Rethinking NLU Text Classification for Long Documents and Resource-Constrained Models

Anonymous ACL submission

Abstract

Encoder models have excelled at Natural Language Understanding (NLU) classification tasks for shorter texts. As recent datasets for NLU tasks, such as Sentiment Analysis, increasingly involve longer texts, the traditional 512-token context window of encoder models poses a challenge. To address this, we present sentence-level text selection methods, including heuristics and learned models, that enable context-limited encoders to effectively process longer documents while maintaining computational efficiency. Concurrently, we seek to optimize sub-10B parameter decoder models for NLU classification tasks in resource-constrained settings. We propose applying the pairwise comparison training method for such tasks, adapting the Bradley-Terry model, which significantly enhances model performance. Our evaluation primarily on the Norwegian Entity-Level Sentiment Analysis (ELSA) dataset, featuring texts with a mean length of 650 tokens, and on Norwegian and English EuroEval benchmarks, validates our approaches. Results show that text selection reduces training times by half and improves performance for encoder models on the longer document ELSA task. Furthermore, pairwise comparison training enables gemma-2-9b to achieve 83.3% weighted F1 on ELSA and establishes new performance benchmarks for sub-10B models with the EuroEval NLU classification datasets for sentiment analysis and linguistic acceptability.

1 Introduction

Encoder models trained for masked language modeling paved the way for Natural Language Processing (NLP) with transformer models. In recent years the generative decoder models have surpassed the encoder models in popularity and capability in many tasks within NLP. However, for text classification tasks within Natural Language Understanding (NLU) where training data is present, encoder models remain ahead of decoder models that are an order of magnitude larger, as shown in the EuroEval leaderboard.[1] With their smaller size, the encoder models also allow for NLU training and inference with data that need to be protected and therefore must remain on-premise.

One limitation of encoder models is their limited context window, typically capped at 512 subword tokens. While NLU tasks such as sentiment analysis (SA), linguistic acceptability (LA) and named entity recognition (NER) are frequently conducted on brief inputs, e. g., sentences, microblog posts and user reviews, emerging applications increasingly involve the analysis of substantially longer documents. These new tasks seek to leverage the expanded context windows offered by recent decoder models, which enable the processing of extended textual spans (Pi et al., 2024; Cai et al., 2024; Luo et al., 2022).

Our work focuses on enhancing NLU classification for both encoder models and decoder models with less than 10B parameters. We limit the GPU memory requirements to one 64GB instance, to mimic limitations that may be encountered, in particular when sensitive data can not leave the premises for modeling.

To overcome the limitations of the encoder models' traditional context window, we experiment with selection methods for extracting the relevant parts of a text for classification. This way, we can extend the models' operational range, avoiding any naïve chunking or sliding window approach with subsequent aggregation of partial results. We hypothesize that such text selection methods may be beneficial when employing generative models for classification as well. The "Lost-in-the-middle" effect on generative models has been studied, showing that too much irrelevant text degrades the models performance (Liu et al., 2024; Hsieh et al.,

[1] https://euroeval.com/leaderboards/Multilingual/european/#__tabbed_1_4, accessed July 2025.

2024). Moreover, longer prompts also incur high computational costs (McDonald et al., 2025). Reducing computational resource consumption is desirable, and allows for models to be run on lighter hardware.

We selected one novel dataset for an in-depth study of the effect of selecting relevant sentences from longer texts, and of other enhancements of the tested models. The Norwegian dataset for entity-level SA (ELSA) (Rønningstad et al., 2024) contains some attractive features for our study. The texts in this dataset have a mean tokenized length of 650 tokens. 60% of the texts are longer than the traditional encoder models' 512 token limit. Each text is annotated for sentiment regarding each entity, both at the document- and sentence level, establishing a ground truth for sentence relevance.

For decoder models, we test the effect of the same relevant text selection methods, and we experiment with a new method of fine-tuning such models through pairwise comparison. This method is common in the field of reinforcement learning from human feedback (RLHF), but to the best of our knowledge, our work is the first to apply this method to NLU classification tasks. We evaluate this method further on English and Norwegian NLU tasks, and find that this method surpasses previously recorded results for our tested decoder models, and yields results above any other evaluation on EuroEval (Saattrup Nielsen et al., 2025) for any sub-10B parameter model.

Our main contributions are twofold: First, we demonstrate that **sentence-wise text selection** reduces training times considerably and enables encoder models with limited context windows to effectively analyze longer documents.

Second, we propose the **pairwise comparison training method** for sub-10B decoder models that yields strong results on the ELSA dataset and establishes new state-of-the-art on the EuroEval platform.

2 Related Work

We here present previous work on extending NLP tasks to longer texts, and how to overcome the new challenges these longer texts pose.

2.1 Encoder Models in the LLM Era

The shortcomings, both in terms of resource efficiency and performance scores, of generative models for certain NLP tasks have been studied by

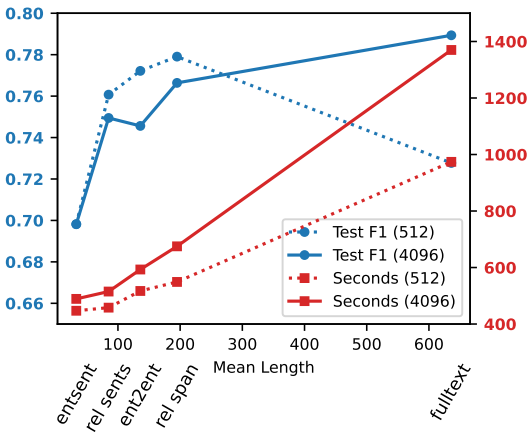

Figure 1: Weighted F_1 scores and training time for NorBERT4-large, max textlength set to 512 and 4096 tokens, on various ELSA dataset versions. We see how training time is doubled from the optimized dataset versions to the fulltext version. Sentence selection through the ent2ent heuristics and relevant span modeling improves performance for a model with a context window of 512. However, a context window of 4096 allows the model to learn best from the fulltext version.

Kocoń et al. (2023) in general, by Saattrup Nielsen et al. (2025) for NLU tasks, and by Martinelli et al. (2024) for coreference resolution. At the time of writing, the encoder model NorBERT3-large is ranked highest among the 262 models of any size at the EuroEval NLP leaderboard that have been fully tested for Norwegian NLU tasks. The DeBERTa-large (He et al., 2021) is likewise ranked highest among all 327 models ranked on the English NLU leaderboard, including up to 500B parameters instruction-tuned decoder models.

