– WNNLP 2026 –
The 8th IN5550 Teaching Workshop on Neural Natural Language Processing

University of Oslo

June 1st 2026



WNNLP 2026 at a glance

- 29 accepted papers submitted by 44 students (39 MSc + 5 PhDs)
- All-time record in the number of papers!

WNNLP 2026 at a glance

- 29 accepted papers submitted by 44 students (39 MSc + 5 PhDs)
- All-time record in the number of papers!
- 6 different thematic tracks:
  1. Machine-generated code detection (Track Chair: Vladislav Mikhailov)
  2. Steering LLMs' Responses Towards Moral Foundations (Track Chairs: Hannah Clausen and Andrey Kutuzov)
  3. English–Sámi Machine Translation (Track Chairs: David Samuel and Yves Scherrer)
  4. Predicting Emotional Valence and Arousal Over Time (Track Chair: Shenbin Qian)
  5. LLM Reasoning (Track Chair: Meihan Tong)
  6. Open-Ended Regional Question Answering with RAG (Track Chair: Jindřich Helcl)
- All papers peer-reviewed by two students + one track chair.
- Workshop proceedings: TBA

Program

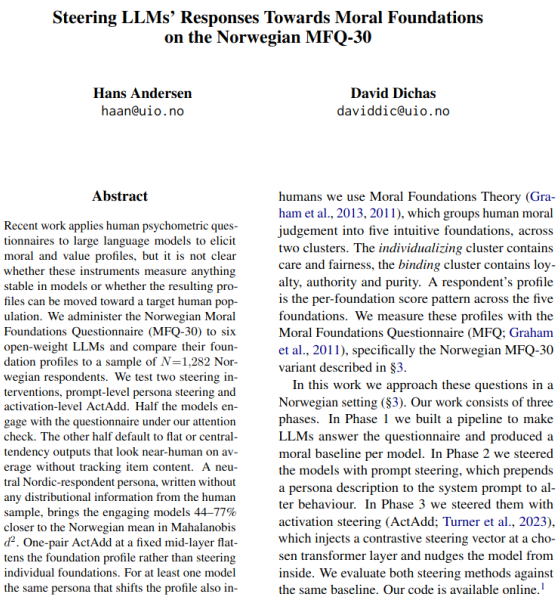
- 10:15 - 10:25 Welcome and awards
- 10:25 - 10:26 Machine-Generated Code Detection task overview
  - 10:26 - 10:38 lightning talks
- 10:38 - 10:39 Steering LLMs' Responses Towards Moral Foundations task overview
  - 10:39 - 10:54 lightning talks
- 10:54 - 10:55 English–Sámi Machine Translation task overview
  - 10:55 - 11:00 lightning talks
- 11:00 - 11:15 Break
- 11:15 - 11:16 Predicting Emotional Valence and Arousal Over Time task overview
  - 11:16 - 11:28 lightning talks
- 11:28 - 11:29 LLM reasoning task overview
  - 11:29 - 11:41 lightning talks
- 11:41 - 11:42 Open-Ended Regional Question Answering with RAG task overview
  - 11:42 - 11:56 lightning talks
- 11:56 - 12:00 Closing

Awards

- Only MSc students considered for awards, not PhD fellows.

Awards

- Only MSc students considered for awards, not PhD fellows.



Steering LLMs' Responses Towards Moral Foundations on the Norwegian MFQ-30

Hans Andersen
haan@uio.no

David Dichas
daviddic@uio.no

Abstract

Recent work applies human psychometric questionnaires to large language models to elicit moral and value profiles, but it is not clear whether these instruments measure anything stable in models or whether the resulting profiles can be moved toward a target human population. We administer the Norwegian Moral Foundations Questionnaire (MFQ-30) to six open-weight LLMs and compare their foundation profiles to a sample of $N=1{,}282$ Norwegian respondents. We test two steering interventions, prompt-level persona steering and activation-level ActAdd. Half the models engage with the questionnaire under our attention check. The other half default to flat or central-tendency outputs that look near-human on average without tracking item content. A neutral Nordic-respondent persona, written without any distributional information from the human sample, brings the engaging models 44–77% closer to the Norwegian mean in Mahalanobis d^2. One-pair ActAdd at a fixed mid-layer flattens the foundation profile rather than steering individual foundations. For at least one model the same persona that shifts the profile also inhumans we use Moral Foundations Theory (Graham et al., 2013, 2011), which groups human moral judgement into five intuitive foundations, across two clusters. The *individualizing* cluster contains care and fairness, the *binding* cluster contains loyalty, authority and purity. A respondent's profile is the per-foundation score pattern across the five foundations. We measure these profiles with the Moral Foundations Questionnaire (MFQ; Graham et al., 2011), specifically the Norwegian MFQ-30 variant described in §3.

In this work we approach these questions in a Norwegian setting (§3). Our work consists of three phases. In Phase 1 we built a pipeline to make LLMs answer the questionnaire and produced a moral baseline per model. In Phase 2 we steered the models with prompt steering, which prepends a persona description to the system prompt to alter behaviour. In Phase 3 we steered them with activation steering (ActAdd; Turner et al., 2023), which injects a contrastive steering vector at a chosen transformer layer and nudges the model from inside. We evaluate both steering methods against the same baseline. Our code is available online.[1]

Best paper

- Authors: Hans Andersen and David Dichas
- Title: Steering LLMs' Responses Towards Moral Foundations on the Norwegian MFQ-30

Awards

- Only MSc students considered for awards, not PhD fellows.



Steering LLMs' Responses Towards Moral Foundations on the Norwegian MFQ-30

Hans Andersen
haan@uio.no

David Dichas
daviddic@uio.no

Abstract

Recent work applies human psychometric questionnaires to large language models to elicit moral and value profiles, but it is not clear whether these instruments measure anything stable in models or whether the resulting profiles can be moved toward a target human population. We administer the Norwegian Moral Foundations Questionnaire (MFQ-30) to six open-weight LLMs and compare their foundation profiles to a sample of $N=1{,}282$ Norwegian respondents. We test two steering interventions, prompt-level persona steering and activation-level ActAdd. Half the models engage with the questionnaire under our attention check. The other half default to flat or central-tendency outputs that look near-human on average without tracking item content. A neutral Nordic-respondent persona, written without any distributional information from the human sample, brings the engaging models 44–77% closer to the Norwegian mean in Mahalanobis d^2. One-pair ActAdd at a fixed mid-layer flattens the foundation profile rather than steering individual foundations. For at least one model the same persona that shifts the profile also in-

humans we use Moral Foundations Theory (Graham et al., 2013, 2011), which groups human moral judgement into five intuitive foundations, across two clusters. The *individualizing* cluster contains care and fairness, the *binding* cluster contains loyalty, authority and purity. A respondent's profile is the per-foundation score pattern across the five foundations. We measure these profiles with the Moral Foundations Questionnaire (MFQ; Graham et al., 2011), specifically the Norwegian MFQ-30 variant described in §3.

In this work we approach these questions in a Norwegian setting (§3). Our work consists of three phases. In Phase 1 we built a pipeline to make LLMs answer the questionnaire and produced a moral baseline per model. In Phase 2 we steered the models with prompt steering, which prepends a persona description to the system prompt to alter behaviour. In Phase 3 we steered them with activation steering (ActAdd; Turner et al., 2023), which injects a contrastive steering vector at a chosen transformer layer and nudges the model from inside. We evaluate both steering methods against the same baseline. Our code is available online.[1]

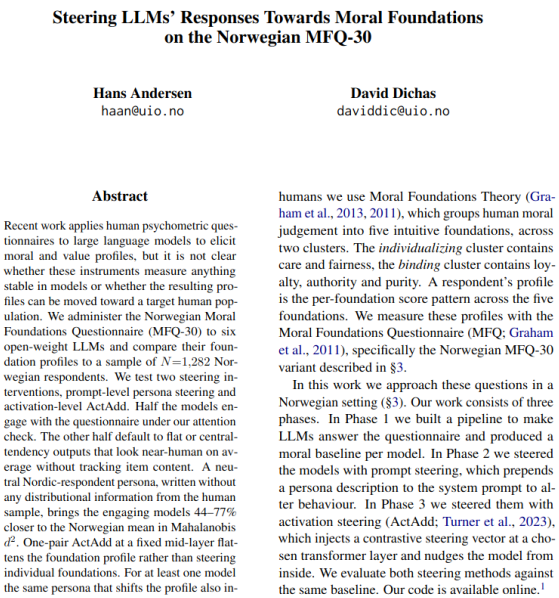
Best paper

- Authors: Hans Andersen and David Dichas
- Title: Steering LLMs' Responses Towards Moral Foundations on the Norwegian MFQ-30

Highest scores across obligatory assignments

Awards

- Only MSc students considered for awards, not PhD fellows.



Steering LLMs' Responses Towards Moral Foundations on the Norwegian MFQ-30

Hans Andersen
haan@uio.no

David Dichas
daviddic@uio.no

Abstract

Recent work applies human psychometric questionnaires to large language models to elicit moral and value profiles, but it is not clear whether these instruments measure anything stable in models or whether the resulting profiles can be moved toward a target human population. We administer the Norwegian Moral Foundations Questionnaire (MFQ-30) to six open-weight LLMs and compare their foundation profiles to a sample of $N=1{,}282$ Norwegian respondents. We test two steering interventions, prompt-level persona steering and activation-level ActAdd. Half the models engage with the questionnaire under our attention check. The other half default to flat or central-tendency outputs that look near-human on average without tracking item content. A neutral Nordic-respondent persona, written without any distributional information from the human sample, brings the engaging models 44–77% closer to the Norwegian mean in Mahalanobis d^2. One-pair ActAdd at a fixed mid-layer flattens the foundation profile rather than steering individual foundations. For at least one model the same persona that shifts the profile also inhumans we use Moral Foundations Theory (Graham et al., 2013, 2011), which groups human moral judgement into five intuitive foundations, across two clusters. The *individualizing* cluster contains care and fairness, the *binding* cluster contains loyalty, authority and purity. A respondent's profile is the per-foundation score pattern across the five foundations. We measure these profiles with the Moral Foundations Questionnaire (MFQ; Graham et al., 2011), specifically the Norwegian MFQ-30 variant described in §3.

In this work we approach these questions in a Norwegian setting (§3). Our work consists of three phases. In Phase 1 we built a pipeline to make LLMs answer the questionnaire and produced a moral baseline per model. In Phase 2 we steered the models with prompt steering, which prepends a persona description to the system prompt to alter behaviour. In Phase 3 we steered them with activation steering (ActAdd; Turner et al., 2023), which injects a contrastive steering vector at a chosen transformer layer and nudges the model from inside. We evaluate both steering methods against the same baseline. Our code is available online.[1]

Best paper

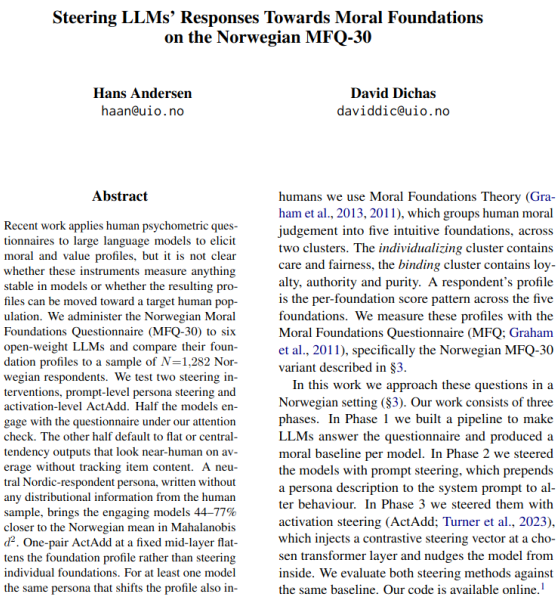
- Authors: Hans Andersen and David Dichas
- Title: Steering LLMs' Responses Towards Moral Foundations on the Norwegian MFQ-30

Highest scores across obligatory assignments

- Arthur Spalanzani, Elias Trana and Ola Lømo Ellingsen, with full 32 points each
- Also featured in almost all the leaderboards.
- Great job!

Awards

- Only MSc students considered for awards, not PhD fellows.



Steering LLMs' Responses Towards Moral Foundations on the Norwegian MFQ-30

Hans Andersen
haan@uio.no

David Dichas
daviddic@uio.no

Abstract

Recent work applies human psychometric questionnaires to large language models to elicit moral and value profiles, but it is not clear whether these instruments measure anything stable in models or whether the resulting profiles can be moved toward a target human population. We administer the Norwegian Moral Foundations Questionnaire (MFQ-30) to six open-weight LLMs and compare their foundation profiles to a sample of $N=1{,}282$ Norwegian respondents. We test two steering interventions, prompt-level persona steering and activation-level ActAdd. Half the models engage with the questionnaire under our attention check. The other half default to flat or central-tendency outputs that look near-human on average without tracking item content. A neutral Nordic-respondent persona, written without any distributional information from the human sample, brings the engaging models 44–77% closer to the Norwegian mean in Mahalanobis d^2. One-pair ActAdd at a fixed mid-layer flattens the foundation profile rather than steering individual foundations. For at least one model the same persona that shifts the profile also inhumans we use Moral Foundations Theory (Graham et al., 2013, 2011), which groups human moral judgement into five intuitive foundations, across two clusters. The *individualizing* cluster contains care and fairness, the *binding* cluster contains loyalty, authority and purity. A respondent's profile is the per-foundation score pattern across the five foundations. We measure these profiles with the Moral Foundations Questionnaire (MFQ; Graham et al., 2011), specifically the Norwegian MFQ-30 variant described in §3.

In this work we approach these questions in a Norwegian setting (§3). Our work consists of three phases. In Phase 1 we built a pipeline to make LLMs answer the questionnaire and produced a moral baseline per model. In Phase 2 we steered the models with prompt steering, which prepends a persona description to the system prompt to alter behaviour. In Phase 3 we steered them with activation steering (ActAdd; Turner et al., 2023), which injects a contrastive steering vector at a chosen transformer layer and nudges the model from inside. We evaluate both steering methods against the same baseline. Our code is available online.[1]

Best paper

- Authors: Hans Andersen and David Dichas
- Title: Steering LLMs' Responses Towards Moral Foundations on the Norwegian MFQ-30

Highest scores across obligatory assignments

- Arthur Spalanzani, Elias Trana and Ola Lømo Ellingsen, with full 32 points each
- Also featured in almost all the leaderboards.
- Great job!

And now.... the presentations!

Track Overview: Machine-generated code detection

Main info

Track chairs: Vladislav Mikhailov

Learn more: `https://github.uio.no/in5550/2026/tree/main/exam/generated_code_detection`

Track Overview: Machine-generated code detection

Background

- Modern generative LMs are widely used on a daily basis for software development: code completion, debugging, documentation, etc.
- However, such models can be misused for harmful purposes, e.g., generating vulnerable code, obfuscated malware, plagiarized solutions, or bypassing security and licensing constraints.

Track Overview: Machine-generated code detection

Background

- Modern generative LMs are widely used on a daily basis for software development: code completion, debugging, documentation, etc.
- However, such models can be misused for harmful purposes, e.g., generating vulnerable code, obfuscated malware, plagiarized solutions, or bypassing security and licensing constraints.

Machine-generated code detection

- A specialized area focused on mitigating the risks associated with the misuse of code-generating LMs.
- Developing datasets, benchmarks, and detection systems to assist humans in identifying machine-generated code.

Track Overview: Machine-generated code detection

Goal

- Develop your own system for a task formulation of your choice.

Track Overview: Machine-generated code detection

Goal

- Develop your own system for a task formulation of your choice.

Data

- SemEval-2026 Task 13: Detecting Machine-Generated Code with Multiple Programming Languages, Generators, and Application Scenarios
- Robustness across different programming languages, domains, and generation techniques.

Track Overview: Machine-generated code detection

Goal

- Develop your own system for a task formulation of your choice.

Data

- SemEval-2026 Task 13: Detecting Machine-Generated Code with Multiple Programming Languages, Generators, and Application Scenarios
- Robustness across different programming languages, domains, and generation techniques.

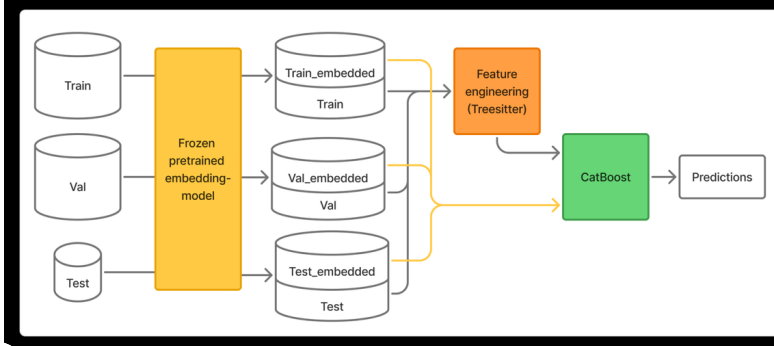
Task Formulations

- Subtask A: is the code snippet created by a human or an LM?
- Subtask B: if latter, which LM generated the code?
- Subtask C: is the code snippet multi-authored?

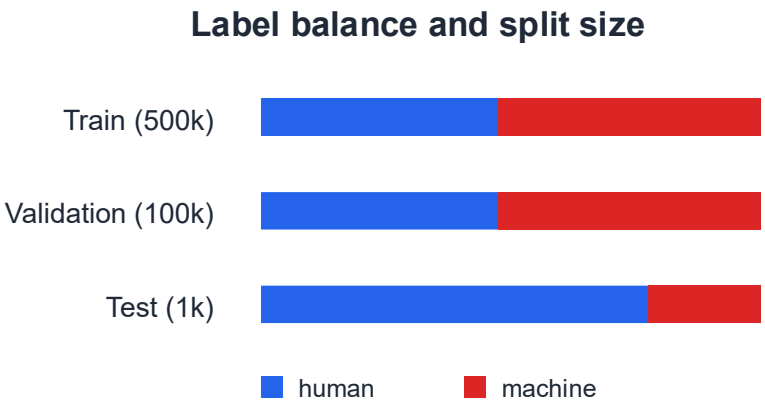
Track Overview: Machine-generated code detection

General Workflow

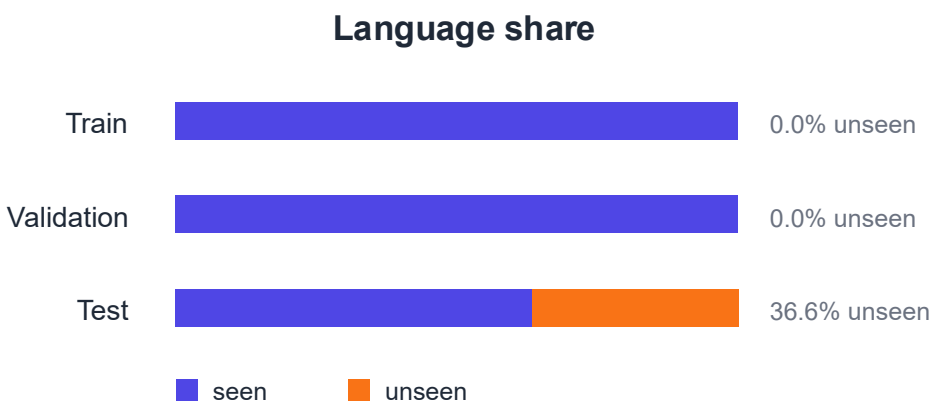
- Select one or more task formulations.
- Analyze the participants' submissions, think about how they can be improved or come up with your own solutions.
- Try various types of detectors (zero-shot, feature-based, and Transformer-based detectors) and their ensembles.
- Conduct ablation studies to analyze the performance w.r.t. robustness, training data mixtures, dataset criteria.
- Compare the performance of your resulting system against the corresponding public leaderboards.



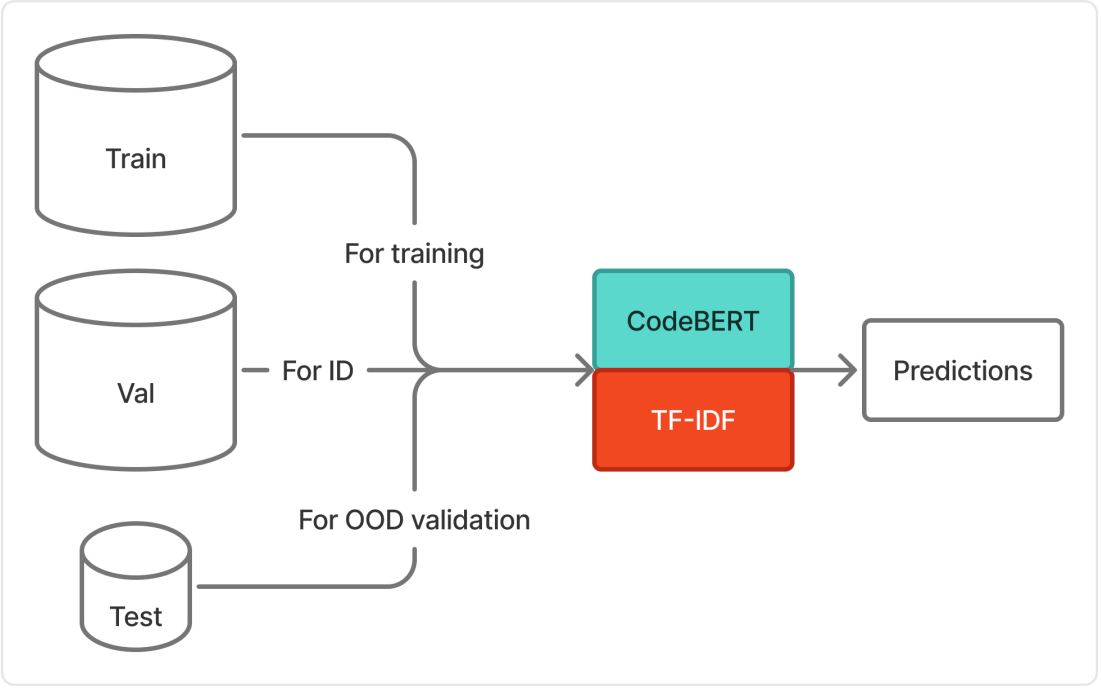
Frozen Embeddings and Parser Features for Machine-Generated Code Detection Under Distribution Shift

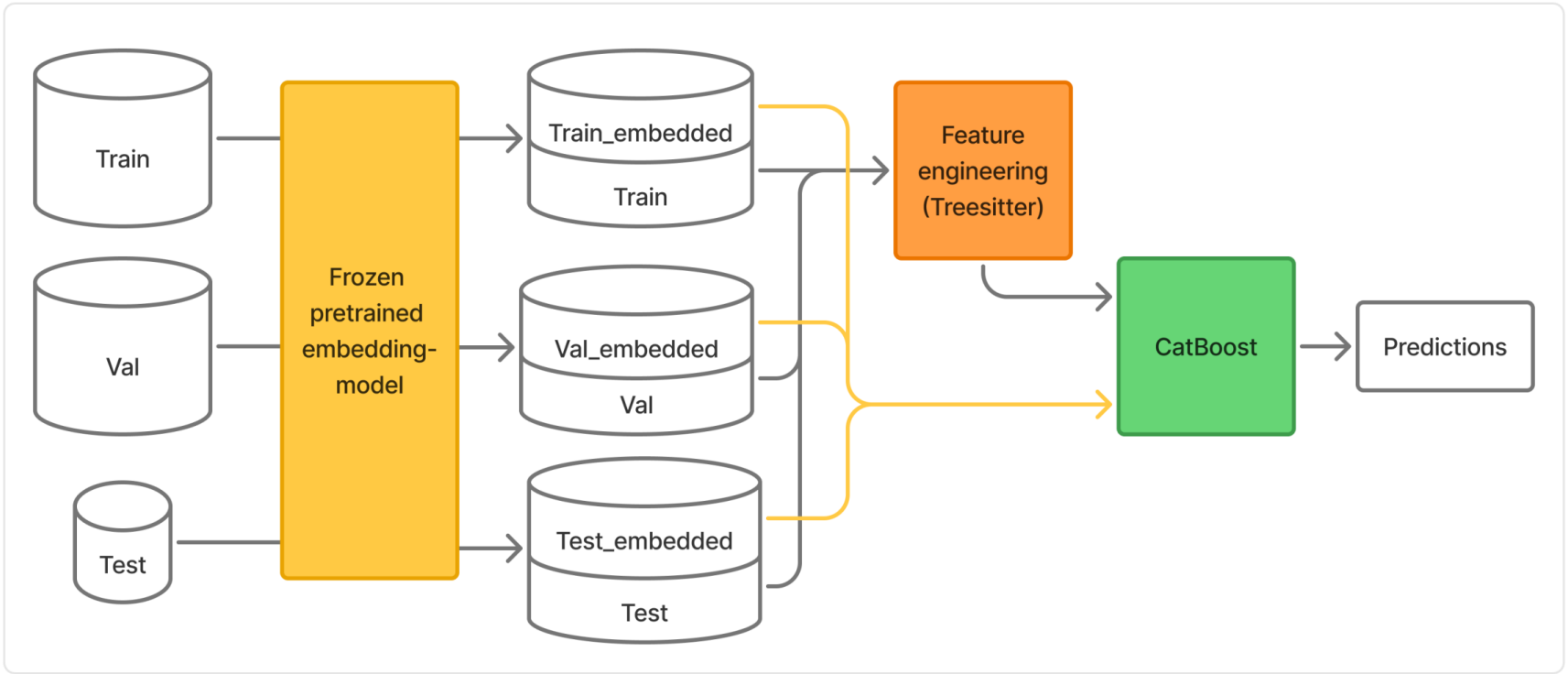

Label balance and split size
Train (500k)
Validation (100k)
Test (1k)
human
machine

