LTZGLUE: Luxembourgish General Language Understanding Evaluation

Alistair Plum[1], Felicia Körner[2,3], Anne-Marie Lutgen[1], Laura Bernardy[1], Fred Philippy[1], Emilia Milano[1], Nils Rehlinger[1], Cédric Lothritz[4], Tharindu Ranasinghe[5], Barbara Plank[2,3], Christoph Purschke[1]

[1]University of Luxembourg, Luxembourg, [2]LMU Munich, Germany
[3]Munich Center for Machine Learning, Germany
[4]LIST, Luxembourg, [5]Lancaster University, UK

Correspondence: alistair.plum@uni.lu

Abstract

This paper presents LTZGLUE, the first Natural Language Understanding (NLU) benchmark for Luxembourgish (LTZ) based on the popular GLUE benchmark for English. Although NLU tasks are available for many European languages nowadays, LTZ is one of the official national languages that is often overlooked. We construct new tasks and reuse existing ones to introduce the first official NLU benchmark and accompanying evaluation of encoder models for the language. Our tasks include common natural language processing tasks in binary and multi-class classification settings, including named entity recognition, topic classification, and intent classification. We evaluate various pre-trained language models for LTZ to present an overview of the current capabilities of these models on the LTZ language.

1 Introduction

Language models now support Natural Language Processing (NLP) tasks in more languages than ever before (Park et al., 2021a). Advances since the introduction of the Transformer architecture (Vaswani et al., 2017; Devlin et al., 2019; Tay et al., 2022) have led to substantial performance gains, enabling Large Language Models (LLMs) to achieve state-of-the-art results across a wide range of tasks. As a result, both closed and open-weight LLMs have become the models of choice in NLP and related fields. Owing to their architecture and exposure to large-scale multilingual pre-training data, these models often demonstrate strong performance across many languages. Moreover, their ability to be fine-tuned for a wide variety of downstream tasks enhances their multilingual capabilities.

The perceived support for a wide range of languages has created an unprecedented need for language-specific evaluation of language models. As access to LLMs becomes increasingly widespread, so too does the belief that these models perform well across all languages, an assumption that does not always hold in practice (Zhang et al., 2023). In the interest of transparency and responsible deployment, it is therefore essential to systematically evaluate the Natural Language Understanding (NLU) capabilities of language models (Hettiarachchi et al., 2026).

Small and under-researched languages are particularly difficult to evaluate, as is the case with Luxembourgish (LTZ), the national language of Luxembourg, with around 400k speakers. In English, multiple benchmarks for NLU exist, including GLUE (Wang et al., 2019b) and SuperGLUE (Wang et al., 2019a), and more for other large and small languages alike (Park et al., 2021b; Basile et al., 2023; Hardalov et al., 2023; Shavrina et al., 2020). However, this is not the case for LTZ, which only has a handful of NLU tasks available (Lothritz et al., 2022; Philippy et al., 2024; Plum et al., 2026). As most of these are in the news domain, and the majority of the down-stream tasks comprise less than a thousand instances, model evaluation is not always dependable. Additional factors, such as the ongoing standardisation of the language (Gilles, 2019), vast amounts of variation (Lutgen et al., 2025), and decentralised resources, make it extremely challenging to evaluate LTZ language understanding in language models.

To address these gaps, we introduce LTZGLUE, a general language understanding benchmark for LTZ that includes new and existing NLU tasks. Moreover, we evaluate various pre-trained language models for LTZ to ascertain the current state of the art for the language. Our contributions are:

(1) LTZGLUE: the first unified GLUE benchmark for LTZ, with 8 tasks.[1]
(2) LTZ-E1 (mini/base): 2 new encoder language models for LTZ, which achieve competitive performance when fine-tuned on LTZGLUE.[2]

[1] https://github.com/plumaj/ltzGLUE
[2] https://huggingface.co/instilux

Findings of the Association for Computational Linguistics: ACL 2026, pages 9791–9807

(3) A systematic evaluation of new and existing models for LTZ.

2 Related Work

Work on language understanding has progressed along two largely separate lines: large-scale benchmarking for high-resource languages and emerging efforts to build resources for smaller ones. The first has produced influential frameworks which have shaped evaluation practices for pre-trained models across domains. The second has focused on adapting NLP methods to under-researched languages, where data scarcity and linguistic variation remain major challenges.

The General Language Understanding Evaluation (GLUE) benchmark (Wang et al., 2019b) became a cornerstone of NLU research by consolidating diverse tasks such as sentiment analysis (Socher et al., 2013), textual entailment (Williams et al., 2018), and paraphrase detection (Dolan and Brockett, 2005) into a unified evaluation framework. It established a shared reference point for pre-trained models like BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), and XLNet (Yang et al., 2019), and allowed for systematic comparison across architectures and training regimes. SuperGLUE (Wang et al., 2019a) addressed some shortcomings by introducing more challenging tasks such as COPA (Roemmele et al., 2011), WSC (Levesque et al., 2012), and MultiRC (Khashabi et al., 2018), shifting focus toward commonsense inference and multi-sentence reasoning. While both benchmarks remained English-only, the methodological influence extended widely, shaping later evaluation design in terms of robustness, transparency, and reproducibility.

Multilingual benchmarks in the GLUE style have also been developed, including XGLUE (Liang et al., 2020), XTREME (Hu et al., 2020), and XTREME-R (Ruder et al., 2021), as well as language-specific adaptations such as KLUE (Park et al., 2021b), RussianSuperGLUE (Shavrina et al., 2020), bgGLUE (Hardalov et al., 2023) and SinhalaGlue (Ranasinghe et al., 2025). These efforts highlighted the limits of cross-lingual transfer, reinforcing the need for careful, language-specific evaluation beyond high-resource settings.

2.1 Luxembourgish NLP

LTZ, the focus of this benchmark, is regarded as under-researched, and research is ongoing. Joshi et al. (2020) classify Luxembourgish as one of the "scraping-by" languages: although some unlabeled data exists, meaningful progress will require coordinated efforts to raise awareness and collect labeled datasets, as such resources are currently almost nonexistent. Nevertheless, the first computational tools and corpora were introduced by Adda-Decker et al. (2008), followed by orthographic studies such as contextual n-deletion in transcribed speech (Snoeren et al., 2010). Lavergne et al. (2014) later provided one of the earliest annotated datasets for mixed-language processing. These efforts, though limited, established the foundations for subsequent large-scale data creation and model development.

Since then, the range of tasks has expanded considerably. Work on sentiment analysis (Sirajzade et al., 2020; Gierschek, 2022), orthographic correction (Purschke, 2020), and syntactic annotation (Plum et al., 2024) has broadened the empirical basis for LTZ NLP. Additional datasets have targeted topic classification (Philippy et al., 2024), comment moderation (Ranasinghe et al., 2023), and orthographic normalisation (Lutgen et al., 2025), alongside the generative benchmark set LuxGen (Plum et al., 2025). A manually annotated classification resource was introduced by (Lothritz et al., 2022), which covers a variety of classification tasks.

Model development has explored different transfer and training strategies. LuxGPT (Bernardy, 2022) applied cross-lingual transfer from German, LuxemBERT (Lothritz et al., 2022) used data augmentation with generated samples, and LuxT5 (Plum et al., 2025) extended multilingual pretraining with a balanced language representation.

Yet progress remains uneven across tasks, and existing resources vary widely in size, domain, and annotation quality. No unified benchmark currently exists to evaluate LTZ language understanding consistently, a gap we aim to fill.

3 Tasks

In this section, we introduce the eight tasks for LtzGLUE. The set spans binary and multi-class sentence and token-level classification tasks. Together, these tasks cover a broad spectrum of linguistic and semantic phenomena and provide the first unified benchmark for evaluating LTZ NLP models. See Table 7 in the Appendix for task examples.

Unless stated otherwise, the textual data used across most tasks stems from two main sources: (i)

RTL[3] is the main news broadcaster in Luxembourg, and the only one that is completely in LTZ. RTL provides news articles from the time span 2008 until 2024. (ii) Wikipedia has a growing set of articles in LTZ, around 66k at the time of writing.