2.2 Pairwise Comparison in Supervised NLP

Pairwise comparison (pc) is a key technique for evaluating generated text, as introduced by Stiennon et al. (2020). The method is popularized in reward models training for language models instruction tuning (Ouyang et al., 2022). We further find the method implemented in evaluation through ranking, as for assessing open-domain dialogue systems (Park et al., 2024). Beyond evaluation, its application has been extended to direct supervised learning for document-level ranking tasks such as readability assessment (Lee and Vajjala, 2022). Our work builds on this paradigm by applying it to sentiment classification and linguistic acceptability and demonstrating its effectiveness on sub-10B models.

2.3 Relevant Text Selection

Liu et al. (2024) show how LLMs (Instruction-tuned large decoder models) can struggle to find information buried within the middle of irrelevant texts. See also Vodrahalli et al. (2024); Liskavets et al. (2025). Hsieh et al. (2024) Provide the RULER benchmark to better assess how well LLMs with large stated context windows, really manage to utilize this large space and extract the relevant information.

Padmakumar et al. (2025) show how context selection improves LLM summarization on the DiverseSumm benchmark (Huang et al., 2024). Sheng et al. (2025) discuss the dangers of breaking apart semantically coherent text when analyzing texts by chunks. These observations support our motivations for finding methods for text selection that improve over naïve text chunking.

2.4 Entity-Level SA

Ben-Ami et al. (2015) motivate the task of *Entity-level Sentiment Analysis* (ELSA). We apply their task description of identifying the document-level sentiment for each entity mentioned in each text. Rønningstad et al. (2024) released the Norwegian ELSA dataset, annotated by human experts, and provide initial baselines for modeling ELSA sentiment classification.

A work related to this is reported by Kuila and Sarkar (2024) who use the PerSenT dataset (Bastan et al., 2020) in their task of determining the overall sentiment polarity expressed towards a target entity in news texts. In contrast to the ELSA dataset, the PerSenT dataset is annotated for only one entity per document and the text length is limited to 16 sentences.

3 Datasets

Our primary evaluation is conducted on the Norwegian dataset for Entity-Level Sentiment Analysis (ELSA) . We further evaluate our method for fine-tuning decoder models on a subset of the datasets on NLU in the EuroEval evaluation suite, for Norwegian and English.

3.1 The ELSA Dataset

Each document in this dataset contains a review written by professional authors. Noteworthy features of the dataset include its annotation scheme, where each document is annotated for multiple person or organization entities at both the sentence and document level. Entities can co-occur in sentences, and sentiment is often directed via coreference or bridging, not just explicit mentions. Critically for our work, 60% of texts in the training split exceed 512 tokens after tokenization. The ELSA dataset has the label distribution per entity as seen in Table 1. The initial baselines for modeling as reported by Rønningstad et al. (2024) are a weighted average F_1 of 68.1% with encoder models, versus 73.3% with GPT-4.

1. Simon and Garfunkel were a duo featuring Paul Simon and Art Garfunkel.
2. Paul Simon's controlling nature often frustrated collaborators during sessions.
3. However, his extraordinary songwriting genius created timeless classics that elevated the duo.
4. Art Garfunkel's pretentious style created tension within the partnership.

Entity	Doc sentiment	Sentences
Simon & Garfunkel	Neutral	(1 neu)
Paul Simon	Positive	(1 neu),(2 neg),(3 pos)
Art Garfunkel	Negative	(1 neu),(4 neg)

Figure 2: Fictional text exemplifying the ELSA dataset, containing three entities with sentiment-relevant sentences and overall sentiment annotated for each entity, and entity-specific sentiment per sentence.

label split	Negative	Neutral	Positive	Count
dev	41	145	138	324
test	21	132	94	247
train	241	1014	653	1908
Pct	12.22	52.08	35.70	

Table 1: Distribution of the 2479 sentiment-annotated entities in the ELSA dataset across splits and polarities.

3.2 EuroEval Datasets

In order to further evaluate our method of pairwise comparison for NLU classification, we select the Norwegian and English datasets for linguistic acceptability (LA) and sentiment analysis (SA) in the EuroEval benchmarking framework. The SA datasets contain sentences labeled as either Positive, Neutral or Negative. There is one dataset for Norwegian SA in the test suite, a subset of NoReC (Velldal et al., 2018) and one English SA dataset, a subset of SST-5 (Socher et al., 2013). For both datasets, 1024 sentences are sampled from their train split, 256 sentences are sampled from the val-

idation split, and 2048 sentences are sampled from the test split.

The LA datasets contain sentences from the Universal Dependencies Treebank. These are either used as-is and labeled as grammatically correct, or they are carefully corrupted and labeled as grammatically incorrect (Nielsen, 2023).There is one such LA dataset for English (en) and one for each of the two written standards for the Norwegian language: Bokmål (nb) and Nynorsk (nn).

4 Methods

We here present our methods for relevant text selection by heuristics and by modeling, as well as our method for text classification with decoder models through pairwise comparison.

4.1 Preparing a Classification Dataset for Pairwise Comparison

When training a model for classification through pairwise comparison, we need two texts that are contrasted. Since we for the ELSA dataset will need to classify the same text with regards to various entities, this entity in question needs to be identified. We therefore prepend the text to classify with a sentence identifying the entity in question, and append the true sentiment label for the chosen text, and an incorrect label for the rejected text. A full example is found in Appendices A and B. To our knowledge, we are the first to apply this training regime to both sentiment classification and relevant text selection.

4.2 Pairwise Comparison Learning

To fine-tune a model on these pairs of chosen and rejected texts, a regression head is attached to the decoder model, and the output for the chosen text is compared to the output for the rejected text. The loss function $\mathcal{L}(\theta)$ is derived from the Bradley–Terry model (Bradley and Terry, 1952), which models the probability that one item is preferred over another as:

$$P(y_w \succ y_l|x) = \sigma\left(r_\theta(x, y_w) - r_\theta(x, y_l)\right) \quad (1)$$

where σ is the sigmoid function, $r_\theta(x, y)$ is the scalar output of the reward model for input x and response y, y_w is the winning (chosen) response, and y_l is the losing (rejected) response.

The loss function is formulated as the negative log-likelihood, implemented as the mean loss over batches of size B:

$$\mathcal{L}_B(\theta) = -\frac{1}{B}\sum_{i=1}^{B} \log \sigma(\Delta r_i) \quad (2)$$

where $\Delta r_i = r_\theta(x_i, y_{w,i}) - r_\theta(x_i, y_{l,i})$ represents the reward difference for the i-th training pair.