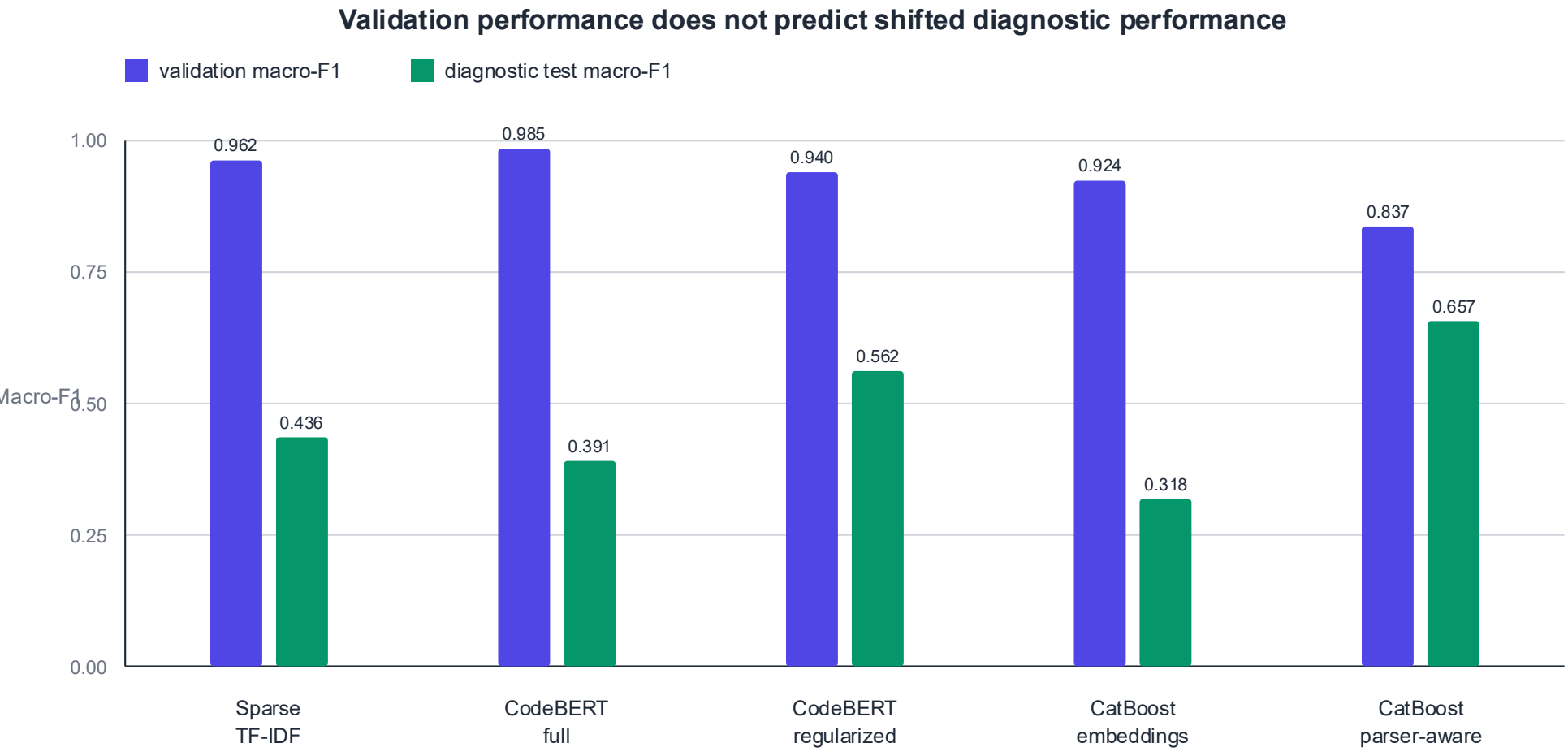

Median code length
Train
Validation
Test
319
726
317
728
803
994
0
500
1000
Characters


Language share
Train
Validation
Test
0.0% unseen
0.0% unseen
36.6% unseen
seen
unseen

Baseline and “modern” comparison

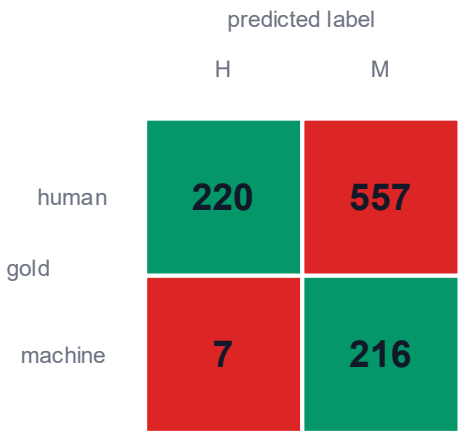



Train
Val
Test
Frozen pretrained embedding-model
Train_embedded
Train
Val_embedded
Val
Test_embedded
Test
Feature engineering (Treesitter)
CatBoost
Predictions


Validation performance does not predict shifted diagnostic performance
validation macro-F1
diagnostic test macro-F1
1.00
0.75
0.50
0.25
0.00
Macro-F1
0.962
0.436
0.985
0.391
0.940
0.562
0.924
0.318
0.837
0.657
Sparse TF-IDF
CodeBERT full
CodeBERT regularized
CatBoost embeddings
CatBoost parser-aware

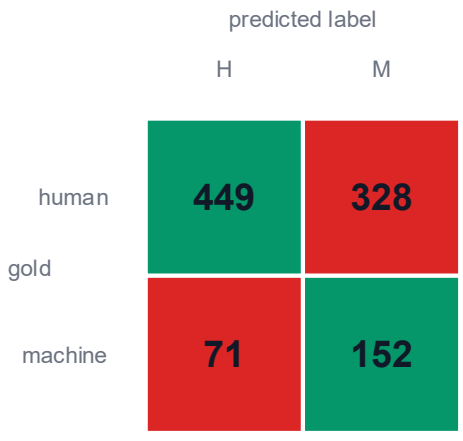
Diagnostic test confusion matrices

Sparse TF-IDF



H recall 0.28, M recall 0.97

Regularized CodeBERT

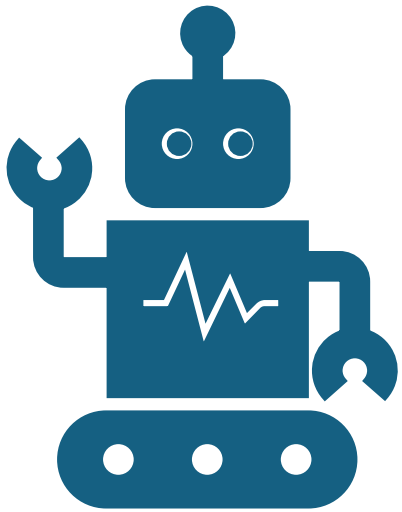


H recall 0.58, M recall 0.68

Parser-aware CatBoost

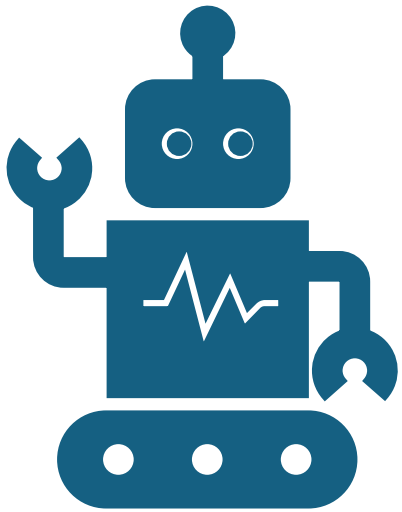


H recall 0.75, M recall 0.63



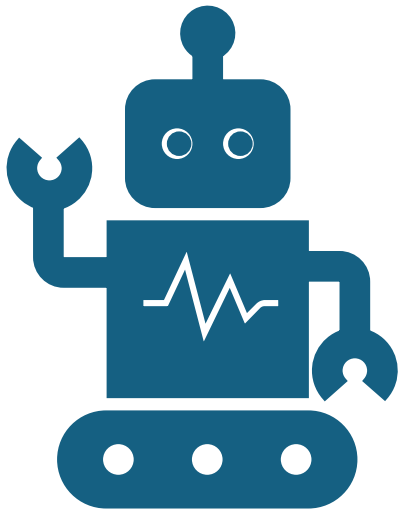
Key takeaway

- AST features are helpful in cross language code detection



Key takeaway

- AST features are helpful in cross language code detection
- Gradient boosted forest models + feature engineering is still a useful low-cost alternative to transformers



Key takeaway

- AST features are helpful in cross language code detection
- Gradient boosted forest models + feature engineering is still a useful low-cost alternative to transformers
- Shortcut-resistant features and robust validation design can matter as much as model capacity.

Detecting Machine-Generated Code with Fine-Tuned CodeBERT

Juni Søreng · University of Oslo · IN5550 Exam

Track: Machine-generated code detection

Four detectors compared:

1. TF-IDF char n-grams + Logistic Regression
2. TF-IDF word n-grams + Logistic Regression
3. Frozen CodeBERT (head only, 0.2% params trained)
4. Fully fine-tuned CodeBERT (all params)

Results: Validation vs. Test Sample

In-distribution performance ≠ out-of-distribution performance

Model	Val F1	Test F1
TF-IDF word n-grams	0.8993	0.3143
TF-IDF char n-grams	0.9374	0.2999
Frozen CodeBERT	0.9579	**0.4120**
Full CodeBERT	**0.9942**	0.2571

Key finding

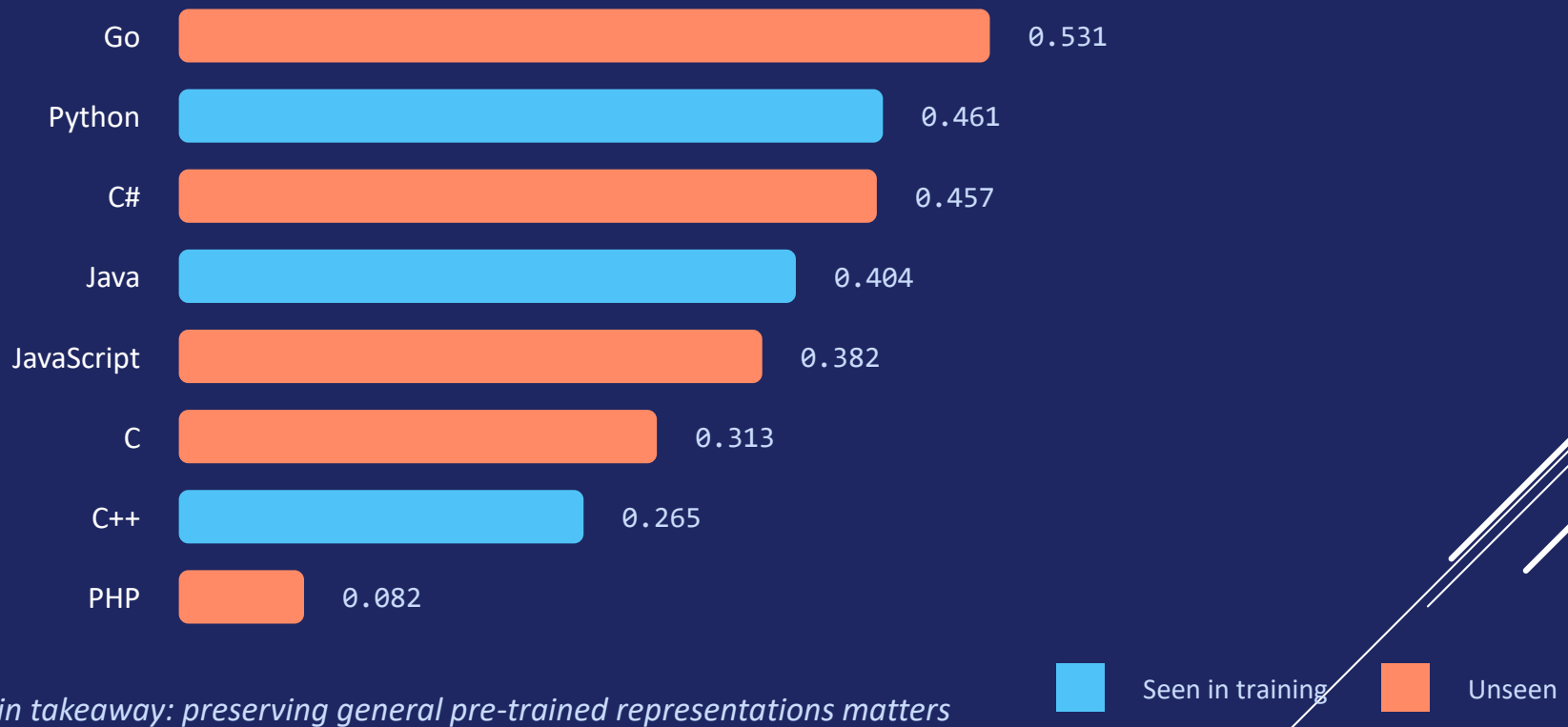
The frozen model (0.2% params trained) generalizes BETTER than the fully fine-tuned model.

Full fine-tuning overfits to training patterns that don't transfer to unseen languages and domains.

Seen vs. unseen language gap (test):

```
Full CodeBERT:  0.275 → 0.168  (gap: 0.107)
Frozen CodeBERT:  0.376 → 0.353  (gap: 0.023)
```

Per-Language Test Performance (Frozen CodeBERT)



Main takeaway: preserving general pre-trained representations matters more than maximizing validation accuracy.

Detecting Machine-Generated Code

A Study of Out-Of-Domain Generalization

Sigurd Nordbye · Muhammad Zeeshan Shahid · Ola Gabriel Huus Skeie

SemEval-2026 Task 13, Subtask A · IN5550 WNNLP 2026

Method

Neural encoders

- CodeBERT
  - Pre-trained on code and matched many dataset languages.

Gradient Boosted Trees

- Gradient boosted trees build decision trees sequentially, where each tree fixes previous errors.

- XGBoost
  - Handcrafted stylometric features (52)
  - Character n-gram TF-IDF (3–5 grams)
  - Word n-gram TF-IDF (1–2 grams)

- LightGBM
  - UniXcoder embeddings (768-dim)
  - Handcrafted stylometric features (24)
  - 9 TF-IDF expert probability scores

Results

F1 0.648

XGBoost

Best clean system

F1 0.498

LightGBM

With UniXcoder

F1 0.260

CodeBERT

Val F1 was 0.995

System Comparison

System	Test F1	Kaggle F1	Note
Majority class	0.437	0.437	All human
CodeBERT fine-tuned	0.260	0.260	Val 0.995 → collapse
GPT-4o zero-shot	0.632	0.632	No training needed
XGBoost (ours)	**0.648**	**0.630**	OOD validation
ABC combined†	–	*0.941†*	*Contaminated*

Discussion

Why neural encoders fail

- CodeBERT val F1 0.995 → test F1 0.260, below majority class baseline
- Memorises generator-specific patterns that don't transfer to GPT-4o or Gemini

Why XGBoost does better

- Character n-grams and empty line ratio capture more general patterns
- OOD validation (C++ + Gemma held out) picks better models than standard val

Leaderboard contamination

- 67.4% of Subtask A test set appears verbatim in B+C training data
- A lookup oracle with no ML scores F1 0.940 — enough for 2nd place

Detecting Machine-Generated Code Across Languages and Domains

A Comparative Study

Berk Hadzhamolla

Motivation

Problem:

- Large Language Models can generate increasingly realistic code.
- Difficult to distinguish human-written from machine-generated code.
- Important for:
  - Academic integrity
  - Software quality
  - Security and trustworthiness

Research Question

Can language-balanced training and improved model selection improve out-of-distribution generalization for machine-generated code detection?

Task Overview

Training Data

- Languages: Python, Java, C++
- Domain: Algorithmic programming tasks

Evaluation Data

- 8 programming languages
- Multiple domains
  - Algorithmic
  - Research & Production

Challenge: Generalization under distribution shift.

Exploratory Data Analysis

Language Distribution

Language	Percentage
Python	91.5%
C++	4.7%
Java	3.9%

Observation

- Training data is heavily dominated by Python.
- Models may learn Python-specific patterns.
- Risk of poor cross-language generalization.

Exploratory Data Analysis

Code Length Analysis

- Machine-generated code is generally longer.
- However:
  - C++: Human code tends to be longer.
  - Java: Human code tends to be longer.

Implication

Code length is not a universal indicator of machine generation.

Simple heuristics are unlikely to generalize across languages.

Compared Methods

1. TF-IDF + Logistic Regression
2. Frozen Encoder CodeBERT
3. Fine-tuned CodeBERT (Pre-trained on 6 languages)
4. Fine-tuned StarEncoder (Pre-trained on 86 languages)
5. Ensemble

Goal

Compare:

- Traditional machine learning
- Frozen representations
- Fine-tuned transformers
- Ensemble learning

Validation Results

Model	Macro-F1
Frozen Encoder	0.3360
TF-IDF + LR	0.9480
CodeBERT	0.9952
StarEncoder	0.9968
Ensemble	**0.9971**

Observation

Fine-tuned transformer models achieve near-perfect in-distribution performance.

Out-of-Distribution Results

Model	Kaggle Score
Frozen Encoder	**0.4974**
CodeBERT	0.2546
StarEncoder	0.2484
Ensemble	0.2406

Unexpected Result

The frozen encoder substantially outperforms all fine-tuned models.

Why Did Fine-Tuning Fail?

Shortcut Learning

- Learning superficial patterns.

Generator Overfitting

- Memorizing generator-specific artifacts.

Domain Shift

- Research and production code differ from training data.

Language Bias

- Python dominates the training set.

Main Findings

- StarEncoder + balancing improves validation performance.
- Ensemble achieves best in-distribution score.
- All fine-tuned models collapse under distribution shift.
- Frozen CodeBERT achieves the best leaderboard score.

Key Finding

Fine-tuning on a narrow training distribution can harm out-of-distribution generalization.

Conclusion

Takeaway

High validation performance does not imply robust generalization.

The strongest result of this study is that:

A frozen encoder outperformed all fine-tuned models on out-of-distribution data.

This highlights the importance of robustness and distribution-aware evaluation for machine-generated code detection.

The fine-tuned encoder doesn't just collapse. It inverts.

Setup. Train: 91.4% Python. Test: 5 unseen languages, class balance flipped to 77.7% Human.

Result. Validation F_1 0.993 $\rightarrow$ test F_1 0.240.

Both encoders we tested have **negative MCC** on the labeled proxy:

Model	Proxy MCC
CodeBERT	−0.113
UniXcoder	−0.191
Constant predictor	0.000

The models learned a rule. The rule is wrong more often than right.

Why? Train is 91.4% Python. Encoders associated Python-specific surface patterns with *Human*. On unseen languages, those cues are absent and the models default to *AI*.

Both encoders predict AI on ∼87% of test samples vs. a true rate of 22.3%.

The "Illusion of Competence"

High in-distribution validation reflects *distribution overlap*, not authorship learning. Memorized boundaries do not transfer.

ASC: strip the vocabulary, keep the rhythm

Abstract Structural Cadence (ASC). Map every character to one of five classes:

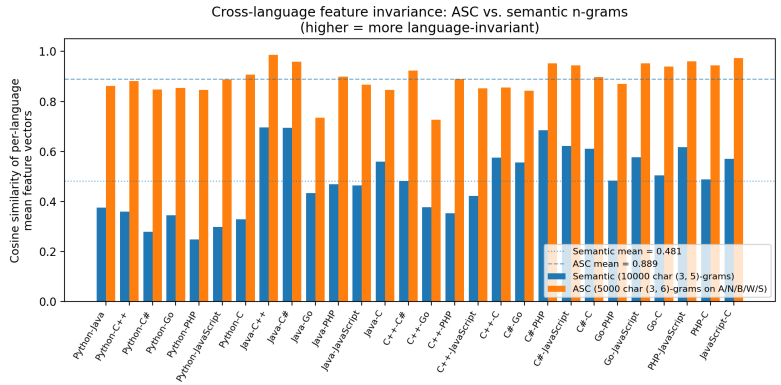
- `A` alphabetical
- `N` numeric
- `B` brackets
- `W` whitespace
- `S` symbols/operators

Then character *n*-grams (3 to 6) + logistic regression. Trains in 10 minutes on a single CPU.

Cross-language feature invariance, measured directly:

Mean cosine sim.	across 28 lang. pairs
Semantic *n*-grams	0.48
ASC *n*-grams	**0.89**

ASC beats semantic on **all 28 pairs**, no exceptions.



Cosine similarity between per-language mean feature vectors. ASC stays high; semantic features fall apart between Python and other languages.

Results: ASC works, but the honest picture is humbling

Leaderboard Macro F_1:

System	Test F_1
CodeBERT (fine-tuned)	0.240
UniXcoder (fine-tuned)	0.228
Semantic TF-IDF only	0.306
Dual-Stream (sem.+ASC)	0.309
ASC-only (ours)	**0.361**
All-Human (trivial)	**0.438**
FITUS-Simple (external)	$\sim$0.715

ASC is 50% above CodeBERT, but **no trained system clears the all-Human baseline**.

The honest reading.

Under F_1: nothing we trained beats predicting "Human" for every sample.

Under MCC: only the ASC-bearing systems are positively correlated with the truth.

Takeaways:

- Encoder fine-tuning on a narrow language distribution fails *worse* than "OOD collapse."
- Structural abstraction is a real axis of cross-language robustness, but not sufficient alone.
- Next: ASC + handcrafted stylometry + non-linear classifier.

Machine-generated code detection

Outstanding paper in this track

Machine-generated code detection

Outstanding paper in this track

Frozen Embeddings and Parser Features for Machine-Generated Code Detection Under Distribution Shift

Samuel Valland Lyngset
Department of Informatics, University of Oslo
Oslo, Norway
samuelvl@uio.no

Abstract

This paper studies binary machine-generated code detection for SemEval-2026 Task 13, Subtask A, a task designed to evaluate generalization across programming languages, domains, and generator families. The experiments compare sparse term frequency–inverse document frequency (TF-IDF) features, fine-tuned CodeBERT (a pretrained Transformer encoder for source code and natural language), and frozen code embeddings with engineered CatBoost (a gradient-boosted decision-tree model) features. Validation performance is a poor proxy for shifted performance: full transformer fine-tuning and broad feature engineering both learn brittle shortcuts, while a narrower parser-aware CatBoost model gives the most balanced diagnostic behavior. The results support parser-aware CatBoost, optionally augmented with frozen embeddings, as a practical alternative when generalization, interator families. A detector that works only on the source distribution is therefore not sufficient.

This generalization requirement motivates the main modeling question: can a parser-aware CatBoost pipeline, optionally enriched with frozen embeddings, provide a practical alternative to full transformer fine-tuning when shifted robustness matters more than in-distribution validation performance? The paper uses the exposed 1,000-example labeled test split only as a diagnostic split, since later modeling decisions were informed by it.

The contributions are: (1) an empirical comparison of sparse, fine-tuned transformer, and parser-aware CatBoost detectors under the task's intended distribution shift; (2) feature ablations showing that broad formatting features can harm generalization; (3) an input-source comparison separating encoder representations, code features,

Track Overview: LLMs and Moral Foundations

Track chairs: Hannah Clausen and Andrey Kutuzov

Research problem

- You are given a questionnaire and human responses.
- Can we make LLMs respond to the questionnaire in a human-like way?
- Can we steer their responses using activation and prompt steering?
- How uncertain the models are about their responses?

Track Overview: LLMs and Moral Foundations

Track chairs: Hannah Clausen and Andrey Kutuzov

Research problem

- You are given a questionnaire and human responses.
- Can we make LLMs respond to the questionnaire in a human-like way?
- Can we steer their responses using activation and prompt steering?
- How uncertain the models are about their responses?

Moral Foundations Questionnaire (MFQ-30)

- An instrument measuring five moral foundations: Care/Harm, Fairness/Cheating, Loyalty/Betrayal, Authority/Subversion, Sanctity/Degradation
- Human sample from a Norwegian population (N=1282)
- Can we observe human-like patterns in model responses?
  - E.g., human respondents who score high on care and fairness, tend to score low on loyalty, authority, and sanctity, and vice versa.