3.1 Headline Acceptability

We formulate headline acceptability (HA) as a binary classification task where the model must decide whether a given headline matches the accompanying article body. To construct this dataset, we use RTL news articles. We keep only documents from the twenty most frequent categories. We then filter articles by body length and title length, remove exact duplicate titles, randomly shuffle the remaining instances, and retain a fixed subset of 30k examples. This subset is split equally, with one half serving as the positive class with original headlines, and the other half providing the article bodies for which we assign swapped headlines.

The swapping itself is based on a document level similarity space constructed over the full corpus. We compute TF–IDF representations of the article texts using unigrams and bigrams, an LTZ stopword list, a minimum document frequency of two, and a large feature cap to preserve topical detail. On this representation, we build a cosine nearest neighbour index that returns the 100 most similar articles for any given source. In parallel, we derive a set of content tokens for each article by extracting tokens of length 3+, and removing stopwords. This set is used as a proxy for the article topic. The size of the intersection between two content token sets gives a simple but effective measure of topic overlap.

For every article body in the negative half, we search its nearest neighbours to identify a donor headline, with a minimum 30-day distance so that we avoid headlines tied to the same event. We score candidates by their word overlap, which is computed as the intersection of content-word sets, and use cosine distance as a secondary tiebreaker, stopping early when the overlap reaches at least five tokens. To prevent trivial matches, we reject candidates whose headlines show high positional similarity, measured as the fraction of identical tokens in aligned positions (threshold 0.25). If no neighbour passes all criteria, we fall back to the first viable option, or ultimately to the first non-identical neighbour. We store original and swapped titles, reshuffle, and split into train (20k), development (3k), and test (6k) sets. The resulting negative examples remain topically related but are temporally and structurally mismatched, forcing models to attend to article content rather than surface cues.

3.2 Sentiment Analysis

We formulate the sentiment analysis (SA) task as a classification task where the model has to predict *positive*, *negative*, and *neutral* sentiment. We use articles from RTL, randomly selected from the *commentary* and *letter to the editor* sections. We chose these two specific sections since these pieces could be written by every reader of the journal or expert of a given topic and are usually comments to national or international events. Therefore, there is no required objectivity or impartiality in the writing.

In total, we extract 4,583 sentences, which are then annotated by two native speakers of LTZ. Annotators are instructed to label each sentence, and to use *unsure* only when they would otherwise randomly use the other labels. We calculated Cohen's Kappa at 0.45. For the final set, the annotators agree on a label in cases of label disagreement.

Sentiment	Train	Dev	Test
Neutral	2,339	382	679
Positive	136	43	54
Negative	547	172	193
Total	3,022	597	926

Table 1: Sentiment label distribution per split.

3.3 Linguistic Acceptability

We introduce a linguistic acceptability dataset consisting of four distinct linguistic subtypes, which can either be used as a binary (LA (BINARY)) or multiclass (LA (MULTI)) classification dataset. The sentences are derived from the Luxembourgish Online Dictionary (LOD) and are manipulated using the tags available in the dataset.[4]

The first class interferes with the subject-verb agreement by changing the conjugated form of the main verb or auxiliary verb. The second class similarly modifies the declined form of the adjective and therefore violates the agreement in case, number, and gender. For the third class, we manipulate the syntax by deleting 2-3 random words from the sentence, depending on the length. The last class impacts the orthography, which is achieved

[3] https://rtl.lu

[4] https://lod.lu

by using data provided by *Spellchecker.lu*,[5] a semi-automatic spellchecking website frequently used in Luxembourg. We changed one random word in the sentence by using the least frequent variant in the spellchecker data. The multiclass dataset and binary dataset have a 70-10-20 split, and the distribution is shown in Table 2. The binary dataset distinguishes between correct (1) and incorrect (0), for which the label 0 encompasses the categories Verb, Adj, Syntax and Ortho.

Category	Train	Dev	Test
Verb	2,969	709	405
Adj	2,388	357	673
Syntax	2,327	333	664
Ortho	2,328	333	666
Correct	4666	666	1333
Total	14,678	2,094	4,045

Table 2: Linguistic acceptability categories per split.

3.4 Named Entity Recognition

The JUDGEWEL dataset (Plum et al., 2026) introduces an automatically constructed corpus for named entity recognition (NER) in LTZ, derived from Wikipedia and Wikidata. Using Wikipedia's hyperlink structure, entities are matched to their corresponding Wikidata types and labelled in BIO format. Candidate sentences are selected to maximise diversity, and a set of quality heuristics filters incomplete or overlapping entities. The resulting sentences are then evaluated using LLMs acting as judges, with minimal human verification to calibrate quality thresholds. The final dataset contains roughly 27k sentences across five entity types (see Table 3). Models trained on JUDGEWEL achieve performance comparable to human-annotated data, demonstrating that automatically constructed resources can provide effective supervision.

The NER dataset introduced by Lothritz et al. (2022), by contrast, is a fully human-annotated corpus derived from RTL online news comments. It covers a wider range of text types and registers, including informal and code-mixed writing, and focuses on four primary entity categories (PER, ORG, LOC, GPE). Annotation was conducted manually, yielding a smaller but high-precision dataset.

The two datasets are merged to increase both coverage and domain balance. To ensure compatibility, the tag set is harmonised by merging the GPE and LOC categories into a single location label, while retaining PER, ORG, and MISC unchanged. This unified resource thus aligns the structured reliability of JUDGEWEL with the domain and stylistic breadth of the NER set by (Lothritz et al., 2022), providing a large-scale, multi-domain NER dataset for LTZ. See entity type counts in Table 3.

Entity Type	Train	Dev	Test
PER	11,961	1,587	1,449
ORG	3,323	423	385
LOC	11,701	1,503	1,425
DATE	11,355	1,414	1,523
MISC	511	116	40
Total	38,851	5,043	4,822

Table 3: Entity type counts per split.

3.5 Topic Classification

To construct the news topic classification (TC) dataset, we collected news articles from RTL, which provides content pre-assigned to editorial categories. We applied a series of preprocessing steps to ensure data quality. Specifically, we removed articles identified as non-Luxembourgish by OpenLID (Burchell et al., 2023), as well as those containing fewer than 40 words or more than 400 words. From the available categories, we focused on five principal domains: SPORTS, CULTURE, TECHNOLOGY, BUSINESS, and ANIMALS. Given the substantial over-representation of the SPORTS category, we performed downsampling to mitigate class imbalance. The resulting dataset was split into training, development, and test sets (category distribution is summarized in Table 4).

Category	Train	Dev	Test
Sports	4,000	500	500
Culture	2,984	373	374
Business	1,111	138	140
Technology	1,027	128	129
Animals	810	101	102
Total	9,932	1,240	1,245

Table 4: News topics per split.

3.6 Intent Detection

We constructed a new LTZ dataset for intent detection (ID) by translating the English xSID test and validation datasets (van der Goot et al., 2021). The translations were performed by an LTZ native speaker. In cases of uncertainty, additional native LTZ speakers were consulted. Since LTZ is linguistically closely related to German, the German

[5] https://spellchecker.lu

dataset (van der Goot et al., 2021) occasionally served as a reference point. Since this task is originally intended to be crosslingual, we use the machine translated German training set (van der Goot et al., 2021).

The main challenge in translating the English dataset stems from its register. The source segments consist of user commands for a voice-controlled AI assistant, representing a specialised spoken register for which there is no equivalent reference corpus in LTZ. This register is marked by domain-specific terminology and collocations (e.g., *set an alarm*, *set a reminder*, *add to playlist*), as well as non-standard spelling (e.g., all lowercase, missing punctuation). Due to the lack of LTZ references in this register, it was not possible to systematically verify the translated terminology.

After translating the dataset, we transferred the BIO tags by first using token-level fuzzy matching between the LTZ and the German dataset, followed by manual verification. Table 5 shows the label distribution and size of each data split.

Category	Train	Dev	Test
AddToPlaylist	1,842	1	2
BookRestaurant	1,873	8	11
PlayMusic	1,900	3	5
RateBook	1,856	4	3
SearchCreativeWork	1,854	0	9
SearchScreeningEvent	1,859	6	4
alarm/cancel_alarm	2,069	0	1
alarm/modify_alarm	439	0	0
alarm/set_alarm	4,816	4	4
alarm/show_alarms	1,142	1	0
alarm/snooze_alarm	432	0	0
alarm/time_left_on_alarm	384	0	0
reminder/cancel_reminder	1,151	0	0
reminder/set_reminder	4,743	1	3
reminder/show_reminders	1,006	0	0
weather/checkSunrise	124	0	0
weather/checkSunset	168	0	0
weather/find	15,947	25	24
Total	36,605	53	66

Table 5: Intent distribution per data split.

