A}_m \in \mathbb{R}^{N \times d_m}$. To address the challenges of disparate embedding spaces and mismatched dimensionalities, we use PIP matrices that have been successfully used in prior work on meta-embedding learning (Bollegala, 2022) to compare different embedding spaces. PIP computes the pairwise semantic similarities[3] among a fixed set of sentences according to a given encoder, which can be seen as capturing the semantic associations according to that encoder.

3.1 Density Matrices of Encoders

The PIP matrix of an encoder f_m, denoted as $\boldsymbol{G}_m$, is defined as the product of the embedding matrix and its transpose as in (1).

$$\boldsymbol{G}_m = \boldsymbol{A}_m \boldsymbol{A}_m^{\top} \in \mathbb{R}^{N \times N} \tag{1}$$

PIP matrices represent all sentence encoders in the same $N \times N$ real space, independently of their output embedding dimensionalities. Note that $\boldsymbol{G}_m$ is Positive Semi-Definite (PSD)[4] with eigenvalues equal to the squared singular values of $\boldsymbol{A}_m$, and the left singular vectors of $\boldsymbol{A}_m$ become the eigenvectors of $\boldsymbol{G}_m$ (see Appendix A for the proof). Moreover, because $d_m \ll N$, PIP matrices are often rank-deficient, having only d_m non-zero eigenvalues at most. Importantly, in practice we can efficiently compute the eigenspace of $\boldsymbol{G}_m$ ($N(N-1)/2$ elements via the SVD of $\boldsymbol{A}_m$ (Nd_m elements), without having to explicitly compute $\boldsymbol{G}_m$, which otherwise requires $\mathcal{O}(N^2)$ memory and $\mathcal{O}(N^3)$ time complexity.

[2] As discussed in § 4.2, $N = 10,000$ is sufficient for the encoders considered in our experiments.

[3] When the embeddings are ℓ_2 normalised, PIP matrix is equivalent to the cosine similarity matrix between sentences.

[4] $\boldsymbol{G}_m$ is a real-symmetric matrix even when $\boldsymbol{A}_m$ is not. However, applying SVD on $\boldsymbol{A}_m$ is more computationally cheaper than applying eigenvalue decomposition on $\boldsymbol{G}_m$.

We normalise PIP matrices by their trace to compute the corresponding density matrices as in (2).

$$\boldsymbol{\rho}_m = \frac{\boldsymbol{G}_m}{\mathrm{Tr}(\boldsymbol{G}_m)} \tag{2}$$

This normalisation ensures that $\mathrm{Tr}(\boldsymbol{\rho}_m) = 1$. As proven in Appendix A, dividing a PIP matrix by a positive scalar preserves its PSD property. Together, these two properties satisfy the requirements for $\boldsymbol{\rho}_m$ to be a density matrix (Nielsen and Chuang, 2010).

3.2 Quantum Relative Entropy

Although KL divergence is a measure of the difference between two probability distributions (Pérez-Cruz, 2008), which is used for the map of decoder LLMs by Oyama et al. (2025a), it is not applicable to sentence encoders in our case. We compare the embedding spaces using density matrices, rather than probability distributions. QRE extends the definition of KL divergence from probability distributions to density matrices and is a natural choice for comparing embedding spaces.

Given two embedding matrices $\boldsymbol{A}$ and $\boldsymbol{B}$, respectively with density matrices $\boldsymbol{\rho}$ and $\boldsymbol{\sigma}$, the QRE of $\boldsymbol{\rho}$ with respect to $\boldsymbol{\sigma}$ is given by (3).

$$S(\boldsymbol{\rho}\|\boldsymbol{\sigma}) = \mathrm{Tr}(\boldsymbol{\rho}\ln\boldsymbol{\rho}) - \mathrm{Tr}(\boldsymbol{\rho}\ln\boldsymbol{\sigma}) \tag{3}$$

Here, $\ln$ denotes the matrix logarithm.[5] Because $\boldsymbol{\rho}$ and $\boldsymbol{\sigma}$ are symmetric PSD matrices, they can be diagonalised into spectral representations by the Spectral Theorem (Strang, 2022) as proven in Appendix A. Therefore, we use spectral decomposition to overcome the numerical instability and the high computational costs involved with power series methods (Higham, 2008) when computing the matrix logarithms in (3). Specifically, we decompose density matrices as in (4).

$$\boldsymbol{\rho} = \sum_{i}^{K_\rho} \lambda_i \boldsymbol{v}_i \boldsymbol{v}_i^\top, \quad \boldsymbol{\sigma} = \sum_{j}^{K_\sigma} \mu_j \boldsymbol{u}_j \boldsymbol{u}_j^\top \tag{4}$$

where $\{(\lambda_i, \boldsymbol{v}_i)\}_{i=1}^{K_\rho}$ and $\{(\mu_j, \boldsymbol{u}_j)\}_{j=1}^{K_\sigma}$ denote the eigenpairs of $\boldsymbol{\rho}$ and $\boldsymbol{\sigma}$, with K_ρ and K_σ representing their respective ranks. The QRE can then be expressed using the eigen components as in (5).

$$S(\boldsymbol{\rho}\|\boldsymbol{\sigma}) = \sum_{i}^{K_\rho} \lambda_i(\ln\lambda_i) - \sum_{i}^{K_\rho} \lambda_i \left(\sum_{j}^{K_\sigma} \left(\boldsymbol{v}_i^\top \boldsymbol{u}_j \right)^2 \ln\mu_j \right) \tag{5}$$

[5]The logarithm $\ln\boldsymbol{\rho}$ is the matrix logarithm, defined as the scalar logarithm of the eigenvalues of $\boldsymbol{\rho}$, and not by the element-wise logarithm of the matrix.

Here, subject to the requirement that for any i, $\sum_{j=1}^{N} \left(\boldsymbol{v}_i^\top \boldsymbol{u}_j \right)^2 = 1$. This requires that the two sets of eigenvectors of $\boldsymbol{\rho}$ and $\boldsymbol{\sigma}$ form complete orthonormal bases spanning $\mathbb{R}^N$. See Appendix C for the derivation of (5) from (3).

In practice, the eigenspaces of two arbitrary encoders rarely align completely or span the same vector space, resulting in a loss of information of the corresponding eigenvector and $\sum_{j=1}^{N} \left| \boldsymbol{v}_i^\top \boldsymbol{u}_j \right|^2 < 1$. Theoretically, QRE goes to infinity when the eigenspace of $\boldsymbol{\rho}$ is not contained in the eigenspace of $\boldsymbol{\sigma}$ (Nielsen and Chuang, 2010). In our case, to maintain a finite and informative divergence, we treat the misaligned subspace as representing a high degree of divergence values where QRE becomes large but finite. To address this misalignment problem, we use Theorem 1 to approximate QRE.

Theorem 1. *Let $\boldsymbol{\rho}$ be the density matrix of an encoder with non-zero orthonormal eigenpairs $\{(\lambda_i, \boldsymbol{v}_i)\}_{i=1}^{K_\rho}$. Let $\boldsymbol{\sigma}$ be the density matrix of a different encoder with non-zero orthonormal eigenpairs $\{(\mu_j, \boldsymbol{u}_j)\}_{j=1}^{K_\sigma}$. By perturbing $\boldsymbol{\sigma}$ with a small noise parameter $\epsilon > 0$ in its null space,[6] we can approximate QRE as follows:*

$$S(\boldsymbol{\rho}\|\boldsymbol{\sigma}_\epsilon) = \sum_{i=1}^{K_\rho} \lambda_i(\ln\lambda_i) - \sum_{i=1}^{K_\rho} \lambda_i(\mathcal{C}_i + r_i \ln\epsilon) \tag{6}$$

where $c_i = \sum_{j=1}^{K_\sigma} (\boldsymbol{v}_i^\top \boldsymbol{u}_j)^2$ is the captured mass, $r_i = 1 - c_i$ is the residual mass, and $\mathcal{C}_i = \sum_{j=1}^{K_\sigma} (\boldsymbol{v}_i^\top \boldsymbol{u}_j)^2 \ln\mu_j$ is the cross-entropy contribution from the aligned subspace.

Theorem 1 enables the computation of QRE values even when the eigenspaces are misaligned. When the captured mass for a specific eigenvector of the encoder represented by $\boldsymbol{\rho}$ is less than 1, we quantify the missing information in this direction of the QRE measurement using ϵ. In this situation, the contribution $-r_i\lambda_i \ln\epsilon$ to the QRE tends to be large, where r_i is the residual mass and $-\lambda_i \ln\epsilon$ represents the cross entropy between the eigenvector of the first encoder ($\boldsymbol{\rho}$) and the misaligned subspace of the second encoder ($\boldsymbol{\sigma}$). See proof of Theorem 1 in Appendix E.

[6]In our experiments in § 5, we set $\epsilon = e^{-16}$. For further discussion, see the ablation study for ϵ in Appendix D.

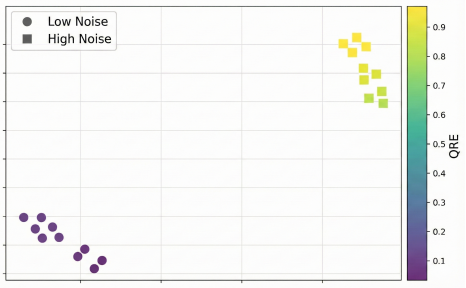


Figure 2: An illustrative example of the visualisation of QRE-based feature vectors. We use the unit base encoder with 10,000 basis vectors and generate two groups of synthetic embeddings with low noise ($\sigma^2 \in [0, 1]$) and high noise ($\sigma^2 \in [3, 4]$), respectively sampled from the normal distribution, $\mathcal{N}(0, \sigma^2\mathbf{I})$. Each group has 10 embedding matrices. We use our proposed QRE method to compute the feature vectors and visualise them using t-SNE. This example clearly shows that the low- and high-noise groups are separated by distinguishable QRE values (sum of feature vectors), validating the faithfulness of our method. See Appendix G for details.

3.3 Representing Sentence Encoders

Recall that QRE measures the divergence of an encoder relative to another encoder, which is problematic for two reasons: (1) the pairwise computation of relative divergences between all encoders ($\mathcal{O}(M^2)$) is expensive, and (2) there is no shared feature representation common to all encoders. To address this, we define a *unit base encoder* f_0, an imaginary encoder that has a density matrix $\boldsymbol{\rho}_0$ where its eigenvectors form a complete orthonormal basis of $\mathbb{R}^N$. Note that we do not explicitly require the embedding (PIP or density) matrices for computing QRE using § 3.2, but only require the eigenspace of its density matrix. We use the unit base encoder to represent each *target encoder* as a *feature vector* in a shared space.

Let $\{(\lambda_i, \boldsymbol{e}_i)\}_{i=1}^N$ denote the eigenpairs of the unit base encoder, where $\boldsymbol{e}_i$ is a unit basis vector with 1 at the i-th dimension and 0 elsewhere, and all $\lambda_i = \frac{1}{N}$. This ensures that the eigenspace of the unit base encoder is *isotropic*. Additionally, this indicates that our unit base encoder corresponds to the eigendecomposition of the density matrix $\boldsymbol{\rho}_0 = \frac{1}{N}\boldsymbol{I}$, where $\boldsymbol{I} \in \mathbb{R}^{N\times N}$ is the identity matrix. Therefore, we can decompose the QRE of the target encoder with respect to the unit base encoder into a feature vector, where each dimension corresponds to the divergence contribution measured along the specific basis vector.

Specifically, we represent the m-th target encoder f_m as an N-dimensional feature vector $\boldsymbol{\phi}_m$ as follows. Let the density matrix corresponding to f_m be $\boldsymbol{\rho}_m$ and its eigenpairs be $\{(\mu_j, \boldsymbol{u}_j)\}_{j=1}^{d_m}$. The w-th dimension ($w \in [1, N]$) of $\boldsymbol{\phi}_m$ is denoted by $\phi_{m,w}$ and represents the contribution by the w-th eigen component of $\boldsymbol{\rho}_0$ to the calculation of QRE in (6). Concretely, $\phi_{m,w}$ is given by (7).

$$\phi_{m,w} = \lambda_w \ln \lambda_w - \lambda_w(\mathcal{C}_w + r_w \ln \epsilon) \quad (7)$$

By construction, the sum of the elements in the feature vector $\boldsymbol{\phi}_m$ is equal to QRE of the target encoder with respect to the unit base encoder as expressed by (8).

$$\sum_{w=1}^{N} \phi_{m,w} = S(\boldsymbol{\rho}_0 \| \boldsymbol{\rho}_m) \quad (8)$$

As an alternative to using the unit base encoder, one might choose one of the sentence encoders as the reference point. However, this choice is likely to skew the visualisation, for example, by making encoders similar to the chosen reference encoder closely mapped to the reference, while pushing the dissimilar ones further away. On the other hand, when we use the proposed uninformative unit base encoder, we refrain from selecting any single encoder as the comparison point, which makes the unit base encoder both *unbiased* and *independent* from the set of encoders that must be mapped. Moreover, as explained above, all target encoders are represented in the same shared N-dimensional feature space. These desirable properties guarantee that any novel encoder can be represented in the same map coordinates, without affecting the positions of the existing encoders.

3.4 Projection onto a 2D Map

Following the prior work on mapping decoder LLMs by Oyama et al. (2025a), we use t-SNE to visualise the map of encoders. t-SNE creates a map by capturing the implicit structures of high-dimensional data with low-dimensional manifolds, showing both global and local relationships in a two-dimensional map and is extensively used for data visualisation (Li et al., 2016; González-Márquez et al., 2022).

As shown by (8), QRE of a target encoder is given by the sum of elements in the corresponding feature vector. Empirically, all feature values are positive due to the residual mass correction and the ℓ_1 norm of $\boldsymbol{\phi}_m$, $||\boldsymbol{\phi}_m||_1$, is equal to $S(\rho_0||\rho_m)$ in

practice. Moreover, the ℓ_1 distance between two target encoder feature vectors reflects the accumulated difference in cross-entropy arising from the distinct alignment of each encoder with the unit base encoder (see Appendix F for the proof). Therefore, we use the ℓ_1 distance measured between encoder feature vectors as the distance metric for finding nearest neighbours (in § 5) and for the t-SNE projections.

Figure 2 shows the result of applying our proposed method to synthetic embeddings. We see that the map perfectly separates the two groups of synthetic embeddings, thus accurately reflecting the inter- and intra-group relationships.

4 Experiments

The creation of feature vectors for the map of encoders requires a set of encoders $\{f_m\}_{m=1}^{M}$ and a set of sentences $\mathcal{S}$ for creating the embedding matrices and their selection is described next.

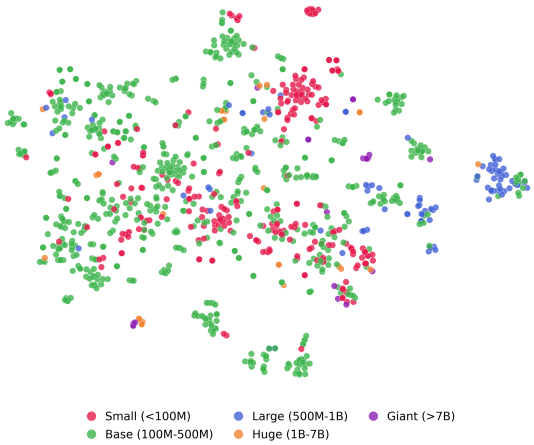
4.1 Selection of Sentence Encoders

We use Hugging Face Hub to select sentence encoders that are compatible with the sentence-transformers library (Hugging Face, 2025). Adopting the model selection strategy by Oyama et al. (2025a), we first select the most-downloaded 2000 sentence encoders (over the past 30 days, as of November 27, 2025). Subsequently, we exclude the sentence encoders that cannot be correctly loaded using the SentenceTransformer class (see detailed discussion in § 7). This filtering process produces a final set of 1101 sentence encoders for the map (see the full model list in Appendix L).

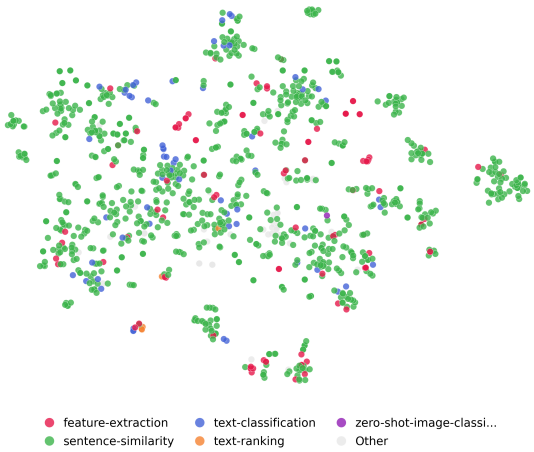
4.2 Selection of Sentence Set

Recall that we use trace normalised PIP matrices as input to our QRE calculations. Therefore, it is important to ensure an injective mapping process in which different encoders are mapped to different PIP matrices. Computation of a PIP matrix is a two step process, where we first represent an encoder using an embedding matrix and then compute the PIP matrix using the embedding matrix.

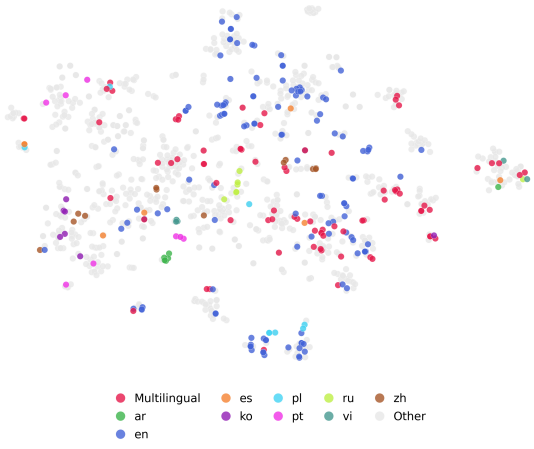
To ensure that the embedding matrices capture the semantic properties of all encoders adequately, we must carefully select an appropriate set of sentences. For example, we must not select a biased sample of sentences such as extremely short or long ones, grammatically incorrect ones or covering only a skewed subset of linguistic and typological structures (e.g. sentences containing negations, counterfactuals or passive voice, etc.). For this purpose, we randomly selected 10,000 high-quality and diverse sentences from a curated high-



(a) Encoder Parameter Size



(b) Training Task



(c) Pre-trained Language

Figure 3: Maps by attributes.

quality-English-sentences (agentlans, 2024) dataset derived from C4 (Raffel et al., 2023) and FineWeb (Penedo et al., 2024). This dataset is filtered by a text-quality assessment classifier, deberta-v3-xsmall-quality, to select a high quality set of sentences that cover a wide range of topics.

According to (1) the rank of the PIP matrix is lower than the smallest between the dimensionality d_m of the encoder and the number of sentences N. To preserve the information represented by all dimensions of all encoders, we must set N at least to be larger than the maximum dimensionality taken over all encoders. The largest dimensionality of any encoder selected in § 4.1 is 4096.

In practice, for the PIP matrix-based encoder representation to act as an injective mapping such that distinct encoders are represented by dissimilar PIP matrices, we must ensure that N remains large. As shown in Appendix B, the probability that two distinct centred PIP matrices having a Pearson correlation coefficient between their off-diagonal elements exceeding τ (>0) is given by $1 - \Phi(\tau\sqrt{N-1})$, where Φ is the standard normal cumulative distribution function. This probability decreases exponentially with increasing N, guaranteeing injectivity in practice. However, using a large set of sentences increases the computation time when creating embedding matrices. As discussed in further detail in Appendix M, we find that selecting 10,000 sentences is sufficient to accurately represent all sentence encoders.

5 Results

5.1 Nearest Neighbours

We investigate the nearest neighbours of the encoders to assess whether our feature vectors can accurately capture the relationships between encoders. We hypothesise that encoders derived from the same primary architectures would exhibit smaller pairwise ℓ_1 values and cluster together within our map. A zoomed-in map at the bottom of Figure 1 visualises the neighbours of the encoders *bge-large* and *bge-base* as an illustrative example, highlighting the proximity of neighbours with the same primary architecture in the map.

Table 1 shows the top 5 nearest neighbours in the ascending order of pairwise ℓ_1 values for 4 randomly-selected target encoders (MPNet, Multi-MPNet, GIST-v0, bge-large-en) (see the full list of nearest neighbours in Appendix H). For all encoders, we inspect their model cards in Hugging Face Hub and have confirmed that the nearest neighbours are indeed fine-tuned versions of the corresponding target models. In particular, from Table 1 we see that the encoders sharing the same name are fine-tuned from the target encoders.

Encoder Name	ℓ_1
Target: MPNet	
optimum-intel-internal-testing/all-mpnet-base-v2	0.0
flax-sentence-embeddings/all_datasets_v4_mpnet-base	$4.96e^{-4}$
spartan8806/atles-champion-embedding	0.0637
wwydmanski/all-mpnet-base-v2-legal-v0.1	0.0655
hojzas/setfit-proj8-all-mpnet-base-v2	0.0931
Target: Multi-MPNet	
meedan/paraphrase-filipino-mpnet-base-v2	0.0776
shtilev/medical_embedded_v5	0.1108
projecte-aina/s-t-NLI-ca_paraphrase-multilingual-mpnet-base	0.1236
shtilev/medical_embedded_v3	0.1325
sdadas/s-t-polish-paraphrase-from-mpnet	0.1344
Target: GIST-v0	
embaas/s-t-gte-small	0.0726
thenlper/gte-small	0.0726
OrcaDB/gte-small	0.0770
JALLAJ/5epo	0.0994
MoralHazard/NSFW-GIST-small	0.1013
Target: bge-large-en	
katanemo/bge-large-en-v1.5	0.0
llmrails/ember-v1	0.1555
ls-da3m0ns/bge_large_medical	0.1932
WhereIsAI/UAE-Large-V1	0.2590
sdadas/mmlw-e5-large	0.2679

Table 1: Nearest neighbours for the encoders sorted in the ascending order of pairwise ℓ_1 values. s-t denotes *sentence-transformers*. 0.0 indicates the value is effectively zero at 12 decimal places.

For **MPNet**, *all-mpnet-base-v2-legal-v0.1* is fine-tuned from MPNet on a legal dataset. *atles-champion-embedding* is also fine-tuned from MPNet on Semantic Textual Similarity (STS) tasks. For **Multi-MPNet**, the neighbour *medical_embedded (v5 and v3)* are fine-tuned from Multi-MPNet on medical and clinical texts. For **GIST-v0**, the neighbour, *5epo*, is fine-tuned from *BAAI/bge-small-en-v1.5*, where GIST-v0 is fine-tuned from the same encoder on a medical dataset. The two neighbours *embaas/s-t-gte-small* and *thenlper/gte-small* have the same ℓ_1 values, indicating that they are highly close to each other and possibly copies of the same encoder. *NSFW-GIST-small* is a fine-tuned version of GIST on NSFW texts. Finally, for **bge-large-en**, the nearest neighbours are copied or directly fine-tuned from bge-large encoders with additional curated datasets and training strategies. We further investigate the hierarchical clustering of the nearest neighbours in Appendix I, with different groups of neighbours clearly identifiable on the map. This shows that our map captures both the local and global relationships between the encoders.

In conclusion, our method accurately represents

encoders reflecting their close relationships, which might not be obvious from the encoder names alone, but are correctly captured in the encoder feature vectors and the map created therewith.

5.2 Analysis of the Map

We observe grouping patterns when the map is coloured by different attributes such as parameter size and encoder types (primary architecture of the model).[7] Figure 1 visualises the encoders by encoder type. Generally, the bert encoders are pervasive in the map, reflecting the wide usage of bert in sentence encoders. The xlm-roberta encoders tend to group in the far right areas, while the mpnet encoders are close together at the top of the map. The roberta and distilbert encoders are close to bert clusters, which is reasonable due to architectural similarity. There are also clear clusters for specific encoder types, such as gemma3_text and model2vec encoders.

From the map showing the encoder parameter sizes (Figure 3a), we see that encoders with smaller parameter sizes (<100M) tend to cluster at the middle and top, whereas encoders with larger parameter sizes (500M-1B) cluster on the far right side. Encoders with parameter sizes (>1B) tend to be scattered across the map, instead of forming tight clusters. This might be because smaller models are often distilled or fine-tuned from the larger foundation models, making them similar to each other on the map. For example, a closer look at the map reveals that the Qwen series encoders with different parameter sizes are closer to each other on the map (e.g. 8B, 4B and 0.6B).

From the map coloured by the fine-tuning tasks (Figure 3b), we see that all encoders are mostly fine-tuned on the sentence similarity tasks, a standard fine-tuning task for sentence encoders (Reimers and Gurevych, 2019), showing its prevalence on the map. Feature extraction and text classification are also commonly used tasks for fine-tuning encoders (Devlin et al., 2019b; Wolf et al., 2020). Although they are not as prevalent as sentence similarity task, they are also spread throughout the map.

Considering pre-trained languages (Figure 3c), encoders trained only on English (en) are mostly located at the central portion of the map spanning from top to bottom, whereas encoders trained on multilingual datasets are located at the centre and right side of the map. Note that most of the multilingual datasets contain English. Monolingual encoders (except English) such as for Arabic (ar) and Korean (ko) languages can be found in concentrated clusters. See Appendix J for maps coloured by the dimensionality of the embeddings and the training datasets.

We further plot a dendrogram to visualise the hierarchical relationships between the top 100 most-downloaded encoders in Figure 4. We see that encoders with exact/derived architectures are closely located (shown in the same colour), which is consistent with our map by encoder type (Figure 1).

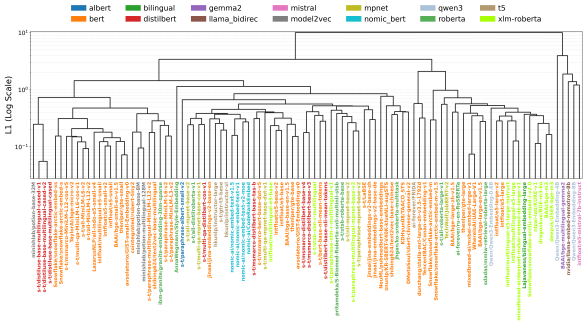


Figure 4: Hierarchical clustering of the top 100 downloaded encoders, coloured by encoder type. ℓ_1 values are reported in log scale for better visualisation. A zoomed-in version shown in Appendix K.

Task	Spearman	Pearson
SciFact	0.900	0.831
NFCorpus	0.881	0.803
BiorxivClusteringS2S	0.879	0.778
ArxivClusteringS2S	0.877	0.814
ArxivClusteringP2P	0.868	0.784
HotpotQA	0.859	0.803
SciDocsRR	0.850	0.792
BiorxivClusteringP2P	0.842	0.739
TwentyNewsgroupsClustering	0.832	0.771
ImdbClassification	0.831	0.775

Table 2: Average Spearman/Pearson correlations between the true and predicted task performance of top 10 tasks, sorted by descending Spearman.

5.3 Correlation with Downstream Tasks

A map that correlates well with the downstream task performance of the encoders is desirable because it shows that the map captures the true performance of encoders and can potentially be used to infer the performance of a novel encoder. For this purpose, we measure the correlation between the *true* performance of an encoder and its performance *predicted* from the feature vector of that

[7] We extract six attributes from Hugging Face API: encoder type, parameter size, training task, pre-trained language, training dataset, and dimensionality.

Task Type	Spearman	Pearson	# Datasets
Clustering	0.824	0.756	11
Retrieval	0.754	0.705	27
Reranking	0.752	0.717	4
PairClassification	0.612	0.530	3
STS	0.565	0.455	10
Classification	0.452	0.402	12

Table 3: Average Spearman/Pearson correlations between the true and predicted task performance by task type, sorted by descending Spearman correlation.