Track Overview: LLMs and Moral Foundations

Making the models respond to the MFQ-30

- Set up an experimental framework
  - Questionnaire presentation
  - Answer option perturbations
  - Generation strategies etc.
- Compare your results to the human sample

Track Overview: LLMs and Moral Foundations

Making the models respond to the MFQ-30

- Set up an experimental framework
  - Questionnaire presentation
  - Answer option perturbations
  - Generation strategies etc.
- Compare your results to the human sample

Estimating model uncertainty

- Are the models consistent in their 'moral values'?
- Or are they producing very different generations with low probabilities?
- Use methods from uncertainty estimation literature

Track Overview: LLMs and Moral Foundations

Making the models respond to the MFQ-30

- Set up an experimental framework
  - Questionnaire presentation
  - Answer option perturbations
  - Generation strategies etc.
- Compare your results to the human sample

Estimating model uncertainty

- Are the models consistent in their 'moral values'?
- Or are they producing very different generations with low probabilities?
- Use methods from uncertainty estimation literature

Steering the models' foundations

- Can you push the model to certain moral foundations?
- **Activation steering** focuses on adjusting the internal activations
- **Prompt steering** focuses on systematically adjusting the prompt

UNIVERSITY OF OSLO

Steering Moral Foundations in Qwen3-8B

Can we reliably shift the model's moral foundations on the MFQ-30 questionnaire using prompt and activation steering?



Christian H. Grusd

31st May 2026

Inspiration: The Golden Gate Eiffel Tower Llama moral philosopher



UNIVERSITY OF OSLO

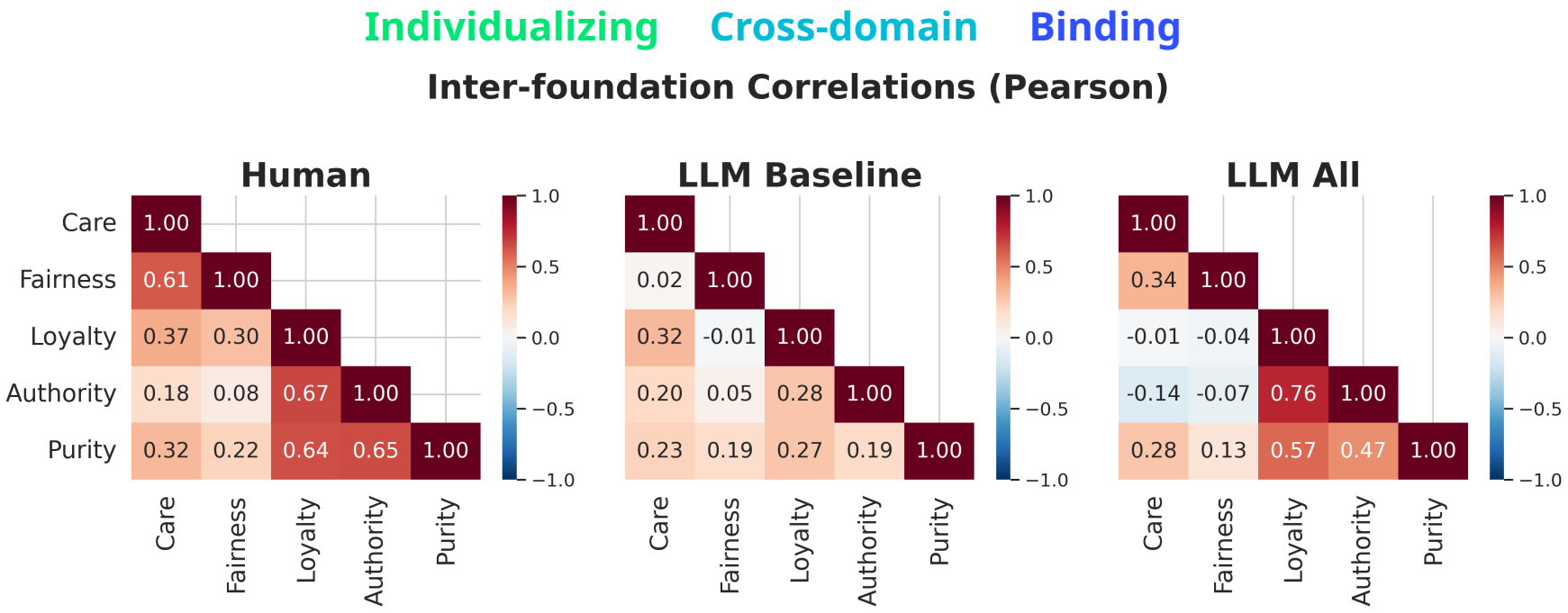
Image generated with Google Gemini, May 2026

Baseline, prompt and activation steering in Qwen3-8B

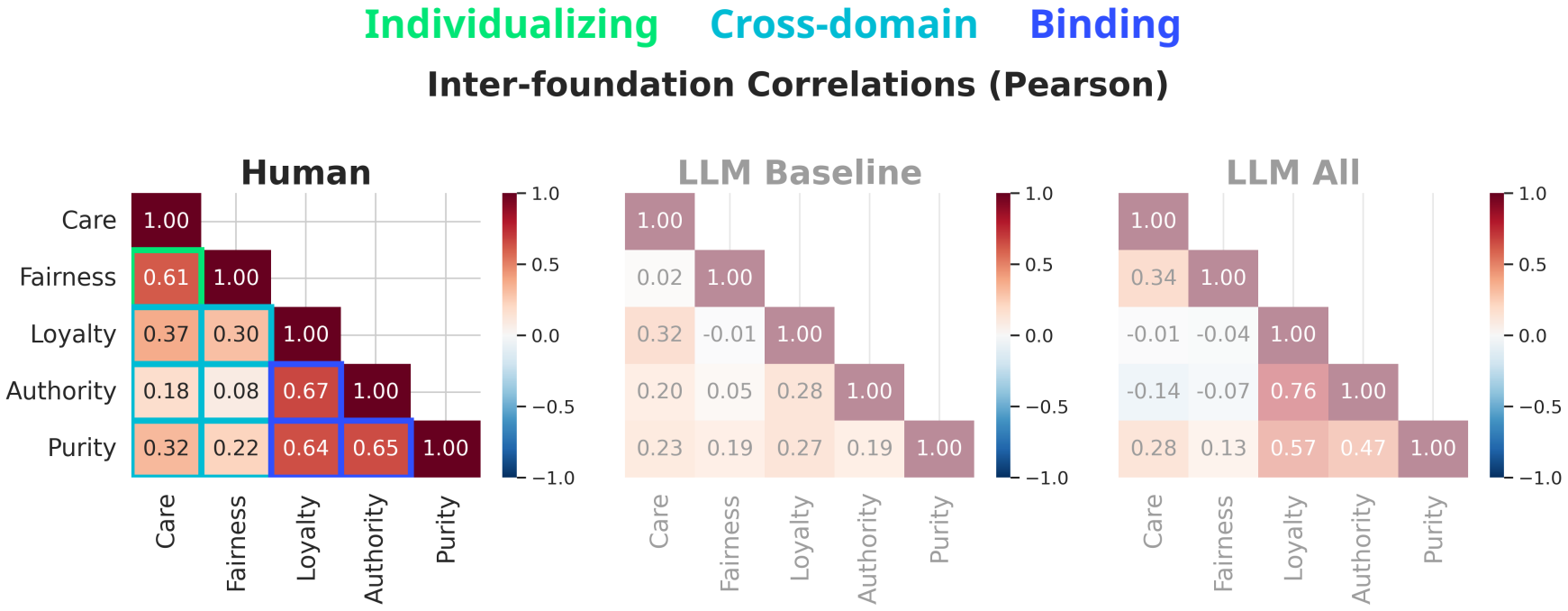
The **MFQ-30** was administered to **Qwen3-8B** under three conditions: unsteered baseline, prompt steering, and activation steering.

1. The two-cluster structure (individualizing and binding) **emerges under steering**
2. Both methods steer **on-target** foundations while prompt steering produces larger shifts

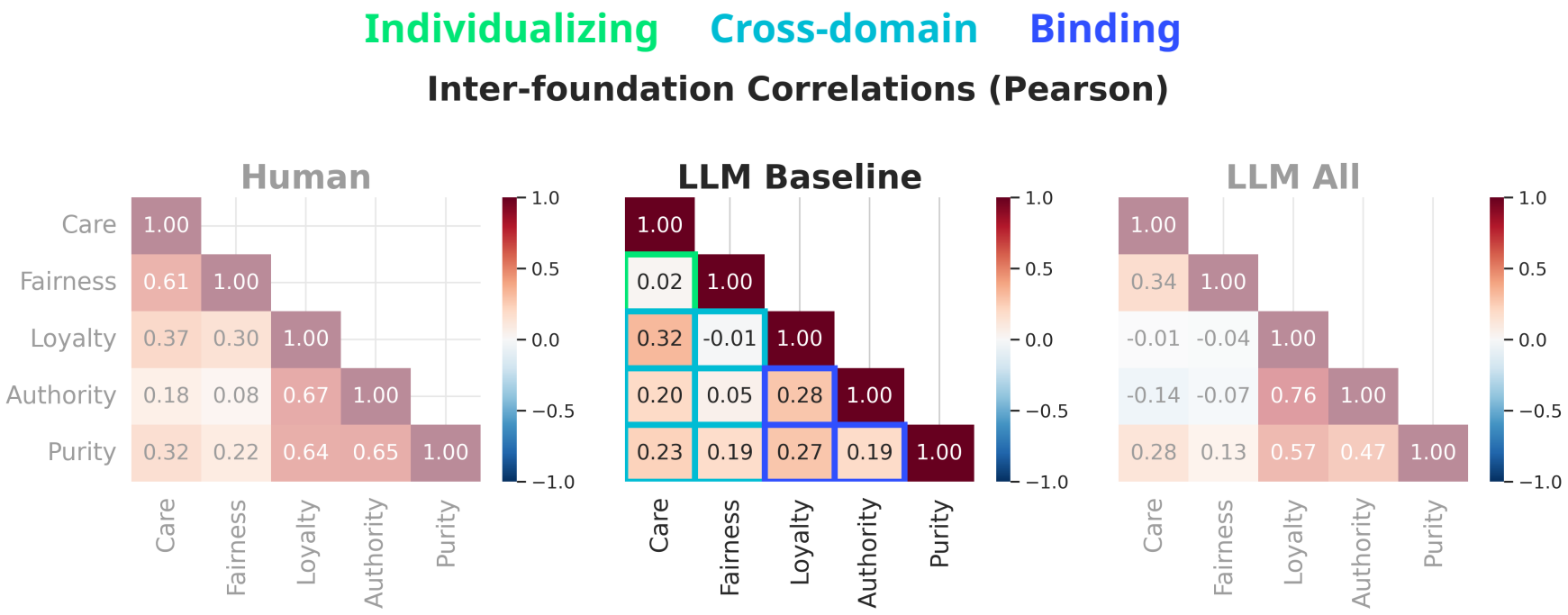
The two-cluster structure emerges under steering



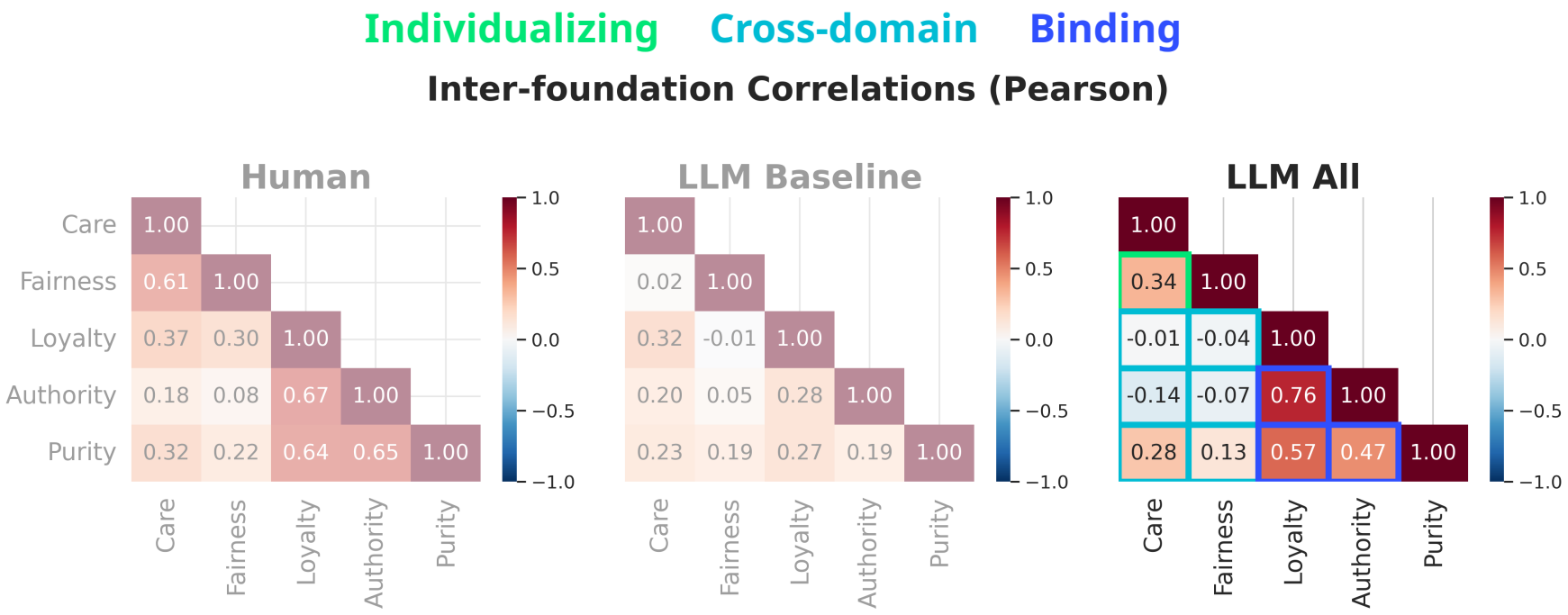
The two-cluster structure emerges under steering



The two-cluster structure emerges under steering



The two-cluster structure emerges under steering



Prompt steering shifts more than activation steering



Prompt steering shifts more than activation steering



Method

1. Framework: Establishing an optimal LLM

Question order / presentation

- block, per item, full
- regular, reversed, mixed

Response generation methods

- Open generation, first-token, constrained

Hyperparameters

- Temperature
- Max new tokens
- Num_samples = 3

2. Steering

Prompting Steering

- Moral Foundations
  - C/F and L/A/P
- Persona Steering
  - Young/Old adult
  - Religious / Non-religous
  - Conservative American Christian

Activation Steering

- Standard ActAdd
  - C/F
  - L/A/P
- First Token ActAdd
  - C/F
  - L/A/P

3. Evaluation Dashboard

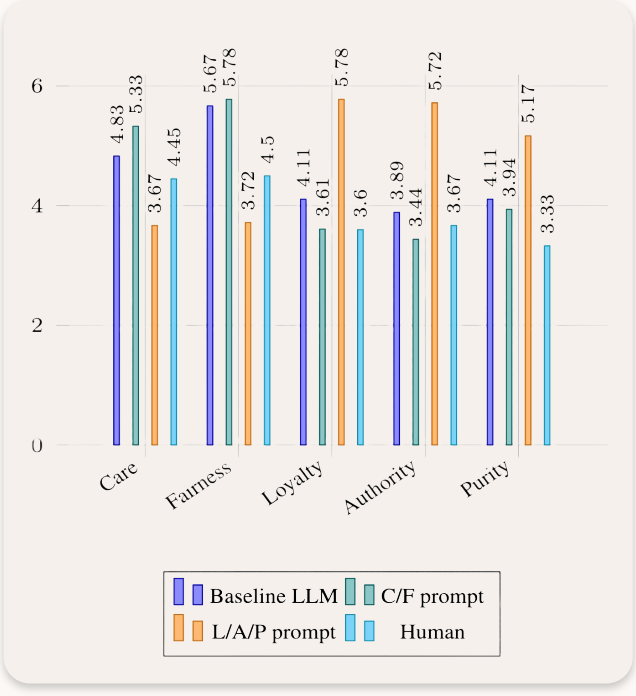
- Automatic parsing
- Human-similarity-score
- LLM-as-a-judge
- Qualitative spot checking

Results

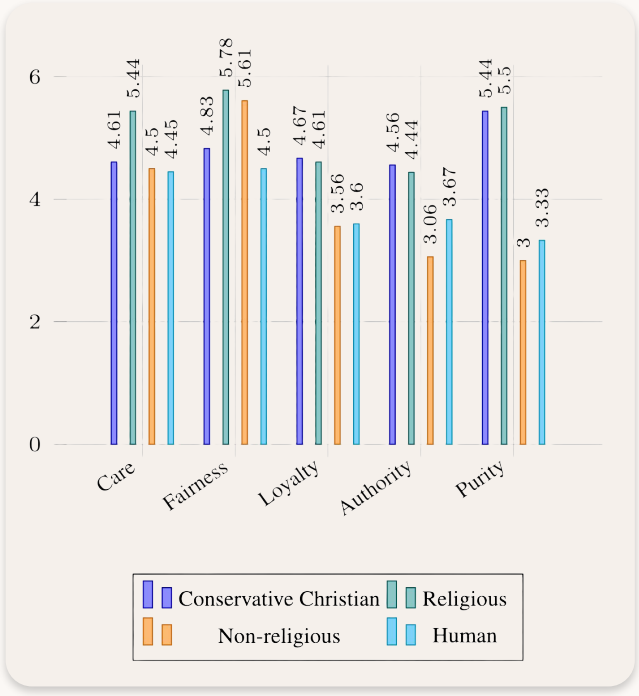
1-6: Fully disagree – Fully agree

Optimal configuration: Qwen3-14B, open generation, regular answer order, per-item presentation

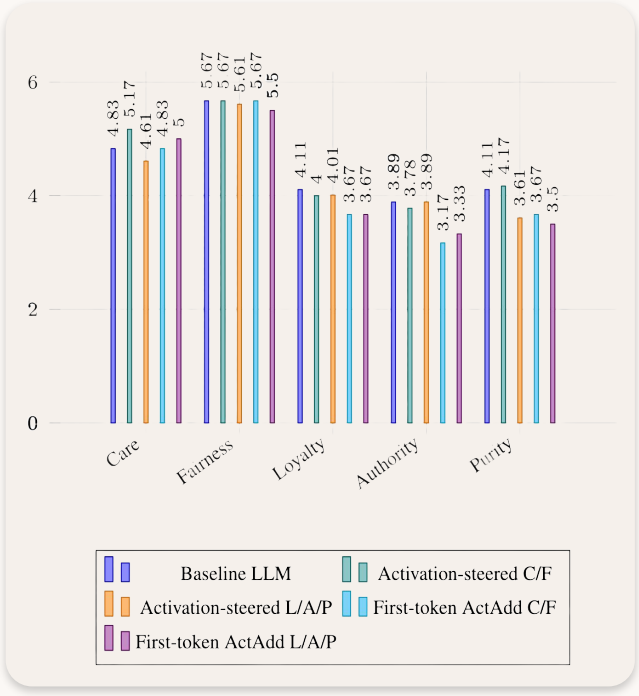
C/F and L/A/P prompts moved the model cleanly along the **individualizing/binding** axis. Confirming a coherent moral stance.



The LLM differentiated a conservative American Christian from a Religious person, by reducing Care and Fairness



All activation configurations resulted in marginal changes, both directions.



Discussion

Main take aways:

Prompt Steering is more effective than the tested implementations of Activation Steering. But Activation Steering would benefit from further testing

Utilizing LLM-as-a-judge and a dashboard allowed for wide testing without needing as many manual checks

Using a vague prompts forced the model to use its own biases, rather than just impersonating its provided persona

Nezar Mahairi
nezarm@uio.no

Steering LLM Responses Towards Moral Foundations

Can LLMs answer a Norwegian moral questionnaire?
And can we change their moral profile?

📋 Questionnaire

Norwegian MFQ-30
5 moral foundations
32 items

🤖 Models

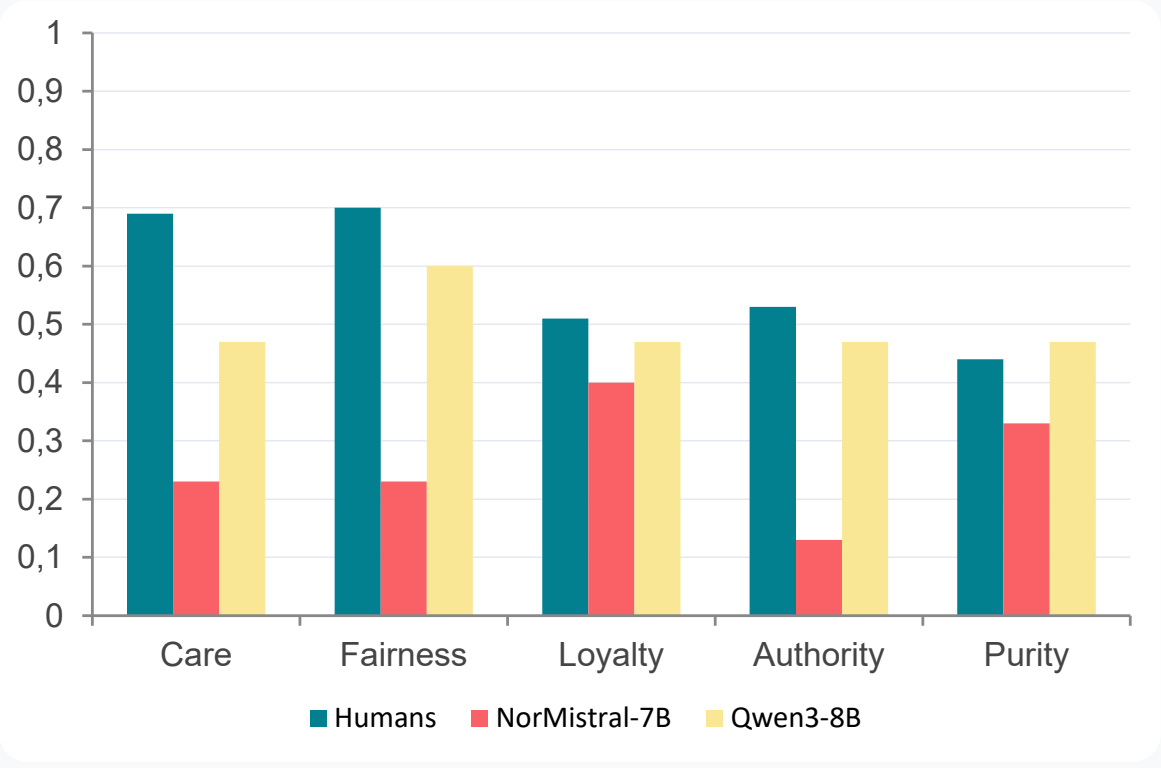
NorMistral-7B
Qwen3-8B
vs 1,282 humans

🎯 Methods

Prompt steering
Activation steering
(ActAdd)

WNNLP 2026 · MFQ Track

Results: Moral Profiles



Key Findings

✅ Humans: Care & Fairness highest

⚠️ Qwen3: flat profile (no differentiation)

⚠️ NorMistral: Loyalty > Care (reversed!)

❌ Neither matches human pattern

Scores normalized to [0,1] scale

Steering & Conclusion

Steering Results

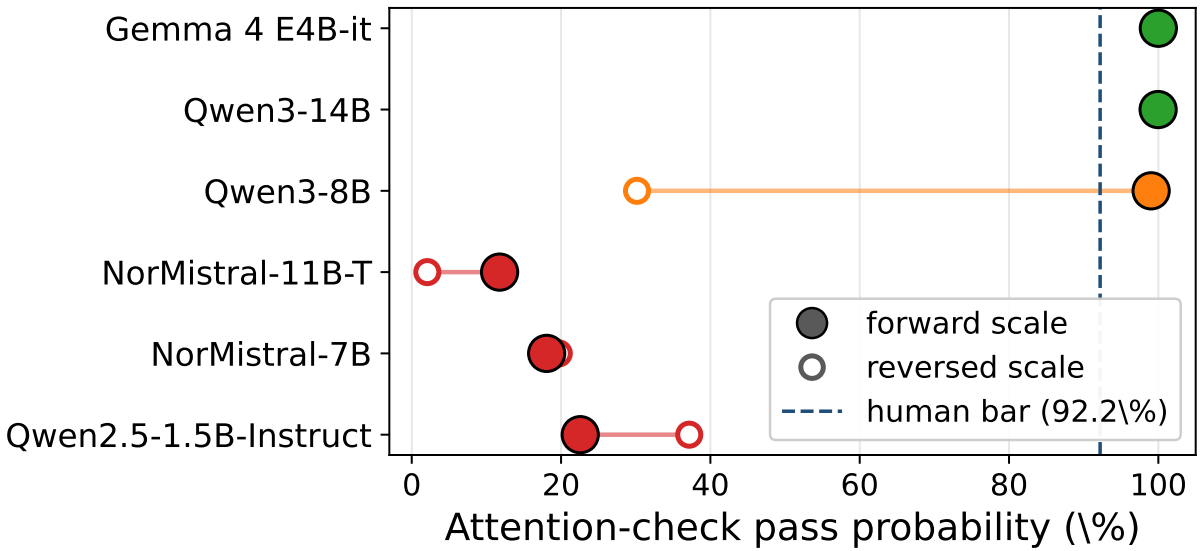
Method	Effect
Prompt (Qwen3)	All scores → 0.60 No differentiation
Prompt (NorMistral)	Unpredictable Sometimes reversed
ActAdd (Qwen3)	All foundations shift together

Takeaways

1. LLMs give stable responses but zero variance across runs
2. Neither model matches the human two-cluster pattern
3. Steering shifts overall level not specific foundations
4. Use caution applying human psychometrics to LLMs

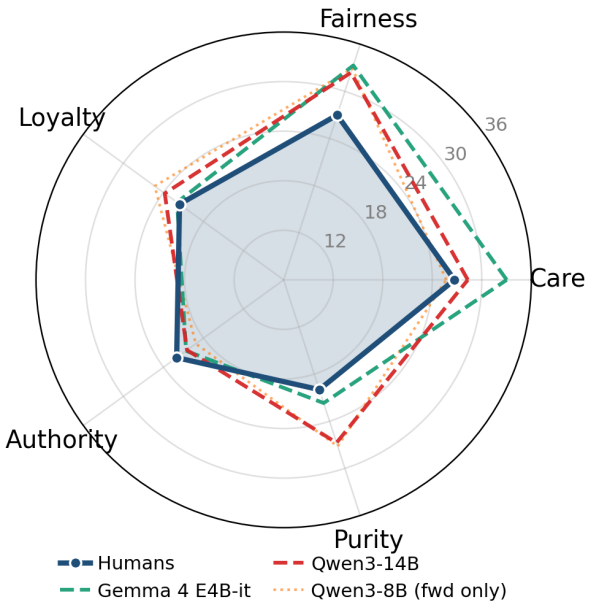
Code: anonymous.4open.science/r/exam-in5550-NLP-E878

Half our models don't actually read the questionnaire



- Six LLMs take the Norwegian moral survey.
- An attention check splits them: **three engage, three don't**.