This formulation is implemented in the Hugging Face TRL RewardTrainer (von Werra et al., 2022).

While our approach shares the pairwise comparison principle with *contrastive learning*, it differs in output structure: traditional contrastive learning learns multidimensional embeddings that are compared via distance metrics in the embedding space, whereas our method directly computes scalar preference scores.

4.3 Relevant Text Selection

Our methods for relevant text selection on the ELSA dataset are based on available anchors for sentence relevance that we seek to exploit. In this section, we present the rationale for these methods, while details of the implementation are provided in Section 5.1.

The first anchor is the presence of a **mention of the entity** in question. The dataset is annotated for sentiment regarding each person and organization, which are standard entity categories in named entity recognition (NER). Suitable NER models exist for various languages, including Norwegian and English. We can therefore label any text containing these entities as part of the pre-processing steps and experiment with alternative heuristics for text selection based on sentences in which an entity is mentioned.

The second anchor is provided by the **sentence-level annotations** of the ELSA dataset. These supply ground truth labels identifying sentiment-relevant text with respect to the entity in question. As this information is not available for new texts, it is necessary to train a model on the annotated data in order to classify new sentences as relevant or irrelevant.

4.4 The Relevant Sentence Modeling Task

Formally, we have as input the ELSA dataset where sentence sentiment is annotated with respect to each entity. These are regarded the relevant sentences that the selector model should learn to distinguish from the non-relevant sentences.

Input: Document collection $\mathcal{D}$, where each

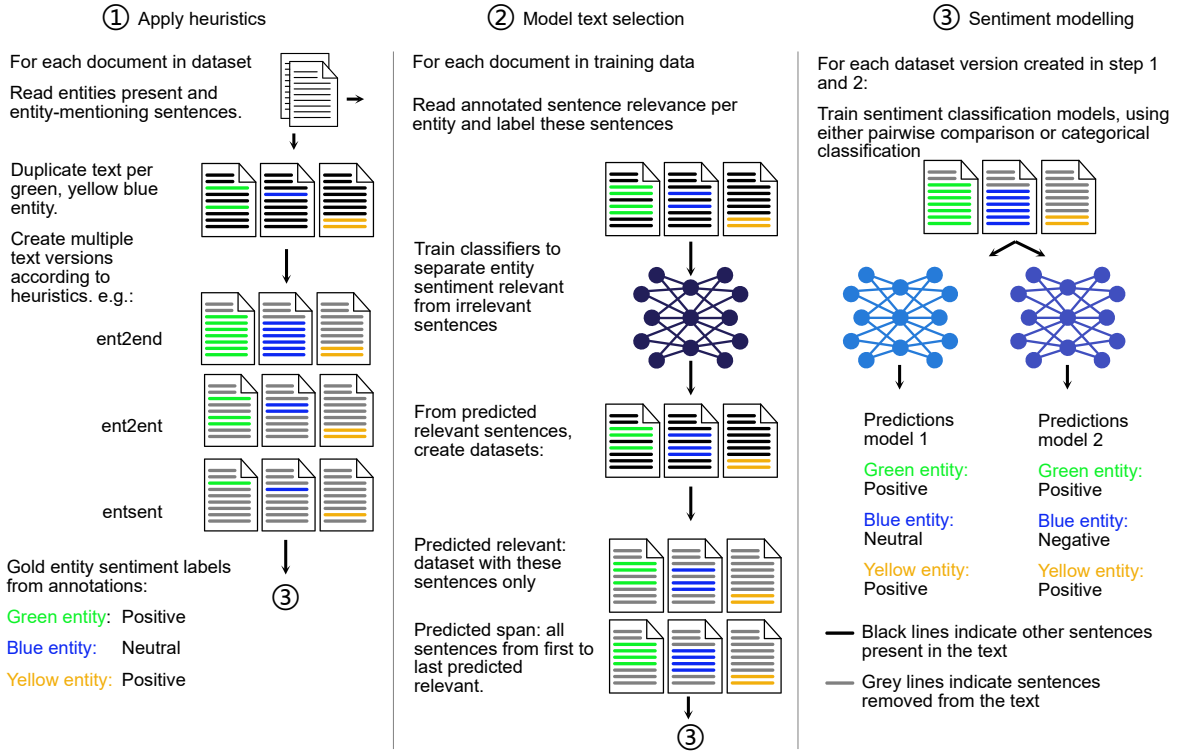

Figure 3: Preparing the ELSA dataset for classification through text selection. Step 1 utilizes only the named entities information in the dataset. Step 2 trains models for selecting relevant text, and uses these results as alternatives to the methods in Step 1. Step 3 trains alternative entity-specific sentiment classifiers based on the datasets from Steps 1 and 2.

$d \in \mathcal{D}$ contains sentences $\{s_1, s_2, \ldots, s_m\}$ and entities $\{e_1, e_2, \ldots, e_k\}$.

Task: For each pair (s_i, e_j) where $s_i \in d$ and $e_j \in d$, predict: relevance$(s_i, e_j) \in \{\text{relevant}, \text{irrelevant}\}$

Goal: Learn function f : Sentence $\times$ Entity $\rightarrow \{0, 1\}$ to classify sentence-entity relevance.

For pairwise comparison learning, a prompt containing the entity in question, preceding relevant text, the sentence in question, and a correct judgment whether this sentence should be included, is the chosen text, while a wrong judgment regarding the sentence's relevance is provided as the rejected text. See Appendix A for further details.

For relevant text selection with encoder models, two texts are passed as input. The first containing the entity in question and preceding relevant text, while the sentence to be classified is the second text. A classifier head with two output nodes are trained to classify the candidate sentence to be either relevant or irrelevant.

5 Experiments

We selected for our experiments the models that were leading among the sub–10B models for Norwegian NLU at the EuroEval language model benchmark (Nielsen, 2023). These are the *NorBERT3-large* (323M params) encoder model (Samuel et al., 2023) and the *Gemma 2 9B* decoder model (Team et al., 2024). We also tested these related models: *NorBERT4-large*[2] (360M params) and *Gemma 2 2B*.

The traditional approach to text classification with encoder models is attaching a classification head to the model with as many output nodes as there are categories for the label. Loss is calculated using cross-entropy loss, and predicted label is found with argmax. We use the term *categorical classification* (cc) for this method, which is used in the classification experiments using encoder models. The cc method is evaluated against decoder models trained with pairwise comparison (pc).