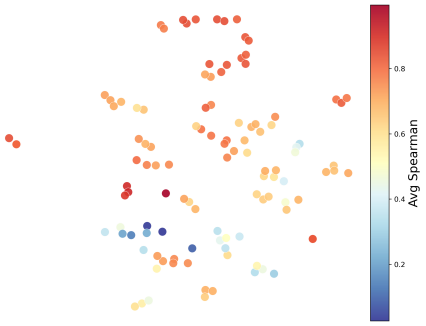


Figure 5: Submap of 112 encoders coloured by the average Spearman correlation between the true and predicted task performance of selected tasks.

encoder.

We use the official results from the MTEB leaderboard as the true encoder performance instead of re-evaluating encoders by ourselves for reproducibility and computational feasibility.[8] Out of all our encoders, 112 appear on the MTEB leader-board (see encoder list in Appendix N and their results in Appendix P), which provide a sufficiently large sample to test for statistical significance when predicting the task performance of encoders from their feature vectors. We first use PCA to project our feature vectors from 10,000 to a 50-dimensional space to mitigate any overfitting. Next, we use ElasticNetCV with 5-fold cross-validation to predict the downstream task performance of the selected encoders using their (low-dimensional) feature vectors (see Appendix O for implementation details).

Table 2 shows the performance for individual tasks. Our encoder feature vectors report significantly high Spearman and Pearson correlations ($>$0.8) (p-value $<$0.05) in multiple tasks, indicating their relatedness to downstream task performance. We further create submaps for the 112 encoders by their average performance on the selected 10 tasks in Table 2.[9] As shown in Figure 5, our map reflects the average Spearman correlation between the true and predicted MTEB performance for encoders in general, which is demonstrated by the proximity of encoders with similar levels of average Spearman correlation. For example, encoders with higher Spearman correlation (red/orange) are grouped in the upper areas of the map, while encoders with lower Spearman correlation (yellow/blue) are grouped in the bottom-middle areas.

Feature vectors are especially connected well with task types such as clustering, retrieval and reranking, as shown in Table 3. In contrast, classification tasks show weaker correlations. We believe this is because they require training classification heads using class-labelled training instances, which are not available to the encoders, and hence not captured in the encoder feature vectors.

STS tasks also report a moderate level of Spearman Correlation. STS is a common fine-tuning task or is used as a training objective to improve sentence encoders. As seen from the performance reported in the MTEB leaderboard, encoders with STS pre-training have a significantly higher performance than those without training on STS tasks. Similar to text classification, STS requires labelled data for the encoders to be trained. However, our QRE-based feature vectors are computed in an unsupervised manner without using any labelled data from STS datasets, which may explain their relatively lower correlate with the downstream STS performance. Nevertheless, we see that our feature vectors have a high Spearman correlation (i.e.$\geq$0.75) on the STS14 and SICK-R tasks.

6 Conclusion

We proposed a QRE-based method to represent a given set of diverse sentence encoders using feature vectors computed in a shared space, which are subsequently projected to 2D to create a map of sentence encoders. Our map has well-founded properties and accurately reflects relationships between encoders, which are verified both theoretically and empirically. Moreover, our map correlates well with downstream task performance of encoders. Given that decoder LLMs have been previously mapped using likelihood vectors, while we mapped encoder LLMs using QRE-based feature vectors, a natural next step would be to map both decoder and encoder models on the same map.

[8] Re-evaluating all encoders on MTEB requires substantial computational resources and many weeks.

[9] We use min-max normalisation for each task's performance and then compute their average.

7 Limitations

Our QRE-based method to create feature vectors for encoders and visualise them in a shared two-dimensional space can be applied to any sentence encoder and any sentence set, which is not specific to English. However, our selected sentence set to create embeddings for encoders as representations of the embedding spaces only covers English, which is a morphologically limited language. Our map is applicable to a multilingual sentence set, while the quality of the maps is based on the ability of the sentence encoders to accurately capture the semantic meaning of the multilingual sentences, which is out of our scope.

We select a set of 10,000 sentences. A larger sentence set (e.g. 100,000 sentences) can be selected to represent the semantic spaces of sentence encoders. However, in Appendix M, our experimental results have shown that 10,000 sentences perform well enough to create high-quality maps. Additionally, our QRE method is scalable to larger sentence sets, because our method does not need to compute PIP matrices explicitly, which instead uses SVD of the embedding matrices as detailed in § 3.

In the selected set of sentence encoders, we have the multilingual sentence encoders and map them successfully based on the English sentence set. The effects of non-English sentence sets for such multilingual encoders in our map remain to be further investigated in future work.

At the time of writing, there are 17,515 sentence encoders in the Hugging Face Hub. However, we only selected 2000 of them based on the most downloaded records, due to computational resource constraints. Among the 2000 selected encoders, we further filter out encoders that cannot be loaded or not used correctly by the SentenceTransformer class such as cross encoders, as explained in § 4.1. Cross encoders require sentence pairs (query, document) as input, while we embed one sentence at a time. Therefore, extending our method to heterogeneous encoders of diverse architectures could further reveal the relationships between sentence encoders.

Our selection strategy for sentence encoders in § 4.1 is based on the number of downloads, introducing a popularity bias. This strategy tends to include fewer encoders that are trained on low-resource languages. Therefore, it would be important to particularly consider low-resource encoders, facilitating a fair and linguistically diverse analysis.

8 Ethical Considerations

We do not annotate or release any datasets in this project. To the best of our knowledge, there are no ethical issues raised regarding the high-quality-english-sentences dataset (§ 4.2) we used. However, it has been reported that unfair social biases are found in some MTEB tasks such as STS (Webster et al., 2021). Although we do not train any sentence encoders, it has been reported that both MLM-based and LLM-based sentence encoders contain various social biases (May et al., 2019; Lin et al., 2025). We have not evaluated the social biases of the sentence encoders and their downstream implications. Therefore, we consider it to be important to measure social biases before integrating any encoder into the map.

Acknowledgements

Danushka Bollegala holds concurrent appointments as a Professor at University of Liverpool and as an Amazon Scholar. This paper describes work performed at the University of Liverpool and is not associated with Amazon.

References

agentlans. 2024. High-quality english sentences. https://huggingface.co/datasets/agentlans/high-quality-english-sentences. Accessed: 2025-12-13.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A simple but tough-to-beat baseline for sentence embeddings. *International Conference on Learning Representations*.

Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. 2024. Llm2vec: Large language models are secretly powerful text encoders. *arXiv preprint arXiv:2404.05961*.

Danushka Bollegala. 2022. Learning meta word embeddings by unsupervised weighted concatenation of source embeddings. *Preprint*, arXiv:2204.12386.

Qian Chen, Wen Wang, Qinglin Zhang, Siqi Zheng, Chong Deng, Hai Yu, Jiaqing Liu, Yukun Ma, and Chong Zhang. 2023. Ditto: A simple and efficient approach to improve sentence embeddings. *Empir Method Nat Lang Process*, abs/2305.10786:5868–5875.

Manlio De Domenico and Jacob Biamonte. 2016. Spectral entropies as information-theoretic tools for complex network comparison. *Physical Review X*, 6(4).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019a. Bert: Pre-training of deep bidirectional transformers for language understanding. *Preprint*, arXiv:1810.04805.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019b. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Stroudsburg, PA, USA. Association for Computational Linguistics.

Rita González-Márquez, Philipp Berens, and Dmitry Kobak. 2022. Two-dimensional visualization of large document libraries using t-sne. In *ICLR 2022 workshop on geometrical and topological representation learning*.

Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. 2005. Measuring statistical dependence with hilbert-schmidt norms. In *International Conference on Algorithmic Learning Theory*, pages 63–77. Springer.

Nicholas J Higham. 2008. *Functions of matrices: theory and computation*. SIAM.

Wassily Hoeffding. 1948. A class of statistics with asymptotically normal distribution. *The Annals of Mathematical Statistics*, 19(3):293–325.

N Houlsby, A Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and S Gelly. 2019. Parameter-efficient transfer learning for NLP. *ICML*, abs/1902.00751:2790–2799.

Hugging Face. 2025. Hugging face models: Sentence transformers. `https://huggingface.co/models?library=sentence-transformers&sort=trending`. Accessed: 2025-12-13.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. Similarity of neural network representations revisited. *Preprint*, arXiv:1905.00414.

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9119–9130, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. 2016. Visualizing and understanding neural models in nlp. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 681–691.

Qiuchi Li, Sagar Uprety, Benyou Wang, and Dawei Song. 2018. Quantum-inspired complex word embedding. In *Proceedings of the Third Workshop on Representation Learning for NLP*, pages 50–57, Melbourne, Australia. Association for Computational Linguistics.

Luyang Lin, Lingzhi Wang, Jinsong Guo, and Kam-Fai Wong. 2025. Investigating bias in llm-based bias detection: Disparities between llms and human perception. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 10634–10649.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Nathan Mantel. 1967. The detection of disease clustering and a generalized regression approach. *Cancer Research*, 27(2 Part 1):209–220.

Chandler May, Alex Wang, Shikha Bordia, Samuel Bowman, and Rachel Rudinger. 2019. On measuring social biases in sentence encoders. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 622–628.

Ari S Morcos, Maithra Raghu, and Samy Bengio. 2018. Insights on representational similarity in neural networks with canonical correlation. *arXiv [stat.ML]*.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2023. Mteb: Massive text embedding benchmark. *Preprint*, arXiv:2210.07316.

Michael A. Nielsen and Isaac L. Chuang. 2010. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press.

Momose Oyama, Ryo Kishino, Hiroaki Yamagiwa, and Hidetoshi Shimodaira. 2025a. Likelihood variance as text importance for resampling texts to map language models. *arXiv [cs.CL]*.

Momose Oyama, Hiroaki Yamagiwa, Yusuke Takase, and Hidetoshi Shimodaira. 2025b. Mapping 1,000+ language models via the log-likelihood vector. *arXiv [cs.CL]*.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. 2024. The fineweb datasets: Decanting the web for the finest text data at scale. *Preprint*, arXiv:2406.17557.

Fernando Pérez-Cruz. 2008. Kullback-leibler divergence estimation of continuous distributions. In *2008 IEEE international symposium on information theory*, pages 1666–1670. IEEE.

Wenbo Qiao, Peng Zhang, and ZengLai Ma. 2024. A quantum-inspired matching network with linguistic theories for metaphor detection. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 1435–1445, Torino, Italia. ELRA and ICCL.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the limits of transfer learning with a unified text-to-text transformer. *Preprint*, arXiv:1910.10683.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. 2017. SVCCA: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992. Association for Computational Linguistics.

Robert J Serfling. 1980. *Approximation Theorems of Mathematical Statistics*. John Wiley & Sons.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Preprint*, arXiv:2004.09297.

Gilbert Strang. 2022. *Introduction to linear algebra*. SIAM.

Peter D Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37:141–188.

Hisaharu Umegaki. 1962. Conditional expectation in an operator algebra, iv (entropy and information). Technical Report 2, Department of Mathematics, Tokyo Institute of Technology.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11):2579–2605.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Improving text embeddings with large language models. *Preprint*, arXiv:2401.00368.

Kellie Webster, Xuezhi Wang, Ian Tenney, Alex Beutel, Emily Pitler, Ellie Pavlick, Jilin Chen, Ed Chi, and Slav Petrov. 2021. Measuring and reducing gendered correlations in pre-trained models.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45. Association for Computational Linguistics.

Sixuan Wu, Jian Li, Peng Zhang, and Yue Zhang. 2021. Natural language processing meets quantum physics: A survey and categorization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3172–3182, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jiahao Xu, Wei Shao, Lihui Chen, and Lemao Liu. 2023. SimCSE++: Improving contrastive learning for sentence embeddings from two perspectives. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12028–12040, Stroudsburg, PA, USA. Association for Computational Linguistics.

Zi Yin and Yuanyuan Shen. 2018. On the dimensionality of word embedding. *Advances in neural information processing systems*, 31.

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.

A Eigenvalue Properties of the PIP Matrix

Theorem 2. *Let $\boldsymbol{A} \in \mathbb{R}^{N \times d}$ be a real matrix with $N \geq d$. Let $\sigma_1, \sigma_2, \ldots, \sigma_d$ denote the eigenvalues of $\boldsymbol{A}$. Let $\boldsymbol{B} = \boldsymbol{A}\boldsymbol{A}^\top$. Let $\lambda_1, \lambda_2, \ldots, \lambda_N$ denote the eigenvalues of $\boldsymbol{B}$. The following properties hold:*

(a) All eigenvalues λ_i are real and non-negative ($\lambda_i \geq 0$).

(b) The first d eigenvalues are equal to the squared singular values of A (i.e. $\lambda_i = \sigma_i^2$ for $1 \leq i \leq d$).

(c) The remaining $(N-d)$ eigenvalues are zero.

(d) For any $c > 0$, the matrix $\boldsymbol{C} = \frac{1}{c}\boldsymbol{B}$ remains PSD.

Proof. **Part (a): Real and Non-negative**

First, we establish that $\boldsymbol{B}$ is symmetric. Using the property $(\boldsymbol{XY})^\top = \boldsymbol{Y}^\top\boldsymbol{X}^\top$:

$$\boldsymbol{B}^\top = (\boldsymbol{A}\boldsymbol{A}^\top)^\top = (\boldsymbol{A}^\top)^\top\boldsymbol{A}^\top = \boldsymbol{A}\boldsymbol{A}^\top = \boldsymbol{B}.$$

Since $\boldsymbol{B}$ is real and symmetric, by the Spectral Theorem, all its eigenvalues are real.

Next, we show that $\boldsymbol{B}$ is PSD. Let λ be an eigenvalue of $\boldsymbol{B}$ and $\boldsymbol{v} \in \mathbb{R}^N$ be the corresponding non-zero eigenvector.

$$\begin{aligned}
\boldsymbol{B}\boldsymbol{v} &= \lambda\boldsymbol{v} \\
\boldsymbol{v}^\top\boldsymbol{B}\boldsymbol{v} &= \boldsymbol{v}^\top(\lambda\boldsymbol{v}) \\
\boldsymbol{v}^\top\boldsymbol{A}\boldsymbol{A}^\top\boldsymbol{v} &= \lambda(\boldsymbol{v}^\top\boldsymbol{v}) \\
(\boldsymbol{A}^\top\boldsymbol{v})^\top(\boldsymbol{A}^\top\boldsymbol{v}) &= \lambda\left|\left|\boldsymbol{v}\right|\right|^2 \\
\left|\left|\boldsymbol{A}^\top\boldsymbol{v}\right|\right|^2 &= \lambda\left|\left|\boldsymbol{v}\right|\right|^2.
\end{aligned}$$

Since the squared Euclidean norm is non-negative, $\left|\left|\boldsymbol{A}^\top\boldsymbol{v}\right|\right|^2 \geq 0$ and $\left|\left|\boldsymbol{v}\right|\right|^2 > 0$, it follows that:

$$\lambda = \frac{\left|\left|\boldsymbol{A}^\top\boldsymbol{v}\right|\right|^2}{\left|\left|\boldsymbol{v}\right|\right|^2} \geq 0.$$

Part (b) and (c): Magnitude of Eigenvalues

We utilize the Singular Value Decomposition (SVD) of $\boldsymbol{A}$. Let $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$, where:

- $\boldsymbol{U} \in \mathbb{R}^{N\times N}$ is an orthogonal matrix.
- $\boldsymbol{V} \in \mathbb{R}^{d\times d}$ is an orthogonal matrix.
- $\boldsymbol{\Sigma} \in \mathbb{R}^{N\times d}$ is a rectangular diagonal matrix containing singular values σ_i.

Substituting the SVD into the expression for $\boldsymbol{B}$:

$$\begin{aligned}
\boldsymbol{B} &= (\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top)(\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top)^\top \\
&= \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top\boldsymbol{V}\boldsymbol{\Sigma}^\top\boldsymbol{U}^\top.
\end{aligned}$$

Since $\boldsymbol{V}$ is orthogonal, $\boldsymbol{V}^\top\boldsymbol{V} = \boldsymbol{I}_d$. Thus:

$$\boldsymbol{B} = \boldsymbol{U}(\boldsymbol{\Sigma}\boldsymbol{\Sigma}^\top)\boldsymbol{U}^\top.$$

This equation represents the eigendecomposition of $\boldsymbol{B}$, where the columns of $\boldsymbol{U}$ are the eigenvectors and the diagonal entries of the matrix $\boldsymbol{\Lambda} = \boldsymbol{\Sigma}\boldsymbol{\Sigma}^\top$ are the eigenvalues.

We calculate $\boldsymbol{\Sigma}\boldsymbol{\Sigma}^\top$ (an $N \times N$ matrix):

$$\begin{aligned}
\boldsymbol{\Sigma}\boldsymbol{\Sigma}^\top = \mathrm{diag}(\sigma_1, \ldots, \sigma_d, 0, \ldots, 0)\times \\
\mathrm{diag}(\sigma_1, \ldots, \sigma_d, 0, \ldots, 0)^\top
\end{aligned}$$

The resulting diagonal entries are:

$$(\boldsymbol{\Sigma}\boldsymbol{\Sigma}^\top)_{ii} = \begin{cases} \sigma_i^2 & \text{for } 1 \leq i \leq d \\ 0 & \text{for } d < i \leq N \end{cases}$$

Thus, the first d eigenvalues are σ_i^2, and the remaining $N-d$ eigenvalues are 0.

Part (d): Positive Semi-Definiteness Preservation

As shown in (a), $\boldsymbol{B}$ is PSD. By definition of PSD matrices, we have for any vector $x \in \mathbb{R}^N s$:

$$x^\top\boldsymbol{B}x \geq 0$$

Given $\boldsymbol{B}$ is symmetric, we first show the symmetry of $\boldsymbol{C}$

$$\boldsymbol{C}^\top = \left(\frac{1}{c}\boldsymbol{B}\right)^\top = \frac{1}{c}\boldsymbol{B}^\top = \frac{1}{c}\boldsymbol{B} = \boldsymbol{C}$$

Then, we show that for any vector x:

$$x^\top\boldsymbol{C}x = x^\top\left(\frac{1}{c}\boldsymbol{B}\right)x = \frac{1}{c}\left(x^\top\boldsymbol{B}x\right) \geq 0$$

because $x^\top\boldsymbol{B}x \geq 0$ and $\frac{1}{c} > 0$. Therefore, $\boldsymbol{C}$ is PSD.

□

B Injectivity of the PIP Transformation

The inputs to the QRE computation are the eigenvectors and eigenvalues of the PIP matrices representing the unit base encoder and a target encoder. Therefore, it is important to discuss how different encoders are represented by their corresponding PIP matrices. The computation of the PIP matrix $\boldsymbol{G}$ of an encoder f is conducted in two steps. First, we must compute the embedding matrix $\boldsymbol{A}$ using the sentence embeddings computed for a fixed set of N sentences. The same set of sentences is used to compute embedding matrices for all encoders that we must map. We must select a set of sentences that is sufficiently larger and diverse such that the semantic space of an encoder is faithfully represented. We select $N = 10,000$ sentences following the sentence selection strategy described in Appendix M. Second, we compute the PIP matrix as $\boldsymbol{G} = \boldsymbol{A}\boldsymbol{A}^\top$ from the embedding matrix computed as described above. In this section, we

discuss the conditions that must be satisfied for this second step of the encoder representation process.

An important property that an encoder representation method must satisfy in our application is that two different embedding matrices must be represented by different PIP matrices.[10] In other words, the PIP-based representation of the embedding matrices must be an *injective* mapping. Otherwise, we will encounter undesirable situations where two inherently different encoders (represented by distinct embedding matrices) are mapped to the same PIP matrix. In such cases, we will incorrectly represent those different encoders by the same point on the encoder map. In this section, we estimate the probability that this degenerative case will occur.

Formally, let us consider two encoders f and f', represented respectively by their embedding matrices $\boldsymbol{A}$ and $\boldsymbol{A}'$, while the corresponding PIP matrices are denoted respectively by $\boldsymbol{G}$ and $\boldsymbol{G}'$. Before we analyse the injectivity of the PIP transformation, it is worth noting that surjectivity does not hold in general and is also not an important consideration in our application. For example, two encoders could be trivially different, one of which being the rotation by an orthogonal matrix. The PIP transformation is invariant under an orthogonal transformation of the embedding matrix. To see this, consider an orthogonal matrix $\boldsymbol{L}$. Then, we have

$$\begin{aligned}\boldsymbol{G}' &= (\boldsymbol{AL})(\boldsymbol{AL})^\top \\ &= \boldsymbol{ALL}^\top\boldsymbol{A}^\top \\ &= \boldsymbol{AA}^\top = \boldsymbol{G} \end{aligned} \tag{9}$$

from the orthogonality of $\boldsymbol{L}$ (i.e. $\boldsymbol{LL}^\top = \boldsymbol{I}$).

On the other hand, the PIP transformation is sensitive to linear translation (or more generally to affine transformations). For example, a linear translation would cause a PIP matrix to have a dominant eigenvector in the direction of the offset vector, resulting in an anisotropic embedding space. In return, this will affect the QRE calculation because it relies on the eigenvectors of the PIP matrix (or more correctly, density matrices obtained by normalising the PIP matrices). If we want to ensure that the PIP transformation remains invariant in such cases, we can do so by first applying centering and whitening[11] as pre-processing steps to the embedding matrices before we compute the PIP matrices. However, in practice it is extremely rare to encounter independently trained encoders that are different only by such linear transformations. In addition, encoders are post-processed by training linear projection heads (Devlin et al., 2019b) or adapters (Houlsby et al., 2019) for downstream tasks. Therefore, applying pre-processing steps on embedding matrices could remove such important differences, thereby mapping such encoders to the same point in the map. Consequently, in our experiments we do not further preprocess embedding matrices.

The following Lemma states that injectivity approximately holds large N values.

Lemma 1. *Let $\rho(\boldsymbol{G}, \boldsymbol{G}')$ be the Pearson correlation between the off-diagonal elements of doubly centred PIP matrices $\boldsymbol{G}$ and $\boldsymbol{G}'$, corresponding to two distinct encoders f and f' (with embedding matrices $\boldsymbol{A}$ and $\boldsymbol{A}'$), respectively. The probability of ρ exceeding a specified threshold τ (>0) is given by (10).*

$$\mathrm{P}(\rho(\boldsymbol{G}, \boldsymbol{G}') > \tau) = 1 - \Phi(\tau\sqrt{N-1}) \tag{10}$$

Here, Φ is the cumulative probability distribution function of the standard normal distribution, $\mathcal{N}(0, 1)$.

Proof. Under the null hypothesis that f and f' are entirely independent, the geometries of their embedding spaces are uncoupled. We evaluate the Pearson correlation $\rho(\boldsymbol{G}, \boldsymbol{G}')$ between the off-diagonal elements of the doubly centered PIP matrices. The unnormalised cross-correlation score S is given by the sum of the element-wise products as follows.

$$S = \sum_{i \neq j} G_{ij} G'_{ij} \tag{11}$$

Because the evaluation of each term $G_{ij}G'_{ij}$ fundamentally requires a pair of sentences (i and j), the test statistic S belongs to the family of U-statistics with a kernel of degree 2 (Hoeffding, 1948). According to Hoeffding's foundational theorem for U-statistics, as the sample size $N \to \infty$, a U-statistic constructed from independent observations

[10] Note that PIP transformation is not a liner mapping represented by the PIP matrix itself in the sense that we are not projecting embeddings using the PIP matrix of an encoder. Therefore, the non-invertability (because PIP matrix is rank deficit) of the PIP matrix does not imply anything about the non-bijectivity of the PIP transformation.

[11] Whitening will be required as a preprocessing step if we want to remove an affine transformation, while centering alone will be sufficient to remove a linear translation.

converges in distribution to a Normal distribution (Serfling, 1980). Since the normalised Pearson correlation ρ is simply a scaled version of S, it follows that ρ is also asymptotically normally distributed.

Under the null hypothesis of independence, the expected correlation is zero ($\mathbb{E}[\rho] = 0$). To determine the variance, we must account for the fact that although ρ is computed over $N(N-1)$ off-diagonal matrix elements, these elements are generated from only N embeddings. As established by the exact moments of the Mantel test for distance matrices (Mantel, 1967), and formally generalised for representational similarity metrics like the Hilbert-Schmidt Independence Criterion (HSIC) (Gretton et al., 2005), the variance is bottlenecked by the effective degrees of freedom of the sample. Consequently, the variance of ρ scales asymptotically with the inverse of the sample size:

$$\mathrm{Var}(\rho) \approx \frac{1}{N-1} \tag{12}$$

Therefore, the asymptotic distribution of our correlation metric is given by:

$$\rho \sim \mathcal{N}\left(0, \frac{1}{N-1}\right) \tag{13}$$

To evaluate the probability that ρ spuriously exceeds the similarity threshold τ, we standardise ρ into a Z-score by dividing by its standard deviation. Let $Z = \rho\sqrt{N-1}$, such that $Z \sim \mathcal{N}(0,1)$. Multiplying both sides of the inequality $\rho > \tau$ by $\sqrt{N-1}$ yields:

$$\mathrm{P}(\rho > \tau) = \mathrm{P}(Z > \tau\sqrt{N-1}) \tag{14}$$

By definition, the cumulative distribution function of the standard normal distribution, $\Phi(z) = \mathrm{P}(Z \leq z)$, gives the lower tail probability. The upper tail probability is simply the complement:

$$\mathrm{P}(\rho > \tau) = 1 - \Phi(\tau\sqrt{N-1}) \tag{15}$$

□

An important practical implication of the above Lamma is that as N grows large (e.g., $N = 10,000$), the probability of a spurious high correlation between distinct PIP matrices decays exponentially, guaranteeing that the PIP transformation remains practically injective.

C Proof for Eigen Decomposition of QRE

Here we prove the derivation from (3) to (5).