Telling a model to “be Norwegian” moves it toward real Norwegians



The gap to the average Norwegian (blue), before any steering.

How much closer to the average Norwegian:

44% Gemma 4

62% Qwen3-14B

77% Qwen3-8B

The persona only says *who you are* — **no hints on how to answer**.

Steering from the inside just flattens — and the survey can mislead

A prompt steers; editing the model's internals doesn't

A persona shifts the model's answers a lot. Editing its internal activations instead just **flattens everything**.

The survey can show a profile that isn't really there (cognitive phantom)

Flip the 1–6 answer scale and one model mostly stopped engaging (∼30%). Add the persona and it engages again (∼98%). The persona didn't just change its answers — it decided **whether the model engaged at all**.

So a survey can show a "moral profile" the model only has because we told it to play one.

Baseline settings

For all four models:

1.1 One question at a time (OOQ)
1.2 All questions at once (AAQ)

2.1 Original question order (OQO)
2.2 Reversed question order (RQO)

3.1 Number-only answer (NOA)
3.2 Probability-Based Answer (PBA)
3.3 Free-text reasoning followed by a number (RNL)
3.4 Number followed by free-text reasoning (NRL)

Baseline filter:

Filter	Purpose
Valid responses	Exclude configurations with more than 10% missing or invalid answers.
Scale usage	Exclude configurations where the model gives nearly identical scores to all items.
Run stability	Prefer configurations with similar foundation scores across repeated runs.
Internal consistency	Prefer configurations where items within the same moral foundation produce coherent scores.
Control questions	Check whether the model gives sensible answers to items with clear expected responses.

Selected baseline

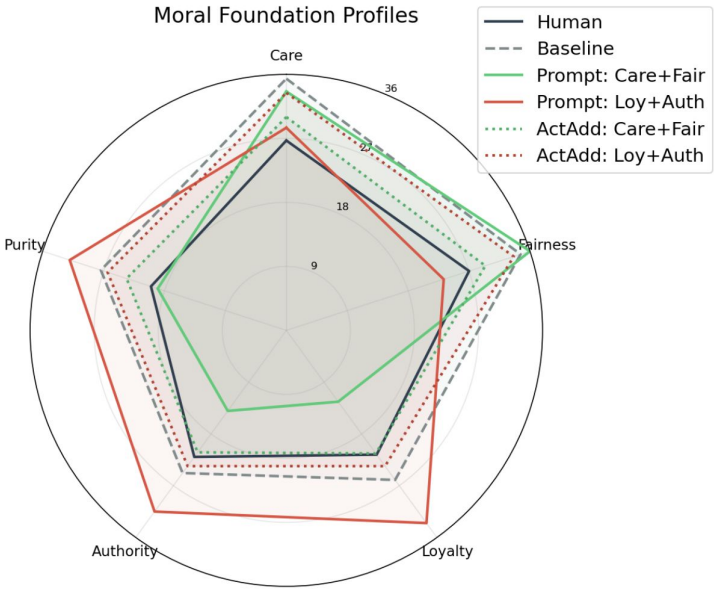
- Model: Qwen3-8B
  - Questionnaire presentation: **AAQ** - all at once
  - Question order: **OQO** - original order
  - Response generation: **NRL** - number before reasoning

- Reliable Numerical answers
- Meaningful scale
- Stable across runs

Baseline vs humans

Condition	Care	Fairness	Loyalty	Authority	Purity
Human	26.7	27.0	21.6	22.0	20.0
Baseline	35.4	34.6	26.0	24.8	27.4

Moral foundation Profiles



Effects of Moral-Foundation Steering

Condition	Care	Fairness	Loyalty	Authority	Purity
Human	26.7	27.0	21.6	22.0	20.0
Baseline	35.4	34.6	26.0	24.8	27.4
Prompt: Care+Fair	33.6	36.0	12.4	14.0	19.0
Prompt: Loy+Auth	28.5	23.2	33.5	31.5	32.0
ActAdd: Care+Fair	30.0	29.4	21.4	21.2	23.6
ActAdd: Loy+Auth	33.4	33.6	23.6	23.6	26.4

IN5550 · FINAL EXAM · MFQ TRACK

Measurement Validity for LLM Responses to the Norwegian MFQ-30

An eight-criterion construct-validity protocol and a steering stress-test

> **A coherent moral profile is not necessarily a valid one.**

Ane Kristine Espeseth · University of Oslo · anekes@uio.no

THE PROBLEM

Coherent ≠ valid

The default pipeline reads the `first-token logprob` over answer labels and sums by foundation. Three response patterns give a **coherent** profile with **no moral content**:

Acquiescence

Agrees with every statement, whatever it says.

Label-token prior

Favours one answer digit regardless of the item.

Centre-locking

Gives nearly the same rating to everything.

THE METHOD

The battery, then the screen

6 open-weight LLMs · Norwegian MFQ-30 × **3 methods** × **24-cell perturbation grid** per item

single-item Likert · battery Likert · forced-choice tournament

Response-process

- **G1** Filler items behave unlike real ones.
- **G7** Ignores which option is shown first.

Construct / content

- **G2** Foundations differ in strength.
- **G3** Items get a spread of scores.
- **G6** Signal survives rewording.

Reliability

- **G4** Stable across both MFQ frames.
- **G5** Stable across reverse-keyed items.

Convergence

- **G8** Agrees with the other models on item order.

FINDING 1 · WHO PASSES

Six models → three classes

	G1	G2	G3	G4	G5	G6	G7	G8
Qwen3-4B	✓ 0.48	✓ 0.24	✓ 0.47	✓ 0.50	✓ 0.50	✓ 1.58	✓ 0.31	✓ 0.75
Llama-3.1-8B	✓ 0.45	✓ 0.18	✓ 0.39	✓ 1.00	✗ 0.17	✓ 1.50	✓ 0.47	✓ 0.77
Gemma-3-12B	✓ 0.50	✓ 0.25	✓ 1.21	✓ 1.00	✓ 0.84	✓ 1.17	✓ 0.32	✓ 0.80
Aya-8B	✓ 0.48	✓ 0.23	✓ 0.38	✓ 0.70	✓ 0.29	✓ 1.18	✓ 0.43	✓ 0.79
NorMistral-11B	✓ 0.38	✗ 0.03	✗ 0.17	✓ 0.90	✓ 0.70	✓ 0.63	✓ 0.60	✓ 0.62
NorMistral-7B	✗ 0.00	✗ 0.00	✗ 0.10	✓ 0.40	✓ 0.84	✓ 1.45	✗ 0.89	✗ 0.00



Foundation-measurable (4)

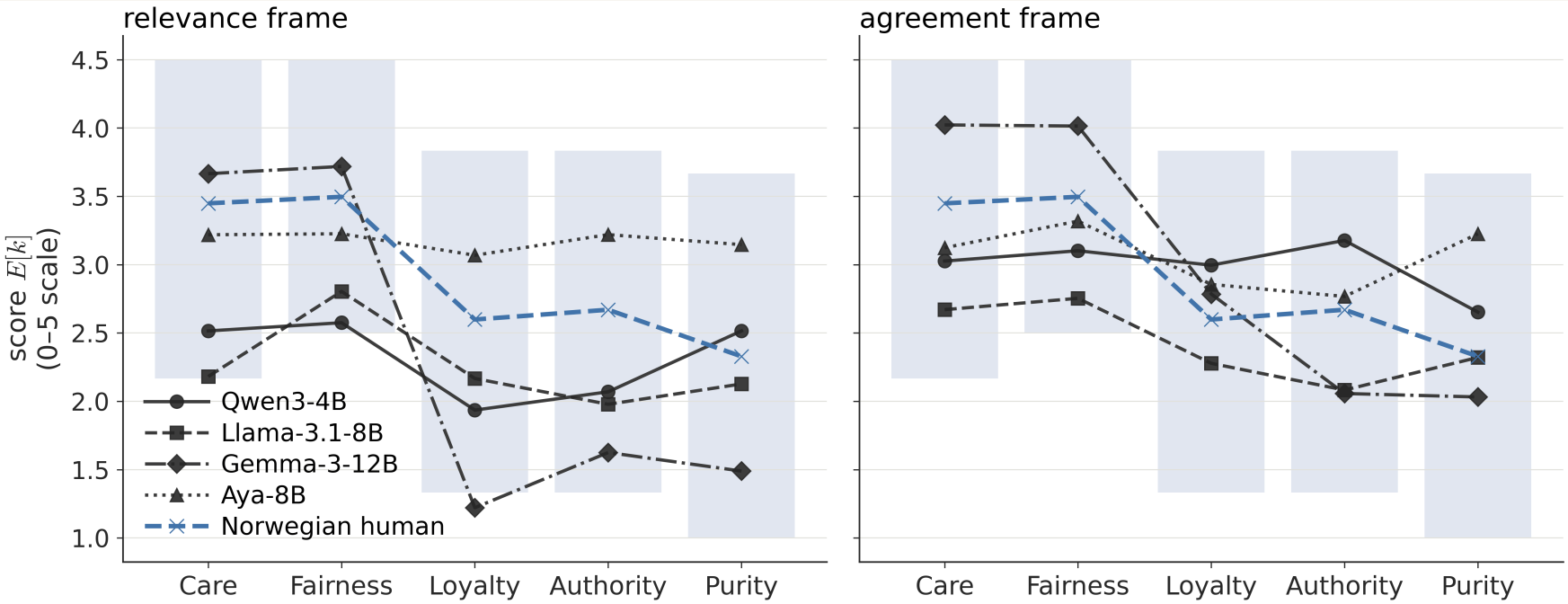
Item-level only (1)

Unmeasurable (1)

The screen sorts the set into **four foundation-measurable**, **one item-level only**, and **one unmeasurable** on this instrument.

FINDING 2 · MODELS' MORAL PROFILES

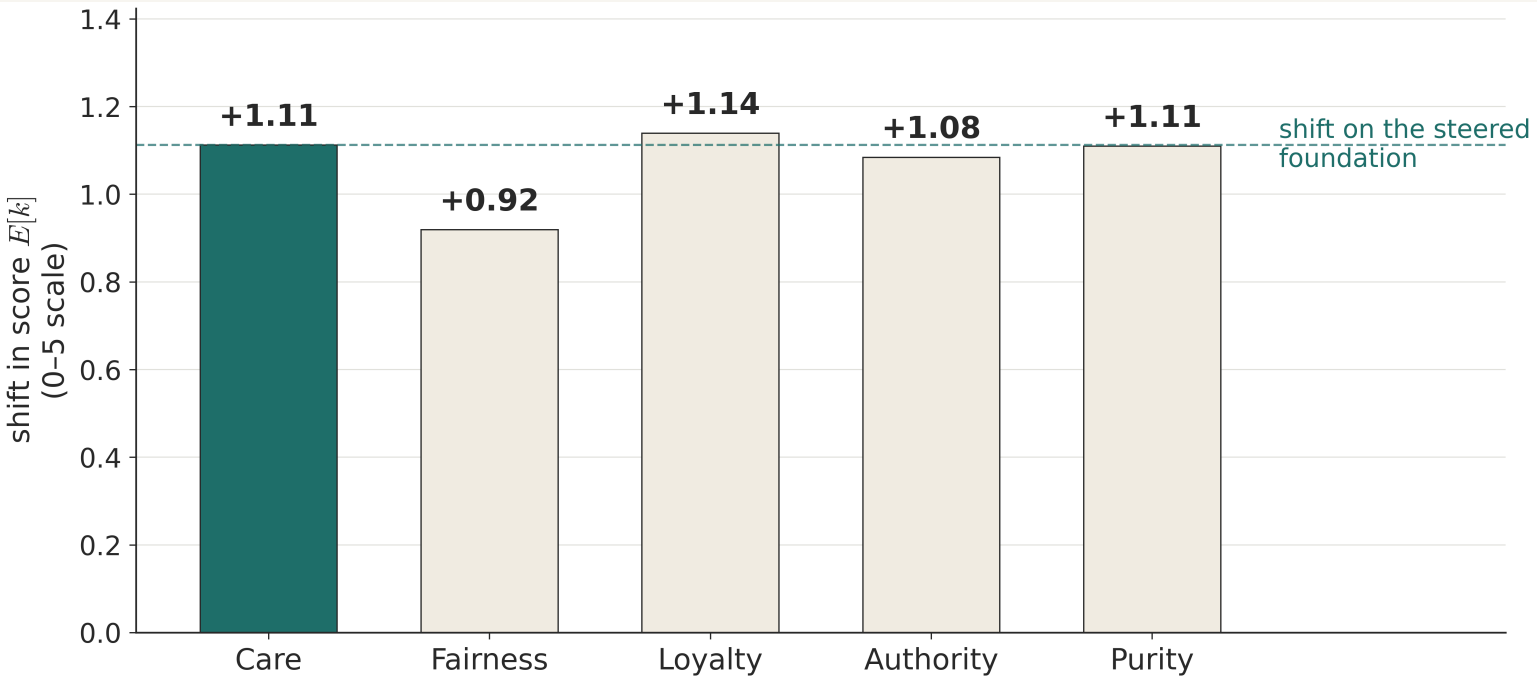
The profile appears, and sharpens



Measurable models share the **WEIRD gradient** — Care and Fairness over the binding three — sharper than the Norwegian human reference.

FINDING 3 · CAN YOU STEER IT?

One volume knob, not five dials



Example: one vector aimed at **Care** in Llama-3.1-8B lifts all five foundations by about the same amount — the off-target four about as much as the target. The steer is one volume knob, not five dials.

TAKEAWAY

Screen before you score

Most cells fail a validity check the default pipeline hides; about **24 prompt evaluations** per item tell a real reading from a **label-prior artefact**.

What passes is then a consistent and **WEIRD-skewed** profile that steers as **one moral volume**, instead of five separable foundations.

LLMs and Moral Foundations

Outstanding paper in this track

Outstanding paper in this track

Steering LLMs' Responses Towards Moral Foundations on the Norwegian MFQ-30

Hans Andersen
haan@uio.no

David Dichas
daviddic@uio.no

Abstract

Recent work applies human psychometric questionnaires to large language models to elicit moral and value profiles, but it is not clear whether these instruments measure anything stable in models or whether the resulting profiles can be moved toward a target human population. We administer the Norwegian Moral Foundations Questionnaire (MFQ-30) to six open-weight LLMs and compare their foundation profiles to a sample of N=1,282 Norwegian respondents. We test two steering interventions, prompt-level persona steering and activation-level ActAdd. Half the models engage with the questionnaire under our attention check. The other half default to flat or central-tendency outputs that look near-human on average without tracking item content. A neutral Nordic-respondent persona, written without any distributional information from the human sample, brings the engaging models 44–77% closer to the Norwegian mean in Mahalanobis d^2. One-pair ActAdd at a fixed mid-layer flattens the foundation profile rather than steering

humans we use Moral Foundations Theory (Graham et al., 2013, 2011), which groups human moral judgement into five intuitive foundations, across two clusters. The *individualizing* cluster contains care and fairness, the *binding* cluster contains loyalty, authority and purity. A respondent's profile is the per-foundation score pattern across the five foundations. We measure these profiles with the Moral Foundations Questionnaire (MFQ; Graham et al., 2011), specifically the Norwegian MFQ-30 variant described in §3.

In this work we approach these questions in a Norwegian setting (§3). Our work consists of three phases. In Phase 1 we built a pipeline to make LLMs answer the questionnaire and produced a moral baseline per model. In Phase 2 we steered the models with prompt steering, which prepends a persona description to the system prompt to alter behaviour. In Phase 3 we steered them with activation steering (ActAdd; Turner et al., 2023), which injects a contrastive steering vector at a chosen transformer layer and nudges the model from

Track Overview: English–Sámi Machine Translation

Main info

Track chairs: Yves Scherrer and David Samuel

Learn more:

`https://github.uio.no/in5550/2026/tree/main/exam/saami_mt`

Track Overview: English–Sámi Machine Translation

WMT is the yearly Conference on Machine Translation and is known for their shared tasks on translation.

Track Overview: English–Sámi Machine Translation

WMT is the yearly Conference on Machine Translation and is known for their shared tasks on translation.

- Each year, WMT shared tasks focus on new language pairs
- In 2026 (and for the first time in its history), an official language of Norway is included: English $\rightarrow$ Northern Sámi
- `https://www2.statmt.org/wmt26/translation-task.html`

Track Overview: English–Sámi Machine Translation

WMT is the yearly Conference on Machine Translation and is known for their shared tasks on translation.

- Each year, WMT shared tasks focus on new language pairs
- In 2026 (and for the first time in its history), an official language of Norway is included: English → Northern Sámi
- `https://www2.statmt.org/wmt26/translation-task.html`

Your task: develop a machine translation system that translates **from English to Northern Sámi**, following the guidelines and requirements of WMT.

Track Overview: English–Sámi Machine Translation

WMT is the yearly Conference on Machine Translation and is known for their shared tasks on translation.

- Each year, WMT shared tasks focus on new language pairs
- In 2026 (and for the first time in its history), an official language of Norway is included: English → Northern Sámi
- `https://www2.statmt.org/wmt26/translation-task.html`

Your task: develop a machine translation system that translates **from English to Northern Sámi**, following the guidelines and requirements of WMT.

- Knowledge of (Northern) Sámi is an advantage, but by no means a requirement!

Track Overview: English–Sámi Machine Translation

Northern Sámi is a very low-resource language.

- There are very few available datasets for training, and they are small and noisy.
- You may need to spend considerable efforts on **curating and augmenting training data**.

Track Overview: English–Sámi Machine Translation

Northern Sámi is a very low-resource language.

- There are very few available datasets for training, and they are small and noisy.
- You may need to spend considerable efforts on **curating and augmenting training data**.

Most pre-trained models don't 'know' Northern Sámi.

- But new language pairs can be added by fine-tuning.
- You will experiment with different base models and adaptation strategies.
- You can use task-specific encoder-decoder models or general-purpose decoder-only LLMs.

Track Overview: English–Sámi Machine Translation

Northern Sámi is a very low-resource language.

- There are very few available datasets for training, and they are small and noisy.
- You may need to spend considerable efforts on **curating and augmenting training data**.

Most pre-trained models don't 'know' Northern Sámi.

- But new language pairs can be added by fine-tuning.
- You will experiment with different base models and adaptation strategies.
- You can use task-specific encoder-decoder models or general-purpose decoder-only LLMs.

WMT has chosen the harder translation direction.

- Translating from Northern Sámi to English is presumably easier, but this is not the goal of the task.

UNIVERSITY
OF OSLO

SáMistral

Improving machine translation from English to Northern Sámi for multilingual Mistral models

Peder M. Førland, Sondre S. Talleraas,
Truls de Lange
{pedermf,sondrta,trulsde}@uio.no

31st May 2026



Data collection

- Pivoting Norwegian-Sámi sentence pairs to English-Sámi
- Backtranslation of Sámi data to English
- Transfer-learning from similar languages
- Combination of datasets:
  - Glot500
  - WMT26
  - Giellatekno
  - Bible

Baseline

		BOUQuET				Tatoeba	
		Val		Test		Test	
Model	# shots	BLEU	chrF	BLEU	chrF	BLEU	chrF
NorMistral-11B-Long	0	14.91	39.16	15.11	39.34	45.69	61.34
NorMistral-11B-Long	3	12.21	36.32	13.78	36.87	30.06	55.36
NorwAI-Mistral-7B	0	10.24	32.80	11.35	34.60	22.22	43.66
NorwAI-Mistral-7B	3	8.95	32.93	10.94	34.37	24.95	49.54

Results

240k = Base data + Backtranslation of Glot500
271k = Base data + Backtranslation of Glot500 + Bible
298k = Base data + Backtranslation of Glot500 + Multilingual Bible

NorMistral-11B-Long		BOUQuET				Tatoeba	
		Val		Test		Test	
Dataset	# shots	BLEU	chrF	BLEU	chrF	BLEU	chrF
71k	0	13.81	36.90	14.90	37.62	52.85	67.59
71k	3	10.28	33.07	11.34	33.87	26.68	56.17
240k	0	13.24	36.49	14.94	38.41	51.01	66.95
240k	3	10.96	32.33	11.53	33.61	30.71	55.62
271k	0	14.52	37.56	15.94	39.64	49.04	65.09
271k	3	11.41	32.57	12.34	33.41	20.65	50.30
298k	0	13.44	36.68	14.96	38.77	54.07	68.35
298k	3	12.02	33.35	12.68	34.32	33.09	56.70
NorwAI-Mistral-7B		BOUQuET				Tatoeba	
		Val		Test		Test	
71k	0	11.69	34.76	13.19	36.64	28.89	52.12
71k	3	10.31	33.01	11.27	34.32	30.38	52.37
240k	0	13.05	37.51	14.97	39.15	30.45	53.46
240k	3	10.91	34.83	12.42	35.97	36.06	57.06
271k	0	13.11	37.28	15.28	39.19	25.95	53.41
271k	3	8.48	32.44	10.47	33.41	29.43	52.36
298k	0	12.68	36.44	14.48	38.58	31.56	53.75
298k	3	9.61	33.43	12.42	35.77	30.47	53.60

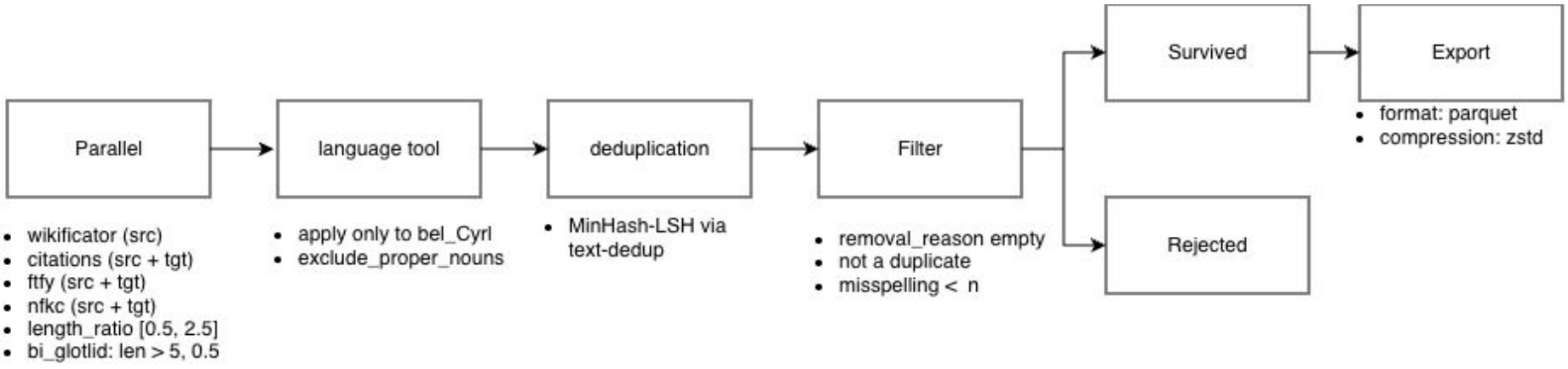
ReMova: Belarusian data-cleaning pipeline

Mikita Pilinka - mikitap@uio.no

Why Belarusian is hard

- Vulnerable language (according to UNESCO)
- ~6,984,000 speakers, low web presence
- **Two orthographies** (Official vs Taraškievica) which is ignored by most of the cleaners
- Available parallel data (OPUS, HPLT) is noisy: cross-language contamination, misalignments, mixed orthographies
- Off-the-shelf MT systems perform poorly out of the box

Remova pipeline



- 3 stages: normalise, annotate, filter
- Wikificator and LanguageTool: belarusian-tailored tools
- Filtered out over 2000 sentence pairs from Tatoeba (crowdsourced translation)

Results

- LoRa for Gemma-3
- Full fine-tuning of NLLB
- 6635 (tatoeba) + 9142 (wikimedia) sentence pairs

		BLEU		chrF++		COMET	
Domain	n	base	+FT	base	+FT	base	+FT
Gemma-3-4b-pt							
Comments	73	24.18	**25.94**	44.81	**47.63**	81.06	**82.47**
Conversation	186	24.19	**25.57**	43.99	**46.03**	83.10	**85.40**
How-to/instr.	107	21.23	21.16	46.62	**46.83**	81.80	**82.77**
Narration	98	21.59	**23.60**	46.87	**49.46**	82.47	**84.56**
Other misc.	95	24.33	24.31	48.60	47.89	86.68	**86.85**
Reflection	88	21.10	19.37	46.95	46.64	86.32	85.36
Social posts	104	24.43	22.42	46.51	45.98	86.32	**87.06**
Web misc.	103	20.01	19.63	46.38	46.07	88.59	**88.83**
Overall	854	22.89	**22.93**	46.37	**47.00**	84.47	**85.50**
NLLB-200-3.3B							
Comments	73	25.22	**32.49**	48.17	**53.40**	83.07	**87.78**
Conversation	186	26.83	**33.96**	49.43	**54.30**	85.87	**88.98**
How-to/instr.	107	25.77	**27.93**	50.65	**53.25**	84.10	**85.63**
Narration	98	27.41	**33.52**	51.68	**56.89**	85.33	**88.55**
Other misc.	95	28.98	**31.28**	53.46	**54.51**	88.99	**90.46**
Reflection	88	25.95	**27.44**	52.39	**53.20**	89.09	**89.96**
Social posts	104	30.35	**32.92**	52.87	**54.61**	88.39	**90.33**
Web misc.	103	24.20	**27.68**	51.17	**54.26**	88.65	**90.72**
Overall	854	27.24	**31.11**	51.35	**54.36**	86.67	**89.05**

Takeaway + future work

- Data quality matters
- We need more of it.
- Future plans include adding more neural metrics for the data cleaning pipeline and training the models with more data

English–Sámi Machine Translation

Outstanding paper in this track

English–Sámi Machine Translation

Outstanding paper in this track

ReMova: Fine-tuning LLMs for English to Belarusian translation*

Mikita Pilinka
University of Oslo
Department of Informatics
mikitap@uio.no

Abstract

This paper presents a Belarusian-specific data-cleaning pipeline and fine-tuned translation models for the English–Belarusian language pair, covering two model paradigms: encoder-decoder models (NLLB) and decoder-only LLMs (Gemma). Our pipeline distinguishes itself from others by employing community-maintained correction tools that help address the issue of the two orthographies of the Belarusian language. By conducting an evaluation on the BOUQuET benchmark, we found that the majority of off-the-shelf open-weight models perform poorly on Belarusian, while fine-tuning of NLLB provides substantial gains and Low-Rank Adaptation (LoRA) fine-tuning of Gemma provides only modest gains concentrated in conversational content. Our results

use in training the most advanced models (Schaffer et al., 2026). The uneven distribution of languages across the web and the wide variation in available linguistic resources have left many languages, including Belarusian, under-prioritised in the training and evaluation of machine translation systems. We present a Belarusian-specific data-cleaning pipeline and use the resulting corpus to fine-tune translation models from both paradigms: NLLB as an encoder-decoder system and Gemma as a decoder-only LLM.