Our experiments with the ELSA dataset follow the following structure, as illustrated in Figure 3:

(1) Employ heuristics to select context for entity-

[2] https://huggingface.co/ltg/norbert4-large

model name	max l	meth	ent2end	ent2ent	entsent	fulltext	rel sents	rel span
gemma-2-9b-it	8192	i	69.15	67.24	59.51	73.14	63.79	64.55
gemma-2-9b	4096	pc		**83.30 (0.3)**		**83.08 (0.9)**	**82.51 (1.5)**	**82.82 (0.3)**
gemma-2-2b	4096	pc		78.84 (1.5)		78.18 (0.6)	77.73 (2.4)	76.76 (0.2)
norbert4-large	4096	cc	76.24 (1.2)	74.56 (1.7)	69.82 (1.9)	78.69 (1.2)	74.95 (2.1)	76.64 (3.4)
norbert4-large	512	cc	72.40 (7.4)	77.22 (2.1)	69.82 (1.9)	72.77 (2.2)	76.07 (3.0)	77.91 (1.6)
norbert3-large	512	cc	73.98 (0.5)	72.20 (1.5)	69.30 (1.7)	67.23 (2.7)	77.04 (0.5)	73.15 (4.2)

Table 2: Performance across models, classification methods and text selection method on the ELSA dataset. Mean weighted F_1 (st. dev.) over three runs. max l: max input tokens; meth: method (pc: pairwise comparison, cc: categorical classification, i: zero-shot inference). Text selection methods are detailed in Section 5.1. We see that gemma-2-9b achieves state-of-the-art when trained with pairwise comparison. Mean train script run times in minutes for gemma-2-9b were: **ent2ent:**247, **fulltext:**703, **rel sents:** 176, **rel span:** 361

task	lng	model	euroeval	ours
LA	en	encoder	**86.49 ±0.83**	
LA	en	g-2-2b	53.22 ±2.17	80.44 ±2.21
LA	en	g-2-9b	71.32 ±1.36	78.07 ±4.85
LA	nb	encoder	**84.91 ±4.26**	
LA	nb	g-2-2b	39.37 ±2.96	65.89 ±3.41
LA	nb	g-2-9b	74.34 ±2.9	77.3 ±7.17
LA	nn	encoder	**85.58 ±1.48**	
LA	nn	g-2-2b	44.84 ±3.99	53.13 ±3.25
LA	nn	g-2-9b	69.01 ±2.75	55.36 ±4.93
SA	no	encoder	71.98 ±1.98	
SA	no	g-2-2b	46.62 ±3.15	74.79 ±0.72
SA	no	g-2-9b	75.82 ±0.92	**79.01 ±0.75**
SA	en	encoder	62.94 ±3.0	
SA	en	g-2-2b	65.91 ±0.99	**71.94 ±0.7**
SA	en	g-2-9b	69.44 ±1.45	**72.92 ±0.87**

Table 3: Evaluation results for our fine-tuned gemma-2 models compared with the reported results on the EuroEval leaderboard. Our 95% confidence interval is calculated through bootstrapping the dataset 10 times, to match the EuroEval method. Our models provide a new state-of-the-art for sub-10B parameter models on the SA datasets. For the LA datasets, our method provides considerable improvements for 4 out of 6 dataset and model combinations.

relevant sentiment classification on longer texts.

(2) Train entity-specific sentence relevance classifiers for context selection.

(3) Train entity-specific sentiment classifiers models using the various contexts selected by the above methods as input.

5.1 Text Selection Implementations

In order to improve text classification of longer texts using smaller models, we experiment with both heuristics and modeling approaches. Only the ELSA dataset has such lengthy texts, the EuroEval datasets are single-sentence classification datasets. The presence of named entities (see Section 4.3) allows us to compare the following text selection heuristics: **fulltext**, which uses the entire document (truncated if necessary); **ent2end**, which includes all text from the first mention of an entity to the end of the document; **ent2ent**, which spans from the first mention of an entity until a different entity is mentioned; and **entsent**, which selects only the first sentence where the entity is mentioned. We speculate that the **ent2ent** heuristic would be a strong candidate, following the intuition that the sentiment-relevant text regarding an entity would start with the sentence where an entity is introduced, and end the sentence before another entity is introduced.

The presence of entity-sentiment annotations at the sentence level allows us to train models for classifying a sentence as relevant or irrelevant for an entity. To validate the potential of this method, we first train a classifier on the gold relevant sentences. We found that training an entity sentiment predictor from the *annotated* relevant sentences yields strong results, achieving over 82% F_1 with NorBERT3-large. This sentence-level information is not available on test data, but these results motivate exploring ways to train models for this relevant sentence selection, based on the dataset's annotations. Through modeling sentence relevance, we create two new dataset versions for entity-specific SA: **relevant sentences:** Concatenate the predicted relevant sentences per entity before classification; and **relevant span:** Concatenate all sentences in the text from the first to the last predicted relevant sentence. This latter approach has the potential of better capturing the semantic coherence, while risking more noise from sentiment signals regarding other entities.

5.2 Training Models for Relevant Text Selection

We trained models for relevant sentence selection from gemma-2-9b and NorBERT3. As the F_1 score was higher using the NorBERT3 model, this model was used to predict sentence relevance on the test set. Appendix A shows the text templates being used, while Algorithm 2 in Appendix C describes the details of preparing a dataset of chosen and rejected sentences for modeling.

5.3 Modeling ELSA with Categorical Classification

We here employ the standard approach for sentiment classification through the HuggingFace AutoModelForSequenceClassification,[3] which attaches a classification head to the model with one output node per label, and calculates the cross-entropy loss. We train models on each of the six text selection criteria presented in Section 5.1.

Encoder Models with Extended Context Window

Regular self-attention scales quadratically with sequence length, therefore increasing the context window quickly becomes infeasible. One approach to mitigate this is the sliding window attention (Beltagy et al., 2020). This is implemented in NorBERT4, the recently released successor of NorBERT3, allowing us to experiment with a context window of 4096, which is adequate for the full text lengths in the ELSA dataset.

5.4 Modeling ELSA with Pairwise Comparison

We fine-tune the gemma-2-2b and gemma-2-9b models using the text templates shown in Appendix B. Using non-instruction-tuned models allow for a subsequent wider selection of language-specific models, and reduces the complexity of the prompt setup.

A selection of design choices were finalized for the pc method (see Appendix D.1 for details). We used Norwegian prompts for Norwegian text; created two chosen-rejected pairs per data item (one for each incorrect label) to provide comprehensive negative examples; trained for two epochs, which we found optimal; and employed 4-bit quantization with LoRA (rank=16, alpha=32) to meet our GPU memory limitations.