Proof. **Derivation of Term 1:** $\mathrm{Tr}(\boldsymbol{\rho} \ln \boldsymbol{\rho})$

First, we compute the matrix logarithm of $\boldsymbol{\rho}$. For a diagonalized matrix, we apply the logarithm to the eigenvalues:

$$\ln \boldsymbol{\rho} = \sum_k \ln(\lambda_k) \boldsymbol{v}_k \boldsymbol{v}_k{}^\top \tag{16}$$

Then, multiply $\boldsymbol{\rho}$ by $\ln \boldsymbol{\rho}$:

$$\boldsymbol{\rho}(\ln \boldsymbol{\rho}) = \left(\sum_i \lambda_i \boldsymbol{v}_i \boldsymbol{v}_i{}^\top\right)\left(\sum_k \ln(\lambda_k) \boldsymbol{v}_k \boldsymbol{v}_k{}^\top\right) \tag{17}$$

We expand the product. Since scalars commute, we can rearrange the terms.

$$= \sum_i \sum_k \lambda_i \ln(\lambda_k) \boldsymbol{v}_i (\boldsymbol{v}_i{}^\top \boldsymbol{v}_k) \boldsymbol{v}_k{}^\top \tag{18}$$

Using the orthonormality, $\boldsymbol{v}_i{}^\top \boldsymbol{v}_k = \delta_{ik}$, the terms are zero unless $i = k$:

$$= \sum_i \lambda_i \ln(\lambda_i) \boldsymbol{v}_i \boldsymbol{v}_i{}^\top \tag{19}$$

Then, we take the matrix Trace on both sides. Note that the Trace of a matrix is linear, and we can take the summation out from the trace:

$$\mathrm{Tr}(\boldsymbol{\rho} \ln \boldsymbol{\rho}) = \sum_i \lambda_i \ln(\lambda_i) \,\mathrm{Tr}(\boldsymbol{v}_i \boldsymbol{v}_i{}^\top) \tag{20}$$

Using the cyclic property of the Trace (i.e. $\mathrm{Tr}(\boldsymbol{AB}) = \mathrm{Tr}(\boldsymbol{BA})$), we have

$$\begin{aligned} \mathrm{Tr}(\boldsymbol{v}_i \boldsymbol{v}_i{}^\top) &= \mathrm{Tr}(\boldsymbol{v}_i{}^\top \boldsymbol{v}_i) \\ &= \boldsymbol{v}_i{}^\top \boldsymbol{v}_i = ||\boldsymbol{v}_i||^2 = 1 \end{aligned} \tag{21}$$

Thus,

$$\mathrm{Tr}(\boldsymbol{\rho} \ln \boldsymbol{\rho}) = \sum_i \lambda_i \ln \lambda_i \tag{22}$$

Derivation of Term 2: $\mathrm{Tr}(\boldsymbol{\rho} \ln \boldsymbol{\sigma})$

First, write out $\ln \boldsymbol{\sigma}$:

$$\ln \boldsymbol{\sigma} = \sum_j \ln(\mu_j) \boldsymbol{u}_j \boldsymbol{u}_j{}^\top \tag{23}$$

Substitute the expansions of $\boldsymbol{\rho}$ and $\ln \boldsymbol{\sigma}$ into the Trace:

$$\mathrm{Tr}(\boldsymbol{\rho} \ln \boldsymbol{\sigma}) = \mathrm{Tr}\left(\left[\sum_i \lambda_i \boldsymbol{v}_i \boldsymbol{v}_i^\top \right] \left[\sum_j \ln(\mu_j) \boldsymbol{u}_j \boldsymbol{u}_j^\top \right] \right) \tag{24}$$

Pull the summations and scalars (λ_i and $\ln \mu_j$) outside the Trace:

$$= \sum_i \sum_j \lambda_i \ln(\mu_j) \cdot \mathrm{Tr}(\boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{u}_j \boldsymbol{u}_j^\top) \tag{25}$$

Focus on the term inside the Trace: $\boldsymbol{v}_i \boldsymbol{v}_i^\top \boldsymbol{u}_j \boldsymbol{u}_j^\top$. Matrix multiplication is associative. We can group the middle terms $(\boldsymbol{v}_i^\top \boldsymbol{u}_j)$. Note that $\boldsymbol{v}_i^\top \boldsymbol{u}_j$ is the dot product of two vectors, resulting in a scalar value.

$$\boldsymbol{v}_i (\boldsymbol{v}_i^\top \boldsymbol{u}_j) \boldsymbol{u}_j^\top = (\boldsymbol{v}_i^\top \boldsymbol{u}_j) \cdot (\boldsymbol{v}_i \boldsymbol{u}_j^\top) \tag{26}$$

Substitute this back into the Trace:

$$\mathrm{Tr}\left((\boldsymbol{v}_i^\top \boldsymbol{u}_j) \cdot \boldsymbol{v}_i \boldsymbol{u}_j^\top \right) = (\boldsymbol{v}_i^\top \boldsymbol{u}_j) \cdot \mathrm{Tr}(\boldsymbol{v}_i \boldsymbol{u}_j^\top) \tag{27}$$

By applying the cyclic property of the Trace again to $\mathrm{Tr}(\boldsymbol{v}_i \boldsymbol{u}_j^\top)$:

$$\mathrm{Tr}(\boldsymbol{v}_i \boldsymbol{u}_j^\top) = \mathrm{Tr}(\boldsymbol{u}_j^\top \boldsymbol{v}_i) \tag{28}$$

Note that for real vectors, the dot product is symmetric, so $\boldsymbol{u}_j^\top \boldsymbol{v}_i = \boldsymbol{v}_i^\top \boldsymbol{u}_j$.

$$= \boldsymbol{v}_i^\top \boldsymbol{u}_j \tag{29}$$

Combine the parts:

$$\text{Trace Term} = (\boldsymbol{v}_i^\top \boldsymbol{u}_j) \cdot (\boldsymbol{v}_i^\top \boldsymbol{u}_j) = (\boldsymbol{v}_i^\top \boldsymbol{u}_j)^2 \tag{30}$$

Therefore, Term 2 becomes:

$$\mathrm{Tr}(\boldsymbol{\rho} \ln \boldsymbol{\sigma}) = \sum_i \lambda_i \sum_j (\boldsymbol{v}_i^\top \boldsymbol{u}_j)^2 \ln \mu_j \tag{31}$$

Subtract Term 2 from Term 1:

$$S(\boldsymbol{\rho} \| \boldsymbol{\sigma}) = \left(\sum_i \lambda_i \ln \lambda_i \right) - \left(\sum_i \lambda_i \sum_j (\boldsymbol{v}_i^\top \boldsymbol{u}_j)^2 \ln \mu_j \right) \tag{32}$$

Rearranging the parenthesis to match the target format:

$$S(\boldsymbol{\rho} \| \boldsymbol{\sigma}) = \sum_i \lambda_i (\ln \lambda_i) - \sum_i \lambda_i \left(\sum_j (\boldsymbol{v}_i^\top \boldsymbol{u}_j)^2 \ln \mu_j \right) \tag{33}$$

□

ϵ	**avg. Spearman**
e^{-12}	0.6564
e^{-16}	0.6712
e^{-20}	0.6703

Table 4: Average Spearman Correlation across 68 tasks for different ϵ values.

D Ablation Study for Epsilon ϵ

QRE is defined to be infinity between misaligned eigenspaces, indicating that the misaligned subspaces differ significantly. In this scenario, the divergence is large for the missing information in the space. As stated in Theorem 1, we introduce a small ϵ to approximate and quantify the large divergence from the null space, which prevents QRE from diverging to infinity. In our experiments, to reduce the noise due to eigenvalues that are small and numerically unstable we drop all eigenvalues less than $\sqrt{\epsilon}$ (corresponding to dropping singular values less than ϵ).

To empirically study the effect of this ϵ thresholding step on the proposed mapping method, we conduct an ablation study for selecting the ϵ value in the set $\{e^{-12}, e^{-16}, e^{-20}\}$. We select these three values to cover a broader numerical range under floating point numerical precision. Specifically, we compare the average Spearman correlation with MTEB performance for 68 tasks as described in § 5.3. Table 4 shows that varying the ϵ values does not affect the performance significantly by obtaining similar average Spearman Correlations. Additionally, we manually check and compare the maps generated based on the three ϵ values, and they all show the same patterns for encoders. These results verify the robustness and effectiveness of our QRE-based method and our setting $\epsilon = e^{-16}$.

E Proof for QRE Estimation with Residual Mass

Theorem. *Let $\boldsymbol{\rho}$ be the density matrix of a base encoder with non-zero orthonormal eigenpairs $\{(\lambda_i, \boldsymbol{v}_i)\}_{i=1}^{K_\rho}$. Let $\boldsymbol{\sigma}$ be the density matrix of a target encoder with non-zero orthonormal eigenpairs $\{(\mu_j, \boldsymbol{u}_j)\}_{j=1}^{K_\sigma}$. By perturbating $\boldsymbol{\sigma}$ with a small noise parameter $\epsilon > 0$ on its null space, the*

QRE can be approximated as follows:

$$S(\boldsymbol{\rho}\|\boldsymbol{\sigma}_\epsilon) = \sum_{i=1}^{K_\rho} \lambda_i(\ln \lambda_i) - \sum_{i=1}^{K_\rho} \lambda_i(\mathcal{C}_i + r_i \ln \epsilon) \tag{34}$$

. *where* $c_i = \sum_{j=1}^{K_\sigma} (\boldsymbol{v}_i{}^\top \boldsymbol{u}_j)^2$ *is the captured mass,* $r_i = 1 - c_i$ *is the residual mass, and* $\mathcal{C}_i = \sum_{j=1}^{K_\sigma} (\boldsymbol{v}_i{}^\top \boldsymbol{u}_j)^2 \ln \mu_j$ *is the cross-entropy contribution from the aligned subspace.*

Proof. We construct a regularised target matrix $\boldsymbol{\sigma}_\epsilon$ by adding a small perturbation ϵ to the zero eigenvalues corresponding to the null space of $\boldsymbol{\sigma}$. Let the subspace spanned by the target encoder be $\mathcal{U} = \text{span}\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{K_\sigma}\}$. Let $\{\boldsymbol{u}_{K_\sigma+1}, \ldots, \boldsymbol{u}_N\}$ be an orthonormal basis for the orthogonal complement $\mathcal{U}^\perp$ (the null space). We define $\boldsymbol{\sigma}_\epsilon$ as:

$$\boldsymbol{\sigma}_\epsilon = \boldsymbol{\sigma} + \epsilon \sum_{k=K_\sigma+1}^{N} \boldsymbol{u}_k \boldsymbol{u}_k{}^\top \tag{35}$$

Without loss of generality, assume $K_\sigma \leq K_\rho$.[12] The spectral decomposition of $\boldsymbol{\sigma}_\epsilon$ is therefore the sum of the original active components and the noise components:

$$\boldsymbol{\sigma}_\epsilon = \sum_{j=1}^{K_\sigma} \mu_j \boldsymbol{u}_j \boldsymbol{u}_j{}^\top + \sum_{k=K_\sigma+1}^{K_\rho} \epsilon \boldsymbol{u}_k \boldsymbol{u}_k{}^\top \tag{36}$$

Note that the set of all eigenvectors $\{\boldsymbol{u}_j\}_{j=1}^{K_\sigma} \cup \{\boldsymbol{u}_k\}_{k=K_\sigma+1}^{K_\rho}$ now forms a complete orthonormal basis for the effective subspace $\mathbb{R}^{K_\rho}$.

Recalling (3), the definition of QRE is:

$$S(\boldsymbol{\rho}\|\boldsymbol{\sigma}_\epsilon) = \text{Tr}(\boldsymbol{\rho} \ln \boldsymbol{\rho}) - \text{Tr}(\boldsymbol{\rho} \ln \boldsymbol{\sigma}_\epsilon) \tag{37}$$

The first term is simply the negative Von Neumann entropy of $\boldsymbol{\rho}$:

$$\text{Tr}(\boldsymbol{\rho} \ln \boldsymbol{\rho}) = \sum_i \lambda_i \ln \lambda_i \tag{38}$$

We focus on the second term (Cross Entropy), expanded using the eigenvectors of $\boldsymbol{\rho}$ (denoted $\{\boldsymbol{v}_i\}$):

$$\text{Tr}(\boldsymbol{\rho} \ln \boldsymbol{\sigma}_\epsilon) = \sum_i \lambda_i \boldsymbol{v}_i{}^\top (\ln \boldsymbol{\sigma}_\epsilon) \boldsymbol{v}_i \tag{39}$$

As we already clarified in § 3.1, ln is the matrix logarithm, not the element-wise logarithm. For a symmetric PSD matrix $\boldsymbol{\rho}$ with its spectral decomposition $\boldsymbol{\rho} = \sum \lambda_i \boldsymbol{v}_i \boldsymbol{v}_i{}^\top$, the matrix logarithm is defined by the logarithm of its eigenvalues: $\ln \boldsymbol{\rho} = \sum (\ln \lambda_i) \boldsymbol{v}_i \boldsymbol{v}_i{}^\top$.

Using the spectral decomposition of $\boldsymbol{\sigma}_\epsilon$, the matrix logarithm $\ln \boldsymbol{\sigma}_\epsilon$ is:

$$\ln \boldsymbol{\sigma}_\epsilon = \sum_{j=1}^{K_\sigma} (\ln \mu_j) \boldsymbol{u}_j \boldsymbol{u}_j{}^\top + \sum_{k=K_\sigma+1}^{K_\rho} (\ln \epsilon) \boldsymbol{u}_k \boldsymbol{u}_k{}^\top \tag{40}$$

Substituting this into the quadratic form $\boldsymbol{v}_i{}^\top (\ldots) \boldsymbol{v}_i$:

$$\boldsymbol{v}_i{}^\top (\ln \boldsymbol{\sigma}_\epsilon) \boldsymbol{v}_i = \underbrace{\sum_{j=1}^{K_\sigma} (\ln \mu_j) \boldsymbol{v}_i{}^\top (\boldsymbol{u}_j \boldsymbol{u}_j{}^\top) \boldsymbol{v}_i}_{\text{Aligned Term}} + \underbrace{\sum_{k=K_\sigma+1}^{K_\rho} (\ln \epsilon) \boldsymbol{v}_i{}^\top (\boldsymbol{u}_k \boldsymbol{u}_k{}^\top) \boldsymbol{v}_i}_{\text{Orthogonal Term}} \tag{41}$$

Given that $\{\boldsymbol{v}_i\}$ and $\{\boldsymbol{u}_j\}$ are orthonormal bases, we have the properties that $||\boldsymbol{v}_i||^2 = ||\boldsymbol{u}_j||^2 = 1$, and they satisfy the orthogonality condition:

$$\boldsymbol{v}_i{}^\top \boldsymbol{v}_k = \delta_{ik} \quad \text{and} \quad \boldsymbol{u}_j{}^\top \boldsymbol{u}_l = \delta_{jl}$$

where δ_{ik} is the Kronecker delta, which equals 1 if $i = k$ and 0 otherwise.

Recognizing that

$$\boldsymbol{v}_i{}^\top \boldsymbol{u}_j \boldsymbol{u}_j{}^\top \boldsymbol{v}_i = (\boldsymbol{v}_i{}^\top \boldsymbol{u}_j)(\boldsymbol{u}_j{}^\top \boldsymbol{v}_i) = (\boldsymbol{v}_i{}^\top \boldsymbol{u}_j)^2 \tag{42}$$

Then,

$$\boldsymbol{v}_i{}^\top (\ln \boldsymbol{\sigma}_\epsilon) \boldsymbol{v}_i = \sum_{j=1}^{K_\sigma} (\ln \mu_j)(\boldsymbol{v}_i{}^\top \boldsymbol{u}_j)^2 + \sum_{k=K_\sigma+1}^{K_\rho} (\ln \epsilon)(\boldsymbol{v}_i{}^\top \boldsymbol{u}_k)^2 \tag{43}$$

Since the basis $\{\boldsymbol{u}_k\}_{k=1}^{K_\rho}$ forms a complete orthonormal basis for $\mathbb{R}^{K_\rho}$,[13] **Parseval's identity** asserts that the squared norm of any unit vector $\boldsymbol{v}_i$ equals the sum of its squared projection coefficients:

$$||\boldsymbol{v}_i||^2 = \sum_{k=1}^{K_\rho} (\boldsymbol{v}_i{}^\top \boldsymbol{u}_k)^2 = 1 \tag{44}$$

[12] With unit base encoder used as $\boldsymbol{\rho}$, we always have $K_\sigma \leq K_\rho$.

[13] This is the case for our proposed unit base encoder.

We decompose this sum into components lying in the aligned subspace $\mathcal{U}$ (spanned by $\{\boldsymbol{u}_j\}_{j=1}^{K_\sigma}$) and the orthogonal null space $\mathcal{U}^\perp$ (spanned by $\{\boldsymbol{u}_k\}_{k=K_\sigma+1}^{K_\rho}$). This decomposition corresponds to the **generalized Pythagorean theorem**, which states that $||\boldsymbol{v}_i||^2 = ||\mathcal{P}_{\mathcal{U}}(\boldsymbol{v}_i)||^2 + ||\mathcal{P}_{\mathcal{U}^\perp}(\boldsymbol{v}_i)||^2$, where $\mathcal{P}$ denotes the projection operator. Substituting the projection coefficients into the theorem yields:

$$\underbrace{\sum_{j=1}^{K_\sigma}(\boldsymbol{v}_i{}^\top \boldsymbol{u}_j)^2}_{\text{Squared projection on } \mathcal{U}} + \underbrace{\sum_{k=K_\sigma+1}^{K_\rho}(\boldsymbol{v}_i{}^\top \boldsymbol{u}_k)^2}_{\text{Squared projection on } \mathcal{U}^\perp} = 1 \tag{45}$$

Recognizing that the first term is the captured mass c_i, we rearrange the equation to solve for the projection onto the null space:

$$\sum_{k=K_\sigma+1}^{K_\rho}(\boldsymbol{v}_i{}^\top \boldsymbol{u}_k)^2 = 1 - \sum_{j=1}^{K_\sigma}(\boldsymbol{v}_i{}^\top \boldsymbol{u}_j)^2 = 1 - c_i = r_i \tag{46}$$

Substituting m_i and r_i back into the expansion in (43):

$$\boldsymbol{v}_i{}^\top(\ln \boldsymbol{\sigma}_\epsilon)\boldsymbol{v}_i = \left(\sum_{j=1}^{K_\sigma}(\boldsymbol{v}_i{}^\top \boldsymbol{u}_j)^2 \ln \mu_j\right) + r_i \ln \epsilon \tag{47}$$

Combining the terms, we arrive at the computational form used in (6). We define the cross-entropy contribution from the aligned subspace as $\mathcal{C}_i = \sum_{j=1}^{K_\sigma}(\boldsymbol{v}_i{}^\top \boldsymbol{u}_j)^2 \ln \mu_j$.

$$S(\boldsymbol{\rho}\|\boldsymbol{\sigma}_\epsilon) = \sum_{i=1}^{K_\rho}\lambda_i(\ln \lambda_i) - \sum_{i=1}^{K_\rho}\lambda_i(\mathcal{C}_i + r_i \ln \epsilon) \tag{48}$$

□

F Theoretical Analysis of the QRE-based Encoder Feature Space

In § 3.3, we created a feature vector $\boldsymbol{\phi}_m$ to represent an encoder f_m using the QRE $S(\boldsymbol{\rho}_0||\boldsymbol{\rho}_m)$ of $\boldsymbol{\rho}_m$ (i.e. the density matrix corresponding to f_m) with respect to $\boldsymbol{\rho_0}$ (i.e. the density matrix corresponding to the unit base encoder, f_0). Given that our goal is to create a map where each target encoder is represented with respect to its relationship to the other encoders, we consider it to be insightful to mathematically analyse the encoder feature vector space further.

For this purpose, let us consider two target encoders f_m and f'_m with their respective density matrices $\boldsymbol{\rho}_m$ and $\boldsymbol{\rho}'_m$. For simplicity of the disposition, we assume both $\boldsymbol{\rho}_m$ and $\boldsymbol{\rho}'_m$ to be full rank in $\mathbb{R}^N$. Therefore, we can denote the eigenpairs for $\boldsymbol{\rho}_m$ and $\boldsymbol{\rho}'_m$ respectively as $\{(\mu_j, \boldsymbol{u}_j)\}_{j=1}^N$ and $\{(\mu'_k, \boldsymbol{u}'_k)\}_{k=1}^N$. For rank deficient cases we can add a small random perturbation to the eigenvalues to make the density matrices full rank.

Following (7), we can write the w-th dimensions of $\boldsymbol{\phi}_m$ and $\boldsymbol{\phi}'_m$ as follows.

$$\phi_{m,w} = \lambda_w \ln \lambda_w - \lambda_w \sum_{j=1}^{N}\left(\boldsymbol{v}_w{}^\top \boldsymbol{u}_j\right)^2 \ln \mu_j \tag{49}$$

$$\phi_{m',w} = \lambda_w \ln \lambda_w - \lambda_w \sum_{k=1}^{N}\left(\boldsymbol{v}_w{}^\top \boldsymbol{u}'_k\right)^2 \ln \mu'_k \tag{50}$$

Because the unit base encoder has all of its eigenvalues $\lambda_w = 1/N$ and eigenvectors $\boldsymbol{v}_w$ as the w-th unit vector, we can substitute those values in (49) and (50) to further simplify as follows.

$$\phi_{m,w} = -\frac{1}{N}\ln N - \frac{1}{N}\sum_{j=1}^{N}\left(\boldsymbol{v}_w{}^\top \boldsymbol{u}_j\right)^2 \ln \mu_j \tag{51}$$

$$\phi_{m',w} = -\frac{1}{N}\ln N - \frac{1}{N}\sum_{k=1}^{N}\left(\boldsymbol{v}_w{}^\top \boldsymbol{u}'_k\right)^2 \ln \mu'_k \tag{52}$$

In particular, we are interested in the *offset*, $(\phi_{m,w} - \phi_{m',w})$, between the w-th feature of f_m and f'_m, which is given by (53).

$$-\frac{1}{N}\sum_{j=1}^{N}\left(\boldsymbol{v}_w{}^\top \boldsymbol{u}_j\right)^2 \ln \mu_j + \frac{1}{N}\sum_{k=1}^{N}\left(\boldsymbol{v}_w{}^\top \boldsymbol{u}'_k\right)^2 \ln \mu'_k \tag{53}$$

We see that the self-entropy terms for the unit base encoder cancel out in this offset.

Moreover, recall that the inner-product between the w-th unit vector $\boldsymbol{v}_w$ and $\boldsymbol{u}_j$ is simply selecting the w-th dimension $u_{j,w}$ of $\boldsymbol{u}_j$. Therefore, cross-entropy terms can be further simplified as follows:

$$-\frac{1}{N}\sum_{j}^{N}\left(\boldsymbol{v}_w{}^\top \boldsymbol{u}_j\right)^2 \ln \mu_j = -\frac{1}{N}\sum_{j}^{N}(u_{j,w})^2 \ln \mu_j \tag{54}$$

$$= \frac{1}{N}\sum_{j}^{N}(u_{j,w})^2 (-\ln \mu_j) \tag{55}$$

$$= \frac{1}{N}\sum_{j}^{N}\left(u_{j,w}\sqrt{(-\ln \mu_j)}\right)^2 \tag{56}$$

$$= \frac{1}{N}\left|\left|\bar{\boldsymbol{u}_w}\right|\right|^2. \tag{57}$$

In (56) we used the fact that all eigenvalues of density matrices are in $[0, 1]$, hence $(-\ln \mu_j) > 0$ (in practice $\mu_j \in (0, 1)$). Moreover, we define $\bar{\boldsymbol{u}}_w$ as a vector in $\mathbb{R}^N$ whose j-th component is $u_{j,w}\sqrt{-\ln \mu_j}$.

Plugging these results back, we can evaluate the offset as given by (58).

$$\phi_{m,w} - \phi_{m',w} = \frac{1}{N}\left(||\bar{\boldsymbol{u}_w}||^2 - \left|\left|\bar{\boldsymbol{u}'_w}\right|\right|^2\right) \quad (58)$$

From (58), we see that the offset is independent of the density matrix of the unit base encoder and depends only on the weighted projection of the eigenvectors of the target encoders onto the w-th basis vector of the unit base encoder.

G Synthetic Example

To show the faithfulness of our method, we provide an example of synthetic embeddings. We create synthetic embeddings with different levels of noise added to the 10,000-dimensional identity density matrix of the unit base encoder. For the embedding matrix of the base encoder, we add noise perturbation to it as follows.

For each embedding vector x in the base encoder matrix, we generate a noise vector $\boldsymbol{n}$ from a multivariate Gaussian distribution:

$$\boldsymbol{n} \sim \mathcal{N}\left(0, \sigma^2 \mathbf{I}_d\right) \quad (59)$$

Recall that embedding matrices are normalised. To ensure the noise is scaled and does not dominate the original signal, we calculate the perturbed synthetic embedding $\hat{x}$ as:

$$\hat{\boldsymbol{x}} = \boldsymbol{x} + 0.5 \cdot \frac{\boldsymbol{n}}{||\boldsymbol{n}||} \quad (60)$$

We set two noise levels with intervals $[0, 1]$ and $[3, 4]$ (low noise and high noise), and uniformly at random select 10 values for σ^2 from the two intervals respectively.

For the synthetic embeddings, we compute the feature vectors using the method defined in § 3.3 and use t-SNE to visualise as described in § 3.4. Figure 2 shows that the synthetic representations with low noise are close to each other and are spatially located from the group with high noise. Additionally, the low noise group has low QRE values based on the base encoder, while the high noise group has high QRE values. These two results validate that our map can distinguish the similar encoders and accurately visualise them in the same space.

H Full Nearest Neighbours Table

We randomly select the 16 enocoders from our encoder list detailed in § 4.1. For each selected encoder, taken as the target encoder, we retrieve its top 10 nearest neighbours based on the lowest pairwise ℓ_1 value relative to that encoder. Table 5 shows the full table of the nearest neighbours.

I Hierarchical Clustering of Nearest Neighbours

We draw a dendrogram for 10 randomly-selected encoders from the 30 encoders for computing nearest neighbours with the top 5 nearest neighbours for each to visualise their connections, shown in Figure 6. It is clear that all the neighbourhoods based on different core sentence encoders shown in the same colour are separable with neighbours close to each other. This validates that our map captures both the local and global relationships between encoders.

J More Maps by Training Datasets and Dimensionalities

Figure 7 shows two maps based on training datasets and the dimensionality of the embeddings.

For the map coloured by the training datasets Figure 7a, the encoders trained with multiple datasets tend to be located across the map. The encoders trained with single datasets exist more in the left areas of the map

For the map coloured by the output dimensionality of the embeddings corresponding to the encoder Figure 7b, we see a pattern that encoders with lower dimensionality (e.g. 384) are more located in the upper-middle area of the map, with similar size groups of 128 and 512 close together. Encoders with a higher dimensionality of 1024 have clear groups on the right and bottom side in the map. Dimensinality of 768 is the most common dimensionality for sentence encoders, where encoders with dimensionality of 768 are spread out in the map.