3.7 Recognizing Textual Entailment

Recognizing Textual Entailment (RTE) (Haim et al., 2006) is a classic NLU task featured in the original GLUE benchmark. Given a pair of texts A and B, the task consists of determining whether A is a logical premise of B. Lothritz et al. (2023) released a machine-translated Luxembourgish version of the dataset using Google Translate. However, due to numerous grammar and vocabulary related mistakes introduced in this process, we set out to improve the quality of the dataset.

Specifically, we first prompted CHATGPT-5.1 to assess and improve the translated sentence pairs unless they were already of very high quality, while explicitly keeping the original meaning to avoid label conflicts (see Appendix 7.4). In addition, we perform two verification steps to make sure that (a) the quality of the improved texts is high enough and (b) that the labels are correct.

To achieve (a), we prompted CHATGPT-5-MINI to judge the texts in the improved data and label their quality as either *low*, *medium*, or *high*, keeping only data rated at least *medium*, removing nearly 25% of the entire dataset (see Appendix 7.5).

For (b), we prompted CHATGPT-5-MINI to verify whether the dataset labels remained correct after the first translation and improvement, outputting *true* or *false* for each sentence pair (see Appendix 7.6). Nearly 10% of the labels were *false*. We found that the quality improvement step often corrected intentional logical contradictions or factual inaccuracies rather than keeping the original semantics. We therefore adjusted the sentences manually such that they corresponded to the ground truth again, while keeping false positives intact.

The filtering reduced between 22 and 28% of instances in the data, resulting in a final dataset of 1,876, 197, and 626 sentence pairs for the training, development, and test set, respectively.

3.8 Summary

Together, the eight tasks in LTZGLUE form a broad and balanced evaluation suite, covering four binary and four multi-class settings, sentence- and document-level inputs, as well as a token-level sequence-labelling task. Despite the low-research status of LTZ, this places LTZGLUE in the same general range as the original English GLUE benchmark, which comprises nine diverse NLU tasks (Wang et al., 2019b). In addition, a substantial proportion of the LTZGLUE tasks are newly created for LTZ rather than direct translations or simple repackaging, allowing the benchmark to reflect phenomena and usage patterns specific to the language.

Compared to recent GLUE-style benchmarks for other non-English languages, LTZGLUE also offers competitive, and in some respects stronger, task coverage. SINHALA-GLUE, introduced as part of the Sinhala encoder-only language models and evaluation suite (Ranasinghe et al., 2025), bundles six datasets into a single NLU bench-

mark, while UINAUIL provides six harmonised Italian NLU tasks drawn from existing shared-task resources (Basile et al., 2023). For Bulgarian, BGGLUE defines nine NLU tasks, combining sequence labelling, document-level classification, and regression over established datasets (Hardalov et al., 2023). In this landscape, supporting eight tasks for LTZ, including token-level NER and several newly constructed text-level tasks, is a strong indicator of the maturity and breadth of the emerging LTZ NLP ecosystem.

4 Models

This section presents the models we trained and evaluated with LTZGLUE. We cover both supervised encoder-based architectures fine-tuned on the benchmark tasks and prompt-based large language models. This design allows us to assess current LTZ NLU performance across fundamentally different modelling paradigms, while maintaining a clear separation between task-specific supervision and general-purpose language understanding.

4.1 LTZ-E1

We train two encoder language models for LTZ: LTZ-E1-mini with 68M and LTZ-E1-base with 110M non-embedding parameters. We closely follow the Ettin recipe (Weller et al., 2026), which is based on MODERNBERT (Warner et al., 2025) (see Appendix 7.2 for detailed settings).

The pre-training set is compiled from a variety of sources of LTZ. A large portion of the data stems from RTL (see Section 3), including news articles (News), transcribed radio interviews (Radio), and user comments (Comments). We also include transcribed podcasts (Podcasts) and transcribed political speeches and debates from the Chambre des Députés (Chamber). In addition, we use 1M sentences from the web crawl of the Leipzig Collection (Web, this excludes RTL), text crawled from LTZ chat rooms (Webchat), a Wikipedia crawl from October 2023 (Wikipedia), and finally, example sentences from the LOD retrieved in March 2024. We filter out sentences containing fewer than three words (as tokenized by whitespace), totalling 11.7M sentences, which corresponds to roughly 233M tokens using our tokenizer. Token counts per source can be found in Table 9 in the Appendix.

4.2 Supervised

We evaluate a set of supervised encoder-based models that explicitly support LTZ, either through direct pre-training or multilingual coverage. As a representative baseline, we include multilingual BERT (MBERT-base) (Devlin et al., 2019), which still remains widely used for multilingual transfer and low-resource evaluation. We additionally evaluate a more recent multilingual BERT (MMBERT-base) variant with updated pre-training data and tokenisation.

To complement these general-purpose multilingual models, we include LUXEMBERT, a language-specific model trained on LTZ data (Lothritz et al., 2022), which provides a stronger inductive bias for the language's lexical and orthographic properties. Finally, we evaluate XLM-RoBERTa (XLM-R-base) (Conneau et al., 2020), a large-scale multilingual model trained on substantially more data and languages than MBERT-base, and commonly used as a strong reference point for multilingual NLU.

4.3 Unsupervised

In addition to supervised encoder-based models, we evaluate a set of LLMs in a prompt-based zero-shot setting. This group includes QWEN3-235B, LLAMA-3.3, GEMMA-3-27B, and GPT-5-MINI, which represent a range of model sizes, training regimes, and degrees of multilingual coverage. None of these models are fine-tuned on LTZGLUE, although some of the text data (RTL, Wikipedia) is very likely to have been processed during training. The models are evaluated using prompts that describe each task, allowing us to assess their ability to generalise to LTZ without task-specific supervision (see Appendix 7.7 and 7.8 for further details). We did not use a Multiple Choice Question Answering (MCQA)-setup, but provided the labels that should be used as output.

This evaluation setting reflects the growing use of LLMs as general-purpose language understanding systems, particularly in scenarios where annotated data is scarce or unavailable. However, prompt-based evaluation introduces additional sources of variability, including prompt sensitivity and differences in instruction-following behaviour across models. As a result, performance should be interpreted as indicative rather than directly comparable to supervised results. Nevertheless, including these models provides a complementary perspective on the current capabilities of large-scale multilingual and instruction-tuned systems for LTZ NLU.

5 Evaluation

We evaluate the models described in Section 4 across all tasks in the benchmark. For encoder-based models, results are reported as averages over multiple runs (see Appendix 7.2 for more details). Prompted LLMs do not always produce well-formed outputs and may return an incorrect number of predictions for a given task; such outputs are discarded prior to evaluation. All reported scores are computed on the remaining valid predictions per model. For the supervised models, since the linguistic acceptability and sentiment analysis datasets are highly imbalanced, when fine-tuning on these tasks we use class-balanced loss based on effective size (Cui et al., 2019) with a beta of 0.99. Table 6 shows F_1 scores for all models across all tasks (see Appendix 7.9 for full results).

Overall, our evaluation reveals consistent trends across tasks. Encoder-based models perform strongly across most settings, particularly on structurally complex and label-sensitive tasks, confirming findings from prior work on multilingual and low-resource NLU (Wu and Dredze, 2019; Conneau et al., 2020). Prompted large language models, by contrast, show more variable behaviour and perform competitively only on a set of semantically coarse-grained tasks, consistent with recent observations that prompting alone is often insufficient for strong performance on structured NLU tasks (Wei et al., 2022; Liu et al., 2023).

HA Results on the headline acceptability task show substantial variation across encoder-based models, both in absolute performance and in stability. MMBERT-base achieves the highest mean F_1 score with comparatively low variance, indicating robust performance across runs. In contrast, MBERT-base reaches competitive average performance but exhibits very high standard deviation, suggesting sensitivity to initialisation and training dynamics. The LTZ-specific encoders, LUXEMBERT and LTZ-E1-mini, perform moderately well but remain clearly below MMBERT-base, while LTZ-E1-base and XLM-R-base lag behind in both performance and consistency. Among the prompted LLMs, GPT achieves the strongest single-run result, approaching the performance of MMBERT-base, followed by QWEN. GEMMA and LLAMA perform noticeably worse. However, these results are based on a single evaluation and therefore do not allow conclusions about stability.