[3] https://huggingface.co/docs/transformers/en/model_doc/auto

5.5 Key Findings from the ELSA Dataset

Figure 1 shows that we are able to mitigate long text lengths for an encoder model with limited context window through our methods for text selection. Classification results improve and training time is kept short. We also find that the NorBERT4 model with a larger context window is able to utilize the entire document text with a mean tokenized text length of 636 and achieve higher F_1 scores, at the cost of longer training times.

Table 2 shows that when fine-tuning gemma-2-9b for entity-specific SA, this model achieves a new state-of-the-art for the ELSA dataset, yielding a mean weighted F_1 score of over 83% across 3 seeds. The highest mean was achieved with the ent2ent selection method. The scores using the fulltext version are slightly behind, a not statistically significant difference. However, mean training script running time for gemma-2-9b was reduced from 703 minutes training on the fulltext versions, to 247 minutes when training on the ent2ent versions, a 65% reduction of training time without any performance loss. We also see that NorBERT4 performs on par with gemma-2-2b when trained with a context window of 4096, having less than 20% of the parameter count of gemma-2-2b.

5.6 Other ELSA Classification Methods

Experiments attaching a traditional classification head to gemma-2 models with LoRA did not surpass baseline levels. A full exploration of these instabilities is beyond the scope of this work, so these results are not discussed further.

We performed zero-shot inference with gemma-2-9b-it, with a prompt similar to what is described in Appendix B. The results are included in Table 2. The best performance was obtained using the full document text as input, and the scores were noticeably weaker than those of the fine-tuned models.

5.7 Evaluating the Pairwise Comparison Method on the EuroEval Benchmark

We replicate the EuroEval tests[4] for the two NLU tasks *sentiment analysis* and *linguistic acceptability*, for Norwegian and English, five datasets in total. See Table 3. The datasets are prepared for pairwise comparison training applying the euroeval prompt templates. Since all five datasets are single-sentence classification tasks, we do not experiment with sentence selection on these datasets,

[4] https://euroeval.com/methodology/

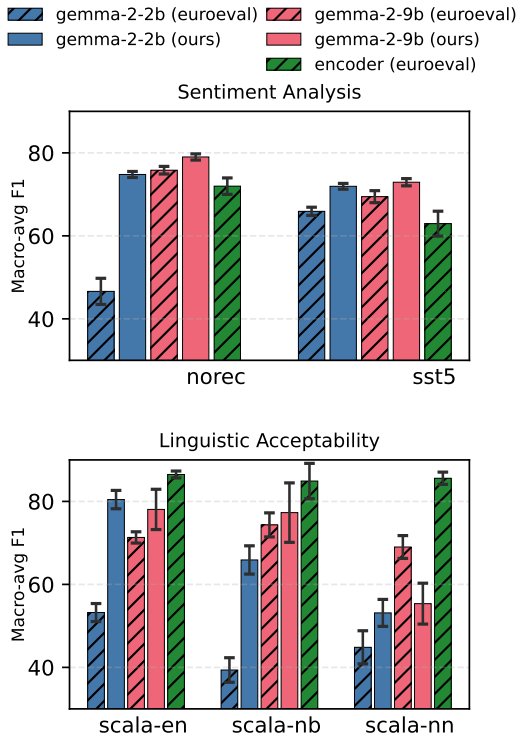

Figure 4: F_1 scores across EuroEval sentiment analysis and linguistic acceptability tasks. Striped bars show EuroEval baselines; solid bars show our fine-tuned models. Error bars indicate 95% confidence intervals based on 10 bootstrapped dataset versions.

but explore the general applicability of our method of classification through pairwise comparison.

We follow the test setup reported for the EuroEval test scores and bootstrap each dataset tenfold. For each bootstrapped version train and test three models using three fixed seeds. The ten mean macro-averaged F_1 scores over three seeds are averaged, and a 95% confidence interval is calculated from these ten values. For simplicity, we consider the results where there is no confidence interval overlap to be significant.

All EuroEval results are copied from the leaderboard where the best encoder models are DeBERTa-large for English and NorBERT3-large for Norwegian.

5.8 Key Findings from the EuroEval Datasets

As shown in Figure 4 and Table 3, our fine-tuned 2B and 9B gemma-2 models both performed significantly better than the EuroEval results for the same model, on five out of six datasets. For the two SA datasets, our finetuned gemma-2-9b models perform better than the EuroEval fine-tuned encoder models, and thus provide a new state-of-the-art for all sub-10B models. Our scores of 79.01% for Norwegian SA (norec) using gemma-2-9b, surpass any other scores on the leaderboard, including up to 500B models.

This extended testing shows that our suggested method for NLU classification through pairwise comparison is a viable alternative across a wider range of tasks, besides the state-of-the-art results on the ELSA dataset.

6 Conclusion

This work has presented two distinct strategies for advancing NLU classification in resource-constrained environments.

First, for encoder models constrained by short context windows, we demonstrated the efficacy of relevant text selection. Our heuristic and model-based approaches allow models with a 512-token limit to achieve substantial performance gains on the ELSA dataset, where most documents exceed this length. These methods effectively reduce the computational load while preserving the information critical for classification.

Second, for sub-10B parameter decoder models, we introduced the pairwise comparison training methodology for classification. This approach, adapted from reward modeling, proved highly effective for NLU classification. Our experiments show that gemma-2-9b, trained with this method, achieves a state-of-the-art weighted F1 score of 83.3% on the ELSA dataset. Text selection methods were also evaluated with this model. The ent2ent method allowed training time to be reduced by 65% for gemma-2-9b without any performance degradation.

Performance on the EuroEval benchmarks provides evidence for the generalizability of the pairwise comparison method. The approach yielded significant improvements for gemma-2 models on both sentiment analysis and linguistic acceptability tasks in Norwegian and English, establishing new performance benchmarks for sub-10B models. Notably, on the Norwegian sentiment analysis task, our result surpassed scores from models of any size.

7 Limitations

Our method for fine-tuning decoder models for NLU classification is tested on two Germanic languages, Norwegian and English, and two NLU tasks, sentiment analysis and linguistic acceptabil-

ity. The relevance for other languages would depend on how well the language is represented in the base models, and on labeled data. The relevance for other tasks remains an area for future work.