2 Related Studies

2.1 Multilingual Encoder-Decoder Systems

While the high-resource languages benefit from large parallel corpora suitable for training and eval-

Track Overview: Emotional Valence and Arousal

Main info

Track chairs: Shenbin Qian

Learn more:

`https://github.uio.no/in5550/2026/tree/main/exam/valence`

Track Overview: Emotional Valence and Arousal

Goal

Given a short personal text, predict two emotional scores:

- Valence: how positive/negative (-2 to $+2$)
- Arousal: how energized/calm (0 to 2)

Track Overview: Emotional Valence and Arousal

Goal

Given a short personal text, predict two emotional scores:

- Valence: how positive/negative (-2 to $+2$)
- Arousal: how energized/calm (0 to 2)

Data: Ecological Momentary Assessment (EMA)

- Participants wrote texts over time describing their emotional state
- Two text types: **essays** ($\sim$59 words) and **feeling-word lists** ($\sim$5 words)
- Train: 2,764 texts from 137 authors; Test: 1,737 texts from 91 authors
- Longitudinal: multiple entries per user across up to 7 collection phases

Track Overview: Emotional Valence and Arousal

Goal

Given a short personal text, predict two emotional scores:

- Valence: how positive/negative (-2 to $+2$)
- Arousal: how energized/calm (0 to 2)

Data: Ecological Momentary Assessment (EMA)

- Participants wrote texts over time describing their emotional state
- Two text types: **essays** ($\sim$59 words) and **feeling-word lists** ($\sim$5 words)
- Train: 2,764 texts from 137 authors; Test: 1,737 texts from 91 authors
- Longitudinal: multiple entries per user across up to 7 collection phases

What makes this different from regular sentiment analysis?

- Two independent dimensions, not just positive/negative
- Evaluation rewards tracking within-person variation over time

Track Overview: Emotional Valence and Arousal

Three correlation metrics (computed for valence and arousal separately)

- Within-person (r_{within}): average of per-user Pearson r
  - How well do you track one individual's emotional fluctuations?
- Between-person (r_{between}): Pearson r across user-level means
  - Can you distinguish generally happier users from sadder ones?
- Composite ($r_{\text{composite}}$): Fisher's z-transform combination
  - $r_{\text{composite}} = \tanh\left(\frac{\operatorname{arctanh}(r_{\text{within}}) + \operatorname{arctanh}(r_{\text{between}})}{2}\right)$
  - **Primary ranking metric**

Origin

Based on SemEval-2026 Task 2 (Subtask 1). Dataset and evaluation scripts provided.

WNNLP '26 · TRACK: emo

Predicting Emotional Valence & Arousal Over Time

A hybrid approach combining lexical & contextual representations

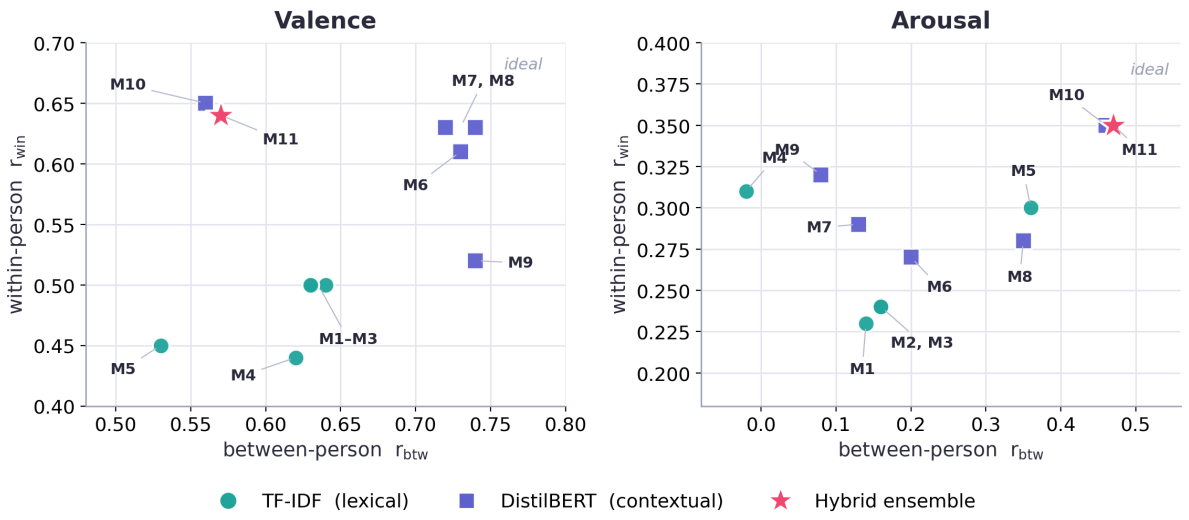
Two emotional signals. Two model families. ***One ensemble that balances both.***



Noor Fatima · **Andre Maharaj Gregussen** · **Mykhailo Karpiuk**
University of Oslo

A structural trade-off — and how we resolve it

Controlled 11-config ablation: **lexical** TF-IDF vs. **contextual** DistilBERT, features added incrementally, then a weighted hybrid ensemble.



Within- vs. between-person Pearson r across all 11 configurations (test set). Ideal = top-right corner.

DistilBERT tracks within-person fluctuations best — top of each plot.

TF-IDF stays competitive on between-person tendencies.

Hybrid ★ recovers both — top-right on arousal, competitive on valence.

Results: the ensemble wins the balance

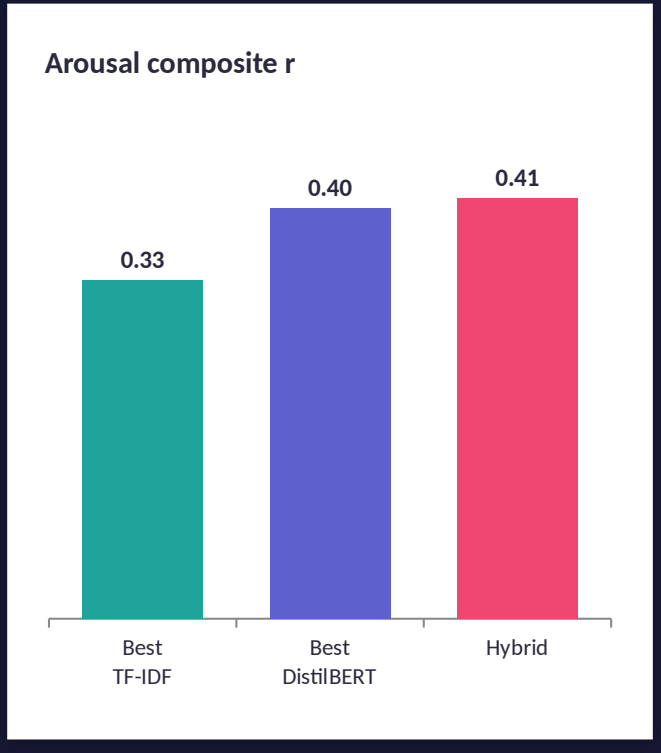
Composite Pearson r — the primary ranking metric (Fisher-z blend of within- & between-person r).

0.61

VALENCE

0.41

AROUSAL



Best arousal of all 11 configurations — beats every single lexical and contextual model; valence remains competitive.

Lexical + contextual representations together balance moment-to-moment emotional change with stable individual differences.

Modelling Emotional Valence and Arousal Over Time: An Ensemble of RoBERTa-Large Fine-Tunes

Baseline

- **Frozen RoBERTa-large**: Few-shot Retrieval
- Embed training data and predict value of unobserved data using knn with 32 neighbors

Reference Model

- **Fine-tuned RoBERTa-large**
- Parameters: learning rate → $2 \cdot 10^{-5}$, max length → 256

Stability Analysis

- **Multi-seed Analysis**
- Train the same model with different initial seeds.
- Note differences due to randomness in the initial point.

Grid Search

- **Different combinations of parameters**
- Learning rate → $1, 2, 3 \times 10^{-5}$, Max length → $256, 384$

MultiTask Learning

Auxiliary head to take advantage of data properties.

Adapt the loss function to calculate correlation between the 2 predicted classes

Temporal and By-User Embedding

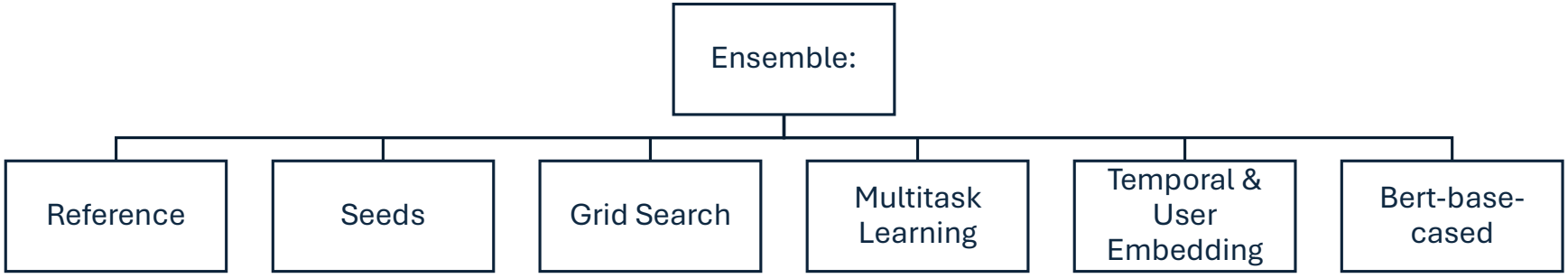
Integrate embedding per user: 32 dimensions

Sinusoidal time encoding

BERT-base-cased

Using a smaller model to compare whether using RoBERTa could be overly complex

Josep Peiró Ramos



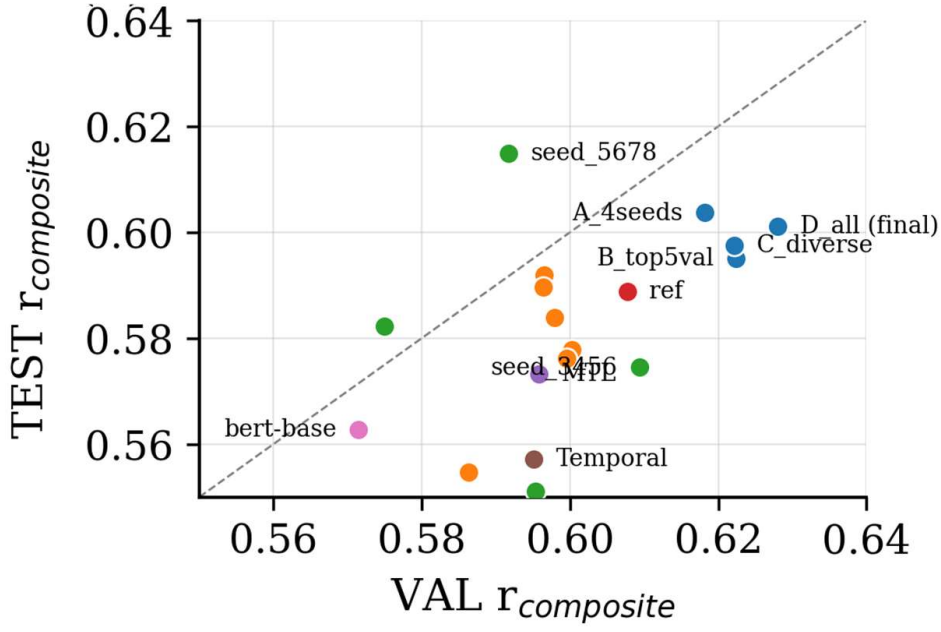
Ensemble Results in test:

- **Arousal: 0.525**
- **Joint: 0.601**
- **Valence: 0.677**

Validation–test gap: below **0.03**

- Ensemble reduces initialization noise

Data is the bottleneck



Predicting Emotional Valence and Arousal from Self-Reported Texts

Research Questions

RQ1: How does a joint multi-output BERT model compare with separate single-task baseline models in predicting valence and arousal?

RQ2: How does training on essay-only data compare with training on the combined essay-and-word data setting?

pretrained encoder: **bert-base-cased**

Chunru Long

Validation results

Full-input setting

Table 3: Validation results in the full-input setting. ST denotes the single-task baseline, and Joint denotes the joint multi-output model. Bold correlation values indicate statistical significance at $p < .05$.

	Valence		Arousal	
Metric	ST	Joint	ST	Joint
r_{within}	**0.451**	**0.485**	0.506	0.560
r_{between}	**0.774**	**0.788**	**0.550**	**0.571**
$r_{\text{composite}}$	0.640	0.662	0.528	0.566
$\text{MAE}_{\text{within}}$	0.726	0.710	0.482	0.495
$\text{MAE}_{\text{between}}$	0.550	0.543	0.378	0.390
$\text{MAE}_{\text{composite}}$	0.647	0.634	0.431	0.444

the two affective dimensions share useful textual information, so a shared encoder can help model them together

Essay-only setting

Table 4: Validation results in the essay-only setting. ST denotes the single-task model, and Joint denotes the joint multi-output model. All models are trained on essay-only data and evaluated on the full validation split. Bold correlation values indicate statistical significance at $p < .05$.

	Valence		Arousal	
Metric	ST	Joint	ST	Joint
r_{within}	**0.532**	**0.545**	**0.348**	0.414
r_{between}	**0.743**	**0.783**	**0.421**	**0.393**
$r_{\text{composite}}$	0.650	0.682	0.385	0.404
$\text{MAE}_{\text{within}}$	0.766	0.744	0.522	0.526
$\text{MAE}_{\text{between}}$	0.591	0.557	0.416	0.418
$\text{MAE}_{\text{composite}}$	0.688	0.661	0.470	0.473

Diagnostic and final test results

Normalization makes the two losses more comparable during training.

Essay-only validation diagnostic

Table 5: Diagnostic comparison of full-validation and essay-only validation results. ST denotes the single-task model, Joint denotes the joint multi-output model, and r_c denotes $r_{\text{composite}}$.

Model	Training data	Evaluation data	Valence r_c	Arousal r_c
Joint	Full input	Full validation	0.662	0.566
Joint	Essay-only	Full validation	0.682	0.404
Joint	Essay-only	Essay validation only	0.729	0.532
ST	Full input	Full validation	0.640	0.528
ST	Essay-only	Full validation	0.650	0.385
ST	Essay-only	Essay validation only	0.713	0.481

arousal was still weaker than in the full-input setting

feeling-words provide useful cues for arousal prediction rather than noise.

Final test results with target normalization

Table 7: Final test results of the selected multi-output BERT model with and without target normalization. The model is trained on the full training set using learning rate 1.5×10^{-5}, batch size 4, and 4 epochs. Bold correlation values indicate statistical significance at $p < .05$.

	Valence		Arousal	
Metric	No norm.	Norm.	No norm.	Norm.
r_{within}	**0.542**	**0.554**	**0.417**	**0.387**
r_{between}	**0.762**	**0.776**	**0.534**	**0.587**
$r_{\text{composite}}$	0.666	0.681	0.478	0.494
$\text{MAE}_{\text{within}}$	0.785	0.775	0.518	0.521
$\text{MAE}_{\text{between}}$	0.415	0.413	0.252	0.248
$\text{MAE}_{\text{composite}}$	0.635	0.627	0.393	0.393

UNIVERSITY OF OSLO

Time, Text and Temperament
Temporal and data-augmented ablation for predicting a 2D emotional state

Jón Arnar Árnarson, Ørjan Oftedal Hanasand & Asgeir Kråkenes
{jonaarna,orjanoh,arkraake}@uio.no
Dept. of Informatics and Mathematics, UiO

EMO track

31st May 2026

Time, Text and Temperament:
A temporal and data-augmented ablation study for predicting a two-dimensional emotional state

Jón Arnar Árnarson **Ørjan Oftedal Hanasand** **Asgeir Kråkenes** *
Department of Informatics, Department of Mathematics
University of Oslo
{jonaarna, orjanoh, arkraake}@uio.no

Abstract

This paper investigates different approaches to model and jointly predict two independent affective dimensions: valence and arousal. We explore methods of integrating temporal and user-specific context, in addition to data-augmentation, to achieve more nuanced emotional modelling than is possible with traditional sentiment analysis. We conducted an ablation study to assess the impact of two distinct approaches: supplementing sentence embeddings with temporal features and augmenting the training data with an LLM. Our findings confirm that the use of certain hand-crafted features can improve the performance of a our architecture on a multitask prediction problem, but also that augmenting the training set with an LLM can further improve performance. However, our experiments with enriching the model with temporal features yielded less promising results, especially in combination with data augmentation.

1 Introduction

When it comes to sentiment analysis, a review by Yadav and Vishwakarma (2020) elucidates how datasets are mainly comprised of user scores, tweets, messages, comments, blogs and reviews. Popular datasets such as *Stanford large movie review*, for instance, contain binary labelled movie reviews from IMDb (Yadav and Vishwakarma, 2020). It thus follows that studies on sentiment analysis build and test models fitting such data. This motivates testing models for ordinal emotion score prediction on the dataset of SemEval (2026c), which contains longitudinal texts with self-annotated valence and arousal scores across two dimensions. In this paper, we build a model for predicting valence and arousal scores based on an input text, which is made possible through the dataset covered in section 2. Our study is constructed as an ablation study, varying **A)** temporal features and **B)** the training dataset used to train our model, enabled by synthetic data augmentation.

*Code available at https://github.uio.no/jonaarna/in5550_gruppe/tree/main/exam

2 Data

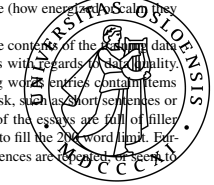
The data for this study originates in a SemEval shared task (SemEval, 2026d). The training dataset consists of 2764 longitudinal texts from 137 users collected over several years, and the testing dataset consists of 1737 longitudinal texts from 91 authors. During hyper-parameter tuning, the training dataset was split 85/15 into a training and validation set. Each instance in these datasets contains the following fields:

- **user_id**: anonymous identifier for author
- **text_id**: identifier for each text instance
- **text**: essay or feeling words entry
- **timestamp**: when the text was written
- **collection_phase**: data collection phase (1–7)
- **is_words**: boolean; `False` for essays, `True` for feeling words
- **valence**: valence score (-2 to $+2$)
- **arousal**: arousal score (0 to 2)

Such split was performed on unique users

Each instance contains a text, which is in the form of an essay *"I've been feeling just fine ..."* or a feeling words list *"Sad, Depressed, Lonely, Broke, Tired"*, flagged by the boolean **is_words**. Each instance is accompanied by a self-annotated **valence score** (how positive or negative they feel), and an **arousal** score (how energized/calm they feel).