K Zoomed in Hierarchical Clustering for Top 100 Most-downloaded Encoders

Figure 8 shows the zoomed-in dendrogram for hierarchical clustering of the top 100 most-downloaded

Encoder Name	ℓ_1	Encoder Name	ℓ_1
Target: BAAI/bge-base-en-v1.5		**Target: nomic-ai/nomic-embed-text-v1.5**	
Shashwat13333/bge-base-en-v1.5_v4	0.0	asmud/nomic-embed-indonesian	0.0077
OrcaDB/bge-base	0.0	CatSchroedinger/nomic-v1.5-financial-matryoshka	0.1285
datasocietyco/bge-base-en-v1.5-course-recommender-v5	0.0218	corto-ai/nomic-embed-text-v1	0.1582
axondendriteplus/Legal-Embed-bge-base-en-v1.5	0.0456	nomic-ai/nomic-embed-text-v1	0.1589
pavanmantha/bge-base-en-bioembed	0.0601	simonosgoode/nomic_embed_fine_tune_law_1.5	0.1854
potsu-potsu/bge-base-biomedical-matryoshka-v3	0.0982	nomic-ai/nomic-embed-text-v1-ablated	0.1921
prashpathak/xlscout_standigger_2_aug	0.1183	nomic-ai/nomic-embed-text-v1-unsupervised	0.2258
iris49/3gpp-embedding-model-v0	0.1400	s-t/sentence-t5-xl	0.2628
avsolatorio/GIST-Embedding-v0	0.1924	hkunlp/instructor-xl	0.2636
infgrad/stella-base-en-v2	0.1976	s-t/sentence-t5-xxl	0.2637
Target: intfloat/multilingual-e5-base		**Target: s-t/msmarco-distilbert-base-tas-b**	
embaas/sentence-transformers-multilingual-e5-base	0.0	s-t/msmarco-distilbert-dot-v5	0.2946
Hiveurban/multilingual-e5-base	0.0	s-t/msmarco-bert-base-dot-v5	0.3134
d0rj/e5-base-en-ru	0.0019	intfloat/e5-base	0.3243
Renan1997/sentence-transformer	0.0397	ValentinaKim/bge-base-automobile-matryoshka	0.3261
antoinelouis/french-me5-base	0.0787	nomic-ai/nomic-embed-text-v2-moe	0.3283
clips/e5-base-trm-nl	0.1078	d0rj/e5-base-en-ru	0.3322
djovak/embedic-base	0.1703	embaas/sentence-transformers-multilingual-e5-base	0.3323
Lajavaness/bilingual-embedding-base	0.2027	Hiveurban/multilingual-e5-base	0.3323
KarBik/legal-french-matroshka	0.2045	intfloat/multilingual-e5-base	0.3323
hiieu/halong_embedding	0.2080	cnmoro/snowflake-arctic-embed-m-v2.0-cpu	0.3323
Target: ostoveland/SBertBaseMittanbudver3		**Target: Qwen/Qwen3-Embedding-0.6B**	
NbAiLab/nb-sbert-base	0.3306	woodx/Qwen3-Embedding-0.6B-SGLang	0.0
Omartificial-Intelligence-Space/Arabic-all-nli-triplet-Matryoshka	0.3938	michaelfeil/Qwen3-Embedding-0.6B-auto	0.0
BlackKakapo/stsb-xlm-r-multilingual-ro	0.3945	DeepMount00/Ita-Search	0.3266
s-t/paraphrase-multilingual-mpnet-base-v2	0.3999	tomaarsen/Qwen3-Embedding-0.6B-18-layers	0.3396
s-t/nli-mpnet-base-v2	0.4031	Tarka-AIR/Tarka-Embedding-350M-V1	0.3770
tomaarsen/mpnet-base-nli-matryoshka	0.4032	nlpai-lab/KURE-v1	0.3961
meedan/paraphrase-filipino-mpnet-base-v2	0.4039	seroe/Qwen3-Embedding-0.6B-turkish-triplet-matryoshka-v2	0.3981
s-t/stsb-distilbert-base	0.4063	dragonkue/BGE-m3-ko	0.3984
s-t/distilbert-base-nli-stsb-mean-tokens	0.4063	dragonkue/snowflake-arctic-embed-l-v2.0-ko	0.3987
shtilev/medical_embedded_v5	0.4066	avemio/German-RAG-BGE-M3-MERGED-x-SNOWFLAKE-ARCTIC-HESSIAN-AI	0.4012
Target: s-t/all-MiniLM-L12-v2		**Target: s-t/paraphrase-multilingual-MiniLM-L12-v2**	
flax-sentence-embeddings/all_datasets_v3_MiniLM-L12	0.0694	DataikuNLP/paraphrase-multilingual-MiniLM-L12-v2	0.0
s-t/all-MiniLM-L12-v1	0.0694	annakotarba/sentence-similarity	0.0028
latterworks/ollama-embeddings	0.0931	vahoaka/sentence-transformers-model-vahoaka-v1	0.0210
flax-sentence-embeddings/all_datasets_v4_MiniLM-L6	0.0931	tnguy564/qwen-geospatial-embedder	0.0513
Sbhatti33/sbert_model	0.0931	tikanosa/fine-tuned-sbert-prodi	0.0623
yosuaw/all-MiniLM-L6-v2	0.0931	JoaoVitorr/ifood-classification-model-v5	0.0642
optimum-intel-internal-testing/all-MiniLM-L6-v2	0.0931	gmunkhtur/paraphrase-mongolian-minilm-mn_v2	0.0761
s-t/all-MiniLM-L6-v2	0.0931	Ahmedhisham/queen_of_embedded_egy_20k	0.0770
ozziek/all-MiniLM-L6-v2-lasttoken-false	0.0931	lengocquangLAB/fine-tuned-jobtitle-embed	0.0818
valurank/MiniLM-L6-Keyword-Extraction	0.0931	s-t/paraphrase-MiniLM-L12-v2	0.0900
Target: intfloat/multilingual-e5-small		**Target: BAAI/bge-small-en-v1.5**	
beademiguelperez/s-t-multilingual-e5-small	0.0	optimum-intel-internal-testing/bge-small-en-v1.5	0.0
d0rj/e5-small-en-ru	$8.56e^{-4}$	michaelfeil/bge-small-en-v1.5	0.0
ferrisS/german-english-multilingual-e5-small	0.0016	sdadas/mmlw-e5-small	0.0983
SergeyKarpenko1/multilingual-e5-small-legal-matryoshka_384	0.0103	JALLAJ/5epo	0.0994
antoinelouis/french-me5-small	0.0362	raul-delarosa99/bge-small-en-v1.5-RIRAG_ObliQA	0.1001
clips/e5-small-trm-nl	0.0558	austinpatrickm/finetuned_bge_embeddings_v5_small_v1.5	0.1112
dragonkue/multilingual-e5-small-ko-v2	0.0763	avsolatorio/GIST-small-Embedding-v0	0.1164
djovak/embedic-small	0.0978	baconnier/Finance2_embedding_small_en-V1.5	0.1176
dragonkue/multilingual-e5-small-ko	0.1126	jebish7/MedEmbed-small-v0.1_MNR_1	0.1208
exp-models/dragonkue-KoEn-E5-Tiny	0.1129	thenlper/gte-small	0.1242
Target: s-t/gtr-t5-base		**Target: intfloat/multilingual-e5-large**	
s-t/gtr-t5-large	0.1728	Hiveurban/multilingual-e5-large-pooled	0.0
s-t/gtr-t5-xl	0.1892	embaas/s-t-multilingual-e5-large	0.0
s-t/sentence-t5-base	0.1931	smart-tribune/s-t-multilingual-e5-large	0.0
hkunlp/instructor-base	0.1955	d0rj/e5-large-en-ru	0.0023
hkunlp/instructor-xl	0.1963	antoinelouis/french-me5-large	0.0846
s-t/gtr-t5-xxl	0.2027	mixedbread-ai/deepset-mxbai-embed-de-large-v1	0.1267
krlvi/sentence-t5-base-nlpl-code_search_net	0.2161	DejanX13/Poverenik_embedding_doc_2000	0.1622
s-t/sentence-t5-large	0.2219	DejanX13/ISO_embedding_1000	0.1685
s-t/sentence-t5-xl	0.2262	clips/e5-large-trm-nl	0.2306
s-t/sentence-t5-xxl	0.2310	ng3owb/sentiment-embedding-model	0.2499
Target: malteos/scincl		**Target: s-t/paraphrase-multilingual-mpnet-base-v2**	
andreinsardi/SciBERT-SolarPhysics-Search	0.4968	meedan/paraphrase-filipino-mpnet-base-v2	0.0776
ibm-granite/granite-embedding-125m-english	0.4999	shtilev/medical_embedded_v5	0.1108
nasa-impact/nasa-smd-ibm-st-v2	0.5128	projecte-aina/ST-NLI-ca_paraphrase-multilingual-mpnet-base	0.1236
Sampath1987/EnergyEmbed-v1	0.5155	shtilev/medical_embedded_v3	0.1325
prdev/mini-gte	0.5181	sdadas/st-polish-paraphrase-from-mpnet	0.1344
vaios-stergio/all-mpnet-base-v2-dblp-aminer-180k-pairs	0.5275	s-t/paraphrase-mpnet-base-v2	0.1896
pritamdeka/S-PubMedBert-MS-MARCO-SCIFACT	0.5338	lang-uk/ukr-paraphrase-multilingual-mpnet-base	0.2182
copenlu/spiced	0.5351	Omartificial-Intelligence-Space/Arabic-all-nli-triplet-Matryoshka	0.2184
thenlper/gte-base	0.5387	JoshELambert/illegal	0.2377
hkunlp/instructor-base	0.5388	s-t/nli-mpnet-base-v2	0.2525
Target: s-t/multi-qa-mpnet-base-dot-v1		**Target: BAAI/bge-large-en-v1.5**	
Adarsh921/multi_qa_mpnet	0.0	katanemo/bge-large-en-v1.5	0.0
AI-Growth-Lab/PatentSBERTa	0.0049	llmrails/ember-v1	0.1555
winderfeld/cc-uffs-ppc-ft-test-multiqa	0.1915	ls-da3m0ns/bge_large_medical	0.1932
antonkirk/retrieval-mpnet-dot-finetuned-gpt-4o-mini	0.2264	WhereIsAI/UAE-Large-V1	0.2590
cnmoro/snowflake-arctic-embed-m-v2.0-cpu	0.3032	sdadas/mmlw-e5-large	0.2679
google/embeddinggemma-300m-qat-q4_0-unquantized	0.3157	sdadas/mmlw-retrieval-e5-large	0.2681
MongoDB/mdbr-leaf-mt	0.3162	sdadas/mmlw-roberta-large	0.2712
unsloth/embeddinggemma-300m-qat-q8_0-unquantized	0.3197	avemio/German-RAG-UAE-LARGE-V1-TRIPLES-MERGED	0.2712
h2oai/embeddinggemma-300m-qat-q8_0-unquantized	0.3197	sdadas/mmlw-retrieval-roberta-large	0.2779
google/embeddinggemma-300m-qat-q8_0-unquantized	0.3197	OrcaDB/mxbai-large	0.2960

Table 5: Nearest neighbours for 16 encoders (Top-10 each) sorted in ascending order of pairwise ℓ_1 values. s-t denotes *sentence-transformers*. 0.0 indicates value is effectively zero for 12 decimals.

encoders.

L Full Encoder List for the Map

Table 8 provides the list of 1101 encoders with encoder type, parameter size and dimensionality, which are three attributes used in this paper.

M Evaluating the Sentence Set Selection

To validate selecting 10,000 sentences for creating high-quality maps, we further create feature matrices and maps of encoders based on 1000, 2500 and 5000 sentences as the sentence set $\mathcal{S}$ in § 3.

As shown in Figure 10, maps created with 5000

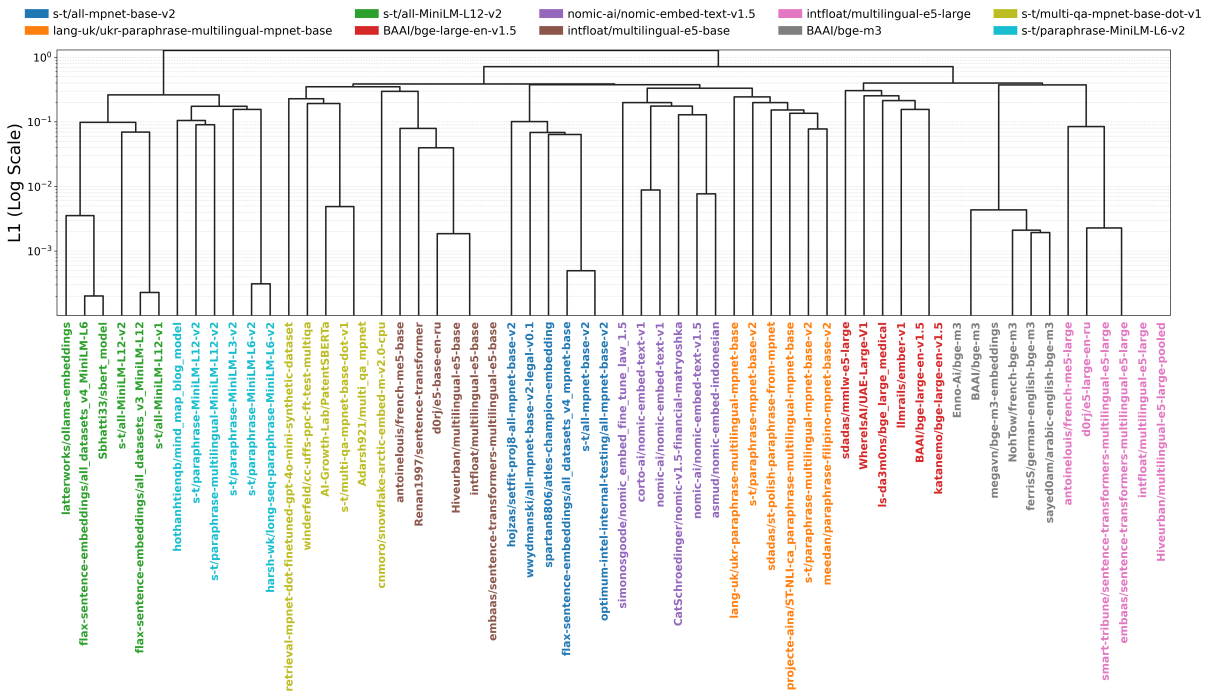


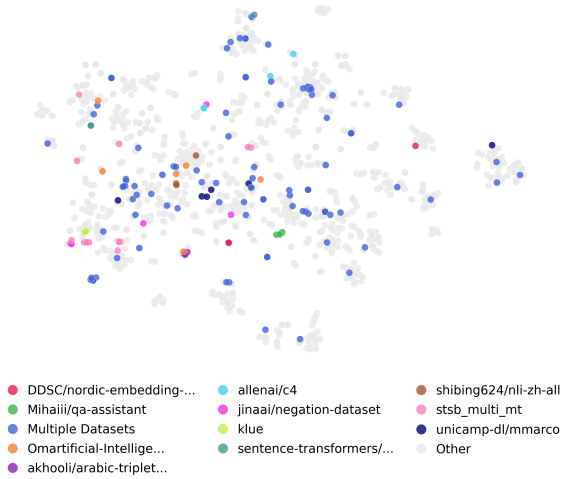
Figure 6: Hierarchical Clustering for 10 randomly selected encoders with top 5 nearest neighbours.

sentences has comparable and clear patterns in the maps as those created with 10,000 sentences in the paper. For example, for the map coloured by encoder type, which is the most complicated map, has clear samll groups such as model2vec and gemma3_text same as the map created with 10,000 sentences. This validates that our selected sentence set has high quality for generating sentence embeddings and thus the maps of encoders.

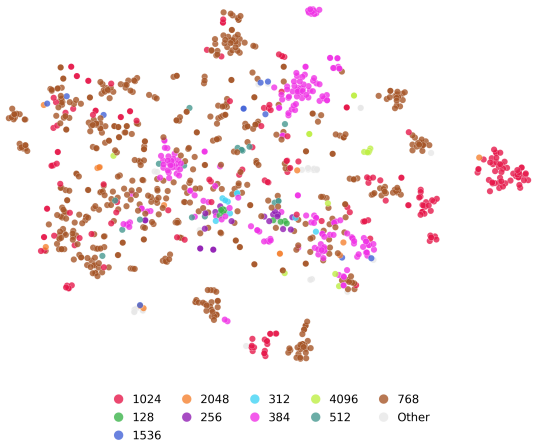
Additionally, we test the effectiveness of the sentence set sizes of 1000, 2500, 5000 and 10000 on the average Spearman Correlation between the true and predicted performance on 68 MTEB tasks, as shown in Figure 9. The average Spearman correlation saturates for a sentence set of 2500 sentences, indicating that 2500 sentences is sufficient to obtain peak average correlation on MTEB and our selection of 10,000 sentences can accurately capture the embedding spaces of all target encoders.

N Full Encoder List for MTEB results

Table 8 shows the full list of 112 encoders with parameter size and dimensionality.



(a) Training Datasets



(b) Dimensionality

Figure 7: Maps based on training datasets and dimensionality of encoders.

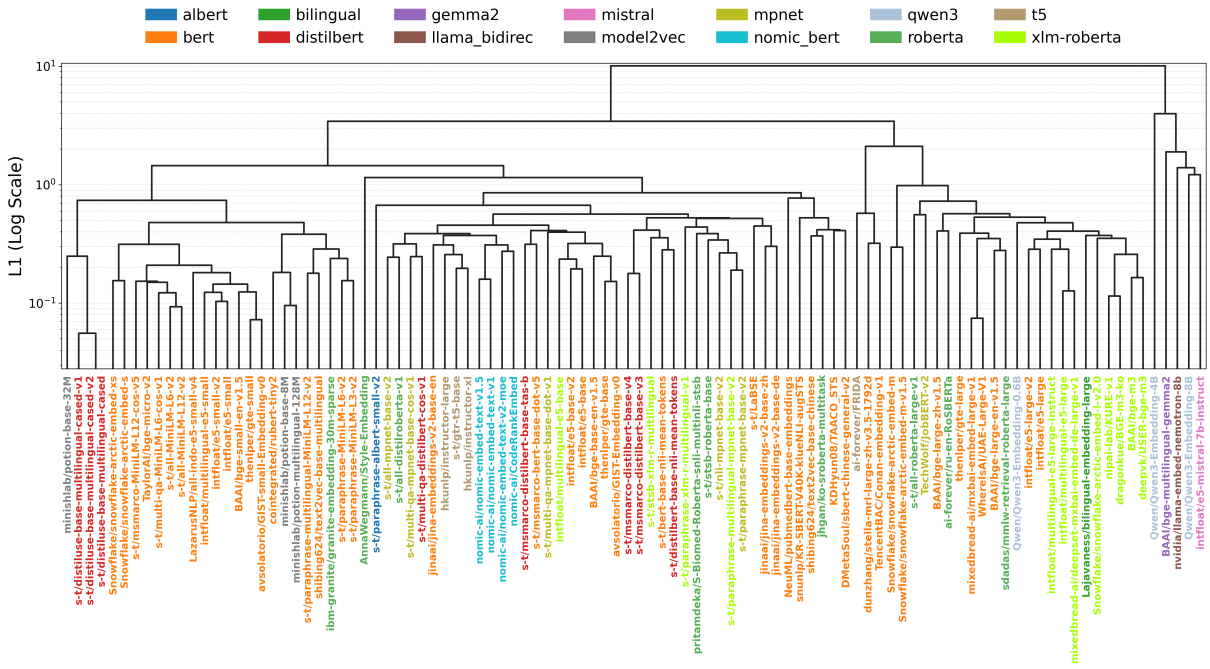


Figure 8: Zoomed-in hierarchical clustering of the top 100 most-downloaded encoders, coloured by model type. ℓ_1 values are reported in log scale for better visualisation.

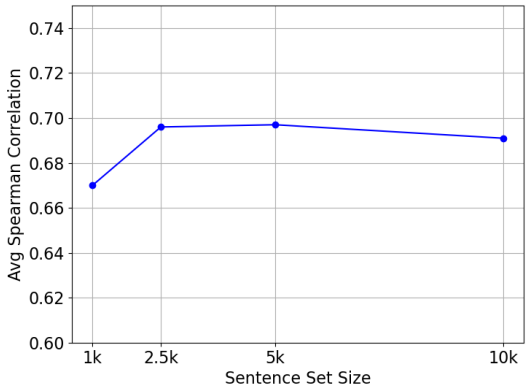
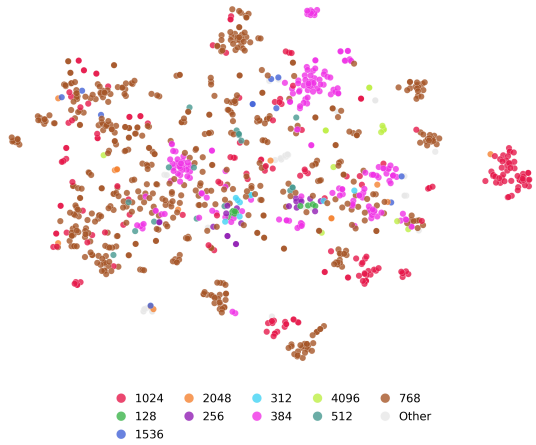
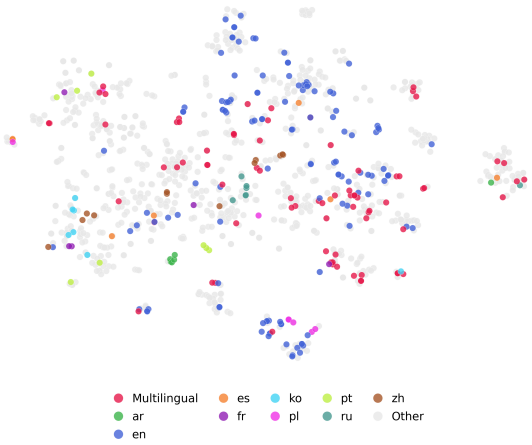


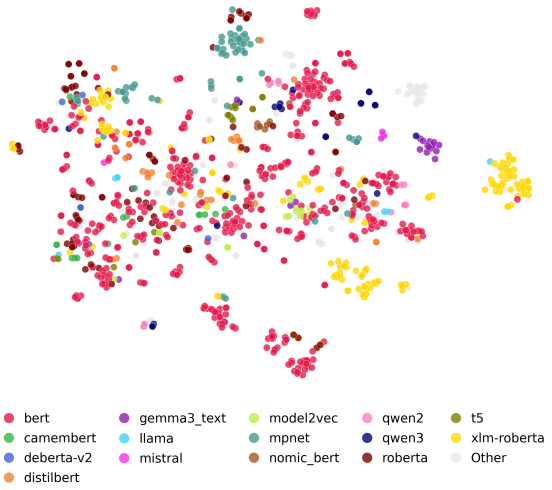
Figure 9: Average Spearman Correlation between the true and predicted performance on 68 MTEB tasks, based on sentence set sizes.



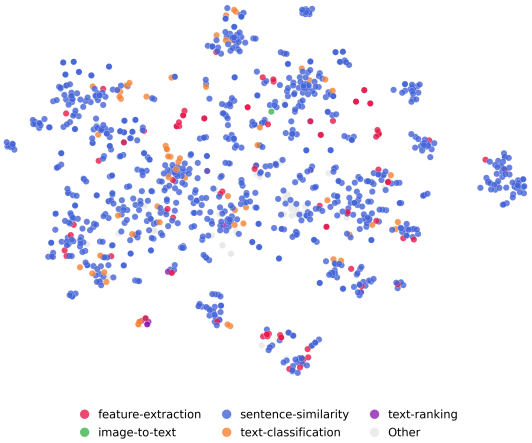
(a) Dimensionality



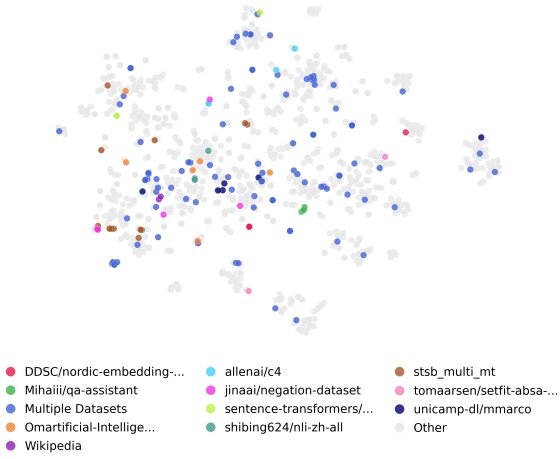
(b) Pre-trained Language



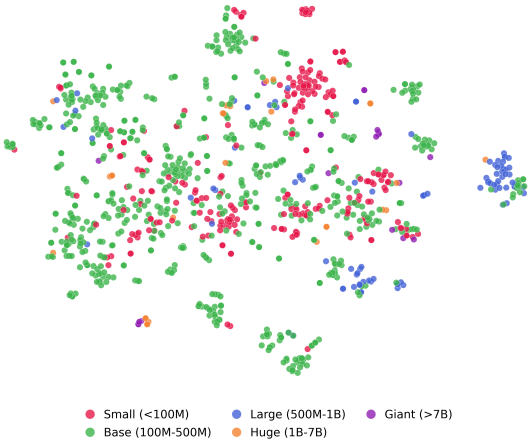
(c) Encoder Type



(d) Training Task



(e) Training Dataset



(f) Encoder Parameter Size

Figure 10: Maps created with a sentence set of size 5000.

#	Encoder Name	# Params	Dimensionality
1	Alibaba-NLP/gte-base-en-v1.5	136.8M	768
2	Alibaba-NLP/gte-large-en-v1.5	434.1M	1024
3	Alibaba-NLP/gte-multilingual-base	305.4M	768
4	BAAI/bge-base-en-v1.5	109.5M	768
5	BAAI/bge-large-en-v1.5	335.1M	1024
6	BAAI/bge-small-en-v1.5	33.4M	384
7	Gameselo/STS-multilingual-mpnet-base-v2	278.0M	768
8	HIT-TMG/KaLM-embedding-multilingual-mini-instruct-v1	494.0M	896
9	HIT-TMG/KaLM-embedding-multilingual-mini-instruct-v1.5	494.0M	896
10	HIT-TMG/KaLM-embedding-multilingual-mini-v1	494.0M	896
11	Lajavaness/bilingual-embedding-base	278.0M	768
12	Lajavaness/bilingual-embedding-large	559.9M	1024
13	Lajavaness/bilingual-embedding-small	117.7M	384
14	Linq-AI-Research/Linq-Embed-Mistral	7.11B	4096
15	Mihaiii/Bulbasaur	17.4M	384
16	Mihaiii/Ivysaur	22.7M	384
17	Mihaiii/Venusaur	15.6M	384
18	Mihaiii/gte-micro-v4	19.2M	384
19	SGPT-1.3B-weightedmean-msmarco-specb-bitfit	1.34B (Est)	2048
20	SGPT-125M-weightedmean-msmarco-specb-bitfit	137.8M (Est)	768
21	SGPT-125M-weightedmean-nli-bitfit	137.8M (Est)	768
22	SGPT-5.8B-weightedmean-msmarco-specb-bitfit	5.87B (Est)	4096
23	Omartificial/Arabert-all-nli-triplet-Matryoshka	135.2M	768
24	Omartificial/Arabic-MiniLM-L12-v2-all-nli-triplet	117.7M	384
25	Omartificia/Arabic-all-nli-triplet-Matryoshka	278.0M	768
26	Omartificial/Arabic-labse-Matryoshka	470.9M	768
27	Omartificial/Arabic-mpnet-base-all-nli-triplet	109.5M	768
28	Omartificial/Marbert-all-nli-triplet-Matryoshka	162.8M	768
29	OrcaDB/cde-small-v1	281.1M	768
30	OrcaDB/gte-base-en-v1.5	136.8M	768
31	Salesforce/SFR-Embedding-2_R	7.11B	4096
32	Salesforce/SFR-Embedding-Mistral	7.11B	4096
33	SmartComponents/bge-micro-v2	8.7M (Est)	384
34	Snowflake/snowflake-arctic-embed-l	334.1M	1024
35	Snowflake/snowflake-arctic-embed-l-v2.0	567.8M	1024
36	Snowflake/snowflake-arctic-embed-m	108.9M	768
37	Snowflake/snowflake-arctic-embed-m-long	136.7M	768
38	Snowflake/snowflake-arctic-embed-m-v1.5	108.9M	768
39	Snowflake/snowflake-arctic-embed-s	33.2M	384
40	Snowflake/snowflake-arctic-embed-xs	22.6M	384
41	TaylorAI/bge-micro	17.4M	384
42	TaylorAI/bge-micro-v2	17.4M	384
43	TaylorAI/gte-tiny	22.7M	384
44	WhereIsAI/UAE-Large-V1	335.1M	1024
45	aari1995/German_Semantic_STS_V2	335.7M	1024
46	abhinand/MedEmbed-small-v0.1	33.4M	384
47	ai-forever/ru-en-RoSBERTa	403.7M	1024
48	arkohut/jina-embeddings-v2-base-en	137.4M	768
49	avsolatorio/GIST-Embedding-v0	109.5M	768
50	avsolatorio/GIST-all-MiniLM-L6-v2	22.7M	384
51	avsolatorio/GIST-large-Embedding-v0	335.1M	1024
52	avsolatorio/GIST-small-Embedding-v0	33.4M	384
53	bigscience/sgpt-bloom-7b1-msmarco	7.07B (Est)	4096
54	cointegrated/rubert-tiny2	29.4M	312
55	corto-ai/nomic-embed-text-v1	136.7M	768
56	deepvk/USER-base	124.0M	768
57	ggrn/e5-small-v2	33.4M (Est)	384
58	hkunlp/instructor-base	110.2M (Est)	768
59	hkunlp/instructor-large	335.7M (Est)	768
60	hkunlp/instructor-xl	1.24B (Est)	768
61	ibm-granite/granite-embedding-125m-english	124.6M	768
62	ibm-granite/granite-embedding-278m-multilingual	278.0M	768
63	ibm-granite/granite-embedding-30m-english	30.3M	384
64	infgrad/stella-base-en-v2	54.8M (Est)	768
65	intfloat/e5-base	109.5M	768
66	intfloat/e5-base-v2	109.5M	768
67	intfloat/e5-large	335.1M	1024
68	intfloat/e5-large-v2	335.1M	1024

#	Encoder Name	# Params	Dimensionality
69	intfloat/e5-mistral-7b-instruct	7.11B	4096
70	intfloat/e5-small	33.4M	384
71	intfloat/e5-small-v2	33.4M	384
72	intfloat/multilingual-e5-base	278.0M	768
73	intfloat/multilingual-e5-large	559.9M	1024
74	intfloat/multilingual-e5-large-instruct	559.9M	1024
75	intfloat/multilingual-e5-small	117.7M	384
76	jinaai/jina-embedding-b-en-v1	109.6M (Est)	768
77	jinaai/jina-embedding-l-en-v1	335.0M (Est)	1024
78	jinaai/jina-embedding-s-en-v1	35.3M (Est)	512
79	jinaai/jina-embeddings-v2-base-en	137.4M	768
80	jinaai/jina-embeddings-v2-small-en	32.7M	512
81	jxm/cde-small-v1	281.1M	768
82	katanemo/bge-large-en-v1.5	335.1M	1024
83	khoa-klaytn/bge-base-en-v1.5-angle	109.5M	768
84	liddlefish/privacy_embedding_rag_10k_base_15_final	109.5M	768
85	llmrails/ember-v1	335.1M	1024
86	minishlab/M2V_base_output	7.6M	256
87	minishlab/potion-base-2M	1.9M	64
88	minishlab/potion-base-4M	3.8M	128
89	minishlab/potion-base-8M	7.6M	256
90	mixedbread-ai/mxbai-embed-2d-large-v1	335.1M	1024
91	mixedbread-ai/mxbai-embed-large-v1	335.1M	1024
92	nomic-ai/nomic-embed-text-v1	136.7M	768
93	nomic-ai/nomic-embed-text-v1-ablated	136.7M (Est)	768
94	nomic-ai/nomic-embed-text-v1-unsupervised	136.7M (Est)	768
95	nomic-ai/nomic-embed-text-v1.5	136.7M	768
96	sdadas/mmlw-e5-base	278.0M	768
97	sdadas/mmlw-e5-large	559.9M	1024
98	sdadas/mmlw-e5-small	117.7M	384
99	sdadas/mmlw-roberta-base	124.4M	768
100	sdadas/mmlw-roberta-large	435.0M	1024
101	sentence-transformers/LaBSE	470.9M	768
102	sentence-transformers/all-MiniLM-L12-v2	33.4M	384
103	sentence-transformers/all-MiniLM-L6-v2	22.7M	384
104	sentence-transformers/all-mpnet-base-v2	109.5M	768
105	paraphrase-multilingual-MiniLM-L12-v2	117.7M	384
106	paraphrase-multilingual-mpnet-base-v2	278.0M	768
107	sergeyzh/LaBSE-ru-turbo	128.3M	768
108	sergeyzh/rubert-tiny-turbo	29.2M	312
109	shibing624/text2vec-base-multilingual	117.7M	384
110	thenlper/gte-base	109.5M	768
111	thenlper/gte-large	335.1M	1024
112	thenlper/gte-small	33.4M	384

Table 6: Full list of 112 encoders used in MTEB evaluation with parameter size and dimensionality.