SA On the sentiment analysis task, differences between encoder models are smaller than for HA, though consistent trends remain. LUXEMBERT achieves the highest mean F_1 score with low variance, followed closely by MMBERT-base, although with considerable variance across runs. LTZ-E1-base, LTZ-E1-mini, and MBERT-base perform worse and exhibit increased variability, while XLM-R-base performs weakest among the encoders. Prompted LLMs perform roughly equal to the fine-tuned encoders in this setting. GPT achieves the strongest single-run F_1 score among the LLMs, marginally outperforming GEMMA.

LA (BINARY) For the binary linguistic acceptability task, all encoder models achieve relatively high F_1 scores, with MMBERT-base and LTZ-E1-mini performing best and showing limited variance across runs. LUXEMBERT also performs competitively, suggesting that coarse-grained acceptability judgments are well captured by language-specific representations. In contrast, LTZ-E1-base exhibits notably higher variance despite a reasonable mean score, complicating direct comparison. XLM-R-base performs substantially worse than the other encoders. Prompted LLMs trail the encoder models by a clear margin: GPT achieves the highest single-run performance, followed by QWEN, while GEMMA and LLAMA perform considerably worse. These results indicate that even binary acceptability judgments benefit from task-specific supervision.

LA (MULTI) The multi-class linguistic acceptability task proves considerably more challenging and reveals larger performance differences. Among the encoders, MMBERT-base again leads, combining strong performance with moderate variance. LUXEMBERT and LTZ-E1-mini follow closely but show increased instability across runs, while LTZ-E1-base exhibits particularly high standard deviation, suggesting difficulty in consistently modeling fine-grained acceptability distinctions. MBERT-base performs slightly worse than the LTZ-specific encoders, and XLM-R-base remains the weakest. Prompted LLM performance drops sharply in this setting: although GPT achieves the highest single-run score, all LLMs perform well below the encoders, with LLAMA approaching chance-level behaviour. This highlights the difficulty of multi-class linguistic judgments without supervised adaptation.

NER Results on the named entity recognition task show strong performance across all encoder-

Model	HA	SA	LAB	LAM	NER	TC	ID	RTE
LxBRT	66.37 $_{\pm 0.00}$	**58.66** $_{\pm 0.73}$	89.17 $_{\pm 0.18}$	87.96 $_{\pm 0.71}$	87.43 $_{\pm 1.22}$	98.68 $_{\pm 0.17}$	**91.71** $_{\pm 0.11}$	46.51 $_{\pm 5.16}$
mBRT	77.91 $_{\pm 10.26}$	41.25 $_{\pm 4.87}$	81.26 $_{\pm 0.56}$	81.20 $_{\pm 1.02}$	83.06 $_{\pm 2.06}$	97.80 $_{\pm 0.67}$	60.65 $_{\pm 4.71}$	42.57 $_{\pm 6.13}$
ltzE1b	62.81 $_{\pm 4.98}$	46.59 $_{\pm 5.88}$	83.17 $_{\pm 8.38}$	78.63 $_{\pm 11.55}$	88.01 $_{\pm 1.07}$	**98.95** $_{\pm 0.36}$	73.32 $_{\pm 11.66}$	39.38 $_{\pm 6.04}$
ltzE1m	72.69 $_{\pm 1.33}$	45.39 $_{\pm 6.79}$	89.31 $_{\pm 2.89}$	86.62 $_{\pm 4.64}$	88.95 $_{\pm 0.61}$	98.50 $_{\pm 0.23}$	80.13 $_{\pm 3.00}$	45.35 $_{\pm 5.71}$
mmBRT	**85.59** $_{\pm 1.61}$	53.37 $_{\pm 4.47}$	**89.97** $_{\pm 0.09}$	**88.83** $_{\pm 1.23}$	**90.41** $_{\pm 0.55}$	98.92 $_{\pm 0.28}$	78.26 $_{\pm 7.22}$	**52.81** $_{\pm 3.01}$
XLMR	72.09 $_{\pm 9.90}$	36.40 $_{\pm 0.57}$	70.25 $_{\pm 4.55}$	73.40 $_{\pm 5.61}$	79.58 $_{\pm 2.77}$	97.24 $_{\pm 0.50}$	62.75 $_{\pm 8.76}$	35.24 $_{\pm 7.73}$
GPT	**88.88**	**56.44**	**75.24**	**51.45**	67.15	**89.27**	**38.26**	**88.51**
Qwen	86.08	51.63	67.77	39.69	**70.73**	88.37	34.90	84.17
Gemma	80.53	55.60	58.68	43.28	48.44	48.67	7.25	73.12
Llama	77.66	45.84	41.44	11.17	50.50	12.47	35.75	72.01

Table 6: **Test F_1 scores across all ltzGLUE tasks.** Encoder results are averaged over three runs with standard deviations as subscripts. Prompted LLMs were evaluated once; we report macro-F_1 only.

based models, with comparatively small differences in mean F_1 scores. mmBERT-base achieves the highest score with very low variance, indicating both high accuracy and stability. ltz-E1-mini and ltz-E1-base perform similarly well, while luxembert remains competitive but slightly behind. mBERT-base and XLM-R-base trail the other encoders. In contrast, prompted LLMs perform substantially worse than all fine-tuned encoders. Qwen achieves the strongest LLM performance, followed by GPT, but both remain far below the encoder models, underscoring the importance of token-level supervision for this task.

TC The topic classification task emerges as the easiest overall. All encoder models achieve very high F_1 scores with extremely low variance, indicating a stable and largely language-agnostic task. Differences between encoders are minimal, with ltz-E1-base and mmBERT-base marginally outperforming the others. Prompted LLMs perform competitively in this setting: GPT and Qwen approach encoder-level performance in a single run. However, Gemma and especially Llama perform poorly, suggesting that strong topic classification performance is not guaranteed without either fine-tuning or robust multilingual pre-training.

ID Results on the intent detection task reveal a clear separation between models. Among the encoders, luxembert achieves the strongest performance with very low variance, highlighting the benefit of language-specific pre-training. ltz-E1-mini and mmBERT-base perform well but exhibit higher variability, while ltz-E1-base shows both lower mean performance and substantial deviation across runs. mBERT-base and XLM-R-base perform considerably worse. Prompted LLMs struggle substantially with this task: all LLMs achieve low F_1 scores, with Gemma performing particularly poorly. This suggests that intent classification in LTZ relies on supervised task-specific training.

RTE The recognising textual entailment task is the most challenging overall, with low F_1 scores and high variance across encoder models. mmBERT-base clearly outperforms the other encoders, achieving the highest mean performance with relatively controlled variance. luxembert and ltz-E1-mini follow but show notable instability, while ltz-E1-base and XLM-R-base perform poorly, making reliable inference difficult. Prompted LLMs perform relatively well in comparison to most encoders: GPT and Qwen achieve strong single-run F_1 scores, exceeding all encoder models except mmBERT-base. This suggests that entailment reasoning may benefit from broader semantic representations encoded in large generative models, although the lack of variance estimates warrants caution.

Overall Taken together, the results reveal three overall patterns. First, mmBERT-base consistently achieves the strongest or near-strongest performance across almost all tasks, combining high mean F_1 scores with comparatively low variance, suggesting that broad multilingual pre-training with sufficient LTZ exposure yields stable and transferable representations. Second, LTZ-specific encoders such as luxembert and ltz-E1-mini are particularly competitive on lexically grounded or task-specific settings (e.g., intent detection and acceptability), but exhibit greater instability on structurally complex inference tasks such as multi-class acceptability and textual entailment. Third, prompted LLMs display substantially more task-dependent behaviour and generally underperform fine-tuned encoders, except on semantically coarse-

grained tasks such as topic classification. Overall, tasks requiring structured prediction or fine-grained linguistic discrimination benefit strongly from supervised fine-tuning, underscoring the importance of task-specific adaptation in LTZ NLU.

6 Conclusion

This paper makes two central contributions to LTZ NLU. First, we introduce a new benchmark that provides the first comprehensive GLUE-style evaluation suite for LTZ. Second, we present a systematic evaluation of encoder-based models and prompted large language models across all tasks, offering concrete guidance on model choice in such a low-resource setting.

