

1. The importance of openness in the era of generative AI
 - LLM-as-a-service
 - Open software and open LLMs
 - Aspects of openness
 - LTG and the open LLM ecosystem
2. Web-derived LLM training data for Norwegian
3. Training data for Norwegian LLM research
- 4. Developing NorMistral-11B**
5. NorEval: Native benchmarking for Norwegian LLMs
6. The language quality of the Norwegian output in language models
7. Outlook

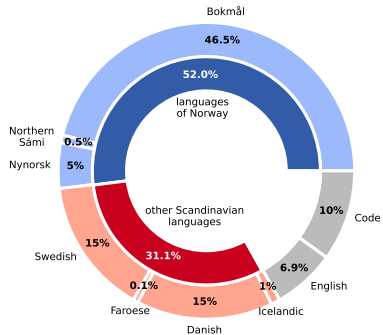
- Learning from training 7B Norwegian models, we moved to training a better and larger model
- We started from 12B Mistral model and **continually trained** on additional 250B tokens
- The training was done on LUMI with 256 GPUs ran for about a week



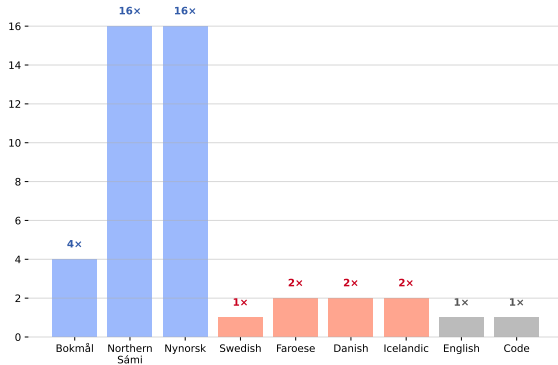
Base NorMistral 11B – training corpus



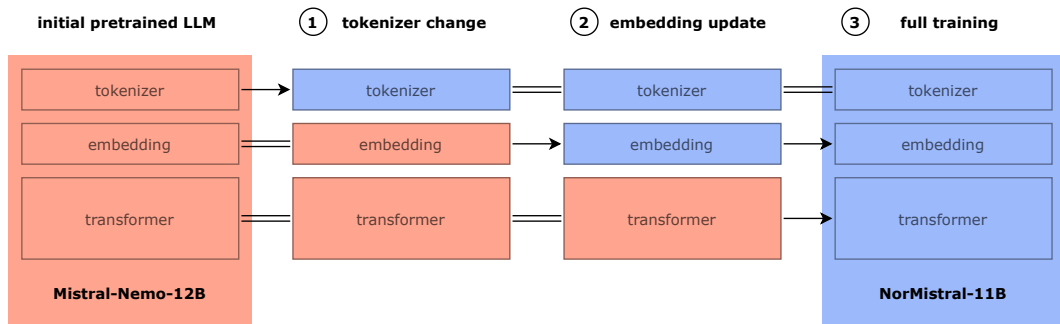
Corpus mixture



Language upsampling factors



Base NorMistral 11B – continual training



Base NorMistral 11B – openly available on HuggingFace



- License: **Apache 2.0**
- NorMistral 11B: <https://hf.co/norallm/normistral-11b-warm>



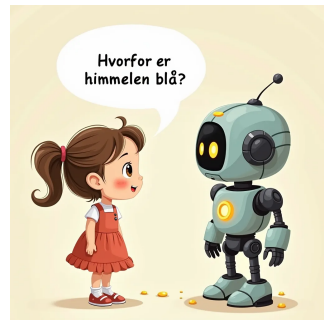
Length-extended NorMistral 11B – newly available on HuggingFace



- Base NorMistral extended to 32 768 context length
- License: **Apache 2.0**
- NorMistral 11B long: <https://hf.co/norallm/normistral-11b-long>



- Base language models, such as NorMistral, are useful for researchers, but not for general public
- We need to turn base models into **general conversational assistants**
- But this is hard to do **without any Norwegian training datasets**



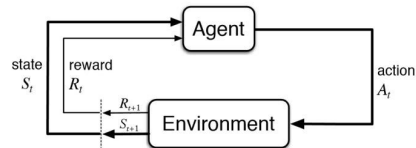
Norwegian post-training – the problem with fluency



- Previously, we solved the data issue with **machine translation** of English datasets
- This works, but it has a noticeable impact on the **fluency** of the resulting models
- The problem is that the translation models introduce *translationese* – unnatural artifacts from translating too literally
- LLMs are very good in learning language patterns – so the fluent base model is easily steered away into generating disfluent texts



- The trick to preserve fluency is obvious: never train on any translated data
- **Hypothesis 1:** This can be achieved by moving from off-policy training to **on-policy reinforcement learning**
 - Instead of telling the model what to generate, we let the model generate a response and give it a feedback
- However, we need another model to provide feedback – **Hypothesis 2:** The reward model does not have to be fluent in Norwegian, it just has to understand it well enough