8 Ethical Considerations

Bias and Fairness: The pre-trained models and datasets used in this work may encode and amplify existing societal biases. For sentiment analysis, this could lead to models that assign unfairly negative sentiment to text associated with certain demographic groups. For linguistic acceptability, there is a risk that models trained on standard language corpora could penalize or misclassify non-standard dialects, sociolects, or text produced by non-native speakers, potentially leading to technological marginalization.

Intended Use and Potential for Misuse: Our methods are developed for research purposes. The sentiment analysis models could be used beneficially to study and reveal media bias. Conversely, they could also be misused for large-scale censorship or automated surveillance of online expression. The linguistic acceptability models, while useful for grammatical error correction, could be improperly deployed in contexts such as hiring or education to discriminate against individuals based on their adherence to a standard linguistic norm.

Use of AI Assistants. During various stages of research and writing, we used AI-assisted tools to help with editing, and clarifying text. Such tools were also used for developing code, primarily for data aggregation and analysis steps. All scientific claims, experimental design, and dataset analysis were conducted and verified by the authors.

Researcher Responsibility: We acknowledge the dual-use nature of this technology. Researchers and developers building upon this work have a responsibility to conduct thorough bias audits and to consider the potential societal impact of the applications they enable. We recommend that any deployment, particularly in high-stakes domains, be accompanied by safeguards and human oversight.

References

Jason Ansel, Edward Yang, Horace He, Natalia Gimelshein, Animesh Jain, Michael Voznesensky, Bin Bao, Peter Bell, David Berard, Evgeni Burovski, et al. 2024. Pytorch 2: Faster machine learning through dynamic python bytecode transformation and graph compilation. In Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2, pages 929–947.

Mohaddeseh Bastan, Mahnaz Koupaee, Youngseo Son, Richard Sicoli, and Niranjan Balasubramanian. 2020. Author's sentiment prediction. In Proceedings of the 28th International Conference on Computational Linguistics, pages 604–615, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. arXiv preprint arXiv:2004.05150.

Zvi Ben-Ami, Ronen Feldman, and Benjamin Rosenfeld. 2015. Exploiting the focus of the document for enhanced entities' sentiment relevance detection. In 2015 IEEE International Conference on Data Mining Workshop (ICDMW), pages 1284–1293.

Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. Biometrika, 39(3/4):324–345.

Hongjie Cai, Heqing Ma, Jianfei Yu, and Rui Xia. 2024. A joint coreference-aware approach to document-level target sentiment analysis. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 12149–12160, Bangkok, Thailand. Association for Computational Linguistics.

Charles R Harris, K Jarrod Millman, Stéfan J Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J Smith, et al. 2020. Array programming with numpy. nature, 585(7825):357–362.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pretraining with gradient-disentangled embedding sharing. Preprint, arXiv:2111.09543.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. 2024. Ruler: What's the real context size of your long-context language models? Preprint, arXiv:2404.06654.

Kung-Hsiang Huang, Philippe Laban, Alexander Fabbri, Prafulla Kumar Choubey, Shafiq Joty, Caiming Xiong, and Chien-Sheng Wu. 2024. Embrace divergence for richer insights: A multi-document summarization benchmark and a case study on summarizing diverse information from news articles. In Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 570–593, Mexico City, Mexico. Association for Computational Linguistics.

J. D. Hunter. 2007. Matplotlib: A 2d graphics environment. Computing in Science & Engineering, 9(3):90–95.

Jan Kocoń, Igor Cichecki, Oliwier Kaszyca, Mateusz Kochanek, Dominika Szydło, Joanna Baran, Julita Bielaniewicz, Marcin Gruza, Arkadiusz Janz, Kamil Kanclerz, et al. 2023. Chatgpt: Jack of all trades, master of none. Information Fusion, 99:101861.

Alapan Kuila and Sudeshna Sarkar. 2024. Deciphering political entity sentiment in news with large language models: Zero-shot and few-shot strategies. In Proceedings of the Second Workshop on Natural Language Processing for Political Sciences @ LREC-COLING 2024, pages 1–11, Torino, Italia. ELRA and ICCL.

Justin Lee and Sowmya Vajjala. 2022. A neural pairwise ranking model for readability assessment. In Findings of the Association for Computational Linguistics: ACL 2022, pages 3802–3813, Dublin, Ireland. Association for Computational Linguistics.

Barys Liskavets, Maxim Ushakov, Shuvendu Roy, Mark Klibanov, Ali Etemad, and Shane K. Luke. 2025. Prompt compression with context-aware sentence encoding for fast and improved llm inference. Proceedings of the AAAI Conference on Artificial Intelligence, 39(23):24595–24604.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the Middle: How Language Models Use Long Contexts. Transactions of the Association for Computational Linguistics, 12:157–173.

Yun Luo, Hongjie Cai, Linyi Yang, Yanxia Qin, Rui Xia, and Yue Zhang. 2022. Challenges for open-domain targeted sentiment analysis. Preprint, arXiv:2204.06893.

Giuliano Martinelli, Edoardo Barba, and Roberto Navigli. 2024. Maverick: Efficient and accurate coreference resolution defying recent trends. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 13380–13394, Bangkok, Thailand. Association for Computational Linguistics.

Tyler McDonald, Anthony Colosimo, Yifeng Li, and Ali Emami. 2025. Can we afford the perfect prompt? balancing cost and accuracy with the economical prompting index. In Proceedings of the 31st International Conference on Computational Linguistics, pages 7075–7086, Abu Dhabi, UAE. Association for Computational Linguistics.

Dan Nielsen. 2023. ScandEval: A benchmark for Scandinavian natural language processing. In Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa), pages 185–201, Tórshavn, Faroe Islands. University of Tartu Library.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In Advances in Neural Information Processing Systems, volume 35, pages 27730–27744. Curran Associates, Inc.

Vishakh Padmakumar, Zichao Wang, David Arbour, and Jennifer Healey. 2025. Principled content selection to generate diverse and personalized multi-document summaries. arXiv preprint arXiv:2505.21859.

The pandas development team. 2020. pandas-dev/pandas: Pandas.

ChaeHun Park, Minseok Choi, Dohyun Lee, and Jaegul Choo. 2024. Paireval: Open-domain dialogue evaluation with pairwise comparison. arXiv preprint arXiv:2404.01015.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12:2825–2830.

Qiankun Pi, Jicang Lu, Taojie Zhu, Yepeng Sun, Shunhang Li, and Jiaxing Guo. 2024. Enhancing cross-evidence reasoning graph for document-level relation extraction. PeerJ Computer Science, 10.