O Implementation Details for Spearman Correlation between True and Predicted MTEB Performance

We conduct the performance prediction using elastic net regression (ElasticNetCV) from the scikit-learn library with a 5-fold cross-validation strategy. ElasticNetCV is linear regression with both ℓ_1 and ℓ_2 regularisation. The feature vectors are first normalised using standard scaler to ensure zero mean and unit variance. To mitigate overfitting, given the high dimensionality of our feature vectors (10,000) relative to the number of encoders (112), we first use PCA to project the input feature vectors from 10,000 to 50 dimensions. We rely on the internal 5-fold cross-validation mechanism of ElasticNetCV to automatically tune the regularisation strength α, which is selected from a log-scale grid of 100 values starting from α_{max} (the smallest value penalising all coefficients to zero) down to $\alpha_{max} \times 10^{-3}$.

P Full Results for MTEB Correlation With Feature Vectors

Table 7 shows the full results of the correlation between 68 MTEB task performance and our feature vectors of 112 encoders, along with p-value. We visualise task performance of the top 10 tasks Table 2 by min-max normalising the predicted and actual performance in the same figure Figure 11. The Spearman correlation between the predicted and average performance is 0.766, indicating a strong correlation. This shows our encoders in the map are connected to their downstream task performance.

Figure 12 visualise the 112 encoders in the map coloured by the QRE value (sum of feature vector) for each encoder. There are group patterns in the map, where encoders of similar QRE values tend to be close to each other.

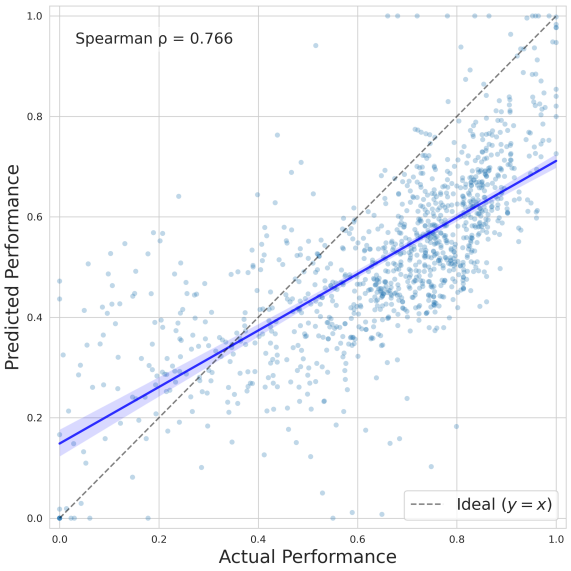


Figure 11: Predicted performance vs. actual performance for 112 encoders on top 10 tasks in Table 2.

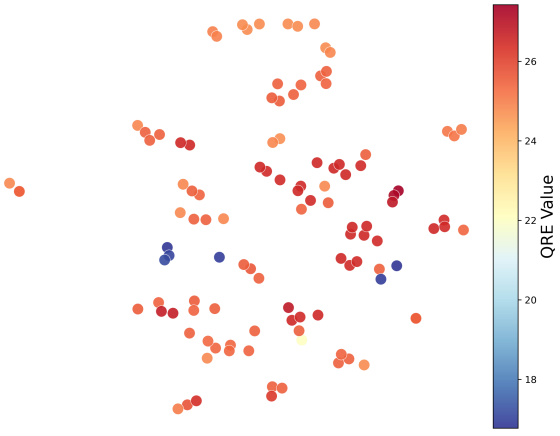


Figure 12: Submap of 112 encoders coloured by QRE values.

Task	Spearman	p-value	Pearson	p-value	Task Type
SciFact	0.900	2.14×10^{-41}	0.831	9.50×10^{-30}	Retrieval
NFCorpus	0.881	1.35×10^{-37}	0.803	1.61×10^{-26}	Retrieval
BiorxivClusteringS2S	0.879	3.02×10^{-37}	0.778	5.63×10^{-24}	Clustering
ArxivClusteringS2S	0.877	7.50×10^{-37}	0.814	9.78×10^{-28}	Clustering
ArxivClusteringP2P	0.868	2.94×10^{-35}	0.784	1.50×10^{-24}	Clustering
HotpotQA	0.859	9.50×10^{-34}	0.803	1.95×10^{-26}	Retrieval
SciDocsRR	0.850	1.93×10^{-32}	0.792	2.48×10^{-25}	Reranking
BiorxivClusteringP2P	0.842	2.74×10^{-31}	0.739	1.45×10^{-20}	Clustering
TwentyNewsgroupsClustering	0.832	7.39×10^{-30}	0.771	2.64×10^{-23}	Clustering
ImdbClassification	0.831	9.36×10^{-30}	0.775	1.23×10^{-23}	Classification
RedditClustering	0.829	1.39×10^{-29}	0.760	2.37×10^{-22}	Clustering
RedditClusteringP2P	0.827	3.10×10^{-29}	0.767	6.59×10^{-23}	Clustering
StackExchangeClustering	0.823	9.79×10^{-29}	0.760	2.58×10^{-22}	Clustering
StackOverflowDupQuestions	0.819	2.73×10^{-28}	0.778	6.33×10^{-24}	Reranking
CQADupstackPhysicsRetrieval	0.816	6.82×10^{-28}	0.752	1.19×10^{-21}	Retrieval
FiQA2018	0.814	1.13×10^{-27}	0.789	5.45×10^{-25}	Retrieval
CQADupstackGamingRetrieval	0.802	2.37×10^{-26}	0.754	8.04×10^{-22}	Retrieval
ArguAna	0.796	1.10×10^{-25}	0.731	6.11×10^{-20}	Retrieval
MedrxivClusteringS2S	0.792	2.48×10^{-25}	0.724	1.96×10^{-19}	Clustering
DBPedia	0.780	3.66×10^{-24}	0.744	5.75×10^{-21}	Retrieval
TwitterSemEval2015	0.775	1.14×10^{-23}	0.628	1.32×10^{-13}	PairClassification
STS14	0.771	2.60×10^{-23}	0.599	2.95×10^{-12}	STS
SCIDOCS	0.771	2.60×10^{-23}	0.712	1.43×10^{-18}	Retrieval
CQADupstackEnglishRetrieval	0.768	4.64×10^{-23}	0.720	3.64×10^{-19}	Retrieval
SICK-R	0.767	6.54×10^{-23}	0.541	7.46×10^{-10}	STS
CQADupstackMathematicaRetrieval	0.764	1.09×10^{-22}	0.701	7.27×10^{-18}	Retrieval
CQADupstackAndroidRetrieval	0.760	2.34×10^{-22}	0.733	3.81×10^{-20}	Retrieval
StackExchangeClusteringP2P	0.758	3.91×10^{-22}	0.737	2.03×10^{-20}	Clustering
CQADupstackRetrieval	0.757	4.24×10^{-22}	0.717	5.77×10^{-19}	Retrieval
CQADupstackProgrammersRetrieval	0.756	5.31×10^{-22}	0.715	7.89×10^{-19}	Retrieval
CQADupstackGisRetrieval	0.755	6.37×10^{-22}	0.728	9.96×10^{-20}	Retrieval
ClimateFEVER	0.754	7.79×10^{-22}	0.658	3.12×10^{-15}	Retrieval
Banking77Classification	0.752	1.17×10^{-21}	0.700	9.00×10^{-18}	Classification
CQADupstackStatsRetrieval	0.749	2.11×10^{-21}	0.682	1.22×10^{-16}	Retrieval
CQADupstackUnixRetrieval	0.748	2.47×10^{-21}	0.716	7.52×10^{-19}	Retrieval
AskUbuntuDupQuestions	0.748	2.79×10^{-21}	0.757	5.13×10^{-22}	Reranking
QuoraRetrieval	0.745	4.30×10^{-21}	0.673	4.38×10^{-16}	Retrieval
MedrxivClusteringP2P	0.740	1.10×10^{-20}	0.677	2.38×10^{-16}	Clustering
STS16	0.736	2.45×10^{-20}	0.609	1.10×10^{-12}	STS
FEVER	0.731	6.07×10^{-20}	0.682	1.32×10^{-16}	Retrieval
CQADupstackWordpressRetrieval	0.728	9.34×10^{-20}	0.711	1.64×10^{-18}	Retrieval
CQADupstackTexRetrieval	0.726	1.24×10^{-19}	0.686	7.39×10^{-17}	Retrieval
CQADupstackWebmastersRetrieval	0.721	3.05×10^{-19}	0.700	8.67×10^{-18}	Retrieval
STS13	0.713	1.14×10^{-18}	0.679	1.98×10^{-16}	STS
NQ	0.685	8.34×10^{-17}	0.664	1.41×10^{-15}	Retrieval
AmazonPolarityClassification	0.681	1.48×10^{-16}	0.584	1.34×10^{-11}	Classification
TRECCOVID	0.667	1.01×10^{-15}	0.629	1.14×10^{-13}	Retrieval
STS12	0.652	6.54×10^{-15}	0.430	2.25×10^{-6}	STS
BIOSSES	0.645	1.60×10^{-14}	0.613	6.50×10^{-13}	STS
STS15	0.635	5.25×10^{-14}	0.510	9.14×10^{-9}	STS
MindSmallReranking	0.592	6.02×10^{-12}	0.542	6.50×10^{-10}	Reranking
MSMARCO	0.578	2.46×10^{-11}	0.497	2.51×10^{-8}	Retrieval
SprintDuplicateQuestions	0.540	7.89×10^{-10}	0.526	2.60×10^{-9}	PairClassification
Touche2020	0.538	9.90×10^{-10}	0.506	1.23×10^{-8}	Retrieval
EmotionClassification	0.528	2.14×10^{-9}	0.282	0.003	Classification
TwitterURLCorpus	0.522	3.68×10^{-9}	0.437	1.43×10^{-6}	PairClassification
TweetSentimentExtractionClassification	0.500	1.99×10^{-8}	0.320	0.001	Classification
ToxicConversationsClassification	0.465	2.40×10^{-7}	0.368	6.48×10^{-5}	Classification
AmazonReviewsClassification	0.446	8.38×10^{-7}	0.440	1.20×10^{-6}	Classification
STS22	0.321	5.59×10^{-4}	0.278	0.003	STS
MTOPIntentClassification	0.296	0.002	0.319	6.05×10^{-4}	Classification
MTOPDomainClassification	0.275	0.003	0.287	0.002	Classification
STSBenchmark	0.248	0.008	0.132	0.165	STS
AmazonCounterfactualClassification	0.247	0.009	0.260	0.006	Classification

Task	Spearman	p-value	Pearson	p-value	Task Type
MassiveScenarioClassification	0.209	0.027	0.252	0.007	Classification
MassiveIntentClassification	0.198	0.036	0.240	0.011	Classification
SummEval	0.159	0.094	0.196	0.039	Summarization
STS17	0.157	0.098	0.158	0.096	STS

Table 7: Spearman and Pearson correlations between the true and predicted task performance of 68 tasks, sorted in a descending order of Spearman correlation.

#	Encoder Name	Encoder Type	# Params	Dimensionality
1	sentence-transformers/all-MiniLM-L6-v2	bert	22.7M	384
2	sentence-transformers/all-mpnet-base-v2	mpnet	109.5M	768
3	sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2	bert	117.7M	384
4	sentence-transformers/gtr-t5-base	t5	109.6M	768
5	BAAI/bge-m3	xlm-roberta	567.8M (Est)	1024
6	sentence-transformers/paraphrase-multilingual-mpnet-base-v2	xlm-roberta	278.0M	768
7	sentence-transformers/multi-qa-mpnet-base-dot-v1	mpnet	109.5M	768
8	Qwen/Qwen3-Embedding-0.6B	qwen3	595.8M	1024
9	BAAI/bge-large-en-v1.5	bert	335.1M	1024
10	Alibaba-NLP/gte-large-en-v1.5	new	434.1M	1024
11	BAAI/bge-base-en-v1.5	bert	109.5M	768
12	BAAI/bge-small-en-v1.5	bert	33.4M	384
13	sentence-transformers/all-MiniLM-L12-v2	bert	33.4M	384
14	sentence-transformers/msmarco-distilbert-base-tas-b	distilbert	66.4M	768
15	sentence-transformers/paraphrase-MiniLM-L6-v2	bert	22.7M	384
16	intfloat/multilingual-e5-large	xlm-roberta	559.9M	1024
17	intfloat/multilingual-e5-small	bert	117.7M	384
18	mixedbread-ai/mxbai-embed-large-v1	bert	335.1M	1024
19	intfloat/multilingual-e5-base	xlm-roberta	278.0M	768
20	nomic-ai/nomic-embed-text-v1.5	nomic_bert	136.7M	768
21	sentence-transformers/paraphrase-mpnet-base-v2	mpnet	109.5M	768
22	Alibaba-NLP/gte-multilingual-base	new	305.4M	768
23	intfloat/e5-base-v2	bert	109.5M	768
24	Snowflake/snowflake-arctic-embed-l-v2.0	xlm-roberta	567.8M	1024
25	sentence-transformers/multi-qa-MiniLM-L6-cos-v1	bert	22.7M	384
26	WhereIsAI/UAE-Large-V1	bert	335.1M	1024
27	intfloat/multilingual-e5-large-instruct	xlm-roberta	559.9M	1024
28	sentence-transformers/stsb-xlm-r-multilingual	xlm-roberta	278.0M	768
29	minishlab/potion-base-32M	model2vec	32.3M	512
30	intfloat/e5-large-v2	bert	335.1M	1024
31	thenlper/gte-large	bert	335.1M	1024
32	sentence-transformers/LaBSE	bert	470.9M	768
33	sentence-transformers/distiluse-base-multilingual-cased-v2	distilbert	134.7M	512
34	dunzhang/stella-mrl-large-zh-v3.5-1792d	bert	325.5M	1792
35	sentence-transformers/distiluse-base-multilingual-cased-v1	distilbert	134.7M	512
36	nomic-ai/nomic-embed-text-v1	nomic_bert	136.7M	768
37	thenlper/gte-small	bert	33.4M	384
38	sentence-transformers/all-roberta-large-v1	roberta	355.4M	1024
39	BAAI/bge-large-zh-v1.5	bert	325.6M (Est)	1024
40	cointegrated/rubert-tiny2	bert	29.4M	312
41	google/embeddinggemma-300m	Unknown	307.6M (Est)	768
42	sentence-transformers/paraphrase-MiniLM-L3-v2	bert	17.4M	384
43	Qwen/Qwen3-Embedding-8B	qwen3	7.57B	4096
44	BAAI/bge-multilingual-gemma2	gemma2	9.24B	3584
45	sentence-transformers/multi-qa-mpnet-base-cos-v1	mpnet	109.5M	768
46	Snowflake/snowflake-arctic-embed-m	bert	108.9M	768
47	sentence-transformers/bert-base-nli-mean-tokens	bert	109.5M	768
48	sentence-transformers/msmarco-MiniLM-L12-cos-v5	bert	33.4M	384
49	snunlp/KR-SBERT-V40K-klueNLI-augSTS	bert	116.8M (Est)	768
50	hkunlp/instructor-xl	t5	1.24B (Est)	768
51	intfloat/e5-large	bert	335.1M	1024
52	nomic-ai/nomic-embed-text-v2-moe	nomic_bert	475.3M	768
53	Qwen/Qwen3-Embedding-4B	qwen3	4.02B	2560
54	sentence-transformers/msmarco-bert-base-dot-v5	bert	109.5M	768
55	sentence-transformers/all-distilroberta-v1	roberta	82.1M	768
56	Alibaba-NLP/gte-base-en-v1.5	new	136.8M	768
57	sentence-transformers/stsb-roberta-base	roberta	124.6M	768
58	TencentBAC/Conan-embedding-v1	bert	325.5M	1792
59	shibing624/text2vec-base-chinese	bert	102.3M	768
60	minishlab/potion-base-8M	model2vec	7.6M	256
61	jhgan/ko-sroberta-multitask	roberta	110.6M (Est)	768

#	Encoder Name	Encoder Type	# Params	Dimensionality
62	pritamdeka/S-Biomed-Roberta-snli-multinli-stsb	roberta	124.7M (Est)	768
63	sentence-transformers/distilbert-base-nli-mean-tokens	distilbert	66.4M	768
64	intfloat/e5-mistral-7b-instruct	mistral	7.11B	4096
65	intfloat/e5-small-v2	bert	33.4M	384
66	hkunlp/instructor-large	t5	335.7M (Est)	768
67	TechWolf/JobBERT-v2	mpnet	109.5M	1024
68	jinaai/jina-embeddings-v2-base-zh	bert	160.8M	768
69	intfloat/e5-small	bert	33.4M	384
70	sentence-transformers/paraphrase-albert-small-v2	albert	11.7M	768
71	OrcaDB/gte-base-en-v1.5	new	136.8M	768
72	avsolatorio/GIST-small-Embedding-v0	bert	33.4M	384
73	sdadas/mmlw-retrieval-roberta-large	roberta	435.0M	1024
74	KDHyun08/TAACO_STS	bert	110.6M (Est)	768
75	sentence-transformers/paraphrase-xlm-r-multilingual-v1	xlm-roberta	278.0M	768
76	mixedbread-ai/deepset-mxbai-embed-de-large-v1	xlm-roberta	487.1M	1024
77	sentence-transformers/msmarco-distilbert-base-v4	distilbert	66.4M	768
78	LazarusNLP/all-indo-e5-small-v4	bert	117.7M	384
79	Snowflake/snowflake-arctic-embed-xs	bert	22.6M	384
80	ai-forever/FRIDA	t5	823.4M	1536
81	minishlab/potion-multilingual-128M	model2vec	128.1M	256
82	nomic-ai/CodeRankEmbed	nomic_bert	136.7M	768
83	ai-forever/ru-en-RoSBERTa	roberta	403.7M	1024
84	jinaai/jina-embeddings-v2-base-en	bert	137.4M	768
85	sentence-transformers/distiluse-base-multilingual-cased	distilbert	134.7M	512
86	NeuML/pubmedbert-base-embeddings	bert	109.5M	768
87	Snowflake/snowflake-arctic-embed-s	bert	33.2M	384
88	TaylorAI/bge-micro-v2	bert	17.4M	384
89	nlpai-lab/KURE-v1	xlm-roberta	567.8M	1024
90	Lajavaness/bilingual-embedding-large	bilingual	559.9M	1024
91	sentence-transformers/multi-qa-distilbert-cos-v1	distilbert	66.4M	768
92	intfloat/e5-base	bert	109.5M	768
93	dragonkue/BGE-m3-ko	xlm-roberta	567.8M	1024
94	DMetaSoul/sbert-chinese-general-v2	bert	102.3M (Est)	768
95	thenlper/gte-base	bert	109.5M	768
96	nvidia/llama-embed-nemotron-8b	llama_bidirec	7.50B	4096
97	deepvk/USER-bge-m3	xlm-roberta	359.0M	1024
98	sentence-transformers/nli-mpnet-base-v2	mpnet	109.5M	768
99	AnnaWegmann/Style-Embedding	roberta	124.7M (Est)	768
100	shibing624/text2vec-base-multilingual	bert	117.7M	384
101	ibm-granite/granite-embedding-30m-sparse	roberta	30.3M	384
102	Snowflake/snowflake-arctic-embed-m-v1.5	bert	108.9M	768
103	avsolatorio/GIST-Embedding-v0	bert	109.5M	768
104	jinaai/jina-embeddings-v2-base-de	bert	160.8M	768
105	sentence-transformers/msmarco-distilbert-base-v3	distilbert	66.4M	768
106	jhgan/ko-sbert-nli	bert	110.6M (Est)	768
107	ibm-granite/granite-embedding-30m-english	roberta	30.3M	384
108	hiieu/halong_embedding	xlm-roberta	278.0M	768
109	moka-ai/m3e-base	bert	102.3M	768
110	mlsa-iai-msu-lab/sci-rus-tiny	roberta	23.4M	312
111	sentence-transformers-testing/stsb-bert-tiny-safetensors	bert	4.4M	128
112	pritamdeka/S-PubMedBert-MS-MARCO	bert	109.5M (Est)	768
113	sentence-transformers/multi-qa-MiniLM-L6-dot-v1	bert	22.7M	384
114	Snowflake/snowflake-arctic-embed-m-long	nomic_bert	136.7M	768
115	sentence-transformers/stsb-roberta-large	roberta	355.4M	1024
116	embaas/sentence-transformers-e5-large-v2	bert	335.2M (Est)	1024
117	sentence-transformers/clip-ViT-B-32-multilingual-v1	distilbert	134.7M	512
118	dariolopez/bge-m3-es-legal-tmp-6	xlm-roberta	567.8M	1024
119	BAAI/bge-base-zh-v1.5	bert	102.3M (Est)	768
120	krlvi/sentence-msmarco-bert-base-dot-v5-nlpl-code_search_net	bert	109.5M (Est)	768
121	sentence-transformers/paraphrase-MiniLM-L12-v2	bert	33.4M	384
122	jhgan/ko-sbert-sts	bert	110.6M (Est)	768

#	Encoder Name	Encoder Type	# Params	Dimensionality
123	hiiamsid/sentence_similarity_spanish_es	bert	109.9M	768
124	sentence-transformers/bert-large-nli-max-tokens	bert	335.1M	1024
125	jinaai/jina-embeddings-v2-base-code	bert	160.9M	768
126	nomic-ai/nomic-embed-code	qwen2	7.07B	3584
127	sentence-transformers/multi-qa-distilbert-dot-v1	distilbert	66.4M	768
128	Salesforce/SFR-Embedding-Mistral	mistral	7.11B	4096
129	moka-ai/m3e-large	bert	325.6M (Est)	1024
130	thenlper/gte-large-zh	bert	325.5M	1024
131	keepitreal/vietnamese-sbert	roberta	135.0M (Est)	768
132	Snowflake/snowflake-arctic-embed-l	bert	334.1M	1024
133	sdadas/mmlw-roberta-base	roberta	124.4M	768
134	AITeamVN/Vietnamese_Embedding	xlm-roberta	567.8M	1024
135	TaylorAI/gte-tiny	bert	22.7M	384
136	sentence-transformers/distilbert-base-nli-stsb-mean-tokens	distilbert	66.4M	768
137	sentence-transformers/msmarco-MiniLM-L6-v3	bert	22.7M	384
138	avsolatorio/GIST-all-MiniLM-L6-v2	bert	22.7M	384
139	kuelumbus/polyBERT	deberta-v2	25.2M (Est)	600
140	sentence-transformers/msmarco-distilbert-cos-v5	distilbert	66.4M	768
141	datasocietyco/bge-base-en-v1.5-course-recommender-v5	bert	109.5M	768
142	pritamdeka/BioBERT-mnli-snli-scinli-scitail-mednli-stsb	bert	108.3M (Est)	768
143	upskyy/bge-m3-korean	xlm-roberta	567.8M	1024
144	OrcaDB/distilbert-base-uncased	distilbert	66.4M	768
145	lier007/xiaobu-embedding-v2	bert	325.5M	1792
146	abhinand/MedEmbed-base-v0.1	bert	109.5M	768
147	jhgan/ko-sbert-multitask	bert	110.6M (Est)	768
148	minishlab/potion-base-2M	model2vec	1.9M	64
149	sentence-transformers/stsb-roberta-base-v2	roberta	124.6M	768
150	sdadas/mmlw-roberta-large	roberta	435.0M	1024
151	huyydangg/DEk21_hcmute_embedding	roberta	135.0M	768
152	llamaindex/vdr-2b-multi-v1	qwen2_vl	2.21B	1536
153	dangvantuan/french-document-embedding	Bilingual	305.4M	768
154	naufalihsan/indonesian-sbert-large	bert	335.2M (Est)	1024
155	michaelfeil/bge-small-en-v1.5	bert	33.4M	384
156	dangvantuan/sentence-camembert-base	camembert	110.6M	768
157	sentence-transformers/paraphrase-distilroberta-base-v1	roberta	82.1M	768
158	sentence-transformers/msmarco-distilbert-dot-v5	distilbert	66.4M	768
159	abdur197/fine-tuned-column-mapper	bert	22.7M	384
160	ibm-granite/granite-embedding-125m-english	roberta	124.6M	768
161	deutsche-telekom/gbert-large-paraphrase-euclidean	bert	335.8M (Est)	1024
162	l3cube-pune/marathi-sentence-similarity-sbert	bert	237.6M (Est)	768
163	davanstrien/headline-similarity	bert	110.6M (Est)	768
164	sentence-transformers/sentence-t5-base	t5	109.6M	768
165	akhooli/Arabic-SBERT-100K	bert	135.2M	768
166	abhinand/MedEmbed-small-v0.1	bert	33.4M	384
167	BAAI/bge-m3-unsupervised	xlm-roberta	567.8M	1024
168	avemio/German-RAG-UAE-LARGE-V1-TRIPLES-MERGED-HESSIAN-AI	bert	335.1M	1024
169	NetherlandsForensicInstitute/robbert-2022-dutch-sentence-transformers	roberta	118.9M	768
170	shibing624/text2vec-base-chinese-paraphrase	ernie	117.9M	768
171	FremyCompany/BioLORD-2023	mpnet	109.5M	768
172	ibm-granite/granite-embedding-278m-multilingual	xlm-roberta	278.0M	768
173	dguzh/geo-all-MiniLM-L6-v2	bert	22.7M	384
174	sentence-transformers/sentence-t5-xl	t5	1.24B	768
175	jinaai/jina-embeddings-v2-small-en	bert	32.7M	512
176	Shashwat13333/bge-base-en-v1.5_v4	bert	109.5M	768
177	sonoisa/sentence-bert-base-ja-mean-tokens	Unknown	110.6M	768
178	sentence-transformers/allenai-specter	bert	109.9M	768
179	bkai-foundation-models/vietnamese-bi-encoder	roberta	135.0M	768
180	abhinand/MedEmbed-large-v0.1	bert	335.1M	1024
181	MoralHazard/NSFW-GIST-small	bert	33.4M	384
182	jinaai/jina-code-embeddings-0.5b	qwen2	494.0M	896
183	optimum-intel-internal-testing/stsb-bert-tiny-safetensors	bert	4.4M	128