FLUENT ALIGNMENT WITH DISFLUENT JUDGES: POST-TRAINING FOR LOWER-RESOURCE LANGUAGES

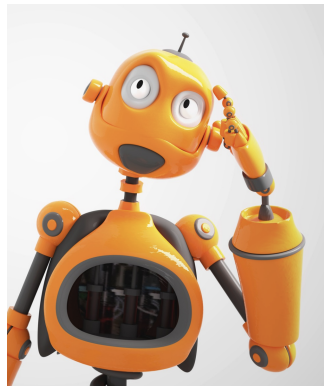
David Samuel Lilja Øvrelid Erik Velldal Andrey Kutuzov
Language Technology Group, University of Oslo
davisamu@ifi.uio.no

ABSTRACT

We propose a post-training method for lower-resource languages that preserves fluency of language models even when aligned by disfluent reward models. Preference-optimization is now a well-researched topic, but previous work has mostly addressed models for English and Chinese. Lower-resource languages lack both datasets written by native speakers and language models capable of generating fluent synthetic data. Thus, in this work, we focus on developing a fluent preference-aligned language model without any instruction-tuning data in the target language. Our approach uses an on-policy training method, which we compare with two common approaches: supervised finetuning on machine-translated data and multilingual finetuning. We conduct a case study on Norwegian Bokmål and evaluate fluency through native-speaker assessments. The results show that the on-policy aspect is crucial and outperforms the alternatives without relying on any hard-to-obtain data.



- The next step was to move out of the clean methodology of an academic paper into **the harsh reality**
- We extended the fluency-preserving post-training **on many fronts**:
 - A much larger and more diverse dataset
 - Reasoning capability
 - Handling of long multi-turn conversations with system prompts
 - Fine-grained rewards evaluating separate quality features
- In the end, a long process full of small tweaks and collapsed training runs



From <https://www.aberdeens.com>

NorMistral 11B Thinking – preliminary generative evaluation



- Yet, our post-trained model is **comparable to the state-of-the-art models of similar size**
- Preliminary evaluation on a generative version of NorEval (that is still work-in-progress)

	NoReC binary	NoReC ternary	Norldiom Bokmål	Norldiom Nynorsk	NorCSQA Bokmål	NorCSQA Nynorsk
NorMistral-11B	86.3	65.2	55.7	27.7	70.7	64.2
Llama-3.1-8B	79.8	52.9	12.7	6.7	64.0	57.9
Mistral-Nemo-12B	67.9	49.1	12.9	8.5	61.6	49.5
Qwen3-15B	83.5	69.6	22.1	13.2	83.8	71.6
Gemma3-12B	85.2	67.1	43.7	23.7	81.9	80.0
OLMo3-7B	72.0	63.3	5.0	2.2	N/A	17.9
OLMo2-13B	32.8	13.2	3.5	2.2	48.0	45.3
Apertus-8B	78.4	58.8	34.3	15.7	69.2	63.2

- The responses (and ‘thinking’ traces) of all evaluated models are also published on [HuggingFace](#) – looking at the responses is usually more informative than trusting some evaluation scores

	NorOBQA Bokmål	NorOBQA Nynorsk	NRK Bokmål	NRK Nynorsk	NorRewrite	NorSummarize
NorMistral-11B	83.0	84.4	58.8	62.3	51.9	54.3
Llama-3.1-8B	78.5	71.1	49.8	46.2	50.0	50.0
Mistral-Nemo-12B	75.3	67.8	47.3	45.0	42.5	39.2
Qwen3-15B	94.4	88.9	63.3	55.9	77.6	83.1
Gemma3-12B	91.5	88.9	59.8	58.4	86.8	77.8.
OLMo3-7B	70.5	54.4	43.3	35.9	7.8	14.2
OLMo2-13B	55.3	56.7	45.3	39.4	48.3	53.7
Apertus-8B	76.1	74.4	50.2	48.3	39.6	42.1

NorMistral 11B Thinking – available on HuggingFace



- License: **Apache 2.0**
- NorMistral 11B thinking: <https://hf.co/norallm/normistral-11b-thinking>
- NorMistral post-training data: <https://hf.co/norallm/normistral-11b-thinking-training>
- NorMistral evaluation data: <https://hf.co/norallm/normistral-11b-thinking-evaluation>



- The post-trained NorMistral is also freely available on our **public chat interface** at <https://chat.llm.sigma2.no/>
- Collaboration between LTG and **Sigma2**
- The model is run on NRIS infrastructure (using `ollama`), which also hosts the chat interface (using `open-webui`).
- The interface also features **API access** to the model