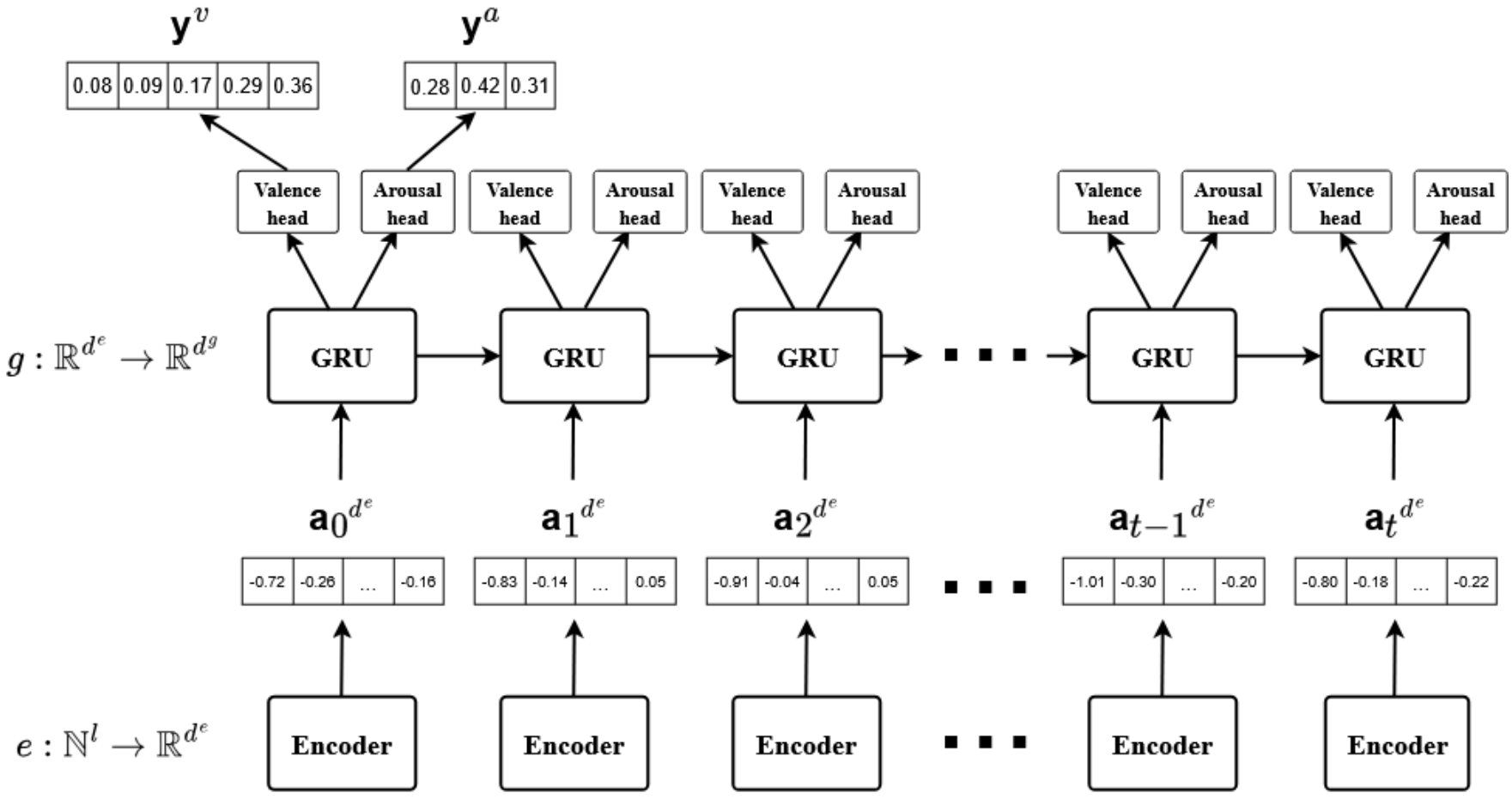
Exploration of the contents of the training data showed some issues with regards to data quality. We note that feeling words entries contain items that do not fit the task, such as short sentences or statements. Some of the essays are full of filler words or characters to fill the 200 word limit. Furthermore, some sentences are repeated, or seem to



Our study in a minute

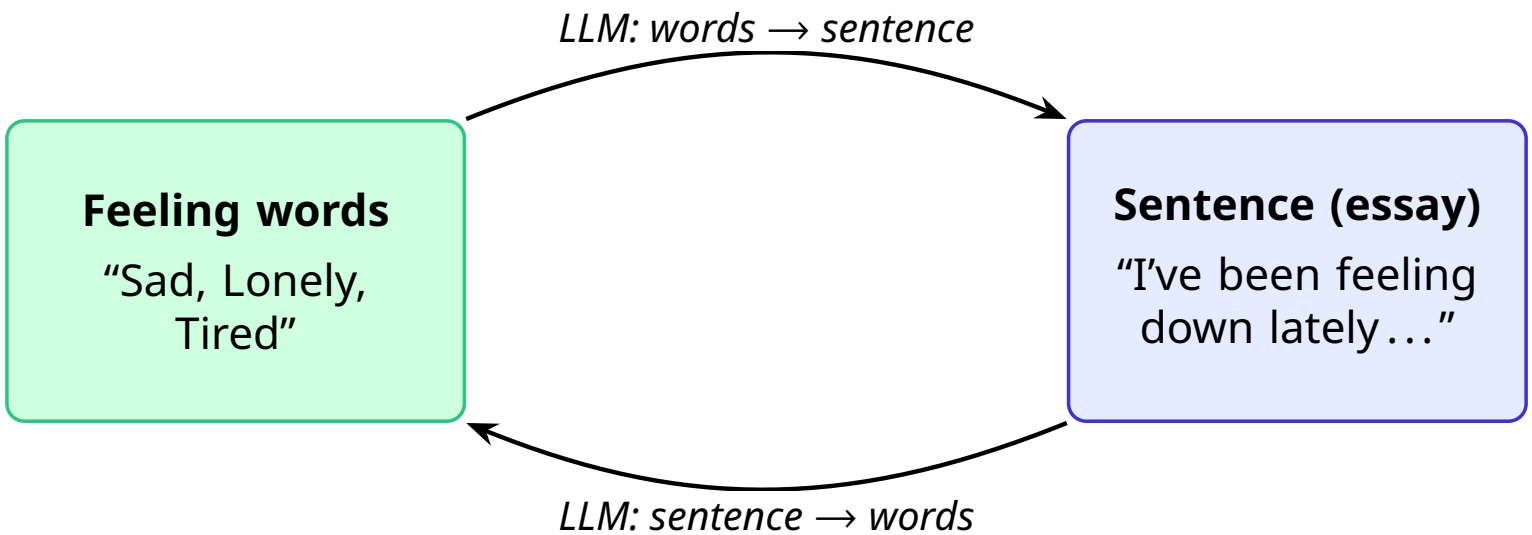
- Make embeddings of texts and word lists (`all-MiniLM-L6-v2` [2]) → predict supervised valence / arousal scores.
- Compare this baseline with:
  - Prediction with temporal information:
    - Process entries sequentially with a GRU unit
    - Hand-crafted feature vectors
  - Augmented training data:
    - Create more training data by converting sentences to words and vice versa
- Assess all results through strict ablation to attribute effects/contributions

Temporal-architectural overview



Making use of the temporal information available to us

Data augmentation: words ↔ sentences



Each real entry is paired with an LLM-generated (`Llama-3.1-8B-Instruct` [1]) version in the *other* format — adding training examples for "free".

What we found

- Fully temporal feature helped partially; no clear prediction gain overall
- Data augmentation could help (essays → words), but possibly because it reduces noise from texts words can better describe
- Valence was more predictable than Arousal
- Between-user was more predictable than that of within-user; difficult to track single user evolvement
- Future work should explore non-RNN approaches and even more granular ablation

References I

[1] Meta. *Llama-3.1-8B-Instruct Model Card*. https://huggingface.co/meta-llama/Llama-3.1-8B. Accessed: 19 May 2026. 2024.

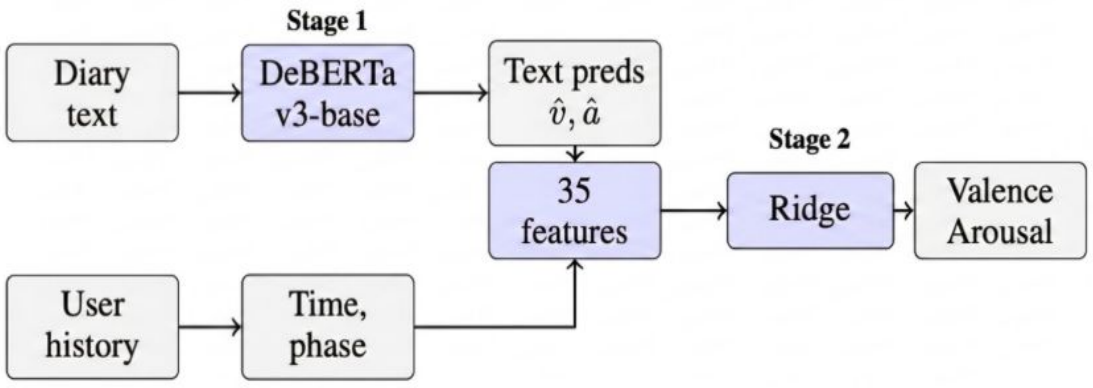
[2] Nils Reimers and Iryna Gurevych. “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 3982–3992.

Predicting Emotional Valence and Arousal Over Time

An exploratory analysis of text encoders and history-augmented Ridge calibration

Adam Jakobsen

METHOD



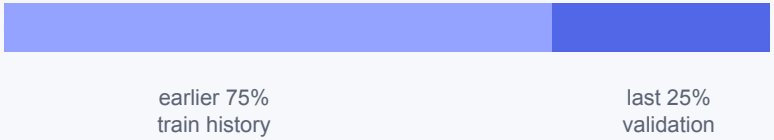
Feature groups:

- Metadata
- History presence
- Most recent targets
- Long-statistics
- Windowed recency
- Trends

Why Ridge?

- Data scale: avg. 20 entries per user
- Ridge is data-efficient
- Explainable
- Very fast in closed form

Validation split

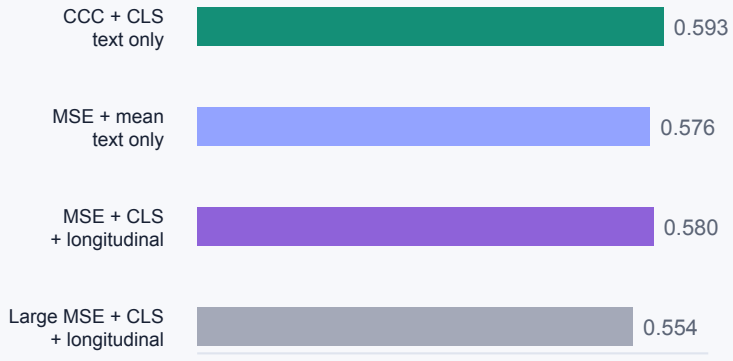


+ 20% users held out entirely

Out-of-fold text predictions for Ridge training

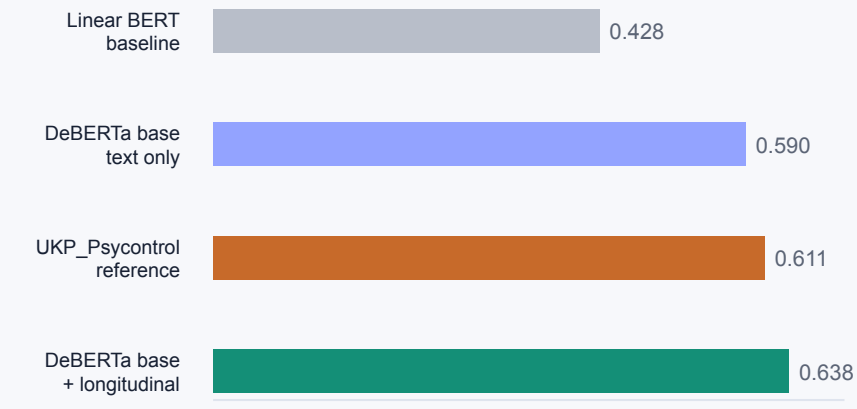
3-fold GroupKFold by user → Longitudinal layer trained on unseen folds

Validation mean r_composite



Hidden-test choice would be: DeBERTa-v3-base, CCC loss, CLS pooling, text-only.

Mean test composite



+0.048
vs. MSE+CLS text-only

0.638
best observed test score

+0.027
vs. reference system

Valence: 0.701 **Arousal: 0.574**

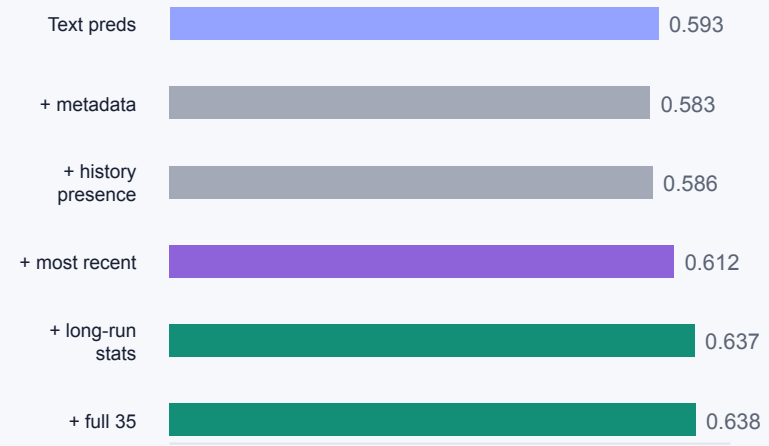
Important caveat

This is after-the-fact analysis on a labelled test set. It demonstrates the value of history tracking, but it is not comparable to a strict hidden-test-leaderboard.

User-group effect

Group	Text	+ Longit.	Δ
All	0.590	0.638	+0.048
Seen users	0.553	0.660	+0.107
Unseen users	0.613	0.606	-0.007

Cumulative feature ablation



Takeaway: long-run personal affect means and most-recent labels explain almost all of the gain. Windows and trends add only +0.001.

Emotional Valence and Arousal

Outstanding paper in this track

Emotional Valence and Arousal

Outstanding paper in this track

Predicting Emotional Valence and Arousal Over Time: An Exploratory Analysis of Text Encoders and History-Augmented Ridge Calibration

Adam Jakobsen
adamj@simula.no

Abstract

We present an exploratory study of emotional valence and arousal prediction from longitudinal ecological momentary assessment (EMA) diary texts, evaluated on the labelled SemEval-2026 Subtask 1 test set. Because gold labels are available for the test set, we distinguish between validation-based model selection and after-the-fact test-set analysis. On the temporal validation split, the best validation model is a DeBERTa-v3-base text-only system trained with CCC loss and CLS pooling (r=0.593). However, after-the-fact analysis on the labelled test set shows that adding explicit user-history features through a Ridge calibration layer strongly improves the MSE+CLS DeBERTa-v3-base model, increasing mean composite correlation from 0.590 to 0.638. While this best test score is higher than the strongest published result known to us, UKP_Psycontrol (r=0.611), we emphasize that the post-hoc analysis serves as an exploratory proof-of-concept rather than a di-

ral structure. SemEval-2026 Task 2 Subtask 1 makes the structure explicit through its evaluation metric, which rewards both tracking within-person emotional changes and recovering between-person baseline differences.

In this study, gold labels are available for the test set. We therefore separate two kinds of conclusions throughout the paper. The first kind is validation-based conclusions, which approximate what would be available under a strict hidden-test competition protocol. The second kind is after-the-fact test-set conclusions, which use the labelled test set for further analysis. This distinction is important because the temporal validation split and the labelled test set do not select the same best-performing model.

We study a two-stage pipeline (Figure 1). Stage 1 fine-tunes DeBERTa-v3-base to predict valence and arousal from text content alone. Stage 2 combines these predictions with 35 features derived from user history and temporal context in a closed-form Ridge regression. The Ridge layer adds explicit

Track Overview: LLM Reasoning

Track chair: Meihan Tong

Goal

- Predict whether a syllogistic argument is formally valid.
- The key challenge is to ignore content plausibility.
- A model should accept valid arguments even when they sound implausible, and reject invalid arguments even when they sound believable.

Example

All dogs are fish. All fish are mammals. Therefore, all dogs are mammals.

This sounds implausible, but it is logically valid.

Reasoning track submissions

- 5 submitted papers, covering 8 students.
- All papers focused on SemEval-2026 Task 11 Subtask 1: English syllogistic validity prediction.
- Main methodological families:
  1. LoRA fine-tuning for content-independent validity prediction.
  2. Abstraction / neutralization to remove real-world content before reasoning.
  3. Neuro-symbolic solving with deterministic validation after parsing.
  4. Robustness analysis under paraphrases, negation, and order changes.
  5. Layer-localized adaptation to study where reasoning behavior emerges.

Takeaways from the reasoning track

- The strongest systems do not simply ask the LLM to 'reason' directly; they often separate structure extraction from validity judgement.
- LoRA is very effective on the official dataset, but several papers show that it may still rely on surface patterns, such as premise order or familiar wording.
- Robust reasoning requires more than high accuracy: we need tests for content effects, paraphrase stability, negation handling, and cross-dataset transfer.
- Overall, this track shows a clear move from benchmark scoring toward understanding what kind of reasoning the model has actually learned.

Abstract-first prompting complements mismatched LoRA

- The same model judges whether the conclusion follows.
- We use Qwen3-8B with LoRA, evaluated on SemEval and NeuBAROCO.
- Metrics: accuracy and content effect.
- Lower content effect (CE) means the model is less influenced by whether the same logical form is paired with plausible or implausible content.

Original

All cats are mammals. Some pets are cats.

Therefore, some pets are mammals.

↓

rewrite content nouns as T1/T2/T3

↓

Abstracted

All T1 are T2. Some T3 are T1.

Therefore, some T3 are T2.

SemEval to NeuBAROCO: CE drops from 24.99 to 9.56

Condition	Acc	CE
SemEval LoRA	97.38	3.19
+ abstract-first	95.29	3.13
SemEval LoRA on NeuBAROCO	89.27	**24.99**
+ abstract-first	89.77	**9.56**

- The main result is in the transfer setting.
- On SemEval, the LoRA is evaluated on the same distribution
- SemEval-trained LoRA is evaluated on NeuBAROCO, CE is much higher
- Abstract-first reduces CE to 9.56

NeuBAROCO-trained LoRA already has low CE

NeuBAROCO LoRA	Acc	CE
vanilla	98.90	0.94
+ abstract-first	94.48	4.96

- This control checks whether NeuBAROCO always benefits from the rewrite
- Trained a LoRA directly on NeuBAROCO
- Adding abstract-first hurts

Takeaway
Abstract-first helped in the SemEval to NeuBAROCO transfer setting. It did not help when the adapter was trained on the target dataset.

Method

We tested four approaches on Qwen3-8B

- Prompt engineering: zero-shot, instruction only, no examples
- In-context learning: 4 most similar training examples in the prompt
- LoRA fine-tuning: parameter-efficient training on the training set
- Syllogism abstraction: content words replaced with symbols A, B, C to force reasoning on logical form

LoRA Fine-Tuning for Syllogistic Validity Prediction with External Benchmark Evaluation

Lars Fredrik Dramdal Imsland, Emil Sandbæk Skari, Kristoffer Sæth Kvam

Results

System	Accuracy	Content Effect	Combined Score
Prompt Engineering	47.12%	34.38	10.32
ICL top-4	49.21%	48.96	10.02
LoRA (epochs=3, r=16)	97.38%	3.19	40.02
LoRA (epochs=5, r=16)	99.48%	1.06	**57.68**
LoRA (epochs=3, r=64)	99.48%	1.06	**57.68**
LoRA (epochs=3, r=16) + Abstraction, inference only	57.59%	41.67	12.12
LoRA (epochs=5, r=16) + Abstraction, train+inference	89.01%	9.62	26.47

Table 10: Results on the official SemEval-2026 dataset (191 examples). Best combined score in bold.

Generalization and findings

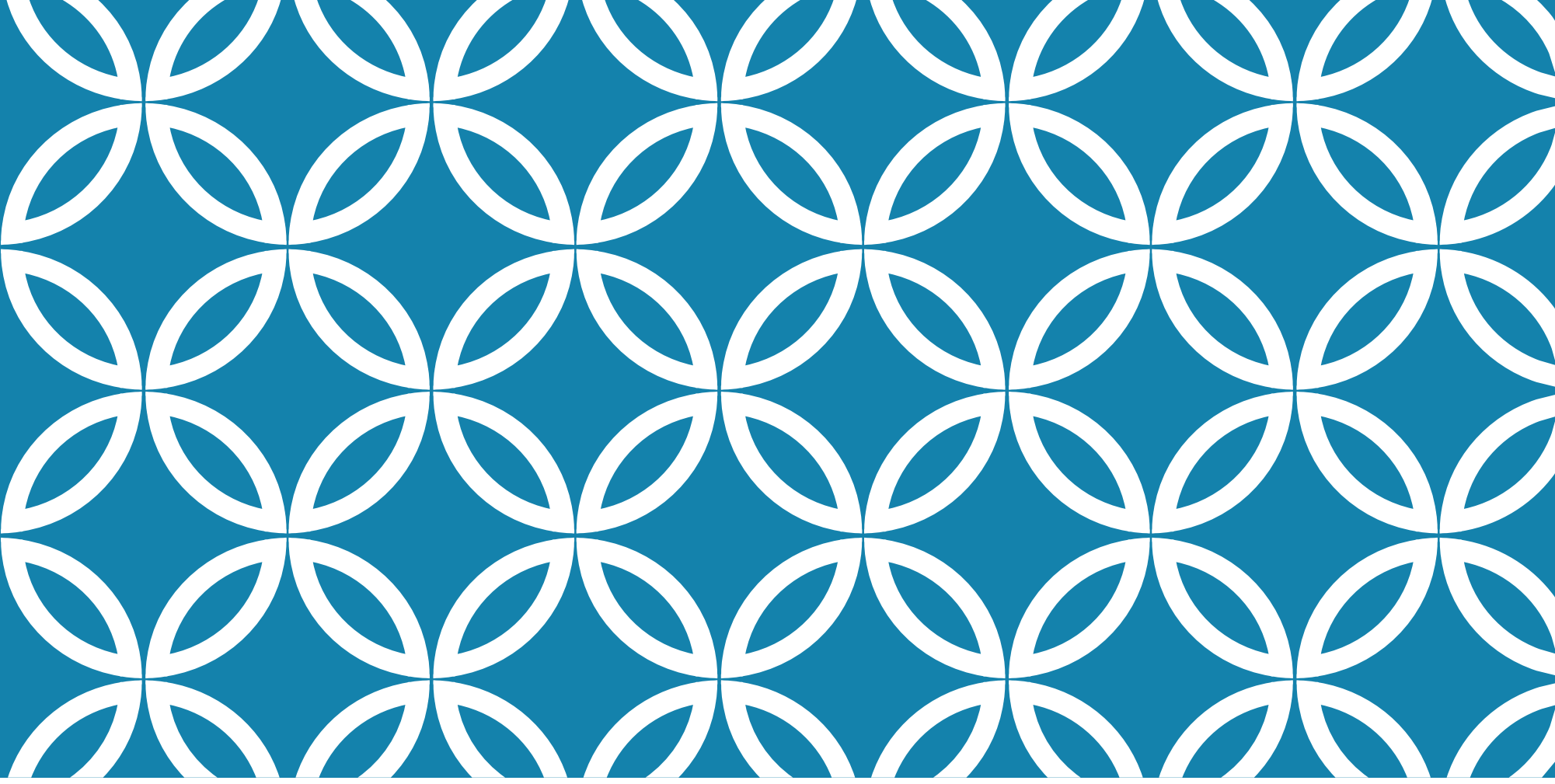
- Evaluated on two external datasets: NeuBAROCO and Logic-in-LLM
- NeuBAROCO - 91.9% accuracy overall. 83.9% on implausible content
- Logic-in-LLM - 72.5% accuracy overall. 62.5% on order switched (22.5% drop)
- Learned from plausibility/positional patterns and not formal reasoning
- Syllogism abstraction: 57.59% without retraining, 89.01% with retraining

Content type	Accuracy	Number
Symbolic (A/B/C)	91.7%	48
Plausible	96.8%	219
Implausible	83.9%	118
Others	86.4%	22
Overall	**91.9%**	**407**

Table 8: Results on NeuBAROCO (407 instances) by content type

Variant	Accuracy	Number
N(Normal words)	85%	40
O (Order switched)	62.5%	40
X (Nonsense words)	77.5%	40
OX (both)	65%	40
Overall	**72.5%**	**160**

Table 9: Logic-in-LLM model accuracy by syllogism variant. Content effect (N vs X): 7.5 pp. Order effect (N vs O): 22.5 pp.



LAYER-LOCALIZED LORA ADAPTATION IN REASONING-ORIENTED FINE-TUNING

Helene Brodin

RESEARCH QUESTION:

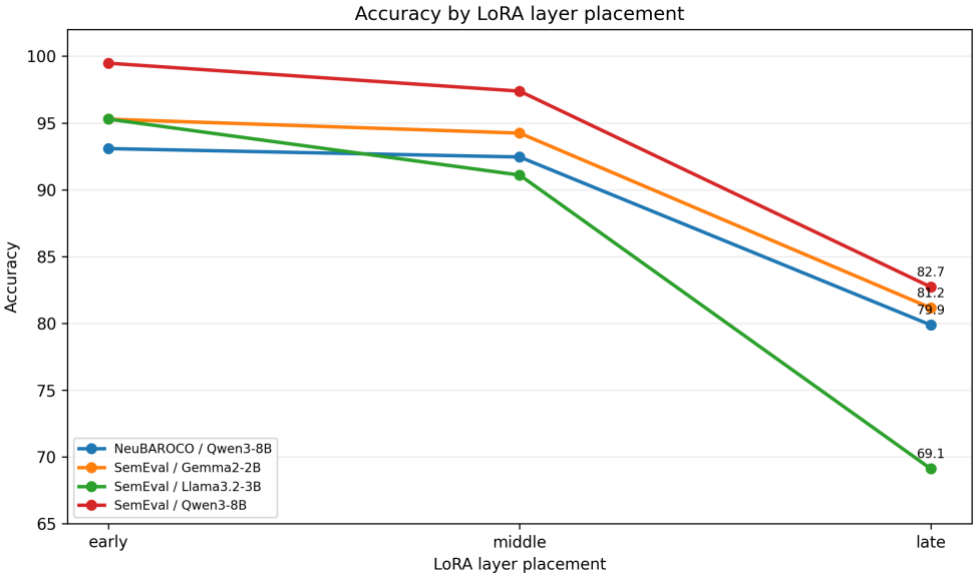
Where does adaptation occur during LoRA fine-tuning for reasoning-oriented tasks?

Models: Qwen3-8B, Gemma2-2B, Llama3.2-3B
Datasets: SemEval-2026 Task 11 and NeuBAROCO
Metrics: Accuracy, Content Effect, LoRA norms

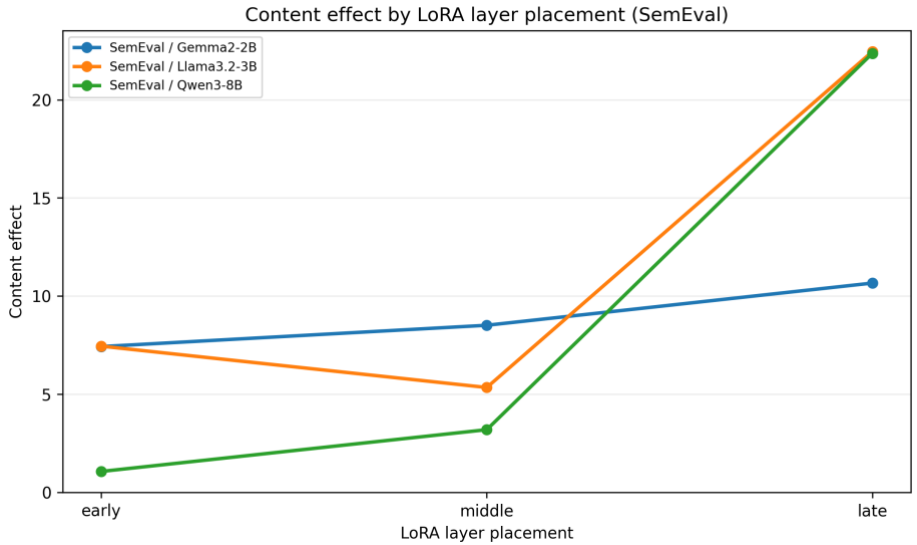
Syllogism	Valid	Plausible
All dogs are animals. All animals are living beings. Therefore, all dogs are living beings.	Yes	Yes
All dogs are fish. All fish are mammals. Therefore, all dogs are mammals.	Yes	No
All birds can fly. Penguins are birds. Therefore, penguins can fly.	No	Yes
All stones are ideas. Some ideas are bicycles. Therefore, all bicycles are stones.	No	No

Table 1: Examples of the four validity-plausibility categories used in reasoning-oriented tasks.

