Model Documentation Form

This Form includes all the information to be documented as part of Measure 1.1 of the Transparency Chapter of the Code of Practice. Crosses on the right indicate whether the information documented is intended for the AI Office (AIO), national competent authorities (NCAs) or downstream providers (DPs), namely providers of AI systems who intend to integrate the general-purpose AI model into their AI systems. Whilst information intended for DPs should be made available to them proactively, information intended for the AIO or NCAs is only to be made available following a request from the AIO, either ex officio or based on a request to the AIO from NCAs. Such requests will state the legal basis and purpose of the request and will concern only items from the Form strictly necessary for the AIO to fulfil its tasks under the AI Act at the time of the request, or for NCAs to exercise their supervisory tasks under the AI Act at the time of the request, in particular to assess compliance of high-risk AI systems built on general-purpose AI models where the provider of the system is different from the provider of the model.

Any elements of information from the Model Documentation Form shared with the AIO and NCAs shall be treated in accordance with the confidentiality obligations and trade secret protections set out in Article 78 AI Act.

Date this document was last updated: 10/04/2026 **Document version number: v1**

General information		AIO	NCAs	DPs
Legal name for the model provider:	Nasjonalbiblioteket	☒	☒	☒
Model name:	nb-sbert-v2-base	☒	☒	☒
Model authenticity:	SHA256: 5cc4842e3a7d0a2c0bff737ce78bb5abb7cbfea3fdbf97dd8955ed080627bb1c	☒	☒	☐
Release date:	13/04/2026 Date when the model was first released through any distribution channel.	☒	☒	☒
Union market release date:	13/04/2026 Date when the model was placed on the Union market.	☒	☒	☒
Model dependencies:	nb-bert-base	☒	☒	☒

Model properties						AIO	NCAs	DPs
Model architecture:	Encoder BERT-based architecture, fine-tuned as a sentence transformer with mean pooling.					☒	☒	☒
Design specifications of the model:	This model is for creating sentence/passage embeddings to be used for retrieval tasks for example.					☒	☒	☐
Input modalities:	☒Text	☐Images	☐Audio	☐Video	If any other please specify:	☒	☒	☒
For each selected modality please include maximum input size or write 'N/A' if not defined	Maximum size: 512	Maximum size: N/A	Maximum size: N/A	Maximum size: N/A	Maximum size: N/A	☒	☐	☒
Output modalities:	☐Text	☐Images	☐Audio	☐Video	A vector of 768 dimensions	☒	☒	☒
For each selected modality please include maximum output size or write 'N/A' if not defined	Maximum size: N/A	Maximum size: N/A	Maximum size: N/A	Maximum size: N/A	Maximum size: 1 vector	☐	☐	☒
Total model size:	$1.8x10^8$					☒	☐	☐
The range within which the total number of parameters falls.	☒1—500M	☐500M—5B	☐ 5B—15B	☐15B—50B		☐	☒	☒
	☐50B—100B	☐100B—500B	☐500B—1T	☐>1T				

Methods of distribution and licenses		AIO	NCAs	DPs
Distribution channels:	The model is available on HuggingFace at https://huggingface.co/NbAiLab/nb-sbert-v2-base	☒	☒	☐
	The model is available on HuggingFace at https://huggingface.co/NbAiLab/nb-sbert-v2-base	☐	☐	☒
License:	Apache 2.0 license (found at https://huggingface.co/datasets/choosealicense/licenses/blob/main/markdown/apache-2.0.md)	☒	☒	☐
	Apache 2.0 (Open-source and free to use and redistribute license)	☐	☐	☒
	All used assets were already available	☒	☐	☒

Use	AIO	NCAs	DPs

Acceptable Use Policy:	none	☒	☒	☒
Intended uses:	Sentence embeddings and retrieval	☒	☒	☒
Type and nature of AI systems in which the general-purpose AI model can be integrated:	Any system that would need a vector representation of text.	☒	☒	☒
Technical means for model integration:	The model is available on HuggingFace. See the model card for example usage with SentenceTransformers and transformer libraries.	☐	☐	☒
Required hardware:	Any GPU with at least 4GB of VRAM or any CPU with at least 4GB of RAM.	☐	☐	☒
Required software:	Transformers v4 and up	☐	☐	☒