The construction of the dataset required a deliberately resource-conscious approach. In the absence of large, task-diverse annotated resources, we combine the reuse of existing datasets with the targeted annotation of new data, carefully aligning annotation schemes across tasks, and using large language models as auxiliary tools. This strategy enables the creation of a benchmark without relying on large-scale annotation efforts. Moreover, our evaluation reveals a clear and consistent pattern: fine-tuned encoder-based models outperform prompted large language models on structurally complex tasks. Prompted large language models perform competitively only on a limited subset of semantically coarse-grained tasks, most notably topic classification and textual entailment. However, prompt-based approaches show limited consistency, as outputs can vary substantially across runs and prompt formulations, making prompting alone an unreliable substitute for fine-tuned models in low-resource NLU settings.

Overall, our findings indicate that, despite rapid progress in generative modelling, encoder-based approaches remain the recommended solution for most LTZ NLU tasks. Nonetheless, LLMs play an important complementary role, both as practical tools during dataset construction and as competitive baselines for selected tasks. By releasing both the dataset and the accompanying evaluation, we aim to support future research on LTZ and to encourage similarly resource-conscious benchmarking efforts for other low-resource languages.

Acknowledgments

We would like to thank the student assistants for their annotation work.

This work is supported by the LLMs4EU project, funded by the European Union through the Digital Europe Programme (DIGITAL) under the grant agreement 10119847. FK and BP are supported by the ERC Consolidator Grant DIALECT 101043235.

Limitations

While LTZGLUE provides the first systematic benchmark for LTZ NLU, the dataset remains constrained by the availability and scope of existing resources. Several tasks rely on relatively small or domain-specific corpora, which limits the ecological validity of the results and restricts the range of linguistic phenomena covered. We therefore view this release as a foundation rather than a comprehensive evaluation suite. In addition, some of the data sources used in this benchmark may already be included, in whole or in part, in the pre-training corpora of the large language models evaluated in this work. While the exact composition of proprietary pre-training datasets is typically not fully disclosed, this potential overlap cannot be entirely ruled out and may inflate performance estimates. We explicitly acknowledge this possibility in the interest of transparency and encourage future evaluations on carefully controlled or newly collected data where feasible.

Coverage across domains, registers, and demographic varieties may also be limited. LTZ displays substantial orthographic and sociolinguistic variation, yet most data sources reflect formal writing or institutional usage and therefore do not fully represent informal and multilingual contexts. Models evaluated on LTZGLUE may therefore overestimate their robustness in real-world applications.

Although we draw on established GLUE-style tasks, some annotation decisions and class distributions are necessarily influenced by resource constraints. Certain tasks exhibit label imbalance or rely on automatic preprocessing, which may introduce biases that we cannot fully quantify. These constraints reflect the current state of LTZ NLP and point to the need for continued data creation and evaluation work.

Ethical Considerations

The datasets included in this work are derived from publicly accessible sources that permit research use, and all preprocessing avoids the inclusion of directly identifying personal information. The data

is licensed under the Creative Commons Attribution (CC BY) licence.

However, some tasks draw on data originally produced in institutional or media contexts, which may reflect societal biases in representation. These patterns can influence model behaviour and should be considered when deploying systems trained on LTZGLUE.

LTZ is a small language community, and linguistic data often originate from a limited set of public domains. As a result, models may reproduce dominant norms while under-representing regional, sociolectal, or multilingual practices. We therefore caution against using benchmark performance as evidence of cultural or demographic coverage.

Finally, although no sensitive content is intentionally included, automated filtering and preprocessing cannot guarantee the complete removal of harmful or offensive material. Researchers using LTZGLUE are encouraged to inspect task-specific subsets and consider downstream implications, especially in public-facing settings.

References

Martine Adda-Decker, Thomas Pellegrini, Eric Bilinski, and Gilles Adda. 2008. Developments of "Lëtzebuergesch" Resources for Automatic Speech Processing and Linguistic Studies. In *Proceedings of LREC*.

Valerio Basile, Livio Bioglio, Alessio Bosca, Cristina Bosco, and Viviana Patti. 2023. UINAUIL: A Unified Benchmark for Italian Natural Language Understanding. In *Proceedings of ACL*.

Laura Bernardy. 2022. A Luxembourgish GPT-2 Approach Based on Transfer Learning. Master's thesis, University of Trier.

Sidney Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, Usvsn Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. 2022. GPT-NeoX-20B: An Open-Source Autoregressive Language Model. In *Proceedings of BigScience*.

Laurie Burchell, Alexandra Birch, Nikolay Bogoychev, and Kenneth Heafield. 2023. An Open Dataset and Model for Language Identification. In *Proceedings of ACL*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of ACL*.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. 2019. Class-balanced loss based on effective number of samples. In *Proceedings of CVPR*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL-HLT*.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of IWP*.

Daniela Gierschek. 2022. *Detection of Sentiment in Luxembourgish User Comments*. Ph.D. thesis, University of Luxembourg.

Peter Gilles. 2019. *39. Komplexe Überdachung II: Luxemburg. Die Genese Einer Neuen Nationalsprache*, pages 1039–1060. De Gruyter Mouton.

R Bar Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The second pascal recognising textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*.

Momchil Hardalov, Todor Mihaylov, Kiril Simov, and Preslav Nakov. 2023. BgGLUE: A Bulgarian General Language Understanding Evaluation Benchmark. In *Proceedings of RANLP*.

Hansi Hettiarachchi, Tharindu Ranasinghe, Alistair Plum, Paul Rayson, Ruslan Mitkov, Mohamed Medhat Gaber, Damith Premasiri, Fiona Anting Tan, and Lasitha Uyangodage. 2026. Overview of the second workshop on language models for low-resource languages (LoResLM 2026). In *Proceedings of the LoResLM*.

J. Edward Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, Melvin Johnson, et al. 2020. XTREME: A Massively Multilingual Multitask Benchmark for Evaluating Cross-lingual Generalization. In *Proceedings of ICML*.

Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, Xinrong Zhang, Zheng Leng Thai, Kaihuo Zhang, Chongyi Wang, Yuan Yao, Chenyang Zhao, Jie Zhou, Jie Cai, Zhongwu Zhai, Ning Ding, Chao Jia, Guoyang Zeng, Dahai Li, Zhiyuan Liu, and Maosong Sun. 2024. Minicpm: Unveiling the potential of small language models with scalable training strategies.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The State and Fate of Linguistic Diversity and Inclusion in the NLP World. In *Proceedings of ACL*.

Daniel Khashabi, Snigdha Chaturvedi, Michael Roth, Shyam Upadhyay, and Dan Roth. 2018. Looking Beyond the Surface: A Challenge Set for Reading

Comprehension over Multiple Sentences. In *Proceedings of NAACL-HLT*.

Thomas Lavergne, Gilles Adda, Martine Adda-Decker, and Lori Lamel. 2014. Automatic language identity tagging on word and sentence-level in multilingual text sources: a case-study on Luxembourgish. In *Proceedings of LREC*.

Hector Levesque, Ernest Davis, and Leora Morgenstern. 2012. The Winograd Schema Challenge. In *Proceedings of KR*.

Yaobo Liang, Yeyun Gong, Weizhen Bian, Nan Jiang, Guoqing Xie, Ruize Lin, Jiuhai Feng, Ruochen Xu, Wenjie Wang, Zhifang Chen, et al. 2020. XGLUE: A New Benchmark Dataset for Cross-lingual Pretraining, Understanding and Generation. In *Proceedings of EMNLP*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Computing Surveys*, 55(9).

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. In *arXiv preprint arXiv:1907.11692*.

Cedric Lothritz, Saad Ezzini, Christoph Purschke, Tegawendé François D Assise Bissyande, Jacques Klein, Isabella Olariu, Andrey Boytsov, Clement Lefebvre, and Anne Goujon. 2023. Comparing Pre-Training Schemes for Luxembourgish BERT Models. In *Proceedings of KONVENS*.

Cedric Lothritz, Bertrand Lebichot, Kevin Allix, Lisa Veiber, Tegawende Bissyande, Jacques Klein, Andrey Boytsov, Clément Lefebvre, and Anne Goujon. 2022. LuxemBERT: Simple and Practical Data Augmentation in Language Model Pre-Training for Luxembourgish. In *Proceedings of LREC*.