#	Encoder Name	Encoder Type	# Params	Dimensionality
184	ai-sage/Giga-Embeddings-instruct	gigarembed	3.45B	2048
185	dragonkue/snowflake-arctic-embed-l-v2.0-ko	xlm-roberta	567.8M	1024
186	aari1995/German_Semantic_STS_V2	bert	335.7M	1024
187	KBLab/sentence-bert-swedish-cased	bert	124.7M	768
188	emrecan/bert-base-turkish-cased-mean-nli-stsb-tr	bert	110.6M (Est)	768
189	tencent/KaLM-Embedding-Gemma3-12B-2511	gemma3_text	11.77B	3840
190	nickprock/sentence-bert-base-italian-xxl-uncased	bert	110.7M	768
191	FremyCompany/BioLORD-2023-M	xlm-roberta	278.0M	768
192	pritamdeka/S-BioBert-snli-multinli-stsb	bert	108.3M (Est)	768
193	llmrails/ember-v1	bert	335.1M	1024
194	mixedbread-ai/mxbai-embed-xsmall-v1	bert	24.1M	384
195	Salesforce/SFR-Embedding-2_R	mistral	7.11B	4096
196	hkunlp/instructor-base	t5	110.2M (Est)	768
197	sentence-transformers/nli-distilroberta-base-v2	roberta	82.1M	768
198	sentence-transformers/roberta-large-nli-stsb-mean-tokens	roberta	355.4M	1024
199	katanemo/bge-large-en-v1.5	bert	335.1M	1024
200	Salesforce/SFR-Embedding-Code-400M_R	new	434.1M	1024
201	FremyCompany/BioLORD-2023-M-Dutch-InContext-v1	xlm-roberta	278.0M	768
202	Hiveurban/multilingual-e5-base	xlm-roberta	278.0M	768
203	sergeyzh/rubert-mini-frida	bert	32.3M	312
204	s2593817/sft-sql-embedding	mpnet	109.5M	768
205	sentence-transformers/embeddinggemma-300m-medical	gemma3_text	302.9M	768
206	dwulff/mpnet-personality	mpnet	109.5M	768
207	minishlab/potion-retrieval-32M	model2vec	32.3M	512
208	sentence-transformers/gtr-t5-large	t5	334.9M	768
209	dariolopez/roberta-base-bne-finetuned-msmarco-qa-es	roberta	124.7M (Est)	768
210	sentence-transformers/sentence-t5-large	t5	334.9M	768
211	embaas/sentence-transformers-multilingual-e5-large	xlm-roberta	559.9M (Est)	1024
212	FinLang/finance-embeddings-investopedia	bert	109.5M	768
213	HIT-TMG/KaLM-embedding-multilingual-mini-v1	qwen2	494.0M	896
214	ssmits/Qwen2-7B-Instruct-embed-base	qwen2	7.07B	3584
215	jinaai/jina-embeddings-v2-base-es	bert	160.9M	768
216	optimum-intel-internal-testing/bge-small-en-v1.5	bert	33.4M	384
217	dell-research-harvard/lt-wikidata-comp-en	mpnet	109.5M	768
218	infly/inf-retriever-v1-1.5b	qwen2	1.54B	1536
219	l3cube-pune/indic-sentence-similarity-sbert	bert	237.6M (Est)	768
220	sentence-transformers/bert-base-nli-stsb-mean-tokens	bert	109.5M	768
221	billatsectorflow/stella_en_400M_v5	new	435.2M	1024
222	dragonkue/multilingual-e5-small-ko	bert	117.7M	384
223	deutsche-telekom/gbert-large-paraphrase-cosine	bert	335.8M (Est)	1024
224	deepvk/USER-base	deberta	124.0M	768
225	sentence-transformers/stsb-mpnet-base-v2	mpnet	109.5M	768
226	FremyCompany/BioLORD-2023-C	mpnet	109.5M	768
227	sergeyzh/BERTA	bert	128.3M	768
228	bowphs/SPhilBerta	roberta	135.2M	768
229	upskyy/e5-large-korean	xlm-roberta	559.9M	1024
230	sentence-transformers/msmarco-MiniLM-L6-cos-v5	bert	22.7M	384
231	BAAI/bge-code-v1	qwen2	1.54B	1536
232	beademiguelperez/sentence-transformers-multilingual-e5-small	bert	117.7M	384
233	sentence-transformers-testing/stsb-bert-tiny-openvino	bert	4.4M	128
234	AI-Growth-Lab/PatentSBERTa	mpnet	109.5M (Est)	768
235	PORTULAN/serafim-335m-portuguese-pt-sentence-encoder-ir	bert	334.4M	1024
236	sentence-transformers/paraphrase-distilroberta-base-v2	roberta	82.1M	768
237	arkohut/jina-embeddings-v2-base-en	bert	137.4M	768
238	mhaseeb1604/bge-m3-law	xlm-roberta	567.8M	1024

#	Encoder Name	Encoder Type	# Params	Dimensionality
239	sentence-transformers/xlm-r-bert-base-nli-stsb-mean-tokens	xlm-roberta	278.0M	768
240	navteca/ms-marco-MiniLM-L-6-v2	bert	22.7M (Est)	384
241	sentence-transformers/roberta-base-nli-stsb-mean-tokens	roberta	124.6M	768
242	Linq-AI-Research/Linq-Embed-Mistral	mistral	7.11B	4096
243	NbAiLab/nb-sbert-base	bert	177.9M	768
244	flax-sentence-embeddings/st-codesearch-distilroberta-base	roberta	82.1M (Est)	768
245	sangmini/msmarco-cotmae-MiniLM-L12_en-ko-ja	bert	118.3M (Est)	1536
246	minishlab/M2V_base_output	model2vec	7.6M	256
247	tomaarsen/mpnet-base-nli-matryoshka	mpnet	109.5M	768
248	Vsevolod/company-names-similarity-sentence-transformer	bert	22.7M (Est)	384
249	malteos/scincl	bert	109.9M	768
250	minishlab/potion-base-4M	model2vec	3.8M	128
251	infgrad/stella-base-en-v2	bert	54.8M (Est)	768
252	TechWolf/JobBERT-v3	xlm-roberta	278.0M	1024
253	imvladikon/sentence-transformers-alephbert	bert	126.0M (Est)	768
254	sentence-transformers/facebook-dpr-ctx_encoder-single-nq-base	bert	109.5M	768
255	jaimevera1107/all-MiniLM-L6-v2-similarity-es	bert	22.7M (Est)	384
256	lealdaniel/comp-embedding-matching	mpnet	109.5M	768
257	sentence-transformers/xlm-r-distilroberta-base-paraphrase-v1	xlm-roberta	278.0M	768
258	krutrim-ai-labs/Vyakyarth	xlm-roberta	278.0M	768
259	bespin-global/klue-sroberta-base-continue-learning-by-mnr	roberta	110.6M	768
260	zeroentropy/zerank-1-small	qwen3	1.72B	2048
261	Lajavaness/bilingual-embedding-small	bilingual	117.7M	384
262	Lajavaness/sentence-camembert-large	camembert	336.7M	1024
263	sergeyzh/rubert-mini-sts	bert	32.4M	312
264	PORTULAN/serafim-100m-portuguese-pt-sentence-encoder-ir	bert	108.9M	768
265	valurank/MiniLM-L6-Keyword-Extraction	bert	22.7M (Est)	384
266	shihab17/bangla-sentence-transformer	xlm-roberta	278.0M	768
267	sentence-transformers/distilbert-multilingual-nli-stsb-quora-ranking	distilbert	134.7M	768
268	sentence-transformers/all-mpnet-base-v1	mpnet	109.5M	768
269	dariolopez/roberta-base-bne-finetuned-msmarco-qa-es-mnrl-mn	roberta	124.7M (Est)	768
270	Omartificial-Intelligence-Space/GATE-AraBert-v1	bert	135.2M	768
271	sentence-transformers/use-cmlm-multilingual	bert	472.0M	768
272	lealdaniel/comp-level-embeddings	bert	33.4M	384
273	jinaai/jina-embedding-b-en-v1	t5	109.6M (Est)	768
274	unsloth/embeddinggemma-300m	gemma3_text	302.9M	768
275	sentence-transformers/paraphrase-TinyBERT-L6-v2	bert	67.0M	768
276	optimum-intel-internal-testing/all-MiniLM-L6-v2	bert	22.7M	384
277	optimum-intel-internal-testing/all-mpnet-base-v2	mpnet	109.5M	768
278	Mihaiii/Venusaur	bert	15.6M	384
279	maidalun1020/bce-embedding-base_v1	xlm-roberta	278.1M (Est)	768
280	MongoDB/mdbr-leaf-mt	bert	22.6M	1024
281	Hiveurban/multilingual-e5-large-pooled	xlm-roberta	559.9M	1024
282	manu/bge-fr-en	xlm-roberta	567.8M	1024
283	infgrad/stella-large-zh-v2	bert	163.1M (Est)	1024
284	mpi-inno-comp/paecter	bert	344.7M	1024
285	OrcaDB/gte-small	bert	33.4M	384
286	DMetaSoul/sbert-chinese-general-v1	bert	102.3M (Est)	768
287	AITeamVN/Vietnamese_Embedding_v2	xlm-roberta	567.8M	1024
288	OpenVoiceOS/ovos-model2vec-intents-LaBSE	Unknown	128.3M	256
289	somosnlp-hackathon-2022/paraphrase-spanish-distilroberta	roberta	124.7M (Est)	768
290	Omartificial-Intelligence-Space/Arabic-Triplet-Matryoshka-V2	bert	135.2M	768
291	OrcaDB/clip-ViT-L-14	clip	427.6M	768
292	nickmuchi/setfit-finetuned-financial-text-classification	mpnet	109.5M (Est)	768

#	Encoder Name	Encoder Type	# Params	Dimensionality
293	sentence-transformers/msmarco-roberta-base-v2	roberta	124.6M	768
294	flax-sentence-embeddings/all_datasets_v3_mpnet-base	mpnet	109.5M (Est)	768
295	ipipan/silver-retriever-base-v1	bert	124.4M	768
296	woong0322/ko-legal-sbert-finetuned	bert	110.6M	768
297	jinaai/jina-code-embeddings-1.5b	qwen2	1.54B	1536
298	sentence-transformers/xlm-r-100langs-bert-base-nli-stsb-mean-tokens	xlm-roberta	278.0M	768
299	nickprock/sentence-bert-base-italian-uncased	bert	109.9M	768
300	sergeyzh/rubert-tiny-turbo	bert	29.2M	312
301	dragonkue/multilingual-e5-small-ko-v2	bert	117.7M	384
302	ValentinaKim/bge-base-automobile-matryoshka	bert	109.5M	768
303	sdadas/st-polish-paraphrase-from-distilroberta	roberta	124.4M	768
304	sdadas/mmlw-retrieval-roberta-base	roberta	124.4M	768
305	annakotarba/sentence-similarity	bert	117.7M	384
306	lang-uk/ukr-paraphrase-multilingual-mpnet-base	xlm-roberta	278.1M (Est)	768
307	sentence-transformers/all-MiniLM-L6-v1	bert	22.7M	384
308	jegormeister/bert-base-dutch-cased-snli	bert	109.2M (Est)	768
309	OrcaDB/cde-small-v1	Unknown	281.1M	768
310	OrcaDB/bge-base	bert	109.5M	768
311	sentence-transformers/bert-large-nli-stsb-mean-tokens	bert	335.1M	1024
312	PORTULAN/serafim-100m-portuguese-pt-sentence-encoder	bert	108.9M	768
313	sentence-transformers/all-MiniLM-L12-v1	bert	33.4M	384
314	avsolatorio/GIST-large-Embedding-v0	bert	335.1M	1024
315	IoannisKat1/all-MiniLM-L6-v2-legal-matryoshka	bert	22.7M	384
316	kamalkraj/BioSimCSE-BioLinkBERT-BASE	bert	108.2M (Est)	768
317	NeuML/pubmedbert-base-embeddings-matryoshka	bert	109.5M	768
318	uer/sbert-base-chinese-nli	bert	102.3M (Est)	768
319	amixh/sentence-embedding-model-InLegalBERT-2	bert	111.8M	768
320	sentence-transformers/facebook-dpr-ctx_encoder-multiset-base	bert	109.5M	768
321	sayed0am/arabic-english-bge-m3	xlm-roberta	362.2M	1024
322	google/embeddinggemma-300m-qat-q8_0-unquantized	Unknown	307.6M (Est)	768
323	uclanlp/keyphrase-mpnet-v1	mpnet	109.5M (Est)	768
324	flax-sentence-embeddings/reddit_single-context_mpnet-base	mpnet	109.5M (Est)	768
325	ricardo-filho/bert-base-portuguese-cased-nli-assin-2	bert	108.9M (Est)	768
326	Lajavaness/sentence-camembert-base	camembert	110.6M	768
327	tstadel/answer-classification-setfit-v2-binary	bert	109.5M	768
328	raul-delarosa99/bge-small-en-v1.5-RIRAG_ObliQA	bert	33.4M	384
329	mukaj/fin-mpnet-base	mpnet	109.5M	768
330	onyx-dot-app/information-content-model	mpnet	109.5M	768
331	sentence-transformers/quora-distilbert-multilingual	distilbert	134.7M	768
332	firqaaa/indo-sentence-bert-base	bert	124.5M (Est)	768
333	sentence-transformers-testing/stsb-bert-tiny-onnx	bert	4.4M	128
334	jhgan/ko-sroberta-nli	roberta	110.6M (Est)	768
335	PORTULAN/serafim-900m-portuguese-pt-sentence-encoder-ir	deberta-v2	884.6M	1536
336	sentence-transformers/msmarco-distilbert-base-dot-prod-v3	distilbert	66.4M	768
337	it-just-works/stella_en_1.5B_v5_bf16	qwen2	1.54B	1024
338	TaylorAI/bge-micro	bert	17.4M	384
339	flax-sentence-embeddings/all_datasets_v4_MiniLM-L6	bert	22.7M (Est)	384
340	PM-AI/bi-encoder_msmarco_bert-base_german	bert	109.9M	768
341	whaleloops/phrase-bert	bert	109.5M (Est)	768
342	zeroentropy/zerank-1	qwen3	4.02B	2560
343	swardiantara/MultiSource-full-crk1-m0.5-e5-b128-L6	bert	22.7M	384
344	thenlper/gte-base-zh	bert	102.3M	768
345	sentence-transformers/msmarco-MiniLM-L12-v3	bert	33.4M	384
346	harsh-wk/long-seq-paraphrase-MiniLM-L6-v2	bert	22.7M	384
347	tgsc/sentence-transformer-ult5-pt-small	t5	51.0M	512

#	Encoder Name	Encoder Type	# Params	Dimensionality
348	iampanda/zpoint_large_embedding_zh	bert	325.5M	1792
349	Fjoralb1/multilingual-e5-small-nli-matryoshka-128	bert	117.7M	128
350	math-similarity/Bert-MLM_arXiv-MP-class_zbMath	bert	109.5M (Est)	768
351	swardiantara/MultiSource-full-crk0-m0.5-e5-b128-L6	bert	22.7M	384
352	sentence-transformers/gtr-t5-xl	t5	1.24B	768
353	Jarbas/ovos-model2vec-intents-distiluse-base-multilingual-cased-v2	Unknown	30.6M	256
354	TurkuNLP/sbert-cased-finnish-paraphrase	bert	124.5M	768
355	sentence-transformers/nli-roberta-base-v2	roberta	124.6M	768
356	SamilPwC-AXNode-GenAI/PwC-Embedding_expr	xlm-roberta	559.9M	1024
357	swardiantara/MultiSource-full-cdk10-m0.5-e5-b128-L6	bert	22.7M	384
358	jinaai/jina-embedding-s-en-v1	t5	35.3M (Est)	512
359	swardiantara/MultiSource-full-crk10-m0.5-e5-b128-L6	bert	22.7M	384
360	swardiantara/MultiSource-full-crk3-m0.5-e5-b128-L6	bert	22.7M	384
361	swardiantara/MultiSource-full-crk5-m0.5-e5-b128-L6	bert	22.7M	384
362	swardiantara/MultiSource-full-cdk5-m0.5-e5-b128-L6	bert	22.7M	384
363	swardiantara/MultiSource-full-cdk3-m0.5-e5-b128-L6	bert	22.7M	384
364	swardiantara/MultiSource-full-cdk1-m0.5-e5-b128-L6	bert	22.7M	384
365	nategro/contradiction-psb	mpnet	109.5M (Est)	768
366	shibing624/text2vec-bge-large-chinese	bert	325.5M	1024
367	SergeyKarpenko1/multilingual-e5-small-legal-matryoshka_384	bert	117.7M	384
368	ozziek/all-MiniLM-L6-v2-lasttoken-false	bert	22.7M	384
369	moka-ai/m3e-small	bert	24.0M (Est)	512
370	pritamdeka/PubMedBERT-mnli-snli-scinli-scitail-mednli-stsb	bert	109.5M (Est)	768
371	PORTULAN/serafim-900m-portuguese-pt-sentence-encoder	deberta-v2	884.6M	1536
372	OrcaDB/e5-large	xlm-roberta	559.9M	1024
373	StyleDistance/styledistance	roberta	124.6M	768
374	sentence-transformers/sentence-t5-xxl	t5	4.86B	768
375	bhavyagiri/InLegal-Sbert	bert	110.1M (Est)	768
376	aari1995/German_Semantic_V3b	bert	335.2M	1024
377	omarelshehy/Arabic-Retrieval-v1.0	bert	135.2M	768
378	djovak/embedic-base	xlm-roberta	278.0M	768
379	AAUBS/PatentSBERTa_V2	mpnet	109.5M (Est)	768
380	corto-ai/nomic-embed-text-v1	nomic_bert	136.7M	768
381	sbhargav/baseline-distilbert-tot24	distilbert	66.4M	768
382	sentence-transformers/msmarco-roberta-base-v3	roberta	124.6M	768
383	ohsuz/k-finance-sentence-transformer	roberta	110.6M	768
384	TurkuNLP/sbert-uncased-finnish-paraphrase	bert	124.5M	768
385	sentence-transformers/msmarco-distilroberta-base-v2	roberta	82.1M	768
386	trmteb/berturk-base_fine_tuned	bert	110.6M	768
387	djovak/embedic-large	xlm-roberta	559.9M	1024
388	Metric-AI/armenian-text-embeddings-1	xlm-roberta	278.0M	768
389	khoa-klaytn/bge-base-en-v1.5-angle	bert	109.5M	768
390	safora/persian-e5-large-scientific-retrieval	xlm-roberta	559.9M	1024
391	BlackKakapo/stsb-xlm-r-multilingual-ro	xlm-roberta	278.0M	768
392	HIT-TMG/KaLM-embedding-multilingual-mini-instruct-v1.5	qwen2	494.0M	896
393	sentence-transformers/stsb-distilbert-base	distilbert	66.4M	768
394	embaas/sentence-transformers-multilingual-e5-base	xlm-roberta	278.1M (Est)	768
395	marroyo777/bge-99GPT-v1	bert	33.4M	384
396	Lajavaness/bilingual-embedding-base	bilingual	278.0M	768
397	KarBik/legal-french-matroshka	xlm-roberta	278.0M	768

#	Encoder Name	Encoder Type	# Params	Dimensionality
398	ricardoz/BERTugues-base-portuguese-cased	bert	110.1M	768
399	PartAI/Tooka-SBERT-V2-Large	bert	353.0M	1024
400	OrcaDB/mxbai-large	bert	335.1M	1024
401	rufimelo/Legal-BERTimbau-sts-large-ma-v3	bert	334.4M	1024
402	OrcaDB/gist-large	bert	335.1M	1024
403	sentence-transformers/msmarco-distilbert-multilingual-en-de-v2-tmp-lng-aligned	distilbert	134.7M	768
404	austinpatrickm/finetuned_bge_embeddings_v5_small_v1.5	bert	33.4M	384
405	embaas/sentence-transformers-gte-small	bert	16.7M (Est)	384
406	shibing624/text2vec-base-chinese-sentence	ernie	117.9M	768
407	KBLab/emotional-classification	bert	124.7M (Est)	768
408	l3cube-pune/indic-sentence-bert-nli	bert	237.6M (Est)	768
409	sentence-transformers/msmarco-roberta-base-ance-firstp	roberta	124.6M	768
410	clips/e5-small-trm-nl	bert	40.8M	384
411	NLBSE/nlbse26_java	bert	22.7M	384
412	minishlab/M2V_multilingual_output	model2vec	128.3M	256
413	rufimelo/bert-large-portuguese-cased-sts	bert	334.4M	1024
414	mihirsingh141/retriever_module	mpnet	109.5M	768
415	jmbrito/ptbr-similarity-e5-small	bert	117.7M (Est)	384
416	Trendyol/TY-ecomm-embed-multilingual-base-v1.2.0	new	305.4M	768
417	xmanii/maux-gte-persian	new	305.4M	768
418	sergeyzh/LaBSE-ru-turbo	bert	128.3M	768
419	Gameselo/STS-multilingual-mpnet-base-v2	xlm-roberta	278.0M	768
420	mrp/simcse-model-m-bert-thai-cased	bert	177.9M (Est)	768
421	Jaume/gemma-2b-embeddings	gemma	2.51B	2048
422	AhmedZaky1/DIMI-embedding-v4	new	305.4M	768
423	sentence-transformers/stsb-distilroberta-base-v2	roberta	82.1M	768
424	mchochlov/codebert-base-cd-ft	roberta	124.7M (Est)	768
425	richinfoai/ritrieve_zh_v1	bert	325.5M	1792
426	TomatenMarc/WRAPresentations	roberta	134.9M (Est)	768
427	mixedbread-ai/mxbai-embed-2d-large-v1	bert	335.1M	1024
428	sentence-transformers-testing/stsb-bert-tiny-openvino-quantized-only	bert	4.4M	128
429	KennethTM/MiniLM-L6-danish-encoder	bert	22.7M	384
430	sentence-transformers/roberta-base-nli-mean-tokens	roberta	124.6M	768
431	ls-da3m0ns/bge_large_medical	bert	335.1M	1024
432	johnpaulbin/jina-embeddings-v3-128	model2vec	32.0M	128
433	antoinelouis/french-me5-base	xlm-roberta	114.6M	768
434	mohamed2811/Muffakir_Embedding_V2	xlm-roberta	362.2M	1024
435	serbog/multilingual-e5-large-skill-job-matcher	xlm-roberta	559.9M (Est)	1024
436	TechWolf/ConTeXT-Skill-Extraction-base	mpnet	109.5M	768
437	infly/inf-retriever-v1	qwen2	7.07B	3584
438	sentence-transformers/bert-large-nli-mean-tokens	bert	335.1M	1024
439	NghiemAbe/Vi-Legal-Bi-Encoder-v2	roberta	135.0M	768
440	nuvocare/WikiMedical_sent_biobert	bert	108.3M	768
441	potsu-potsu/bge-base-biomedical-matryoshka-v3	bert	109.5M	768
442	intfloat/e5-base-unsupervised	bert	109.5M	768
443	PORTULAN/serafim-335m-portuguese-pt-sentence-encoder	bert	334.4M	1024
444	sentence-transformers/facebook-dpr-question_encoder-multiset-base	bert	109.5M	768
445	jinaai/jina-embedding-t-en-v1	bert	14.4M (Est)	312
446	BorisTM/bge-m3_en_ru	xlm-roberta	359.0M	1024
447	efederici/sentence-bert-base	bert	109.9M (Est)	768
448	ssmits/Qwen2.5-7B-embed-base	qwen2	7.07B	3584
449	DejanX13/ISO_embedding_1000	xlm-roberta	559.9M	1024
450	maastrichtlawtech/dpr-legal-french	camembert	110.6M	768
451	Mozilla/smart-tab-embedding	bert	22.7M	384
452	benja-d/paraphrase-spanish-distilroberta-finetuned-chatbot	roberta	124.6M	768
453	sentence-transformers/stsb-bert-base	bert	109.5M	768
454	DMetaSoul/Dmeta-embedding-zh	bert	102.7M (Est)	768
455	newmindai/TurkEmbed4Retrieval	new	305.4M	768
456	aspire/acge_text_embedding	bert	326.0M	1792
457	nickprock/multi-sentence-BERTino	distilbert	67.6M	768

#	Encoder Name	Encoder Type	# Params	Dimensionality
458	myrkur/sentence-transformer-parsbert-fa-2.0	bert	162.8M	768
459	infgrad/stella-base-zh-v2	bert	51.3M (Est)	768
460	nezahatkorkmaz/turkce-embedding-bge-m3	xlm-roberta	567.8M	1024
461	sdadas/st-polish-paraphrase-from-mpnet	roberta	124.4M	768
462	exp-models/dragonkue-KoEn-E5-Tiny	bert	37.5M	384
463	zeroentropy/zerank-2	qwen3	4.02B	2560
464	CatSchroedinger/nomic-v1.5-financial-matryoshka	nomic_bert	136.7M	768
465	d0rj/e5-base-en-ru	xlm-roberta	132.4M	768
466	upskyy/e5-base-korean	xlm-roberta	278.0M	768
467	prdev/mini-gte	distilbert	66.4M	768
468	Krelle/e5-small-v2-imo-pairs	bert	33.4M	384
469	trmteb/turkish-embedding-model	bert	110.6M	768
470	sergeyzh/LaBSE-ru-sts	bert	129.0M	768
471	swardiantara/one-crk10-m0.5-e5-b128-L6	bert	22.7M	384
472	StyleDistance/mstyledistance	xlm-roberta	278.0M	768
473	Lihuchen/pearl_base	bert	109.5M	768
474	tollefj/norbert3-multiloss-embedder	Unknown	126.6M	768
475	xmanii/maux-gte-persian-v2	new	305.4M	768
476	l3cube-pune/hindi-sentence-similarity-sbert	bert	237.6M	768
477	sentence-transformers/facebook-dpr-question_encoder-single-nq-base	bert	109.5M	768
478	h4g3n/multilingual-MiniLM-L12-de-en-es-fr-it-nl-pl-pt	bert	103.5M	384
479	clips/e5-base-trm-nl	xlm-roberta	124.4M	768
480	sdadas/mmlw-e5-large	xlm-roberta	559.9M	1024
481	swardiantara/MultiSource-full-cdk3-m0.5-e5-b128-L12	bert	33.4M	384
482	dell-research-harvard/lt-un-data-fine-fine-en	mpnet	109.5M	768
483	efederici/sentence-BERTino-v2-mmarco-4m	distilbert	66.8M (Est)	768
484	bi-matrix/gmatrix-embedding1	deberta-v2	185.3M	768
485	sentence-transformers/nli-roberta-large	roberta	355.4M	1024
486	NLBSE/nlbse26_python	bert	22.7M	384
487	farhana1996/unsupervised-simcse-bangla-sbert	xlm-roberta	278.0M	768
488	newmindai/TurkEmbed4STS	new	305.4M	768
489	sentence-transformers-testing/all-nli-bert-tiny-dense	bert	4.4M	256
490	arinze/address-match-abp-v2	bert	22.7M (Est)	32
491	anass1209/resume-job-matcher-all-MiniLM-L6-v2	bert	22.7M	384
492	Omartificial-Intelligence-Space/Arabert-all-nli-triplet-Matryoshka	bert	135.2M	768
493	baconnier/Finance2_embedding_small_en-V1.5	bert	33.4M	384
494	denaya/indoSBERT-large	bert	335.4M (Est)	256
495	syang687/FinSentenceBERT	mpnet	109.5M	768
496	NLBSE/nlbse26_pharo	bert	22.7M	384
497	mhiveai/Qwen-Insure	qwen3	595.8M	1024
498	mrhimanshu/finetuned-bge-m3	xlm-roberta	567.8M	1024
499	l3cube-pune/tamil-sentence-similarity-sbert	bert	237.6M (Est)	768
500	Muennighoff/SGPT-1.3B-weightedmean-msmarco-specb-bitfit	gpt_neo	1.34B (Est)	2048
501	intfloat/e5-large-unsupervised	bert	335.1M	1024
502	telepix/PIXIE-Rune-Preview	xlm-roberta	567.8M	1024
503	BAAI/bge-reasoner-embed-qwen3-8b-0923	qwen3	7.57B	4096
504	NYTK/sentence-transformers-experimental-hubert-hungarian	bert	110.6M (Est)	768
505	lwolfrum2/careerbert-jg	bert	109.9M	768
506	Fremtind/norsbert4-large	Unknown	359.0M	960
507	redis/langcache-embed-v3.1	bert	22.6M	384
508	pritamdeka/S-Scibert-snli-multinli-stsb	bert	109.9M (Est)	768
509	alfaneo/bertimbau-base-portuguese-sts	bert	108.9M (Est)	768
510	sentence-transformers/nq-distilbert-base-v1	distilbert	66.4M	768
511	Stern5497/sbert-legal-xlm-roberta-base	roberta	184.4M (Est)	768
512	kinit/slovakbert-sts-stsb	roberta	124.7M (Est)	768
513	sentence-transformers/bert-base-nli-max-tokens	bert	109.5M	768
514	stjiris/bert-large-portuguese-cased-legal-mlm-sts-v1.0	bert	334.4M	1024
515	Wissam42/sentence-croissant-llm-base	llama	1.28B (Est)	2048
516	MongoDB/mdbr-leaf-ir	bert	22.6M	768
517	DeepMount00/Ita-Search	qwen3	595.8M	1024