Training process		AIO	NCAs	DPs
Design specifications of the training process:	The training process employs a contrastive learning methodology to fine-tune NbAiLab/nb-bert-base, optimizing it to map semantically related text into a shared vector space. By utilizing MultipleNegativesRankingLoss (InfoNCE), the model learns to minimize the distance between anchor-positive pairs while maximizing the distance to contradiction-based negatives. A key design choice is the use of in-batch negatives, which treats all other sentences within a training batch as additional negative examples to enhance the efficiency of the embedding signal. To ensure training stability and performance, the configuration incorporates warmup steps to prevent aggressive early weight updates and uses gradient accumulation to simulate larger batch sizes for a stronger contrastive signal. Additionally, the BatchSamplers.NO_DUPLICATES parameter is critical for preventing training noise by ensuring that no identical sentences appear within the same batch.	☒	☒	☐
Decision rationale:	We used best practice for training sentence embedding models.	☒	☒	☐

Information on the data used for training, testing, and validation					AIO	NCAs	DPs
Data type/modality: *Select all that apply.*	☒Text	☐Images	☐Audio	☐Video	If any other please specify:	☒	☒	☒
Data provenance: *Select all that apply*	☐Web crawling	☐Private non-publicly available datasets obtained from third parties	☐User data			☒	☒	☒
For definitions of each listed category, see the Template for the Public Summary of the Training Content of General-Purpose AI models provided by the AI Office	☒Publicly available datasets		☐Data collected through other means					
	☐Synthetic data that is not publicly accessible (when created directly by or on behalf of the provider)		If any other please specify:					
How data was obtained and selected:	The NbAiLab/mnli-norwegian dataset was created by machine-translating the original English Multi-Genre Natural Language Inference (MultiNLI) corpus into Norwegian Bokmål using Google Translate. We transform the MNLI dataset into a triplet format suitable for contrastive learning. We generate training samples by selecting an anchor sentence and pairing it with a randomly sampled positive (entailment) and negative (contradiction) to create the final triplet structure.				☒	☒	☐
Number of data points:	Our subset consists of 527,098 training samples.				☐	☒	☐
	Our subset consists of 527,098 training samples.				☒	☐	☐
Scope and main characteristics:	The original MNLI dataset covers 10 domains. The thematic and geographic context of our subset reflect this original source, but are translated to Norwegian to provide better Norwegian performance.				☒	☒	☐
Data curation methodologies:	The NbAiLab/mnli-norwegian dataset provides information about translation methods. Further processing includes generating training samples by selecting an anchor sentence and pairing it with a randomly sampled positive (entailment) and negative (contradiction) to create the final triplet structure.				☒	☒	☒
Measures to detect unsuitability of data sources:	None				☒	☒	☐
Measures to detect identifiable biases:	None				☒	☒	☐

Computational resources (during training)		AIO	NCAs	DPs
Training time:	Less than 1 month	☐	☒	☐
	Wall time: 3.1×10^{-2} days, GPU-time: 3.1×10^{-2} GB10 days	☒	☐	☐
Amount of computation used for training:	10^{15} FLOPS	☐	☒	☐
	5.0×10^{15} FLOPS	☒	☐	☐
Measurement methodology:	We estimated the number of FLOPS per token using the OpenAI scaling law paper. We then multiply that number by the number of tokens trained on.	☒	☒	☐

Energy consumption (during training and inference)		AIO	NCAs	DPs
Amount of energy used for training:	3.5×10^{-5} MWh	☒	☒	☐
Measurement methodology:	We train our model with Weights & Biases as logging system, this logs the power usage during training. We take the average power usage and then multiply by the compute time.	☒	☒	☐
Benchmarked amount of computation used for inference[1]:	Benchmarked amount of computation used for inference, reported in floating point operations, recorded with at least two significant figures (e.g. 5.1×10^{17} floating point operations).	☒	☒	☐
Measurement methodology:	In the absence of a delegated act adopted in accordance with Article 53(5) AI Act to detail measurement and calculation methodologies, provide a description of a computational task (e.g. generating 100000 tokens) and the hardware (e.g. 64 Nvidia A100s) used to measure or estimate the amount of computation used for inference.	☒	☒	☐

[1] This item relates to energy consumption during inference, which makes up the "energy consumption of the model" (Annex XI, 2(e), AI Act) together with energy consumption during training. Since energy consumption during inference depends on more than just the model itself, the information required for this item is limited to relevant information depending only on the model, namely computational resources used for inference.