Anne-Marie Lutgen, Alistair Plum, Christoph Purschke, and Barbara Plank. 2025. Neural Text Normalization for Luxembourgish Using Real-Life Variation Data. In *Proceedings of VarDial*.

Hyunji Hayley Park, Katherine J. Zhang, Coleman Haley, Kenneth Steimel, Han Liu, and Lane Schwartz. 2021a. Morphology Matters: A Multilingual Language Modeling Analysis. *TACL*.

Sungjoon Park, Joongbo Shin, Yekyung Lee, Jaehyung Lee, Kichang Lee, Kyunghyun Lee, Sang-Woo Kim, and Heuiseok Kim. 2021b. KLUE: Korean Language Understanding Evaluation. In *Proceedings of NAACL-HLT*.

Fred Philippy, Shohreh Haddadan, and Siwen Guo. 2024. Forget NLI, Use a Dictionary: Zero-Shot Topic Classification for Low-Resource Languages with Application to Luxembourgish. In *Proceedings of SIGUL*.

Alistair Plum, Laura Bernardy, and Tharindu Ranasinghe. 2026. Do LLMs Judge Distantly Supervised Named Entity Labels Well? Constructing the JudgeWEL Dataset. In *Proceedings of LREC*.

Alistair Plum, Caroline Döhmer, Emilia Milano, Anne-Marie Lutgen, and Christoph Purschke. 2024. LuxBank: The First Universal Dependency Treebank for Luxembourgish. In *Proceedings of TLT*.

Alistair Plum, Tharindu Ranasinghe, and Christoph Purschke. 2025. Text Generation Models for Luxembourgish with Limited Data: A Balanced Multilingual Strategy. In *Proceedings of VarDial*.

Christoph Purschke. 2020. Attitudes Toward Multilingualism in Luxembourg. A Comparative Analysis of Online News Comments and Crowdsourced Questionnaire Data. *Frontiers in AI*, 3.

Tharindu Ranasinghe, Hansi Hettiarachchi, Nadeesha Chathurangi Naradde Vidana Pathirana, Damith Premasiri, Lasitha Uyangodage, Isuri Nanomi Arachchige, Alistair Plum, Paul Rayson, and Ruslan Mitkov. 2025. Sinhala Encoder-only Language Models and Evaluation. In *Proceedings of ACL*.

Tharindu Ranasinghe, Alistair Plum, Christoph Purschke, and Marcos Zampieri. 2023. Publish or Hold? Automatic Comment Moderation in Luxembourgish News Articles. In *Proceedings of RANLP*.

Melissa Roemmele, Cosmin A. Bejan, and Andrew S. Gordon. 2011. Choice of Plausible Alternatives: An Evaluation of Commonsense Causal Reasoning. In *AAAI Spring Symposium on Logical Formalizations of Commonsense Reasoning*.

Sebastian Ruder, Noah Constant, Jan Botha, Aditya Siddhant, Orhan Firat, Jinlan Fu, Pengfei Liu, Junjie Hu, Dan Garrette, Graham Neubig, and Melvin Johnson. 2021. XTREME-R: Towards More Challenging and Nuanced Multilingual Evaluation. In *Proceedings of EMNLP*.

Tatiana Shavrina, Denis Shevelev, Alena Fenogenova, Irina Nikishina, et al. 2020. RussianSuperGLUE: A Russian Language Understanding Evaluation Benchmark. In *Proceedings of EMNLP*.

Joshgun Sirajzade, Daniela Gierschek, and Christoph Schommer. 2020. An Annotation Framework for Luxembourgish Sentiment Analysis. In *Proceedings of SLTU-CCUR*.

Natalie D. Snoeren, Martine Adda-Decker, and Gilles Adda. 2010. The Study of Writing Variants in an Under-resourced Language: Some Evidence from Mobile N-Deletion in Luxembourgish. In *Proceedings of LREC*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of EMNLP*.

Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. 2022. Efficient Transformers: A Survey. *ACM Computing Surveys*, 55(6).

Rob van der Goot, Ibrahim Sharaf, Aizhan Imankulova, Ahmet Üstün, Marija Stepanović, Alan Ramponi, Siti Oryza Khairunnisa, Mamoru Komachi, and Barbara Plank. 2021. From Masked Language Modeling to Translation: Non-English Auxiliary Tasks Improve Zero-shot Spoken Language Understanding. In *Proceedings of NAACL-HLT*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of NeurIPS*.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019a. SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems. In *Proceedings of NeurIPS*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019b. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *Proceedings of ICLR*.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Griffin Thomas Adams, Jeremy Howard, and Iacopo Poli. 2025. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. In *Proceedings of ACL*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Proceedings of NeurIPS*.

Orion Weller, Kathryn Ricci, Marc Marone, Antoine Chaffin, Dawn Lawrie, and Benjamin Van Durme. 2026. Seq vs Seq: An Open Suite of Paired Encoders and Decoders. In *Proceedings of ICLR*.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2018. A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In *Proceedings of NAACL-HLT*.

Mitchell Wortsman, Tim Dettmers, Luke Zettlemoyer, Ari Morcos, Ali Farhadi, and Ludwig Schmidt. 2023. Stable and low-precision training for large-scale vision-language models. In *Proccedings of NeurIPS*.

Shijie Wu and Mark Dredze. 2019. Beto, Bentz, Becas: The Surprising Cross-Lingual Effectiveness of BERT. In *Proceedings of EMNLP-IJCNLP*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In *Proceedings of NeurIPS*.

Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. 2022. Scaling Vision Transformers. In *Proceedings of CVPR*.

Xiang Zhang, Senyu Li, Bradley Hauer, Ning Shi, and Grzegorz Kondrak. 2023. Don't Trust ChatGPT when Your Question is not in English: A Study of Multilingual Abilities and Types of LLMs. In *Proceedings of EMNLP*.

7 Appendix

7.1 `ltzGLUE` Task Examples

For demonstration purposes, we present an example for each task in `ltzGLUE` in Table 7. The examples are intended to illustrate the task formulations and typical model inputs and outputs.

7.2 LTZ-E1 Training Details

Model Architecture We follow the Ettin recipe (Weller et al., 2026), based on ModernBERT (Warner et al., 2025), for training hyperpameters and model architecture. We train two sizes of LTZ-E1 models, mini and base, with 68M and 110M non-embedding parameters, respectively. Common pre-training configuration parameters for both sizes can be found in Table 8. LTZ-E1-mini has 19 hidden layers, a hidden size of 512, an intermediate size of 768, and 8 attention heads, whereas LTZ-E1-base has 22 hidden layers, a hidden size of 768, an intermediate size of 1152, and 12 attention heads.

Both models share a GPTNeoXTokenizerFast tokenizer (Black et al., 2022) [6], a BPE-based tokenizer, which we train on the entire pre-training set, using a minimum frequency of two and a vocabulary size of 50,368.

Training Details We use a constant batch size of 1024 packed sequences, where both models have a max sequence length of 1024. We follow ModernBERT (Warner et al., 2025) and Ettin (Weller et al., 2026) in using the Warmup-Stable-Decay (WSD) scheduler (Zhai et al., 2022; Hu et al., 2024), though we use a shorter warmup and decay phase of 500 batches each, due to our smaller pre-training dataset size and larger number of epochs (10 vs. one). Again following ModernBERT and Ettin's recipe, we use the StableAdamW optimizer (Wortsman et al., 2023), with a peak learning rate of 3e-3 with a weight decay of 3e-4 for LTZ-E1-mini and 8e-4 with a weight decay of 1e-5 for LTZ-E1-base. As our pre-training set is small, we