ACCURACY

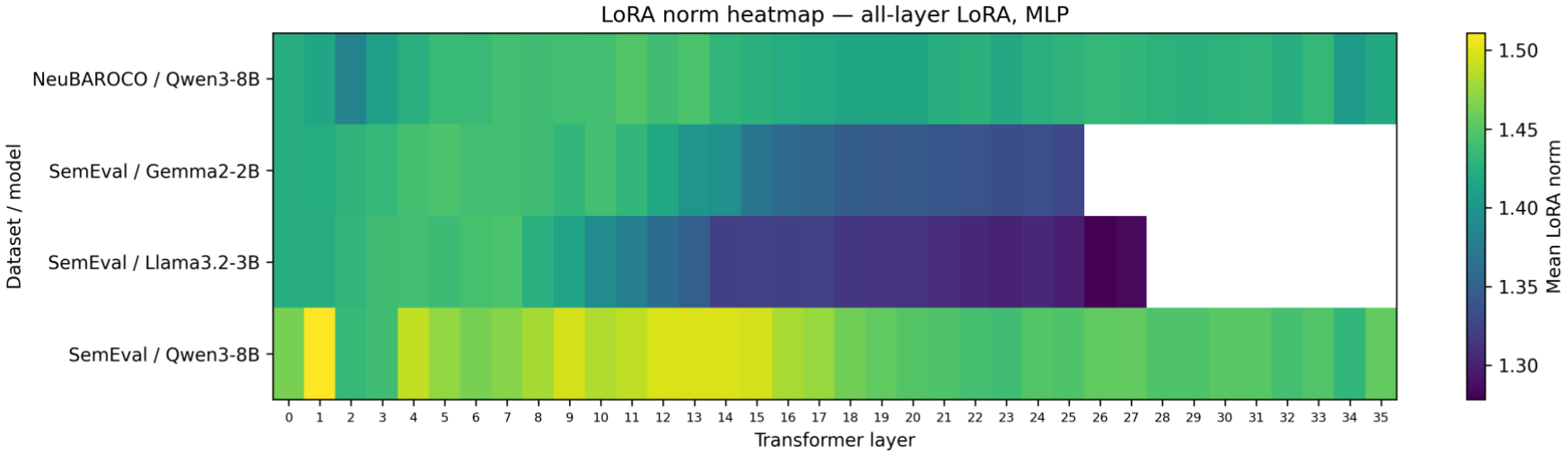


CONTENT EFFECT



Is it biased by real-world plausibility?
Lower content effect is preferable.

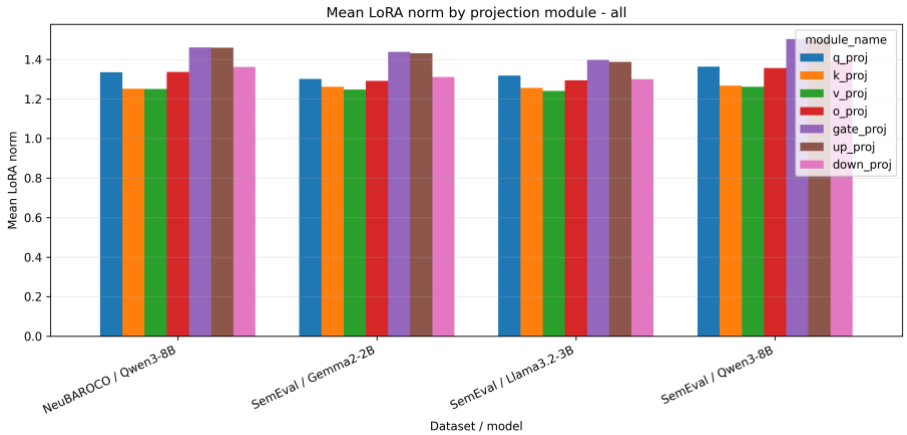
LAYER-WISE LORA NORM



CONCLUSION

Adaptation is concentrated in specific layers and submodules

With stronger adaptation observed in earlier-to-intermediate transformer layers, attention query projections, and MLP gate and up projections.



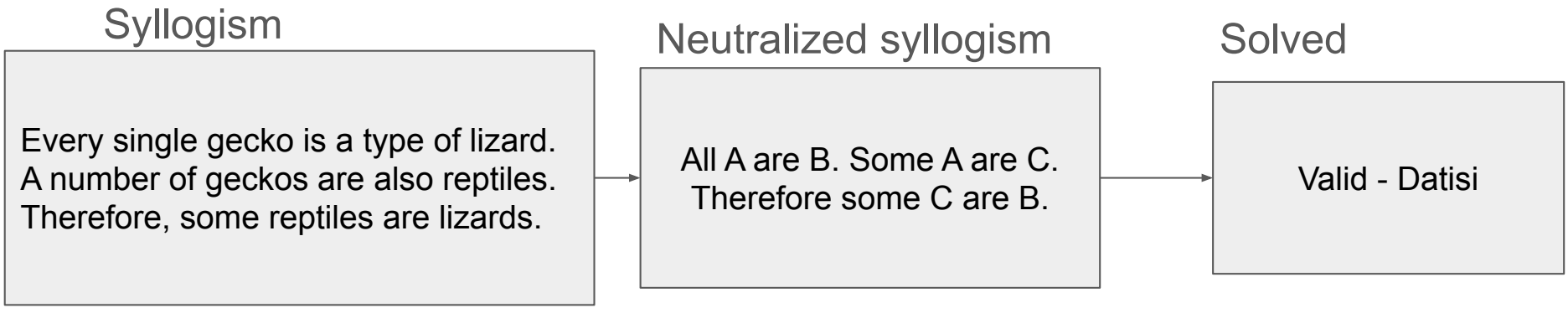
LIMITATIONS

a single random seed

No multi-run evaluation or standard deviation estimates

NeuBAROCO experiments were conducted only for Qwen3-8B.

Method



Name	Pattern	Name	Pattern
Barbara	All A are B; All C are A; therefore All C are B.	Darapti	All A are B; All A are C; therefore Some C are B.
Celarent	No A are B; All C are A; therefore No C are B.	Disamis	Some A are B; All A are C; therefore Some C are B.
Darii	All A are B; Some C are A; therefore Some C are B.	Datisi	All A are B; Some A are C; therefore Some C are B.
Ferio	No A are B; Some C are A; therefore Some C are not B.	Felapton	No A are B; All A are C; therefore Some C are not B.
Barbari	All A are B; All C are A; therefore Some C are B.	Bocardo	Some A are not B; All A are C; therefore Some C are not B.
Celaront	No A are B; All C are A; therefore Some C are not B.	Ferison	No A are B; Some A are C; therefore Some C are not B.
Cesare	No A are B; All C are B; therefore No C are A.	Bamalip	All A are B; All B are C; therefore Some C are A.
Camestres	All A are B; No C are B; therefore No C are A.	Calemes	All A are B; No B are C; therefore No C are A.
Festino	No A are B; Some C are B; therefore Some C are not A.	Dimatis	Some A are B; All B are C; therefore Some C are A.
Baroco	All A are B; Some C are not B; therefore Some C are not A.	Fesapo	No A are B; All B are C; therefore Some C are not A.
Cesaro	No A are B; All C are B; therefore Some C are not A.	Fresison	No A are B; Some B are C; therefore Some C are not A.
Camestros	All A are B; No C are B; therefore Some C are not A.	Calemos	All A are B; No B are C; therefore Some C are not A.

Table D.1: Valid syllogistic forms accepted by the lookup solver.

Results

System	Model / mode	Correct	ACC	CE_{intra}	CE_{inter}	TCE	D_c	Score
Zero-shot	Qwen3-8B Q8, thinking	164/191	85.86	1.06	23.94	12.50	1.06	23.83
Zero-shot	Qwen3-8B Q8, no thinking	128/191	67.02	24.07	55.10	39.58	24.07	14.25
ICL	Qwen3-8B Q8, thinking	176/191	92.15	1.04	15.63	8.33	1.04	28.50
ICL	Qwen3-8B Q8, no thinking	144/191	75.39	26.37	26.37	26.37	26.37	17.49
Neutralize + solve	Qwen3-8B Q8, thinking	187/191	97.91	0.00	4.17	2.08	0.00	46.05
Neutralize + solve	Qwen3-8B Q8, no thinking	134/191	70.16	3.37	17.46	10.42	1.29	20.42
Zero-shot	Qwen3.6-27B MTP Q4, thinking	171/191	89.53	2.11	18.73	10.42	2.11	26.06
Zero-shot	Qwen3.6-27B MTP Q4, no thinking	169/191	88.48	12.70	6.45	9.57	6.45	26.35
ICL	Qwen3.6-27B MTP Q4, thinking	184/191	96.34	5.21	7.29	6.25	5.21	32.32
ICL	Qwen3.6-27B MTP Q4, no thinking	176/191	92.15	5.30	5.12	5.21	1.13	32.61
Neutralize + solve	Qwen3.6-27B MTP Q4, thinking	191/191	100.00	0.00	0.00	0.00	0.00	100.00
Neutralize + solve	Qwen3.6-27B MTP Q4, no thinking	189/191	98.95	0.00	2.08	1.04	0.00	57.74
Zero-shot	Qwen3.6-35B-A3B MTP IQ4, thinking	165/191	86.39	8.40	20.77	14.58	-2.02	23.06
Zero-shot	Qwen3.6-35B-A3B MTP IQ4, no thinking	170/191	89.01	9.55	3.30	6.43	3.30	29.62
ICL	Qwen3.6-35B-A3B MTP IQ4, thinking	185/191	96.86	2.08	6.25	4.17	2.08	36.66
ICL	Qwen3.6-35B-A3B MTP IQ4, no thinking	169/191	88.48	6.38	1.95	4.17	-1.95	33.49
Neutralize + solve	Qwen3.6-35B-A3B MTP IQ4, thinking	191/191	100.00	0.00	0.00	0.00	0.00	100.00
Neutralize + solve	Qwen3.6-35B-A3B MTP IQ4, no thinking	186/191	97.38	3.15	3.15	3.15	3.15	40.20

System	Model / mode	Correct	ACC	CE_{intra}	CE_{inter}	TCE	D_c	Score
Zero-shot	Qwen3-8B Q8, thinking	813/960	84.69	1.47	27.57	14.52	-1.47	22.63
Zero-shot	Qwen3-8B Q8, no thinking	634/960	66.04	17.98	50.46	34.22	17.98	14.48
Neutralize + solve	Qwen3-8B Q8, thinking	946/960	98.54	0.05	2.82	1.43	-0.05	52.15
Neutralize + solve	Qwen3-8B Q8, no thinking	723/960	75.31	2.22	17.71	9.97	2.22	22.18
Zero-shot	Qwen3.6-27B MTP Q4, thinking	857/960	89.27	1.35	20.37	10.86	-1.35	25.70
Zero-shot	Qwen3.6-27B MTP Q4, no thinking	875/960	91.15	8.36	8.36	8.36	8.36	28.17
Neutralize + solve	Qwen3.6-27B MTP Q4, thinking	959/960	99.90	0.20	0.20	0.20	-0.20	84.20
Neutralize + solve	Qwen3.6-27B MTP Q4, no thinking	952/960	99.17	0.40	1.19	0.79	0.40	62.60
Zero-shot	Qwen3.6-35B-A3B MTP IQ4, thinking	850/960	88.54	0.89	20.08	10.49	0.89	25.73
Zero-shot	Qwen3.6-35B-A3B MTP IQ4, no thinking	844/960	87.92	6.48	6.48	6.48	6.48	29.18
Neutralize + solve	Qwen3.6-35B-A3B MTP IQ4, thinking	957/960	99.69	0.22	0.21	0.22	0.21	83.30
Neutralize + solve	Qwen3.6-35B-A3B MTP IQ4, no thinking	944/960	98.33	0.79	1.13	0.96	0.79	58.75

Other findings

Content effect metric

24+ Label errors in dataset

System	Model / mode	Correct	ACC	CE_{intra}	CE_{inter}	TCE	D_c	Score
Neutralize + solve	Qwen-8B Q8, thinking	187/191	97.91	0.00	4.17	2.08	0.00	46.05

Issue type	Example	Formal judgement
Valid EAE pattern	Not a single creature that is a retriever is a dog. Every single animal is a retriever. It must be the case that no animal is a dog.	Valid
Strict three-term failure	Anything that is cutlery is furniture. There are no pieces of cutlery which are also spoons. It is the case that some cutlery is not a spoon.	Invalid
Existential import	It is true that anything that is an apple is an edible thing. Every apple is a fruit. It follows that some fruits are edible things.	Valid
Valid AOO pattern	The set of humans is a subset of the set of mortals. At least one robot is not a mortal. There is at least one robot who is not a human.	Valid

Same Logic, Different Words

A controlled perturbation benchmark for robust syllogistic reasoning

Motivation

High accuracy

LoRA Qwen3-8B reaches 98.06% on SemEval dataset.

The ITLC system which uses normalization and parsing strategy reaches 100% accuracy on the original validity task

Problem

Syllogistic validity should depend on logical form, not wording.

Perhaps the models rely on some heuristics rather than perform a proper logical reasoning?

Robustness question

Will the prediction change if the logic stays the same but is expressed with different wording?

Author: Jacek Węgrzynowski

Benchmark: paired rewrites of the same syllogism

Same structure and validity label, but different wording

Original example

All flowers are stones.

No stones are roses.

Therefore, some roses are not flowers.

Term wrapping

All flowers are stones
→ Every member of the category flower belongs to category stone.

Double negation

All flowers are stones
→ No flower is outside the category stone.

Connected argument form

Three separate sentences
→ One continuous sentence which combines premises and conclusion.

Logical distractor

Add irrelevant truth
→ “One equals one” inserted between premises.

Results: aggregate accuracy can hide failures

LoRA remains strong overall, but double negation exposes instability

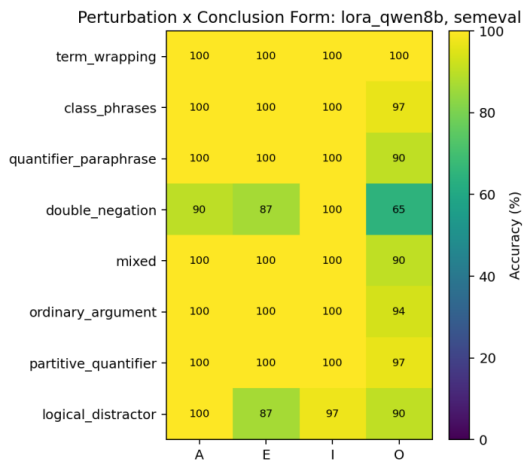
Overall accuracy

Setting	Original	Perturbated	Performance Drop Rate
LoRA SemEval	98.06	96.24	1.86

Where the failure appears

Failure is not random: it concentrates in double negation, especially with O-form conclusions (“Some S are not P”)

Main diagnostic result



- Rows show perturbation types and columns show conclusion forms
- Most combinations stay near 100%
- The only noticeable drops are for double negation and logical distractor perturbations
- Main weakness: double negation with O-form conclusions
  - accuracy reaches only 65%, compared to 93.55% on the original examples
- This shows that the average result can hide a specific weakness in negation handling

Takeaways

What the benchmark reveals

1. Accuracy is not enough

High original accuracy can hide systematic failures under controlled rewrites.

2. Errors are diagnostic

The hardest cases involve double negation and O-form conclusions, metadata helps locate these patterns.

3. Some perturbations can clarify

Some rewrites improve performance, especially on noisier examples.

Negative PDR is informative, not a bug.

LLM Reasoning

Outstanding paper in this track

Outstanding paper in this track

Neutralize, Then Solve: Content-Robust Syllogistic Reasoning for SemEval-2026 Task 11

Daniel Nygården
Department of Informatics
University of Oslo
Oslo, Norway
daniny@uio.no

Torgeir Skovholt
Department of Informatics
University of Oslo
Oslo, Norway
torgesko@uio.no

Abstract

Large language models can solve many syllogistic reasoning problems, but their validity judgements are often entangled with the plausibility of the argument's content. We study SemEval-2026 Task 11, Subtask 1, and evaluate a simple neuro-symbolic pipeline: an LLM first rewrites each natural-language argument as a content-neutral categorical syllogism, after which a deterministic lookup solver decides validity from the abstract form. The pipeline separates parsing from validity judgement, removes real-world content from the reasoning step, and makes failures inspectable. Two final test configurations reach 191/191 correct, and the same architecture gives a large training-set gain over direct zero-shot prompting. We also analyze the official content-effect metric and show that part of it measures valid/invalid class asymmetry rather than content effects. The analysis separates cases where believable con-

We investigate a simpler decomposition. Instead of asking the model to make the final validity judgement, we ask it to remove content. The model converts the input into a canonical form using neutral symbols such as A, B, and C. A deterministic solver then checks whether the resulting categorical syllogism matches one of the valid Aristotelian forms. This design does not require fine-tuning, ensembles, or theorem-prover translation. It also lets us inspect each failure as either a neutralization problem, a solver mismatch, a label issue, or a metric issue.

In this paper, we evaluate whether a simple LLM neutralizer combined with a deterministic syllogism solver can reach maximum scores on SemEval-2026 Task 11, Subtask 1. We also analyze how the task's content-effect metric behaves and inspect the remaining errors exposed by the intermediate neutral form.

Track Overview: Open-Ended Regional QA with RAG

Track chair: Jindřich Helcl

Background

- Most QA benchmarks focus on globally known facts and use multiple-choice formats.
- Regional knowledge is a particularly hard test case: culturally specific facts from less-represented languages, often absent from model training data.
- **CUS-QA**: open-ended QA benchmark in Czech, Slovak, and Ukrainian, grounded in local Wikipedias. Even the best open-weight LLMs achieve only $\approx 60\%$ accuracy.

Task

- Build a cross-lingual RAG system that retrieves relevant Wikipedia passages and generates correct open-ended answers in all three languages.

Track Overview: Open-Ended Regional QA with RAG

Required experiments

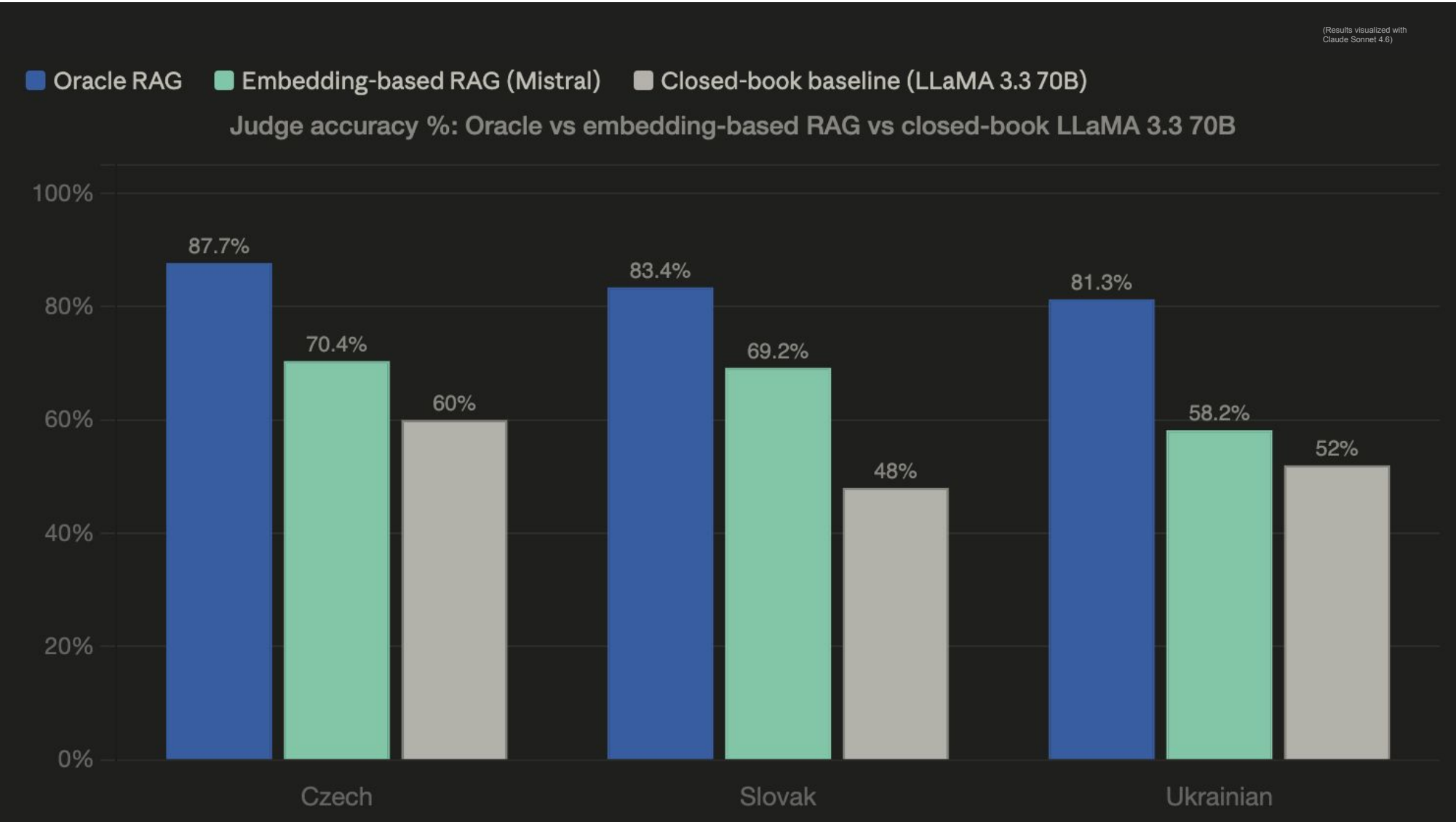
- **Oracle RAG:** use the reference Wikipedia URL as context — upper bound on retrieval quality.
- **Embedding-based RAG:** build a dense index over FineWiki; baseline embedding model is `Multilingual-E5-Large`.
- Compare multiple open-weight LLMs as the generator.

Evaluation

- Primary: LLM-as-a-judge (M-Prometheus) on the dev set.
- Test set scored on Codabench (chrF, BLEU, BERTScore, . . .).

Open-Ended Regional Question Answering with RAG

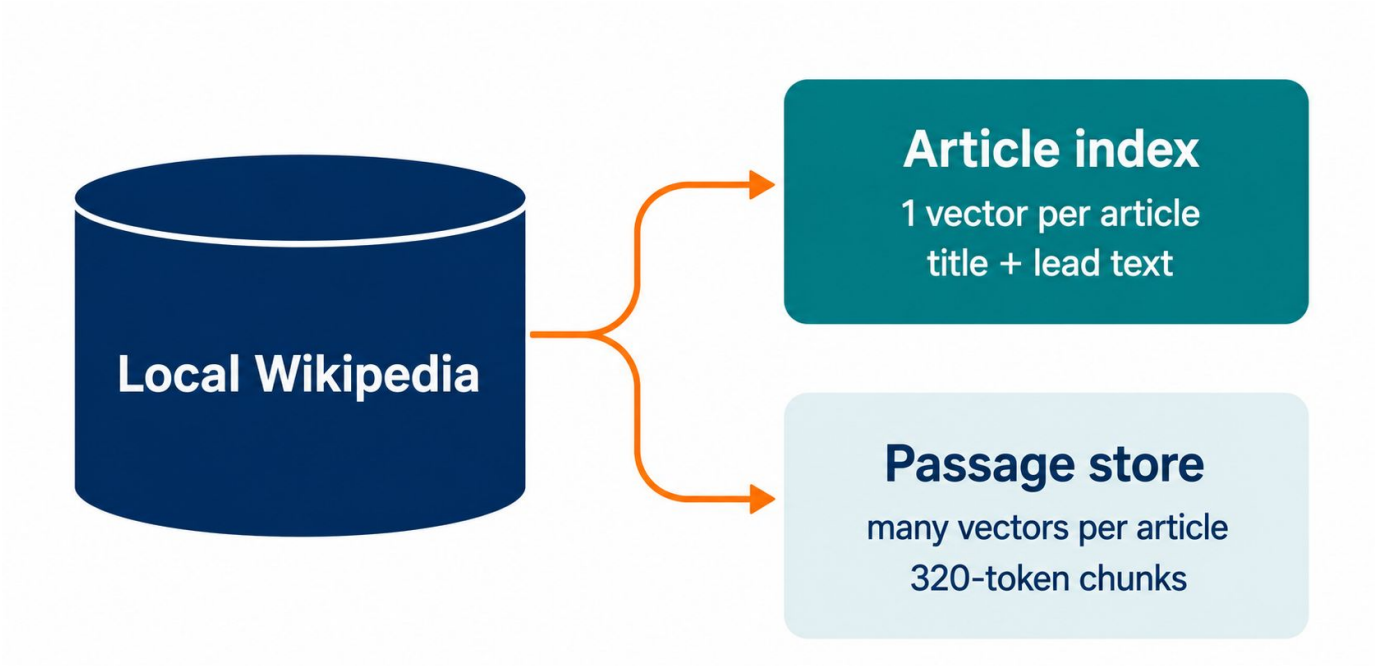
Jarand Akselberg Tomter


(Results visualized with Claude Sonnet 4.6)
Oracle RAG
Embedding-based RAG (Mistral)
Closed-book baseline (LLaMA 3.3 70B)
Judge accuracy %: Oracle vs embedding-based RAG vs closed-book LLaMA 3.3 70B
100%
80%
60%
40%
20%
0%
87.7%
70.4%
60%
83.4%
69.2%
48%
81.3%
58.2%
52%
Czech
Slovak
Ukrainian

Better retrieval, not larger models

Two-level wikipedia index

Local Wikipedia corpus
≈2M articles
≈6.9M passages



Hierarchical retrieval



Did it work?