Egil Rønningstad, Roman Klinger, Lilja Øvrelid, and Erik Velldal. 2024. Entity-level sentiment: More than the sum of its parts. In Proceedings of the 14th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis, pages 84–96, Bangkok, Thailand. Association for Computational Linguistics.

Dan Saattrup Nielsen, Kenneth Enevoldsen, and Peter Schneider-Kamp. 2025. Encoder vs decoder: Comparative analysis of encoder and decoder language models on multilingual NLU tasks. In Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025), pages 561–572, Tallinn, Estonia. University of Tartu Library.

David Samuel, Andrey Kutuzov, Samia Touileb, Erik Velldal, Lilja Øvrelid, Egil Rønningstad, Elina Sigdel, and Anna Palatkina. 2023. NorBench – a benchmark for Norwegian language models. In Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa), pages 618–633, Tórshavn, Faroe Islands. University of Tartu Library.

Boheng Sheng, Jiacheng Yao, Meicong Zhang, and Guoxiu He. 2025. Dynamic chunking and selection for reading comprehension of ultra-long context in large language models. In Proceedings

of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 31857–31876, Vienna, Austria. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. Advances in neural information processing systems, 33:3008–3021.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. arXiv preprint arXiv:2408.00118.

Erik Velldal, Lilja Øvrelid, Eivind Alexander Bergem, Cathrine Stadsnes, Samia Touileb, and Fredrik Jørgensen. 2018. NoReC: The Norwegian review corpus. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan. European Language Resources Association (ELRA).

Kiran Vodrahalli, Santiago Ontanon, Nilesh Tripuraneni, Kelvin Xu, Sanil Jain, Rakesh Shivanna, Jeffrey Hui, Nishanth Dikkala, Mehran Kazemi, Bahare Fatemi, Rohan Anil, Ethan Dyer, Siamak Shakeri, Roopali Vij, Harsh Mehta, Vinay Ramasesh, Quoc Le, Ed Chi, Yifeng Lu, Orhan Firat, Angeliki Lazaridou, Jean-Baptiste Lespiau, Nithya Attaluri, and Kate Olszewska. 2024. Michelangelo: Long context evaluations beyond haystacks via latent structure queries. Preprint, arXiv:2409.12640.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, and Nathan Lambert. 2022. Trl: Transformer reinforcement learning. https://github.com/huggingface/trl.

Binghai Wang, Rui Zheng, Lu Chen, Zhiheng Xi, Wei Shen, Yuhao Zhou, Dong Yan, Tao Gui, Qi Zhang, and Xuanjing Huang. 2024. Reward modeling requires automatic adjustment based on data quality. In Findings of the Association for Computational Linguistics: EMNLP 2024, pages 4041–4064, Miami, Florida, USA. Association for Computational Linguistics.

Wes McKinney. 2010. Data Structures for Statistical Computing in Python. In Proceedings of the 9th Python in Science Conference, pages 56 – 61.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. arXiv preprint arXiv:1910.03771.

A Training the Selector Model for Sentence Selection

Training a selector model on selecting entity-relevant from irrelevant sentences requires text pairs where one is chosen (correct relevance judgment) and one is rejected (incorrect relevance judgment). The following contains the English version of the prompt template:

```
  """{0} is introduced in the Norwegian Main text.
  Is the Additional sentence Relevant or Irrelevant to the sentiment expressed
      toward {0}?
Main text: {1}
Additional sentence: {2}
Response: {3}
"""
0: entity name, 1: Text containing the entity. 2: New sentence which is either
    relevant or irrelevant
3: the judgement of this sentence being "Relevant" or "Irrelevant".
}
```

The Main text contains for training a concatenation of all previous sentences labeled as relevant. During inference, the main text equals previous sentences with the entity mentioned plus previous predicted relevant sentences. The following is a sample training pair with the Norwegian text machine translated into English.

Chosen text

```
 Helene Bøksle is introduced in the Norwegian Main text. Is the Additional sentence
 Relevant or Irrelevant to the sentiment expressed toward Helene Bøksle?
Main text: **When Helene Bøksle sings, it comes from the heart.
That's how I experienced it in the cathedral yesterday evening, and it became as Helene
Bøksle wanted it in the introduction: "a golden hour and a half".
As an artist, she has reached the point where she doesn't need to show off.
She has enough strength within herself, confidence in her voice, and composure on stage to
focus on conveying her message. And she does so convincingly.
This sense of simplicity has likely been developed through the collaboration between
Hotvedt and Bøksle, a collaboration that has now lasted for ten whole years,
long enough to get to know each other's musical taste and qualities.**
Additional sentence: **The consequence is that the lyrics are allowed to stand forth in all
their splendor, and she also selects some magnificent hymns:**
Response: Relevant
```

Rejected text

```
 Helene Bøksle is introduced in the Norwegian Main text. Is the Additional sentence
 Relevant or Irrelevant to the sentiment expressed toward Helene Bøksle?
 Main text: **When Helene Bøksle sings, it comes from the heart.
That's how I experienced it in the cathedral yesterday evening, and it became as Helene
Bøksle wanted it in the introduction: "a golden hour and a half".
As an artist, she has reached the point where she doesn't need to show off.
She has enough strength within herself, confidence in her voice, and composure on stage to
focus on conveying her message. And she does so convincingly.
This sense of simplicity has likely been developed through the collaboration between
Hotvedt and Bøksle, a collaboration that has now lasted for ten whole years,
long enough to get to know each other's musical taste and qualities.**
Additional sentence: **The consequence is that the lyrics are allowed to stand forth in all
their splendor, and she also selects some magnificent hymns:**
Response: Irrelevant
```

B Pairwise Comparison Training for ELSA Sentiment classification

The English prompt template for creating sentiment classification pairs:

```
'The entity {0} is introduced in the Norwegian Main text. We will analyze the
    sentiment expressed regarding this entity.\nMain text: {1}\nThe sentiment
    expressed regarding {0} when chosen from the available labels `["Positive",
    "Neutral", "Negative"]` is {2}'
Entity mention: {0}
Main text: {1}
Label: {2}
```

Each text with its sentiment label is made into two training pairs. The only difference lies in the final label category word. English machine translation of the Norwegian text.