[6] https://huggingface.co/docs/transformers/v4.57.3/en/model_doc/gpt_neox

Task	Content
HA	**Input:** *Headline:* Paschtouer krut 2.500 Euro vun Onéierlechen ofgeknäppt *(Priest robbed of 2500€ by criminals)* *Article:* Déi lescht Wochen hätt een ëmmer méi dacks Kollekte gemellt kritt, déi awer net vun Handicap International an Optrag gi goufen... *(In the past few weeks, an increasing number of charity collections were reported, which were not commissioned by Handicap International)* **Output:** correct
SA	**Input:** Et war den Houfert vun e puer Generatioune Lëtzebuerger. *(It was the pride of a couple of Luxembourgish generations)* **Output:** positive
LAB	**Input:** Dat Bild do ass eng plomper Fälschung! *(That painting is an amateurish forgery!)* **Output:** incorrect
LAM	**Input:** Ech schonn dräimol Usbekistan. *I (have) been (to) Uzbekistan three times.* **Output:** syntax
NER	**Input:** De Mark Cavendish konnt de Massesprint op der Arrivée fir sech entscheeden. *(Mark Cavendish was able to win the mass sprint on the finish line.)* **Output:** "O", "B-PER", "I-PER", "O", "O", "O", "O", "O", "O", "O", "O", "O", "O"
TC	**Input:** Kënnt Dir Iech nach un de Meister Ede a säi Pumuckl erënneren?... *(Do you remember Master Ede and his Pumuckl?)* **Output:** culture
ID	**Input:** Reent et nächst Woch? *(Will it rain next week?)* **Output:** weather/find
RTE	**Input:** IBM huet Geschäftsgeheimnisser geklaut, fir zwee vu senge Programmer ze kopéieren: File-AID, en Dateiemanager, an Abend-AID, e Programm, deen de Benotzer hëlleft, d'Quell vu Feeler ze fannen. *(IBM has stolen confidential business files to copy two programs: File-AID, a data manager, and Abend-AID, a program to detect sources of mistakes.)* Geschäftsgeheimnisser goufen geklaut. *(Business secrets were stolen.)* **Output:** true

Table 7: **Input–output examples for each task.** LTZ inputs are shown with English translations for clarity.

train each model for 10 epochs, following Lothritz et al. (2022).

Computational Resources We use a 20GB MIG partition of an NVIDIA A100-SXM4-80GB to pre-train each model, taking 47 hours for LTZ-E1-mini and 76 hours for LTZ-E1-base. However, we note that compute times were negatively impacted by concurrent jobs on the server cluster with suboptimal CPU thread management.

Pre-training Data Breakdown We show pre-training data token counts per source in Table 9, where sources (described in Section 4.1) are: RTL news articles (News), RTL transcribed radio interviews (Radio), RTL user comments (Comments), transcribed podcasts (Podcasts), transcribed political speeches and debates from the Chambre des Députés (Chamber), 1M sentences from the web crawl of the Leipzig Collection (Web), text from Luxembourgish chat rooms (Webchat), a Wikipedia crawl (Wikipedia), and examples from the Luxembourgish Online Dictionary (LOD).

7.3 Hyperparameter Sweeps

Though we do not aim to optimise performance in our evaluation, we conduct basic hyperparameter sweeps for each model and task combination in order to provide a fairer comparison across models. We use Weights & Biases version 0.23.1. to conduct the sweeps. For each model and task combination, we select the best hyperparameters based on the validation set, and use those parameters to finetune two additional models with differing seeds, resulting in three runs. In order to reduce the computational demand of the sweeps, we use Bayesian search with early stopping after three iterations, and cap each sweep at 30 runs, for 1,440 total runs across all models and tasks (and an additional 96 to finetune the two additional seeds). For each sweep we use the same hyperparameter ranges, shown in Table 10. Best values for each sweep are shown in Table 11. However, we note again that these ranges were kept simple to keep sweeps computationally feasible, thus, these values should not be seen as optimal hyperparameters.

Parameter	Value
Vocabulary Size	50,368
Max Sequence Length	1024
Tokenizer Arch.	GPTNeoXTokenizerFast
Attention Layer	RoPE
Attention Dropout	0.0
Attention Output Bias	false
Attention Output Dropout	0.1
Attention QKV Bias	false
Transformer Layer	prenorm
Embedding Dropout	0.0
Embedding Norm	true
Final Norm	true
Skip First PreNorm	true
Embedding Layer	sans_pos
MLP Dropout	0.0
MLP Input Bias	false
MLP Layer Type	GLU
MLP Output Bias	false
MLM Probability	0.3
Normalization	LayerNorm
Norm Epsilon	1e-5
Norm Bias	false
Hidden Activation	GELU
Head Pred Activation	GELU
Activation Function	GELU
Padding	unpadded
Rotary Embedding Base	10,000.0
Rotary Embedding Interleaved	false
Allow Embedding Resizing	true
Sliding Window	128
Global Attn. every N Layers	3
Unpad Embeddings	true
Masked Prediction	true

Table 8: Common pre-training configuration parameters across both LTZ-E1 models (mini and base).

Computational Resources We use several 20GB MIG partitions of NVIDIA A100-SXM4-80GB GPUs to conduct the sweeps. Depending on model and task dataset size, multiple runs were conducted in parallel on each partition, totalling 59 days of compute, which includes fine-tuning the additional seeds, as well as evaluation on the validation and test sets.

7.4 Prompt to Improve Quality of RTE Task

```
You are an expert for the Luxembourgish
language. I am giving you a sentence in
Luxembourgish. You have to judge its
quality and improve it while keeping
the meaning intact. As output, write only
the improved sentence or the original
sentence if it is of very high quality.
```

7.5 Prompt to Judge the Quality of Improved RTE Dataset

```
You are an expert for the Luxembourgish
language. I am giving you two texts in
Luxembourgish. You have to judge their
quality. As output, simply write 'low',
'medium' or 'high' depending on the
quality of both sentences, nothing else.
```

Source	Tokens (M)
Wikipedia	11.2
LOD	0.7
RTL Radio	24.9
RTL News	51.6
RTL Comments	77.7
Chamber	23.5
Web	22.9
Webchat	20.8
Total	233.4

Table 9: Token counts (M) per source for pretraining data of LTZ-E1.

Parameter	Values
Learning Rate	{1e-5, 3e-5, 5e-5, 8e-5}
Batch Size	{8, 16}
Epochs	{2, 5}
Weight Decay	{0.0, 0.01}
Warmup Ratio	{0.0, 0.1}

Table 10: Hyperparameter sweep ranges used for all task and model combinations.

7.6 Prompt to Verify the Labels of Improved RTE Dataset

```
You are an expert for the Luxembourgish
language. I am giving you two texts
TEXT1 and TEXT2 in Luxembourgish as well
as a LABEL where 1 means that TEXT1
logically entails TEXT2 while 0 means the
opposite. You have to check if the labels
are correct. As output, simply write 'true'
if the label is the correct one or 'false'
if the label is incorrect.
```

7.7 Main prompt for zero-shot testing of LLMs

```
You are a classification and text-processing
model specialized in NLP tasks for
Luxembourgish (lb).
Follow ALL rules strictly:
1. Respond ONLY in valid JSON.
2. Do NOT add explanations, comments or text
outside of JSON.
3. Use field: "output": <model_answer>.
4. Use field: "task": "<task_name>".
5. Use field: "input": "<input example text>".
6. Predict only the requested outputs and
```

```
label(s) in the given formats.
7. If determined labels are 0 and 1 then 0
is used for False, 1 is used for True.
Here is the NLP task definition:
TASK: {task_name}
DESCRIPTION: {task_description}
```

7.8 Task descriptions for zero-shot testing of LLMs

```
headline_classification:
Decide if the given title/headline fits the
text.
Output True or False.

sentiment_analysis:
Classify sentiment of the text.
Allowed labels: positive, neutral, negative.

linguistic_acceptability_binary:
Decide whether the sentence is linguis-
tically acceptable in Luxembourgish.
Output: 0 or 1.

linguistic_acceptability_multilabel:
Detect if the sentence is correct or if
some element is wrong.
If the sentence is correct,
Output: correct.
If it is not, Output the label referencing
the wrong element:
syntax, verb, ortho or adj.

ner:
Perform Named Entity Recognition
on the given sequence of sentence
tokens.
Output tags as lists of ner_tags.
Allowed Tags: O, B-LOC, I-LOC,
B-PER,I-PER, B-DATE, I-DATE,B-ORG,
I-ORG, B-MISC, I-MISC.

topic_classification:
Classify topic of the document
by title and text.
Allowed category_names: sports,
animals, business, culture, technology.

slot_intent_detection:
Detect the intent for
the text given.
Allowed intents:
reminder/show_reminders,
weather/find\,
reminder/set_reminder,
reminder/cancel_reminder,
alarm/snooze_alarm,
alarm/show_alarms,
alarm/set_alarm,
nalarm/cancel_alarm,
nalarm/time_left_on_alarm.

recognizing_textual_entailment:
Determine if the information in the second
sentence is entailed in the first one.
Output: 0 or 1.
```

7.9 Full Results

We show full results (validation and test set performance) for each model and task for HA, SA, LA (BINARY), and LA (MULTI) in Table 12 and for NER, TC, ID, and RTE in Table 13.