#	Encoder Name	Encoder Type	# Params	Dimensionality
518	philschmid/bge-base-financial-matryoshka	bert	109.5M	768
519	DataikuNLP/paraphrase-multilingual-MiniLM-L12-v2	bert	117.7M (Est)	384
520	jxm/cde-small-v1	Unknown	281.1M	768
521	wilsonmarciliojr/bertimbau-embed-nli	bert	108.9M	768
522	ostoveland/SBertBaseMittanbudver3	bert	355.1M (Est)	768
523	Aremaki/sentence-camembert-bio	camembert	110.6M	768
524	google/embeddinggemma-300m-qat-q4_0-unquantized	Unknown	307.6M (Est)	768
525	ipipan/silver-retriever-base-v1.1	bert	124.4M	768
526	Yunika/sentence-transformer-nepali	bert	237.6M	768
527	pierluigic/xl-lexeme	xlm-roberta	559.9M (Est)	1024
528	aari1995/German_Semantic_V3	bert	335.2M	1024
529	PartAI/Tooka-SBERT	bert	353.0M	1024
530	mini1013/master_item_top_bt_flat	roberta	110.6M	768
531	Msobhi/Persian_Sentence_Embedding_v3	xlm-roberta	559.9M	1024
532	oguuzhansahin/bi-encoder-mnrl-dbmdz-bert-base-turkish-cased-margin_3.0-msmarco-tr-10k	bert	110.6M	768
533	prashpathak/xlscout_standigger_2_aug	bert	109.5M	768
534	thenlper/gte-small-zh	bert	30.3M	512
535	iris49/3gpp-embedding-model-v0	bert	109.5M	768
536	mghuibregtse/biolinkbert-large-simcse-rat	bert	333.5M	1024
537	alikia2x/jina-embedding-v3-m2v-1024	model2vec	256.0M	1024
538	sentence-transformers/nli-bert-base-cls-pooling	bert	109.5M	768
539	Kyleiwaniec/COS_TAPT_n_RoBERTa-sts-e3_OnlineContrastiveLoss_2023-10-16	roberta	355.4M (Est)	1024
540	NohTow/french-bge-m3	xlm-roberta	358.4M	1024
541	silma-ai/silma-embedding-sts-v0.1	bert	135.2M	768
542	DejanX13/Poverenik_embedding_doc_2000	xlm-roberta	559.9M	1024
543	qilowoq/bge-m3-en-ru	xlm-roberta	374.9M	1024
544	silma-ai/silma-embedding-matryoshka-v0.1	bert	135.2M	768
545	danfeg/AraBERT_Finetuned-COMB-7443	bert	369.4M	1024
546	xmanii/maux-gte-persian-v3	new	305.4M	768
547	wwydmanski/all-mpnet-base-v2-legal-v0.1	mpnet	109.5M	768
548	h2oai/embeddinggemma-300m-qat-q8_0-unquantized	gemma3_text	302.9M	768
549	LazarusNLP/all-indobert-base-v2	bert	124.4M	768
550	flax-sentence-embeddings/all_datasets_v3_MiniLM-L12	bert	33.4M (Est)	384
551	agentlans/all-MiniLM-L6-v2-nli	bert	22.7M	384
552	dell-research-harvard/lt-wikidata-comp-multi	xlm-roberta	278.0M	768
553	ys7yoo/sentence-roberta-large-kor-sts	roberta	336.7M (Est)	1024
554	clips/e5-large-trm-nl	xlm-roberta	355.1M	1024
555	liddlefish/privacy_embedding_rag_10k_base_15_final	bert	109.5M	768
556	valuesimplex-ai-lab/Fin-Retriever-base	bert	112.9M (Est)	768
557	shtilev/medical_embedded_v5	xlm-roberta	278.0M	768
558	PaDaS-Lab/xlm-roberta-base-msmarco	xlm-roberta	278.0M	768
559	guyhadad01/EncodeRec	bert	22.7M	384
560	hamtaai/bge-m3-hadith	xlm-roberta	567.8M	1024
561	hanhainebula/reason-embed-qwen3-8b-0928	qwen3	7.57B	4096
562	ToolBench/ToolBench_IR_bert_based_uncased	bert	109.5M (Est)	768
563	Nashhz/SBERT_KFOLD_Job_Descriptions_to_Skills	bert	22.7M	384
564	l3cube-pune/telugu-sentence-similarity-sbert	bert	237.6M (Est)	768
565	hamtaai/e5-large-hadith-v2	xlm-roberta	559.9M	1024
566	nomic-ai/nomic-embed-text-v1-unsupervised	nomic_bert	136.7M (Est)	768
567	sentence-transformers/distilbert-base-nli-stsb-quora-ranking	distilbert	66.4M	768
568	mrm8488/multilingual-e5-large-ft-sts-spanish-matryoshka-768-16-5e	xlm-roberta	559.9M	1024
569	Tarka-AIR/Tarka-Embedding-350M-V1	lfm2_bidirec	354.5M	1024
570	Gameselo/french-multilingual-e5-large-instruct	xlm-roberta	342.0M	1024
571	pritamdeka/SapBERT-mnli-snli-scinli-scitail-mednli-stsb	bert	109.5M (Est)	768
572	intfloat/e5-small-unsupervised	bert	33.4M	384
573	nasa-impact/nasa-smd-ibm-st-v2	roberta	62.3M (Est)	768
574	antoinelouis/french-bge-m3	xlm-roberta	349.8M	1024
575	facebook/drama-1b	llama	1.24B	2048
576	d0rj/e5-large-en-ru	xlm-roberta	365.6M	1024

#	Encoder Name	Encoder Type	# Params	Dimensionality
577	sergioburdisso/dialog2flow-joint-bert-base	bert	109.5M	768
578	rbhatia46/financial-rag-matryoshka	new	434.1M	1024
579	sentence-transformers/stsb-bert-large	bert	335.1M	1024
580	imvladikon/sentence_transformers_alephbertgimmel_small	bert	78.7M (Est)	512
581	Muennighoff/SBERT-base-nli-v2	bert	109.5M (Est)	768
582	Manal0809/medical-term-similarity	bert	22.7M	384
583	sentence-transformers/msmarco-distilbert-base-v2	distilbert	66.4M	768
584	sdadas/mmlw-e5-small	bert	117.7M	384
585	super-cinnamon/fewshot-followup-multi-e5	bert	117.7M	384
586	minhquan6203/paraphrase-vietnamese-law	xlm-roberta	278.0M	768
587	pkshatech/RoSEtta-base-ja	retrieva-bert	190.4M	768
588	facebook/drama-base	llama	211.8M	768
589	djovak/embedic-small	bert	117.7M	384
590	sentence-transformers/roberta-large-nli-mean-tokens	roberta	355.4M	1024
591	faodl/20250909_model_g20_multilabel_MiniLM-L12-all-labels-artificial-governance-multi-output	bert	117.7M	384
592	krlvi/sentence-t5-base-nlpl-code_search_net	t5	110.2M (Est)	768
593	sentence-transformers/paraphrase-albert-base-v2	albert	11.7M	768
594	LamaDiab/V7MiniLM-Synonyms-SemanticEngine	bert	22.7M	384
595	friedrichor/Unite-Base-Qwen2-VL-2B	qwen2_vl	2.21B	1536
596	rufimelo/Legal-BERTimbau-sts-base	bert	108.9M	768
597	LazarusNLP/all-indobert-base-v4	bert	124.4M	768
598	Sampath1987/EnergyEmbed-v1	new	305.4M	768
599	HIT-TMG/KaLM-embedding-multilingual-mini-instruct-v1	qwen2	494.0M	896
600	redis/langcache-embed-experimental	bert	22.7M	384
601	sentence-transformers/bert-base-nli-cls-token	bert	109.5M	768
602	deepset/all-mpnet-base-v2-table	mpnet	109.5M	768
603	Lihuchen/pearl_small	bert	33.4M	384
604	mpi-inno-comp/pat_specter	bert	109.9M (Est)	768
605	l3cube-pune/bengali-sentence-similarity-sbert	bert	237.6M (Est)	768
606	minishlab/potion-science-32M	model2vec	31.9M	256
607	jinaai/jina-embedding-l-en-v1	t5	335.0M (Est)	1024
608	smart-tribune/sentence-transformers-multilingual-e5-large	xlm-roberta	559.9M (Est)	1024
609	taxstreem/numens-finbert	bert	109.5M	768
610	bcwarner/PubMedBERT-base-uncased-sts-combined	bert	109.5M	768
611	PartAI/Tooka-SBERT-V2-Small	bert	122.9M	768
612	pritamdeka/S-PubMedBert-MS-MARCO-SCIFACT	bert	109.5M (Est)	768
613	asmud/nomic-embed-indonesian	nomic_bert	136.7M	768
614	dguzh/geo-all-distilroberta-v1	roberta	82.1M	768
615	Tarka-AIR/Tarka-Embedding-150M-V1	gemma3_text	151.0M	768
616	cyberagent/xlm-roberta-large-jnli-jsick	xlm-roberta	559.9M (Est)	1024
617	tss-deposium/gemma-deposium-768d	model2vec	196.4M	768
618	sdadas/mmlw-retrieval-e5-large	xlm-roberta	559.9M	1024
619	XenArcAI/SparkEmbedding-300m	gemma3_text	302.9M	768
620	dmlls/all-mpnet-base-v2-negation	mpnet	109.5M	768
621	antoinelouis/biencoder-camembert-base-mmarcoFR	camembert	110.6M	768
622	KennethEnevoldsen/dfm-sentence-encoder-large-exp2-no-lang-align	bert	355.1M (Est)	1024
623	kakao1513/KURE-legal-ft-v1	xlm-roberta	567.8M	1024
624	Muennighoff/SBERT-large-nli-v2	bert	335.2M (Est)	1024
625	RIOLITE/products_matching_aumet	bert	22.7M (Est)	384
626	textgain/allnli-GroNLP-bert-base-dutch-cased	bert	109.2M (Est)	768
627	uaritm/multilingual_en_uk_pl_ru	xlm-roberta	278.0M	768
628	OpenVoiceOS/ovos-model2vec-intents-potion-32M	Unknown	32.3M	512
629	shawhin/distilroberta-ai-job-embeddings	roberta	82.1M	768
630	mini1013/master_item_top_el_flat	roberta	110.6M	768
631	wjunwei/ecommerce_text_embedding	bert	109.5M	768
632	ng3owb/sentiment-embedding-model	xlm-roberta	559.9M	1024
633	Huffon/sentence-klue-roberta-base	roberta	110.6M (Est)	768

#	Encoder Name	Encoder Type	# Params	Dimensionality
634	mrm8488/distiluse-base-multilingual-cased-v2-finetuned-stsb_multi_mt-es	distilbert	134.7M (Est)	768
635	h4c5/sts-camembert-base	camembert	110.6M	768
636	symanto/sn-xlm-roberta-base-snli-mnli-anli-xnli	xlm-roberta	278.0M	768
637	rfahlevih/sentence-transformer-all-mpnetv2-resume-span-classifier	mpnet	109.5M	768
638	mik3ml/multilingual-e5-large-ita	xlm-roberta	559.9M	1024
639	kasraarabi/finetuned-caption-embedding	mpnet	109.5M	768
640	Muennighoff/SGPT-1.3B-weightedmean-nli-bitfit	gpt_neo	1.34B (Est)	2048
641	altaidevorg/bge-m3-distill-8l	xlm-roberta	366.2M	1024
642	lighteternal/stsb-xlm-r-greek-transfer	xlm-roberta	278.1M (Est)	768
643	antoinelouis/biencoder-distilcamembert-mmarcoFR	camembert	68.1M	768
644	DiTy/bi-encoder-russian-msmarco	bert	177.9M	768
645	sentence-transformers/gtr-t5-xxl	t5	4.86B	768
646	Shitao/bge-m3	xlm-roberta	567.8M	1024
647	manu/sentence_croissant_alpha_v0.4	llama	1.28B	2048
648	kevinkrahn/shlm-grc-en	hlm	94.8M	768
649	kornwtp/ConGen-WangchanBERT-Small	bert	28.8M (Est)	512
650	JoshELambert/illegal	mpnet	109.5M (Est)	768
651	danieleff/hubert-base-cc-sentence-transformer	bert	110.6M (Est)	768
652	tomaarsen/setfit-absa-bge-small-en-v1.5-restaurants-aspect	bert	33.4M (Est)	384
653	odunola/sentence-transformers-bible-reference-final	mpnet	109.5M (Est)	768
654	Dqdung205/medical_vietnamese_embedding	new	305.4M	768
655	sartifyllc/African-Cross-Lingua-Embeddings-Model	bert	470.9M	768
656	JJTsao/fine-tuned_movie_retriever-bge-base-en-v1.5	bert	109.5M	768
657	deepmodal/embeddinggemma-300m-ko	gemma3_text	302.9M	768
658	dimitriz/st-greek-media-bert-base-uncased	bert	112.9M	768
659	maastrichtlawtech/monobert-legal-french	camembert	110.6M	768
660	amin/medical_embedding_1	bert	33.4M	384
661	emillykkejensen/EmbeddingGemma-Scandi-300m	gemma3_text	302.9M	768
662	efederici/sentence-BERTino	distilbert	67.6M (Est)	768
663	ciCic/paraphrase-multilingual-MiniLM-L12-v2-sts-2d-matryoshka	bert	117.7M	384
664	anhtld/VN-Law-Embedding	xlm-roberta	278.0M	768
665	Madnesss/fine-tune-all-MiniLM-L6-v2	bert	22.7M (Est)	384
666	Saegus/sentence-camembert-large-mean-pooling	camembert	336.7M	1024
667	IoannisKat1/legal-bert-base-uncased-new	bert	109.5M	768
668	sentence-transformers/nli-roberta-base	roberta	124.6M	768
669	ferrisS/german-english-bge-m3	xlm-roberta	360.6M	1024
670	xlreator/snomed-biobert	bert	108.3M	768
671	projecte-aina/ST-NLI-ca_paraphrase-multilingual-mpnet-base	xlm-roberta	278.1M (Est)	768
672	am-azadi/bilingual-embedding-small_Fine_Tuned	bilingual	117.7M	384
673	flax-sentence-embeddings/multi-qa_v1-MiniLM-L6-cls_dot	bert	22.7M (Est)	384
674	duongttr/job-candidiate-matching-sentbert	bert	108.6M (Est)	384
675	blemond/RAG_press_multilingual_e5_large	xlm-roberta	559.9M	1024
676	sentence-transformers/nli-bert-large	bert	335.1M	1024
677	DMetaSoul/sbert-chinese-general-v2-distill	bert	45.6M (Est)	768
678	alvperez/skill-sim-model	mpnet	109.5M	768
679	sentence-transformers/bert-large-nli-cls-token	bert	335.1M	1024
680	rainjay/sbert_nlp_corom_sentence-embedding_chinese-base-ecom	bert	102.3M (Est)	768
681	sdadas/mmlw-retrieval-e5-base	xlm-roberta	278.0M	768
682	gmunkhtur/paraphrase-mongolian-minilm-mn_v2	bert	117.7M	384
683	ssmits/Qwen2.5-7B-Instruct-embed-base	qwen2	7.07B	3584
684	l3cube-pune/malayalam-sentence-bert-nli	bert	237.6M (Est)	768
685	tomaarsen/setfit-absa-bge-small-en-v1.5-restaurants-polarity	bert	33.4M (Est)	384
686	quantisan/setfit-minilm-l3-v2-cw-ndc-subsectors-v1	bert	17.4M	384
687	isolation-forest/setfit-absa-polarity	bert	29.2M	312
688	isolation-forest/setfit-absa-aspect	bert	29.2M	312

#	Encoder Name	Encoder Type	# Params	Dimensionality
689	kimseongsan/ko-sbert-384-reduced	bert	110.6M	384
690	stjiris/bert-large-portuguese-cased-legal-mlm-nli-sts-v1	bert	334.4M	1024
691	karakastarik/bert-base-turkish-128k-uncased-spelling-correction	bert	184.4M (Est)	16
692	antonypamo/RRFSAVANTMADE	bert	22.7M	384
693	upskyy/kf-deberta-multitask	deberta-v2	185.3M	768
694	ferrisS/german-english-multilingual-e5-small	bert	40.0M	384
695	michaelfeil/Qwen3-Embedding-0.6B-auto	qwen3	595.8M	1024
696	meedan/paraphrase-filipino-mpnet-base-v2	xlm-roberta	278.1M (Est)	768
697	ronanki/ml_use_512_MNR_10-2022-07-17_14-22-50	distilbert	135.1M (Est)	512
698	jjprietotorres/labse-persuasion-detection-agnostic	bert	470.9M	768
699	KatjaK/gnd_retriever_full	xlm-roberta	567.8M	1024
700	flax-sentence-embeddings/all_datasets_v3_roberta-large	roberta	355.4M (Est)	1024
701	Den4ikAI/sbert_large_mt_ru_retriever	bert	426.9M (Est)	1024
702	nickprock/setfit-italian-hate-speech	bert	109.9M	768
703	pavanmantha/bge-base-en-bioembed	bert	109.5M	768
704	adlumal/auslaw-embed-v1.0	bert	33.4M (Est)	384
705	hetbhagatji09/Job-resume-match-model	bert	22.7M	384
706	cath1616/similar_word_corse_fine_tunig_model	bert	116.8M	768
707	pritamdeka/S-Bluebert-snli-multinli-stsb	bert	109.5M (Est)	768
708	antonkirk/retrieval-mpnet-dot-finetuned-gpt-4o-mini-synthetic-dataset	mpnet	109.5M	768
709	jonny9f/food_embeddings	mpnet	109.5M	768
710	spartan8806/atles-champion-embedding	mpnet	109.5M	768
711	sentence-transformers/xlm-r-large-en-ko-nli-ststb	xlm-roberta	559.9M	1024
712	universalml/Nepali_Embedding_Model	xlm-roberta	559.9M	1024
713	Samizie/avia-MiniLM-L12-v2	bert	33.4M	384
714	jhgan/ko-sroberta-sts	roberta	110.6M (Est)	768
715	sentence-transformers/distilroberta-base-msmarco-v1	roberta	82.1M	768
716	Kwaipilot/OASIS-code-embedding-1.5B	qwen2	1.54B	1536
717	intfloat/mmE5-mllama-11b-instruct	mllama	10.64B	4096
718	bi-matrix/G-MATRIX-Embedding-v1	new	305.4M	768
719	MPA/sambert	bert	184.3M	768
720	omarelsayeed/QA_Search	bert	11.5M	256
721	mini1013/master_item_top_fd_flat	xlm-roberta	278.0M	768
722	cnmoro/portuguese-bge-m3	xlm-roberta	349.3M	1024
723	Roflmax/merged_full-bge_03-07	xlm-roberta	567.8M	1024
724	ddobokki/klue-roberta-small-nli-sts	roberta	68.1M	768
725	haophancs/bge-m3-financial-matryoshka	xlm-roberta	567.8M	1024
726	weizhou03/nomic-embed-text-v1.5	nomic_bert	136.7M	768
727	Muennighoff/SGPT-125M-weightedmean-nli-bitfit	gpt_neo	137.8M (Est)	768
728	TUKE-DeutscheTelekom/slovakbert-skquad-mnlr	roberta	124.6M	768
729	nickprock/mmarco-bert-base-italian-uncased	bert	109.9M	768
730	mjaliz/xml-base-gis-basalam-1MQ	xlm-roberta	559.9M	1024
731	RagN/ABSA-aspect	mpnet	109.5M	768
732	RagN/ABSA-polarity	mpnet	109.5M	768
733	omarelshehy/arabic-english-sts-matryoshka-v2.0	xlm-roberta	559.9M	1024
734	ggrn/e5-small-v2	bert	33.4M (Est)	384
735	AkshitaS/bhasha-embed-v0	bert	237.6M	768
736	Voicelab/sbert-large-cased-pl	bert	355.1M (Est)	1024
737	patent/sbert-all-MiniLM-L6-v2	bert	22.7M (Est)	384
738	CC-AI-Labs/sharks-triplet-hsm-bert-base-uncased-2025-04	bert	109.5M	768
739	rawsh/multi-qa-MiniLM-BERT-Tiny-distill-L-2_H-128_A-cos-v1	bert	4.4M (Est)	128
740	johnnas12/e5-galaxy-finetuned	bert	109.5M	768
741	Lajavaness/bilingual-document-embedding	bilingual	567.8M	1024
742	SteveTran/ob_semantic_model	new	305.4M	768
743	Yoonyoul/fine-tuned-e5-small-drugproduct	bert	117.7M	384
744	nthakur/contriever-base-msmarco	bert	109.5M (Est)	768
745	cnmoro/snowflake-arctic-embed-m-v2.0-cpu	gte	305.4M	768
746	SeppeV/roberta_TSDAE	roberta	355.4M	1024
747	truro7/vn-law-embedding	xlm-roberta	278.0M	768
748	nonola/portuguese-bge-m3	xlm-roberta	349.3M	1024

#	Encoder Name	Encoder Type	# Params	Dimensionality
749	LamaDiab/MiniLM-256BATCH-V6Data-SemanticEngine	bert	22.7M	384
750	Bylaw/convention-collective-sentence-transformer	camembert	336.7M	1024
751	upskyy/gte-base-korean	new	305.4M	768
752	UMCU/SapBERT-from-PubMedBERT-fulltext_bf16	bert	109.5M	768
753	unsloth/embeddinggemma-300m-qat-q8_0-unquantized	gemma3_text	302.9M	768
754	Muennighoff/SGPT-125M-weightedmean-msmarco-specb-bitfit	gpt_neo	137.8M (Est)	768
755	Mihaiii/Ivysaur	bert	22.7M	384
756	gbyuvd/ChemEmbed-v01	bert	22.7M	768
757	LamaDiab/MiniLM-V10Data-128BATCH-SemanticEngine	bert	22.7M	384
758	spartan8806/atles	mpnet	109.5M	768
759	isy-thl/multilingual-e5-base-course-skill-tuned	xlm-roberta	278.0M	768
760	mini1013/master_domain	roberta	110.6M	768
761	marianodo/ContrastiveLoss	mpnet	109.5M (Est)	768
762	stjiris/bert-large-portuguese-cased-legal-tsdae-gpl-nli-sts-MetaKD-v1	bert	334.4M	1024
763	d0rj/e5-small-en-ru	xlm-roberta	44.8M	384
764	smartcat/SRBedding-base-v1	xlm-roberta	278.0M	768
765	ddobokki/klue-roberta-base-nli-sts	roberta	110.6M	768
766	michaelfeil/Qwen3-Embedding-4B-auto	qwen3	4.02B	2560
767	jonny9f/food_embeddings2	mpnet	109.5M	768
768	nthakur/dragon-plus-context-encoder	bert	109.5M (Est)	768
769	desarrolloasesoreslocales/bert-leg-al-setfit	roberta	184.3M	768
770	nthakur/dragon-plus-query-encoder	bert	109.5M (Est)	768
771	dunzhang/stella-large-zh-v3-1792d	bert	326.4M (Est)	1792
772	LazarusNLP/all-nusabert-large-v4	bert	336.7M	1024
773	vaios-stergio/all-mpnet-base-v2-dblp-aminer-180k-pairs	mpnet	109.5M	768
774	danjohnvelasco/filipino-sentence-roberta-v1	roberta	109.1M (Est)	768
775	copenlu/spiced	mpnet	109.5M (Est)	768
776	BlueAvenir/sti_cyber_security_model_updated	xlm-roberta	278.1M (Est)	768
777	dell-research-harvard/lt-wikidata-comp-zh	bert	102.3M	768
778	aminhaeri/risk-embed	bert	108.9M	768
779	kathaem/bert-base-chinese-sentence-transformer-xnli-zh	bert	102.3M (Est)	768
780	hothanhtienqb/mind_map_blog_model	bert	117.7M	384
781	JINSUP/bge-m3-ko-axriv-agent-part-2025	xlm-roberta	567.8M	1024
782	Cloyne/vietnamese-sbert-v3	roberta	135.0M	768
783	redis/langcache-embed-v3-mini-experimental	bert	22.6M	384
784	sobamchan/bert-base-uncased-no-mrl	bert	109.5M	768
785	mboth/distil-eng-quora-sentence	distilbert	66.4M (Est)	768
786	alfaneo/jurisbert-base-portuguese-sts	bert	108.9M	768
787	kornwtp/simcse-model-phayathaibert	camembert	277.5M (Est)	768
788	fgaim/tielectra-bi-encoder	electra	13.5M	256
789	mrm8488/distilroberta-base-ft-allnli-matryoshka-768-64-1e-256bs	roberta	82.1M	768
790	trmteb/turkish-embedding-model-fine-tuned	bert	110.6M	768
791	Day1Kim/Qwen3-Embedding-0.6B-Korean	qwen3	595.8M	1024
792	sentence-transformers/quora-distilbert-base	distilbert	66.4M	768
793	inkoziev/sbert_synonymy	bert	29.2M (Est)	312
794	Hulyyy/req-quality-setfit-hintaware-dict	bert	22.7M	384
795	sobamchan/bert-base-uncased-mrl-768-512-256-128-64	bert	109.5M	768
796	mini1013/master_item_top_ps_flat	xlm-roberta	278.0M	768
797	vrnP66/finetuned-embedding-model	xlm-roberta	597.0M	1024
798	jordyvl/scibert_scivocab_uncased_sentence_transformer	bert	109.9M	768
799	fgaim/tiroberta-bi-encoder	roberta	124.6M	768
800	Maluong/my-retriever-model	roberta	135.0M	768
801	arinze/address-match-abp-v5	bert	22.8M (Est)	64
802	LamaDiab/MiniLM-V6Data-SemanticEngine	bert	22.7M	384
803	Omartificial-Intelligence-Space/Arabic-MiniLM-L12-v2-all-nli-triplet	bert	117.7M	384
804	msbayindir/turkish-legal-bert-base-uncased-stsb-v1-sts	bert	110.6M	768