E5-hier recovered:
≈98% of oracle chrF
≈94% of oracle judge accuracy

Best realistic system:
Gemma-4-31B + E5-hier
55.77 chrF / 81.7% judge

Cross-Lingual RAG for Open-Ended Regional QA

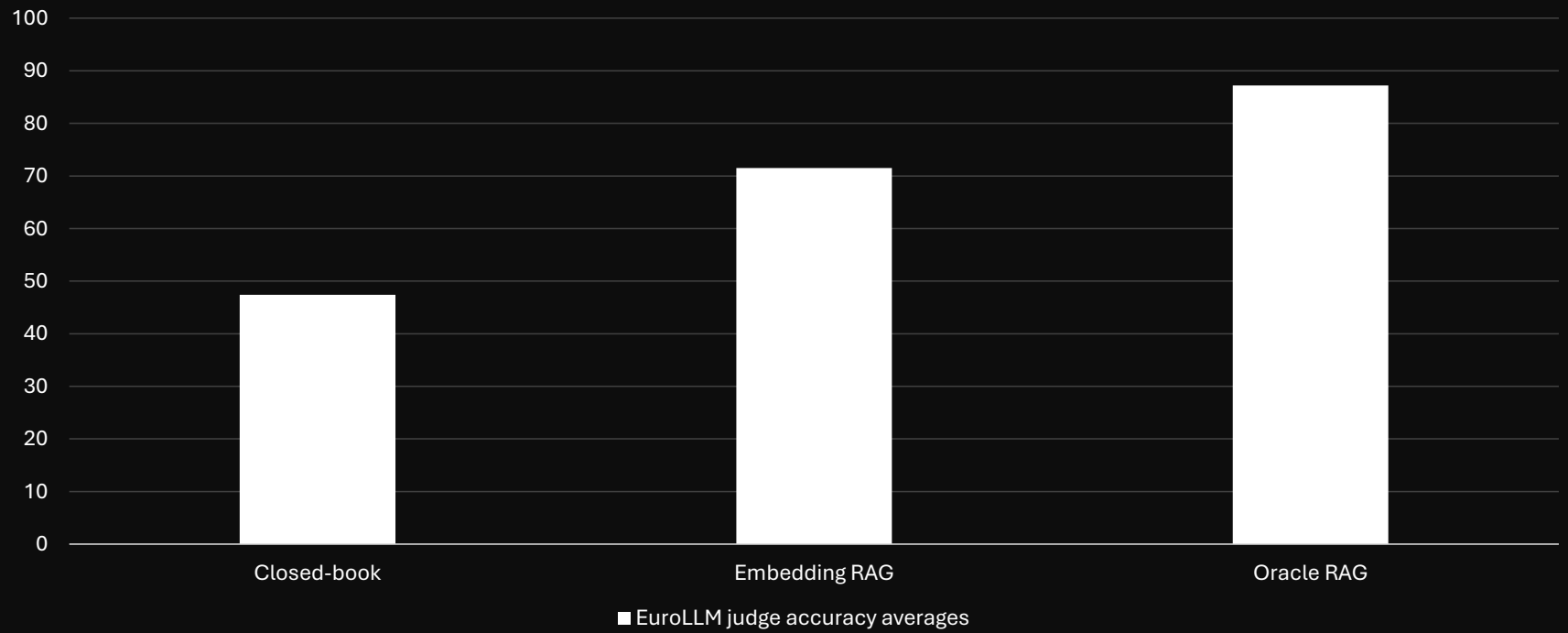
Gard Aanstad - gardaan@uio.no

Pipeline

FineWiki ⟶ Chunk ⟶ Embed

⟶ FAISS ⟶ LLM ⟶ Answer

Evidence Access is the Bottleneck



Cross-lingual retrieval helps

	chrF cs	chrF sk	chrF uk	Judge avg.
Monolingual E5	29.3	42.5	42.2	70.1
Cross-lingual E5	**31.6**	**43.4**	42.2	**71.5**

Results

Overall:

32.98 chrF

Same-language splits:

38.17 chrF

Cross-Lingual RAG for Regional QA

Hugo Hilde

Building and tuning the pipeline — small generators first, then scale

1. **Tune on small generators (oracle first)**
   Experimented with the generator prompt — first run into the chrF ↔ LLM-judge tradeoff.

2. **Build the dense index**
   256-word chunks, 32-word overlap; embedded with E5-Large and BGE-M3 in FAISS.

3. **Study judge prompts**
   Prompt alone moved judged accuracy by ±10 pts on identical predictions.

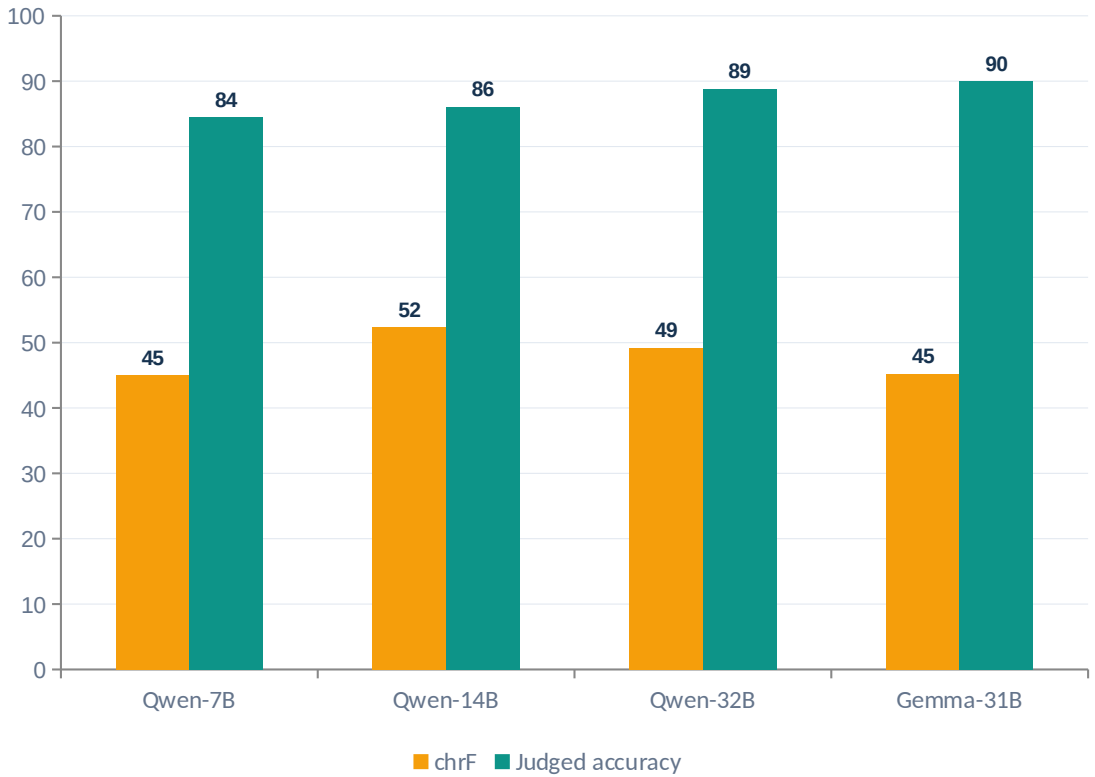
4. **Fix retrieval**
   Swept top-k (3–15), then added a cross-encoder reranker: pool of 50 → top-7. BGE-M3 ≳ E5.

5. **No re-chunking**
   Planned other chunking strategies, but at ~2% below the oracle, re-embedding the corpus each time wasn't worth the cost.

6. **Scale the generator**
   Remaining gap to the oracle score is generation, not retrieval → swap in Qwen2.5-14B/32B and Gemma 4 31B.

Results: scaling the generator

Best retrieval fixed (BGE-M3 + reranker, N=50 → top-7) — dev set



Best (judged accuracy)

BGE-M3 · N=50 → top-7 · Gemma 4 31B

≈ 90% on the full dev set

Best chrF

Qwen2.5-14B (52.3)

ranking reverses vs. the judge

Dev results hold up

Test-set chrF ≈ dev set (Codabench)

dev numbers are a valid indicator

Manual inspection

Judged accuracy seems the better metric

small subset; I don't speak the languages

Minding the Retrieval Gap in RAG: *An Oracle vs. Embedding RAG Analysis on Multilingual Regional QA*

Younes Bouhmidi

Models Tested

Tiny Aya Global & Water (**3.35B** multilingual) · **Llama-3.1-8B** (open-weight baseline) · **Qwen3-30B-A3B** (strongest that I could run reliably)

Tiny Aya Water is specialised for **Europe** and Asia-Pacific; Global is the broad, multilingual version.

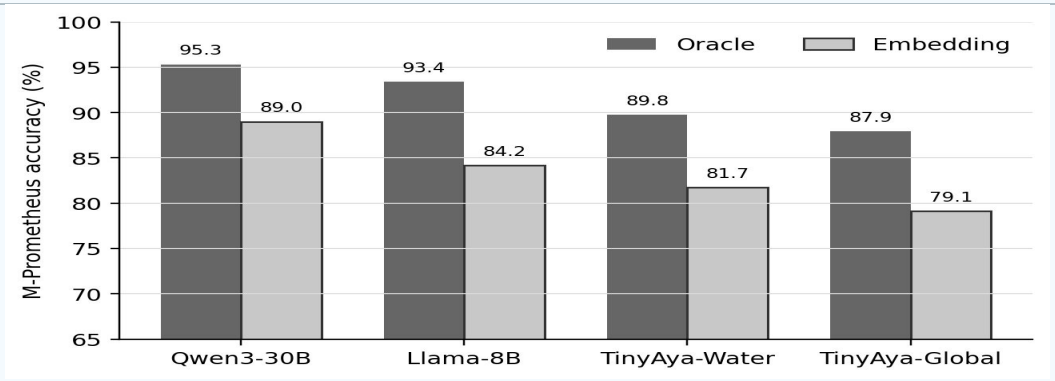
Four Research Questions

- ***Q1:*** *How large is the gap between exact-source Oracle RAG and realistic Embedding RAG?*
- ***Q2:*** *Does increasing retrieval depth from top-5 to top-10 help, or does it distract?*
- ***Q3:*** *Which open-weight generator gives the best quality under runtime and memory constraints?*
- ***Q4:*** *Do semantic LLM judging and lexical chrF evaluation lead to the same conclusions?*



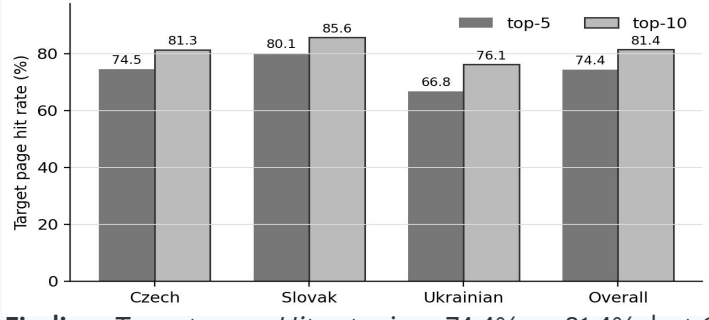
01 Introduction
02 Q1&Q2
03 Q3&Q4

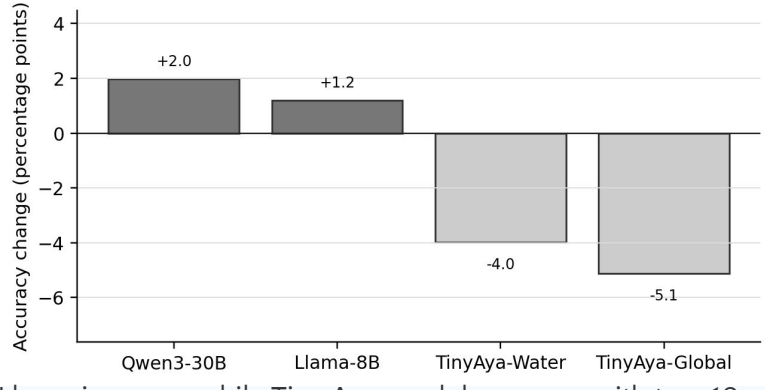
Q1: Oracle vs. Embedding RAG Gap



Finding: For Qwen3-30B, accuracy drops from **0.953** (Oracle) to **0.890** (Embedding). ***Retrieval remains the main bottleneck.***

Q2: Does Top-10 *k* Help or Distract?





Finding: *Target page Hit rate* rises 74.4% → 81.4%, but Qwen/Llama improve while Tiny Aya models worsen with top-10.

Takeaway: Coverage-vs-distraction tradeoff: larger models *use extra evidence*, while smaller models get *distracted* by it.

Q3: Best Generator Under Runtime & Memory Constraints

System (Embedding)	Setting	Acc.	Correct	Runtime	Peak GPU Mem	+Correct Answer
TinyAya-Water	top-5	0.8175	1151 / 1408	0:19:29	7.7 GiB / 1 GPU	—
TinyAya-Global	top-5	0.7912	1114 / 1408	0:19:42	7.7 GiB / 1 GPU	-37
Llama-8B	top-10	0.8416	1185 / 1408	1:11:56	16.4 GiB / 1 GPU	+34
Qwen3-30B	top-10	0.8899	1253 / 1408	2:54:08	60.7 GiB / A100	+68

Finding: Qwen best overall (0.890 acc, 51.09 chrF). **Tiny Aya Water** -2.4 pts behind **Llama**, 3.5x faster, <50% GPU memory.

Q4: chrF vs. M-Prometheus Evaluation Agreement

System	Condition	k	Generator	chrF	M-Prom.	Judge rank / chrF
Oracle Qwen3-30B	Oracle	–	Qwen3-30B	56.52	0.953	1 / 1
Oracle Llama-8B	Oracle	–	Llama-8B	45.89	0.934	2 / 4
Oracle TinyAya-Water	Oracle	–	TinyAya-Water	39.89	0.898	3 / 7
Qwen3-30B top-10	Embedding	10	Qwen3-30B	49.42	0.890	4 / 3
Oracle TinyAya-Global	Oracle	–	TinyAya-Global	39.87	0.879	5 / 8
Qwen3-30B top-5	Embedding	5	Qwen3-30B	49.95	0.870	6 / 2
Llama-8B top-10	Embedding	10	Llama-8B	40.51	0.842	7 / 6
Llama-8B top-5	Embedding	5	Llama-8B	40.52	0.830	8 / 5
TinyAya-Water top-5	Embedding	5	TinyAya-Water	37.10	0.818	9 / 10
TinyAya-Global top-5	Embedding	5	TinyAya-Global	37.29	0.791	10 / 9
TinyAya-Water top-10	Embedding	10	TinyAya-Water	36.56	0.778	11 / 12
TinyAya-Global top-10	Embedding	10	TinyAya-Global	37.07	0.740	12 / 11

Finding: Both agree **Qwen** is strongest, but rank middle systems *differently.* **chrF** rewards character overlap; **M-Prometheus** judges semantic correctness.

Final Takeaway

Retrieval remains the biggest gap: improving RAG is not just about larger generators or more context, but an engineering problem of finding, ranking, and filtering the right evidence.

Evaluating Multilingual RAG for Open-Ended Question Answering on CUS-QA

Mehrnoosh Mollaeian

University of Oslo

Workshop on Neural Natural Language Processing (WNNLP 2026)

Evaluating Multilingual RAG for Open-Ended QA

Task

- Open-ended question answering on the CUS-QA benchmark
- Three languages: Czech, Slovak, Ukrainian
- Questions require regional/local knowledge

Goal

- Compare:
  - Oracle RAG (gold Wikipedia page)
  - Embedding-based RAG (retrieved Wikipedia passages)

Research Question

- How much does retrieval quality affect multilingual QA performance?

Language	Dev	Test
Czech	530	550
Slovak	493	479
Ukrainian	385	370
Total	1408	1399

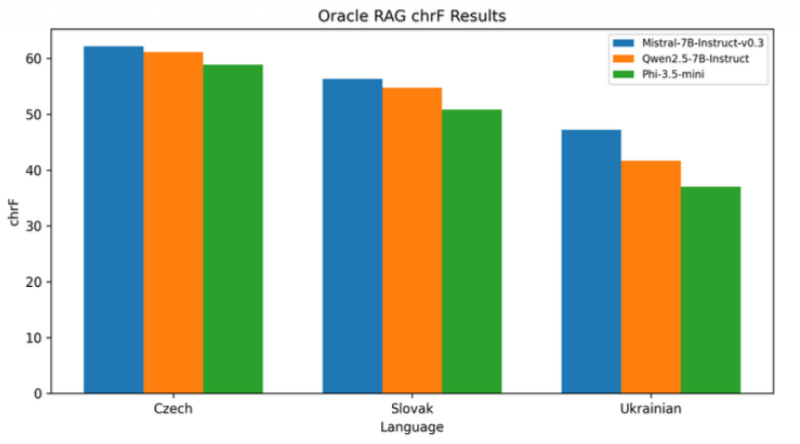
Methodology

- **Data**
  - CUS-QA benchmark
  - FineWiki Wikipedia corpus
- **Chunking**
  - Paragraph-aware chunking
  - 300-word chunks, 50-word overlap
  - Title-enhanced chunks
- **Retrieval**
  - Multilingual-E5-Large embeddings
  - Top-5 retrieved chunks
- **Generation Models**
  - Mistral-7B-Instruct
  - Qwen2.5-7B-Instruct
  - Phi-3.5-mini
- **Evaluation**
  - M-Prometheus (LLM Judge)
  - ChrF

Oracle vs. Retrieval: What Changes

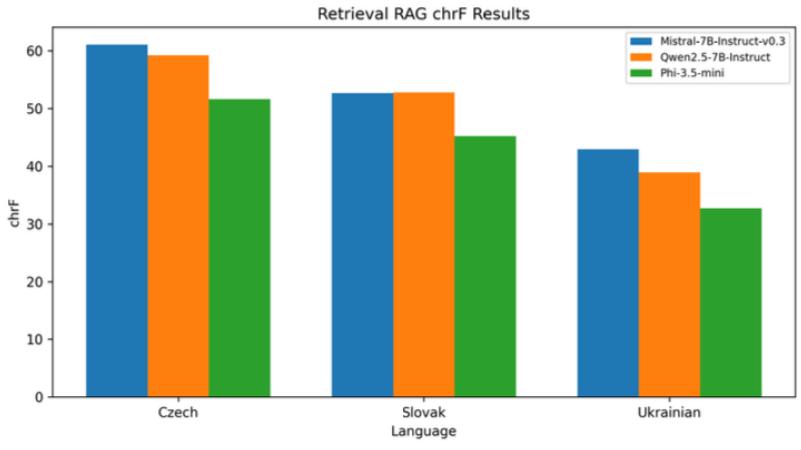
Oracle Retrieval

(Correct Wikipedia page provided)



Embedding Retrieval

(Top-5 retrieved passages)



- Oracle consistently outperforms retrieval.
- Ukrainian shows the lowest performance across settings.
- Qwen and Mistral outperform Phi-3.

Interesting Observation
Slovak achieved the highest retrieval accuracy (78.7%), but Czech achieved the strongest QA performance.

Key Findings & Insights

1. Retrieval quality strongly affects QA performance

Language	Hit@5	Hits/Total
Czech	76.04	403/530
Slovak	78.70	388/493
Ukrainian	69.61	268/385

2. Evaluation depends on the judge

System	M-Prometheus	Mistral Judge	Qwen Judge
Mistral-7B (cs)	86.42	100	79.81

3. Correct retrieval does not guarantee a correct answer

Retrieved:
✓ Madeleine Albright

Prediction:
✗ Hillary Clinton

Discussion & Takeaways

What surprised me?

Better retrieval does not always lead to a correct answer.

- Some questions failed even when relevant evidence was retrieved.

Evaluation itself is challenging.

- Different LLM judges produced substantially different scores.

Retrieval remains the main opportunity for improvement.

- Current Hit@5 is only 70–79%.
- Missing evidence still limits answer quality.

Future Work

- Retrieval reranking
- Stronger multilingual retrieval models
- More robust evaluation strategies

Open-Ended Regional QA with RAG

Outstanding paper in this track

Open-Ended Regional QA with RAG

Outstanding paper in this track

Minding the Retrieval Gap in RAG: An Oracle vs. Embedding RAG Analysis on Multilingual Regional QA

Younes Bouhmidi
Department of Informatics
University of Oslo
younesb@uio.no

Abstract

Open-ended regional question answering asks for locally grounded facts, such as the garden associated with a Czech town or the genre of a Slovak music association, that broad multilingual benchmarks fail to capture and current large language models (LLMs) struggle to answer. We study how retrieval-augmented generation (RAG) can close this gap on CUS-QA, a dataset of native-speaker questions in Czech, Slovak, and Ukrainian. To separate retrieval from generation errors, we compare Oracle RAG, fed the known source page, against Embedding RAG, which retrieves same language chunks via a dense multilingual embedding model. We evaluate four open weight generators, `Tiny Aya Global (3.35B)`, `Tiny Aya Water (3.35B)`, `Llama 3.1 8B`, and `Qwen3-30B-A3B`, with LLM-as-a-judge M-Prometheus and chrF on the hidden test. Retrieval is the dominant bottleneck: replacing exact page context with realistically re-

et al., 2026) was introduced to target this setting through native-speaker written questions grounded in Czech, Slovak, and Ukrainian Wikipedia pages.

Retrieval-augmented generation (RAG; Lewis et al., 2020) is a natural fit for CUS-QA because the questions are tied to Wikipedia evidence. Yet a RAG system can fail at two different points. The retriever may not find the right source page, or the generator may fail to extract the answer even when the evidence is present. These two failures have different practical implications: improving the generator cannot repair a missing document, while improving retrieval may not help if the generator is easily distracted by extra context.

This leads to four questions that guide the experiments:

1. How large is the gap between exact-source Oracle RAG and realistic Embedding RAG?

2. Does increasing retrieval depth from top-5 to

Closing

Thank you all for your presentations
and for your hard work!

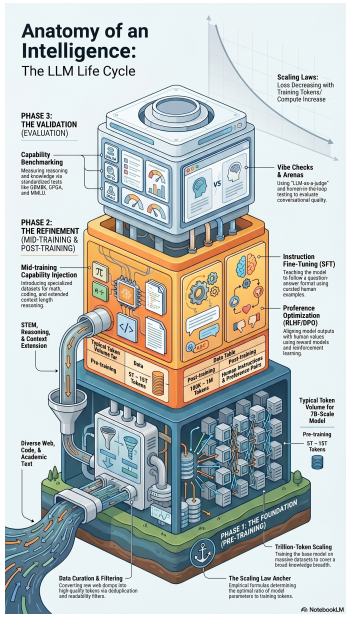
Closing

Thank you all for your presentations
and for your hard work!
You are now done with this course :-)

Closing

What's next?

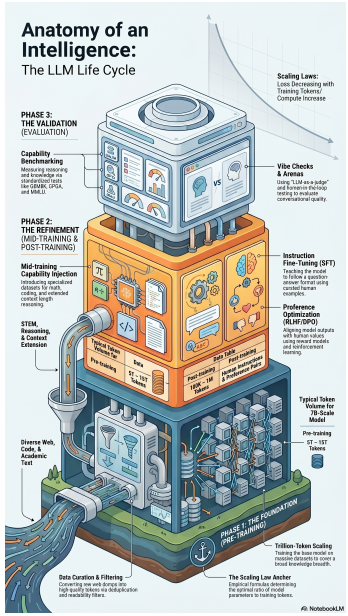
- If you are interested in large language models...
- Join our brand-new course IN5640 – Developing Large Language Models
- Offered first time this fall
- `https://www.uio.no/studier/emner/matnat/ifi/IN5640/`



Closing

What's next?

- If you are interested in large language models...
- Join our brand-new course IN5640 – Developing Large Language Models
- Offered first time this fall
- `https://www.uio.no/studier/emner/matnat/ifi/IN5640/`



Lykke til!

- We wish you luck in all your endeavours
- It was a pleasure to work with you
- Submit your IN5550 papers to 'real' NLP venues!
- Hope to see you again many times more!

References I