Chosen text 1

The entity Norah Jones is introduced in the Norwegian Main text.
We will analyze the sentiment expressed regarding this entity.
Main text: **But Norah Jones does backing vocals.
That's at least something.**
The sentiment expressed regarding Norah Jones when chosen from the available labels
["Positive", "Neutral", "Negative"] is Positive

Rejected text 1

The entity Norah Jones is introduced in the Norwegian Main text.
We will analyze the sentiment expressed regarding this entity.
Main text: **But Norah Jones does backing vocals.
That's at least something.**
The sentiment expressed regarding Norah Jones when chosen from the available labels
["Positive", "Neutral", "Negative"] is Negative

Chosen text 2

The entity Norah Jones is introduced in the Norwegian Main text.
We will analyze the sentiment expressed regarding this entity.
Main text: **But Norah Jones does backing vocals.
That's at least something.**
The sentiment expressed regarding Norah Jones when chosen from the available labels
["Positive", "Neutral", "Negative"] is Positive

Rejected text 2

The entity Norah Jones is introduced in the Norwegian Main text.
We will analyze the sentiment expressed regarding this entity.
Main text: **But Norah Jones does backing vocals.
That's at least something.**
The sentiment expressed regarding Norah Jones when chosen from the available labels
["Positive", "Neutral", "Negative"] is Neutral

C Algorithms

Algorithm 1 describes the relevant text selection heuristics used in the *ent2ent* method. Algorithm 2 describes how a text is prepared for sentence selection model training through pairwise comparison between chosen and rejected text.

Algorithm 1 Extracting ent2ent text spans for classification

ent2enttexts $\leftarrow \emptyset$
for each document d in dataset D **do**
  `all_entitymentions` $\leftarrow$ indices for all entity-mentioning sentences in d
  for each entity e mentioned in d **do**
    selectedsentences $\leftarrow \emptyset$
    `entitymentions` $\leftarrow$ indices for sentences in d mentioning this entity
    for each sentece index i in `entitymentions` **do**
      selectedsentences $\leftarrow$ *selectedsentences* $\cup \{i\}$
      $i \leftarrow i + 1$
      while $i \notin$ `all_entitymentions` **do**
        selectedsentences $\leftarrow$ *selectedsentences* $\cup \{i\}$
        $i \leftarrow i + 1$
      end while
    end for
    ent2enttexts $\leftarrow$ *ent2enttexts* $\cup \{(e, \textit{selectedsentences})\}$
  end for
end for

Algorithm 2 Generating Pairwise Dataset for selection model training

Function: `CreatePrompt`(`context`, `new_sentence`, `relevance`)
  Returns formatted prompt by concatenating inputs within template

for each document d in dataset D **do**
  for each entity e mentioned in d **do**
    `pairwise_dataset` $\leftarrow \emptyset$
    `context` $\leftarrow$ first sentence in d that contains a mention of e
    for each subsequent sentence `new_sentence` **do**
      Assign the annotated relevance label to variable `relevant`
      `chosen_example` $\leftarrow$ `CreatePrompt`(`context`, `new_sentence`, `relevant`)
      `rejected_example` $\leftarrow$ `CreatePrompt`(`context`, `new_sentence`, $\neg$`relevant`)
      `pairwise_dataset` $\leftarrow$ `pairwise_dataset` $\cup \{$(`chosen_example`, `rejected_example`)$\}$
      if `new_sentence` is labeled as relevant **then**
        `context` $\leftarrow$ `context` + `new_sentence`
      end if
    end for
  end for
end for

D More Implementation Details

Our experiments were implemented in Python 3.11. All package versions are available on line,[5] where programming code will be added as well. We made extensive use of the following packages: transformers (Wolf et al., 2019), trl (von Werra et al., 2022), torch (Ansel et al., 2024), numpy (Harris et al., 2020) pandas (Wes McKinney, 2010; pandas development team, 2020), scikit-learn (Pedregosa et al., 2011) and matplotlib (Hunter, 2007).

D.1 Implementation Details for the Pairwise Comparison Training

Norwegian or English prompts A prompt in this context is the standard text shown in Appendix A, which is attached to each text segment and entity. Table 6 shows a slight advantage with keeping the prompt in Norwegian, and this choice is kept for subsequent classification experiments. This is in line with the EuroEval evaluation setup, where prompts are kept in the language of the dataset evaluated on.

One or two negative examples While the selection model has only two labels to distinguish, *Relevant* or *Irrelevant*, for sentiment classification there are three labels to distinguish, *Positive*, *Neutral* or *Negative*. While the chosen text must contain the correct label, there are two options for the negative label. We tested two approaches for this: a) As rejected text, sample from the two incorrect labels according to these labels' distribution in the train set. b) Create two training pairs per entity. Each with the correct label in chosen text, and for the rejected text use the two incorrect labels, one in each pair. This second approach doubles the training set and therefore training time. Table 4 shows that using both incorrect labels gives a slight performance improvement.

Epochs of training While one epoch of training is considered enough for training reward models for RLHF to avoid overfitting (Ouyang et al., 2022; Wang et al., 2024), we check for benefits from training longer, as our models need not respond to the same textual diversity as reward models for RLHF. Table 5 shows that two epochs of training yielded the best evaluation accuracy and the lowest evaluation loss.

Quantization and PEFT: For parameter-efficient fine-tuning (PEFT), we employed Low-Rank Adaptation (LoRA) with a rank parameter of 16 and an alpha scaling factor of 32. The values were chosen according to common practice and initial experiments with higher values that did not yield any improvements. We tested the impact of quantization. The 16-bit precision experiments yielded on average 82.87% accuracy, while the 4-bit quantized counterparts yielded 82.25%.

	Pair per entity	
	1	2
Neg F_1	56.82	58.11
Pos F_1	81.38	82.4
W Avg F_1	80.15	**81.72**
F_1 Std (3 runs)	0.47	0.59

Table 4: Mean F_1 scores in % for training on one pair of chosen and rejected text per entity versus two pairs. The chosen text labels the text with correct label while rejected text labels the text incorrectly. All prompts are Norwegian. Text selection is "Relevant span".

epoch	eval loss	eval accuracy
1	0.4337	0.8472
2	0.3988	**0.8750**
3	1.1496	0.8596
4	1.0391	0.8688

Table 5: Evaluation loss and accuracy for a reward-model trained on preferring the correct sentiment label over an incorrect label.

Text selection	**English**	**Norwegian**
Entire document	83.95%	81.79%
Relevant only	81.48%	83.33%
Relevant span	81.17%	85.19%
Average	82.20%	83.44%

Table 6: Accuracy for predicting ELSA sentiment using pairwise comparison setup with gemma-2-9b. Wrapping the text in a Norwegian prompt yielded best results on average, and we kept this approach for later experiments.

[5] https://anonymous.4open.science/r/nluclassification-FEE5/README.md