#	Encoder Name	Encoder Type	# Params	Dimensionality
805	TimKond/S-PubMedBert-MedQuAD	bert	109.5M	768
806	sdadas/mmlw-e5-base	xlm-roberta	278.0M	768
807	vaios-stergio/all-mpnet-base-v2-dblp-aminer-50k-triplets-64	mpnet	109.5M	768
808	snowdere/trainer_topic	bert	22.7M	384
809	Sinaof1381/Persian_Sentence_Embedding	xlm-roberta	559.9M	1024
810	pablosi/bge-m3-trained	xlm-roberta	567.8M	1024
811	fajayi/nomi-name-encoder	bert	117.7M	384
812	l3cube-pune/tamil-sentence-bert-nli	bert	237.6M (Est)	768
813	mrm8488/multilingual-e5-large-ft-sts-spanish-matryoshka-768-64-5e	xlm-roberta	559.9M	1024
814	Chernofface/fs-setfit-multilable-model	bert	117.7M	384
815	michaelfeil/embeddinggemma-300m	gemma3_text	302.9M	768
816	hetbhagatji09/Job-resume-ner-match-model	bert	22.7M	384
817	bpHigh/Setfit-FakeNews-en	bert	22.7M (Est)	384
818	thuan9889/llama_embedding_model_v1	bert	22.7M	384
819	dbourget/philai-embeddings-v1.1	bert	335.1M	1024
820	LazarusNLP/all-nusabert-base-v4	bert	110.6M	768
821	wilfredomartel/embeddinggemma-300m-legal-spanish-100k	gemma3_text	302.9M	768
822	newmindai/TurkEmbed4STS-Static	model2vec	64.0M	256
823	somosnlp-hackathon-2022/bertin-roberta-base-finetuning-esnli	roberta	124.7M (Est)	768
824	sukantan/all-MiniLM-L6-v2-ftlegal-v1	bert	22.7M (Est)	384
825	vahoaka/sentence-transformers-model-vahoaka-v1	bert	117.7M	384
826	rahmanfadhil/indobert-finetuned-indonli	bert	124.5M (Est)	768
827	jangedoo/all-MiniLM-L6-v2-nepali	bert	22.7M	384
828	C10X/Qwen3-Embedding-TurboX.v2	model2vec	155.3M	1024
829	InstalilyAI/synonym-transformer-3phase	bert	22.7M	384
830	LAMDEC/gte-pgm-pairs	new	305.4M	768
831	ramdane/jurimodel	xlm-roberta	278.0M	768
832	Santp98/SBERT-pairs-bert-base-spanish-wwm-cased	bert	109.9M	768
833	narraticlabs/MiniLM-L6-european-union	bert	107.0M	384
834	antoinelouis/french-me5-large	xlm-roberta	342.0M	1024
835	gromoboy/qwen3_06b_items_matcher	qwen3	595.8M	1024
836	alunadiderot/setfit-e5-base-category-classifier_v2	xlm-roberta	278.0M	768
837	qhoxie/embeddinggemma-model2vec-256d	model2vec	65.5M	256
838	along26/DistillKLDivLoss_manglish-iban-sentence-transformer	roberta	82.1M	768
839	bigscience/sgpt-bloom-7b1-msmarco	bloom	7.07B (Est)	4096
840	nomic-ai/nomic-embed-text-v1-ablated	nomic_bert	136.7M (Est)	768
841	tomerRest/line_item_embeddings	bert	22.7M	384
842	Saidakmal/uz_embeddinggemma-300m	gemma3_text	302.9M	768
843	DejanX13/Javne_Nabavke_embedding_1000	xlm-roberta	559.9M	1024
844	EmmanuelEA/eea-embedding-gemma	gemma3_text	302.9M	768
845	easonanalytica/cnm-multilingual-small-v2	bert	117.7M	384
846	HJUNN/bge-m3b-Art-Therapy-embedding-fine-tuning	xlm-roberta	567.8M	1024
847	LamaDiab/MiniLM-V7-128BATCH-V6Data-SemanticEngine	bert	22.7M	384
848	sentence-transformers/nli-bert-base	bert	109.5M	768
849	Mihaiii/gte-micro-v4	bert	19.2M	384
850	leeloolee/intention	new	306.0M	768
851	tcepi/sts_bertimbau	bert	108.9M	768
852	Leo1212/longformer-base-4096-sentence-transformers-all-nli-stsb-quora-nq	longformer	148.7M	768
853	flax-sentence-embeddings/all_datasets_v4_mpnet-base	mpnet	109.5M (Est)	768
854	menadsa/S-PubMedBERT	bert	109.5M (Est)	768
855	antoinelouis/biencoder-mMiniLMv2-L6-mmarcoFR	xlm-roberta	107.0M	384
856	Byunghwee/roberta_belief_finetuned	roberta	124.6M	768
857	tomaarsen/mpnet-base-nli	mpnet	109.5M	768
858	denniscraandijk/dutch-bge-m3	xlm-roberta	351.1M	1024
859	FremyCompany/BioLORD-STAMB2-v1	mpnet	109.5M (Est)	768
860	upskyy/e5-small-korean	bert	117.7M	384
861	sivarohit2002/qwen06b_bi-e5-ft-weighted	bert	109.5M	768

#	Encoder Name	Encoder Type	# Params	Dimensionality
862	sentence-transformers/msmarco-bert-co-condensor	bert	109.5M	768
863	sentence-transformers/msmarco-distilbert-multilingual-en-de-v2-tmp-trained-scratch	distilbert	134.7M	768
864	avemio/German-RAG-BGE-M3-MERGED-x-SNOWFLAKE-ARCTIC-HESSIAN-AI	xlm-roberta	567.8M	1024
865	lwolfrum2/careerbert-g	bert	109.9M	768
866	sentence-transformers/nli-distilbert-base	distilbert	66.4M	768
867	Omartificial-Intelligence-Space/Arabic-mpnet-base-all-nli-triplet	mpnet	109.5M	768
868	snunlp/KR-SBERT-Medium-extended-klueNLItriplet_PARpair_QApair-klueSTS	bert	117.1M	768
869	woodx/Qwen3-Embedding-0.6B-SGLang	qwen3	595.8M	1024
870	LamaDiab/STSBMiniLM-V9Data-256BATCH-SemanticEngine	bert	22.7M	384
871	Voicelab/sbert-base-cased-pl	bert	124.5M (Est)	768
872	FremyCompany/BioLORD-2023-S	mpnet	109.5M (Est)	768
873	KoonJamesZ/sentence-transformers-nina-thai-v3	xlm-roberta	278.0M	768
874	abragin/ruBert-style-base	bert	178.3M	768
875	joeportnoy/resume-match-ml	bert	22.7M	384
876	EMBO/negative_sampling_pmb	bert	109.5M	768
877	yosuaw/all-MiniLM-L6-v2	bert	22.7M	384
878	WhereIsAI/UAE-Code-Large-V1	bert	335.1M	1024
879	Catchy1282/GradientBoosting_model_1_samples_per_label	mpnet	109.5M	768
880	Omartificial-Intelligence-Space/AraEuroBert-210M	eurobert	211.8M	768
881	rufimelo/Legal-BERTimbau-sts-base-ma-v2	bert	108.9M	768
882	Collab-uniba/github-issues-preprocessed-mpnet-st-e10	mpnet	109.5M (Est)	768
883	latterworks/ollama-embeddings	bert	22.7M	384
884	Catchy1282/GradientBoosting_model_3_samples_per_label	mpnet	109.5M	768
885	Catchy1282/GradientBoosting_model_5_samples_per_label	mpnet	109.5M	768
886	bwang0911/jev2-legal	mpnet	109.5M	768
887	gihong99/qwen3-embedding-ko-v1	qwen3	665.4M	1024
888	Roflmax/bge-m3-legal-ru-updata	xlm-roberta	567.8M	1024
889	efederici/sentence-it5-small	t5	35.3M (Est)	512
890	LazarusNLP/simcse-indoroberta-base	roberta	124.6M	768
891	greatakela/gnlp_hw1_encoder	roberta	82.1M	768
892	truro7/hcmus_handbook	xlm-roberta	278.0M	768
893	Desalegnn/Desu-snowflake-arctic-embed-l-v2.0-finetuned-amharic-45k	xlm-roberta	567.8M	1024
894	sentence-transformers/bert-base-wikipedia-sections-mean-tokens	bert	109.5M	768
895	infgrad/stella-base-zh-v3-1792d	bert	103.5M (Est)	1792
896	manu/sentence_croissant_alpha_v0.2	llama	1.28B	2048
897	CC-AI-Labs/nord-triplet-hsm-bert-base-uncased	bert	109.5M	768
898	gerald29/setfit-bge-small-v1.5-sst2-8-shot-introduction	bert	33.4M	384
899	tomaarsen/Qwen3-Embedding-0.6B-18-layers	qwen3	438.5M	1024
900	Nextcloud-AI/multilingual-e5-large-instruct	xlm-roberta	559.9M	1024
901	potsu-potsu/medembed-base-biomedical-matryoshka	bert	109.5M	768
902	KhaledReda/all-MiniLM-L6-v17-pair_score	bert	22.7M	384
903	cnmoro/custom-model2vec-tokenlearn-small	model2vec	2.6M	256
904	clips/mfaq	xlm-roberta	278.1M (Est)	768
905	lewtun/my-awesome-setfit-model	mpnet	109.5M (Est)	768
906	uaritm/multilingual_en_ru_uk	xlm-roberta	278.1M (Est)	768
907	h2oai/embeddinggemma-300m	gemma3_text	302.9M	768
908	infgrad/Jasper-Token-Compression-600M	qwen3	607.3M	2048
909	JohanHeinsen/Old_News_Segmentation_SBERT_V0.1	bert	109.5M	768
910	KFST/XLMRoberta-en-da-sv-nb	xlm-roberta	278.0M	768
911	AHDMK/Sentence-BioBert-snli	bert	108.3M	768
912	shtilev/medical_embedded_v3	xlm-roberta	278.0M	768
913	yassine123Z/EmissionFactor-mapper2-v2	bert	33.4M	384
914	LAMDEC/gemma-pgm-pairs	gemma3_text	302.9M	768
915	sentence-transformers/distilroberta-base-msmarco-v2	roberta	82.1M	768
916	rufimelo/Legal-BERTimbau-sts-large	bert	334.4M	1024
917	LazarusNLP/simcse-indobert-base	bert	124.4M	768

#	Encoder Name	Encoder Type	# Params	Dimensionality
918	Xenova/jina-embeddings-v2-base-de	bert	80.4M (Est)	768
919	Collab-uniba/fprime-binary-setfit	mpnet	109.5M	768
920	Cloyne/sup-SimCSE-VietNamese-phobert-base	roberta	135.0M	768
921	seregadgl/sts_v11	bert	128.3M	768
922	jegormeister/robbert-v2-dutch-base-mqa-finetuned	roberta	116.8M (Est)	768
923	avsolatorio/all-MiniLM-L6-v2-MEDI-MTEB-triplet-final	bert	22.7M	384
924	Xavarary/mpnet-base-all-medium-triplet	roberta	82.1M	768
925	LamaDiab/MiniLM-V9Data-256BATCH-SemanticEngine	bert	22.7M	384
926	LamaDiab/MiniLM-V10Data-256BATCH-SemanticEngine	bert	22.7M	384
927	eduardofv/stsb-m-mt-es-distiluse-base-multilingual-cased-v1	distilbert	134.7M (Est)	768
928	silverjam/jina-embeddings-v2-base-zh	bert	160.8M	768
929	philipp-zettl/MTGEmb-small	clip	151.3M	512
930	ulysses-camara/legal-bert-pt-br	Unknown	108.9M (Est)	768
931	stjiris/bert-large-portuguese-cased-legal-mlm-mkd-nli-sts-v1	bert	334.4M	1024
932	Omartificial-Intelligence-Space/Arabic-labse-Matryoshka	bert	470.9M	768
933	ontology/EnergyBert	bert	109.8M	768
934	axondendriteplus/Legal-Embed-bge-base-en-v1.5	bert	109.5M	768
935	Albertdebeauvais/all-MiniLM-L6-v2_cotes	bert	22.7M	384
936	Sbhatti33/sbert_model	bert	22.7M	384
937	friedrichor/Unite-Base-Qwen2-VL-7B	qwen2_vl	8.29B	3584
938	Blablablab/multilingual-style-representation	xlm-roberta	559.9M	1024
939	zafonair/e5-turkish-base	xlm-roberta	559.9M	1024
940	Muennighoff/SGPT-125M-weightedmean-msmarco-specb	gpt_neo	137.8M (Est)	768
941	eduardofv/stsb-m-mt-es-distilbert-base-uncased	Unknown	66.4M (Est)	768
942	praisethefool/human_tech-fields-multilabelclassifier	mpnet	109.5M	768
943	tudorizer/distilroberta-ai-job-embeddings	roberta	82.1M	768
944	yosriku/congen-indobert-lite-base	albert	11.7M	768
945	flax-sentence-embeddings/stackoverflow_mpnet-base	mpnet	109.5M (Est)	768
946	recobo/agri-sentence-transformer	bert	220.1M (Est)	768
947	neeva/query2query	bert	22.7M (Est)	384
948	jeonseonjin/embedding_BAAI-bge-m3	xlm-roberta	567.8M	1024
949	bijaygurung/stella_en_400M_v5	new	435.2M	1024
950	Ganaraj/rgveda-embedding-gemma	gemma3_text	302.9M	768
951	faodl/model_cca_multilabel_mpnet-65max-data-augmented-v03	xlm-roberta	278.0M	768
952	IoannisKat1/multilingual-e5-large-ft-new	xlm-roberta	559.9M	1024
953	PrabalAryal/Sentence_Transformer_v0.0.4	bert	117.7M	384
954	andersborges/model2vecdk-stem	Unknown	48.6M	256
955	Ahmedhisham/queen_of_embedded_egy_20k	bert	117.7M	384
956	LazarusNLP/all-indo-e5-small-v3	bert	117.7M	384
957	melino2000/product-torob-matching	distilbert	134.7M	768
958	michaelfeil/Qwen3-Embedding-8B-auto	qwen3	7.57B	4096
959	mancer146/embeddinggemma-300m-haystack-contrastive-thin-fixed	gemma3_text	302.9M	768
960	Enno-Ai/bge-m3	xlm-roberta	567.8M	1024
961	manu/sentence_croissant_alpha_v0.3	llama	1.28B	2048
962	clips/e5-small-v2-t2t-nl	bert	33.2M	384
963	tavakolih/all-MiniLM-L6-v2-pubmed-full	bert	22.7M (Est)	384
964	nixiesearch/nixie-suggest-small-v1	bert	33.4M (Est)	384
965	seroe/bge-m3-turkish-triplet-matryoshka-v2	xlm-roberta	567.8M	1024
966	Tushar0505/fine-tuned-legal-bert	bert	109.5M	768
967	imageomics/trait2vec	mpnet	109.5M	256
968	andersborges/model2vecdk	Unknown	48.0M	256
969	jegormeister/setfit-model	bert	109.2M (Est)	768
970	bongsoo/kpf-sbert-v1.1	bert	114.0M (Est)	768
971	tum-nlp/NegMPNet	mpnet	109.5M (Est)	768
972	dpanea/skill-assignment-transformer	new	434.1M	1024
973	Roflmax/bge-m3-russian-legal	xlm-roberta	567.8M	1024
974	sentence-transformers/xlm-r-base-en-ko-nli-ststb	xlm-roberta	278.0M	768

#	Encoder Name	Encoder Type	# Params	Dimensionality
975	LLukas22/all-MiniLM-L12-v2-embedding-all	bert	33.4M (Est)	384
976	myrkur/sentence-transformer-parsbert-fa	bert	162.8M	768
977	seongil-dn/bge-m3-further-large-v2-150	xlm-roberta	567.8M	1024
978	hilabs/termmap_semantic_model	bert	22.7M	384
979	julian-schelb/SPhilBerta-emb-lat-intertext-v1	roberta	135.2M	768
980	tnguy564/qwen-geospatial-embedder	bert	117.7M	384
981	LazarusNLP/all-indobert-base	bert	124.4M	768
982	pawan2411/address-emnet	mpnet	109.5M	768
983	GreenNode/GreenNode-Embedding-Large-VN-Mixed-V1	xlm-roberta	567.8M	1024
984	seroe/Qwen3-Embedding-0.6B-turkish-triplet-matryoshka-v2	qwen3	595.8M	1024
985	Fremtind/norsbert4-base	Unknown	148.9M	640
986	KhaledReda/all-MiniLM-L6-v18-pair_score	bert	22.7M	384
987	brjezierski/sentence-embeddings-similarity-distill-consistency	bert	33.4M (Est)	384
988	SeanLee97/mxbai-embed-large-v1-nli-matryoshka	bert	335.1M	3072
989	JohanHeinsen/PE_efterlyst_gender	bert	109.5M	768
990	framp/Qwen3-Embedding-TurboX-ie-hf	model2vec	38.8M	256
991	marma/sentence-bert-swedish-cased	bert	124.7M	768
992	LAMDEC/qwen-pgm-pairs	qwen3	595.8M	1024
993	SetFit/test-setfit-sst2	bert	22.7M (Est)	384
994	AhmedBadawy11/bge-base-ar-v1.5-finetuned	bert	109.5M	768
995	winderfeld/cc-uffs-ppc-ft-test-multiqa	mpnet	109.5M	768
996	jebish7/MedEmbed-small-v0.1_MNR_1	bert	33.4M	384
997	nickprock/csr-multi-sentence-BERTino-cv	distilbert	67.6M	768
998	xmanii/maux-gte-persian-v3-fp16	new	305.4M	768
999	Hulyyy/req-quality-setfit-2	mpnet	109.5M	768
1000	l3cube-pune/marathi-sentence-bert-nli	bert	237.6M (Est)	768
1001	antoinelouis/dpr-xm	xmod	852.5M	768
1002	adeshkin/labse-kjh-ru	bert	470.9M	768
1003	RepresentLM/RepresentLM-v1	bert	109.5M	768
1004	seongil-dn/bge-m3-3800_steps_v2_234	xlm-roberta	567.8M	1024
1005	Albertdebeauvais/all-MiniLM-L6-v2_bibliographie	bert	22.7M	384
1006	sentence-transformers/xlm-r-100langs-bert-base-nli-mean-tokens	xlm-roberta	278.0M	768
1007	sentence-transformers/xlm-r-bert-base-nli-mean-tokens	xlm-roberta	278.0M	768
1008	SmartComponents/bge-micro-v2	bert	8.7M (Est)	384
1009	zeta-alpha-ai/Zeta-Alpha-E5-Mistral	mistral	7.11B	4096
1010	alexshah/armembed	llama	1.24B	2048
1011	tikanosa/fine-tuned-sbert-prodi	bert	117.7M	384
1012	mudasir13cs/Field-adaptive-bi-encoder	bert	22.7M	384
1013	dimitriz/st-greek-media-longformer-4096	longformer	150.0M	768
1014	antoinelouis/french-me5-small	bert	35.9M	384
1015	LamaDiab/MiniLM-SemanticEngine	bert	22.7M	384
1016	ceggian/sbert_standard_reddit_softmax	bert	109.5M (Est)	768
1017	ceggian/sbert_pt_reddit_softmax_512	bert	109.5M (Est)	768
1018	textgain/tags-allnli-GroNLP-bert-base-dutch-cased	bert	109.2M (Est)	768
1019	amazon/sm-hackathon-actionability-9-multi-outputs-setfit-all-distilroberta-model-v0.2	roberta	82.1M (Est)	768
1020	stevenluo/bge-large-zh-v1.5-ft-v4	bert	325.5M	1024
1021	SIRIS-Lab/affilgood-dense-retriever	xlm-roberta	559.9M	1024
1022	sachinn1/xl-durel	xlm-roberta	559.9M	1024
1023	Bea-Taylor/objection_fine_tuned_4	bert	22.7M	384
1024	nmixx-fin/nmixx-bge-m3	xlm-roberta	567.8M	1024
1025	GPL/scifact-msmarco-distilbert-gpl	distilbert	66.4M (Est)	768
1026	radlab/polish-sts-v2	roberta	435.0M	1024
1027	syubraj/sentence_similarity_nepali	bert	81.9M	768
1028	amazon/sm-hackathon-setfit-model	mpnet	109.5M (Est)	768
1029	amazon/sm-hackathon-actionability-9-multi-outputs-setfit-all-distilroberta-model-v0.1	roberta	82.1M (Est)	768
1030	hojzas/setfit-proj8-all-mpnet-base-v2	mpnet	109.5M	768
1031	nasa-impact/nasa-ibm-st.38m	roberta	81.6M (Est)	576
1032	webis/tiny-bert-ranker	bert	4.4M	128
1033	bwbayu/sbert_model_jobcv	distilbert	66.4M	768

#	Encoder Name	Encoder Type	# Params	Dimensionality
1034	jarredparrett/all-MiniLM-L6-v2_tuned_on_deepparse_address_mutations_comb_3	bert	22.7M	384
1035	andreaschari/bge-m3-RU_MMARCO_50_MIXED	xlm-roberta	567.8M	1024
1036	yoriis/GTE-tydi-quqa-haqa	bert	135.2M	768
1037	impresso-project/halloween_workshop_ocr_robust_preview	new	305.4M	768
1038	along26/distilbert-base-manglish-multilingual-sentence-transformer	roberta	82.1M	768
1039	yosriku/exp2_ep3_bs128_lr2e5	albert	11.7M	768
1040	Muennighoff/SGPT-5.8B-weightedmean-msmarco-specb-bitfit	gptj	5.87B (Est)	4096
1041	Omartificial-Intelligence-Space/Arabic-all-nli-triplet-Matryoshka	xlm-roberta	278.0M	768
1042	simonosgoode/nomic_embed_fine_tune_law_1.5	nomic_bert	136.7M	768
1043	megavn/bge-m3-embeddings	xlm-roberta	567.8M (Est)	1024
1044	andyP/ro-sentence-transformers-v2	bert	115.1M	768
1045	Ponimash/gpt_text_embd	gpt2	129.1M	1024
1046	BlackBeenie/mdeberta-v3-base-sbert	deberta-v2	278.2M	768
1047	Omartificial-Intelligence-Space/Semantic-Ar-Qwen-Embed-0.6B	qwen3	595.8M	1024
1048	sergeyzh/rubert-tiny-lite	bert	23.0M	256
1049	JALLAJ/5epo	bert	33.4M	384
1050	tntwj/trained_retriever	xlm-roberta	567.8M	1024
1051	faodl/model_cca_multilabel_MiniLM-L12-v01	bert	117.7M	384
1052	CATIE-AQ/camembert-base-embedding	camembert	110.6M	768
1053	Adarsh921/multi_qa_mpnet	mpnet	109.5M	768
1054	LamaDiab/V2MiniLM-SemanticEngine	bert	22.7M	384
1055	amazon/sm-hackathon-actionability-9-multi-outputs-setfit-all-roberta-large-model-v0.1	roberta	355.4M (Est)	1024
1056	nasa-impact/nasa-smd-ibm-st	roberta	62.3M (Est)	768
1057	BlackKakapo/cupidon-base-ro	xlm-roberta	278.0M	768
1058	impresso-project/halloween_workshop_ocr_robust_with_lux_qa	preview	305.4M	768
1059	andreinsardi/SciBERT-SolarPhysics-Search	bert	109.9M	768
1060	lewtun/setfit-ethos-multilabel-example	mpnet	109.5M (Est)	768
1061	amrothemich/sapbert-sentence-transformers	bert	109.5M (Est)	768
1062	amazon/sm-hackathon-actionability-9-multi-outputs-setfit-model-v0.1	mpnet	109.5M (Est)	768
1063	Mihaiii/Bulbasaur	bert	17.4M	384
1064	denniscraandijk/dutch-gte-multilingual-base	new	142.8M	768
1065	dqubit/FRIDA-F16	t5	823.4M	1536
1066	faodl/model_cca_multilabel_MiniLM-L12-75max-data-augmented-labels-desc-v03	bert	117.7M	384
1067	Renan1997/sentence-transformer	xlm-roberta	278.0M	768
1068	JohanHeinsen/ENO_Runaway_Advertisement_classifier_2.0	bert	109.5M	768
1069	Sahajtomar/German-semantic	bert	335.8M (Est)	1024
1070	gabrielloiseau/LUAR-CRUD-sentence-transformers	roberta	82.1M	512
1071	lengocquangLAB/fine-tuned-jobtitle-embed	bert	117.7M	384
1072	yasserrmd/geo-gemma-300m-emb	gemma3_text	302.9M	768
1073	skygudanr/klue-roberta-base-klue-sts	roberta	110.6M	768
1074	IoannisKat1/all-MiniLM-L6-v2-ft-new	bert	22.7M	384
1075	IoannisKat1/bge-m3-ft-new	xlm-roberta	567.8M	1024
1076	vasyz/Giga-Embeddings-instruct	gigarembed	3.45B	2048
1077	stjiris/bert-large-portuguese-cased-legal-tsdae-gpl-nli-sts-MetaKD-v0	bert	334.4M	1024
1078	firqaaa/indo-sentence-bert-large	bert	335.1M	2048
1079	prithivida/miniDense_arabic_v1	bert	117.7M (Est)	384
1080	Saegus/sentence-camembert-base	camembert	110.6M	768
1081	optimum-internal-testing/sentence-transformers-stsb-bert-tiny	bert	4.4M	128
1082	UMCU/SapBERT-UMLS-2020AB-all-lang-from-XLMR-ST	xlm-roberta	278.0M	768
1083	faodl/model_cca_multilabel_MiniLM-L12-70prop-data-augmented-v02	bert	117.7M	384
1084	yosriku/embeddinggemma-300m	gemma3_text	302.9M	768
1085	raph145/paraphrase-multilingual-MiniLM-L12-v2	bert	117.7M	384
1086	CATIE-AQ/distilcamembert-base-embedding	camembert	68.1M	768
1087	along26/distilbert-base-manglish-sentence-transformer	bert	22.7M	384

#	Encoder Name	Encoder Type	# Params	Dimensionality
1088	JoaoVitorr/ifood-classification-model-v5	bert	117.7M	384
1089	sentence-transformers/nli-bert-large-cls-pooling	bert	335.1M	1024
1090	Omartificial-Intelligence-Space/Marbert-all-nli-triplet-Matryoshka	bert	162.8M	768
1091	KennethTM/MiniLM-L6-danish-encoder-v2	bert	22.7M	384
1092	EmanuelOrler/setfit-spanish-event-perspective	bert	117.7M	384
1093	C10X/Qwen3-Embedding-TurboX	model2vec	38.8M	256
1094	tomerRest/line_item_embeddings_	bert	22.7M	384
1095	luiggy2620/jina-v2-es-legal-embeddings	bert	138.8M	768
1096	Roflmax/bge-m3-cocktail-updata-04-06	xlm-roberta	567.8M	1024
1097	MossaabDev/Quran_embed_V2.2	bert	117.7M	384
1098	krlng/sts-GBERT-bi-encoder	bert	335.7M	1024
1099	sentence-transformers/distilbert-base-nli-max-tokens	distilbert	66.4M	768
1100	uaritm/lik_neuro_202	bert	117.7M (Est)	384
1101	Netizine/icis_e5_mistral_embeddings_instruct	mistral	7.11B	4096

Table 8: List of 1101 encoders with encoder type, parameter size and dimensionality.