Task	Model	Learning Rate	Batch Size	Epochs	Weight Decay	Warmup Ratio
HA	LUXEMBERT	5e-5	16	2	0	0
HA	MBERT-base	1e-5	16	2	0.01	0.1
HA	LTZ-E1-base	3e-5	8	5	0	0.1
HA	LTZ-E1-mini	8e-5	8	5	0.01	0.1
HA	MMBERT-base	1e-5	8	5	0	0
HA	XLM-R-base	1e-5	8	5	0	0.1
SA	LUXEMBERT	8e-5	16	2	0	0
SA	MBERT-base	1e-5	16	5	0	0.1
SA	LTZ-E1-base	8e-5	16	5	0	0
SA	LTZ-E1-mini	8e-5	8	5	0	0.1
SA	MMBERT-base	1e-5	16	2	0.01	0.1
SA	XLM-R-base	1e-5	8	5	0.01	0
LAB	LUXEMBERT	8e-5	16	2	0	0
LAB	MBERT-base	1e-5	16	5	0	0.1
LAB	LTZ-E1-base	8e-5	8	5	0.01	0.1
LAB	LTZ-E1-mini	8e-5	16	5	0	0
LAB	MMBERT-base	3e-5	16	5	0	0.1
LAB	XLM-R-base	1e-5	16	5	0.01	0
LAM	LUXEMBERT	5e-5	16	2	0.01	0.1
LAM	MBERT-base	3e-5	16	5	0.01	0.1
LAM	LTZ-E1-base	8e-5	8	5	0.01	0.1
LAM	LTZ-E1-mini	8e-5	8	5	0.01	0.1
LAM	MMBERT-base	3e-5	16	5	0	0.1
LAM	XLM-R-base	1e-5	8	5	0.01	0.1
NER	LUXEMBERT	3e-5	16	5	0	0.1
NER	MBERT-base	3e-5	8	5	0	0
NER	LTZ-E1-base	8e-5	8	5	0.01	0.1
NER	LTZ-E1-mini	8e-5	8	5	0.01	0.1
NER	MMBERT-base	5e-5	16	5	0.01	0.1
NER	XLM-R-base	5e-5	16	5	0	0.1
ID	LUXEMBERT	3e-5	16	2	0.01	0
ID	MBERT-base	3e-5	8	5	0	0
ID	LTZ-E1-base	8e-5	8	5	0	0
ID	LTZ-E1-mini	8e-5	8	5	0	0
ID	MMBERT-base	8e-5	16	2	0	0
ID	XLM-R-base	3e-5	16	5	0	0
TC	LUXEMBERT	3e-5	8	5	0.01	0
TC	MBERT-base	1e-5	8	5	0.01	0.1
TC	LTZ-E1-base	8e-5	16	5	0.01	0.1
TC	LTZ-E1-mini	8e-5	8	5	0.01	0
TC	MMBERT-base	3e-5	8	5	0.01	0.1
TC	XLM-R-base	1e-5	16	5	0.01	0
RTE	LUXEMBERT	8e-5	16	2	0	0
RTE	MBERT-base	1e-5	16	5	0.01	0.1
RTE	LTZ-E1-base	8e-5	8	2	0	0.1
RTE	LTZ-E1-mini	5e-5	16	5	0.01	0
RTE	MMBERT-base	3e-5	8	5	0	0.1
RTE	XLM-R-base	1e-5	16	2	0.01	0.1

Table 11: **Best hyperparameters per model for each task.**

Task	Model	Dev F_1	Test F_1
HA	LUXEMBERT	66.18 $\pm$0.00	66.37 $\pm$0.00
HA	MBERT-base	77.50 $\pm$10.05	77.91 $\pm$10.26
HA	LTZ-E1-base	62.29 $\pm$4.91	62.81 $\pm$4.98
HA	LTZ-E1-mini	72.75 $\pm$1.15	72.69 $\pm$.33
HA	MMBERT-base	84.56 $\pm$2.66	85.59 $\pm$1.61
HA	XLM-R-base	71.91 $\pm$9.92	72.09 $\pm$9.90
SA	LUXEMBERT	58.78 $\pm$2.27	58.66 $\pm$0.73
SA	MBERT-base	43.99 $\pm$10.42	41.25 $\pm$4.87
SA	LTZ-E1-base	46.05 $\pm$9.44	46.59 $\pm$5.88
SA	LTZ-E1-mini	46.45 $\pm$8.10	45.39 $\pm$6.79
SA	MMBERT-base	55.86 $\pm$4.18	53.37 $\pm$4.47
SA	XLM-R-base	38.87 $\pm$1.29	36.40 $\pm$0.57
LAB	LUXEMBERT	89.61 $\pm$0.69	89.17 $\pm$0.18
LAB	MBERT-base	82.74 $\pm$0.73	81.26 $\pm$0.56
LAB	LTZ-E1-base	84.13 $\pm$7.57	83.17 $\pm$8.38
LAB	LTZ-E1-mini	90.00 $\pm$2.64	89.31 $\pm$2.89
LAB	MMBERT-base	89.71 $\pm$ 0.20	89.97 $\pm$0.09
LAB	XLM-R-base	70.86 $\pm$4.47	70. $\pm$4.55
LAM	LUXEMBERT	88.67 $\pm$0.31	87.96 $\pm$0.71
LAM	MBERT-base	82.02 $\pm$1.14	81.20 $\pm$1.02
LAM	LTZ-E1-base	79.19 $\pm$11.94	78.63 $\pm$11.55
LAM	LTZ-E1-mini	86.70 $\pm$4.59	86.62 $\pm$4.64
LAM	MMBERT-base	91.03 $\pm$0.83	90.35 $\pm$0.72
LAM	XLM-R-base	73.58 $\pm$3.31	72.59 $\pm$2.69

Table 12: **Dev and Test F_1 scores for Headline Acceptability (HA), Sentiment Analysis (SA) and Linguistic Acceptability (Binary LAB and Multi LAM**. Results are averaged over three runs, with standard deviations as subscripts.

Task	Model	Dev F_1	Test F_1
NER	LUXEMBERT	89.44 $\pm$0.35	90.57 $\pm$0.09
NER	MBERT-base	90.28 $\pm$0.44	90.83 $\pm$0.28
NER	LTZ-E1-base	88.12 $\pm$0.99	89.17 $\pm$0.70
NER	LTZ-E1-mini	86.86 $\pm$0.79	87.47 $\pm$0.37
NER	MMBERT-base	90.03 $\pm$0.49	90.98 $\pm$0.23
NER	XLM-R-base	88.63 $\pm$1.00	89.80 $\pm$0.50
TC	LUXEMBERT	98.36 $\pm$0.26	98.68 $\pm$0.17
TC	MBERT-base	97.27 $\pm$1.26	97.80 $\pm$0.67
TC	LTZ-E1-base	97.91 $\pm$0.40	98.95 $\pm$0.36
TC	LTZ-E1-mini	98.01 $\pm$0.52	98.50 $\pm$0.23
TC	MMBERT-base	98.38 $\pm$0.29	98.92 $\pm$0.28
TC	XLM-R-base	96.94 $\pm$1.33	97.24 $\pm$0.50
ID	LUXEMBERT	100.00 $\pm$0.00	91.71 $\pm$0.11
ID	MBERT-base	81.02 $\pm$6.52	60.65 $\pm$4.71
ID	LTZ-E1-base	84.82 $\pm$13.43	73.32 $\pm$11.66
ID	LTZ-E1-mini	78.64 $\pm$18.57	80.13 $\pm$3.00
ID	MMBERT-base	83.03 $\pm$15.22	78.26 $\pm$7.22
ID	XLM-R-base	75.85 $\pm$6.98	62.75 $\pm$8.76
RTE	LUXEMBERT	70.31 $\pm$0.71	70.96 $\pm$0.26
RTE	MBERT-base	72.12 $\pm$1.37	73.48 $\pm$1.84
RTE	LTZ-E1-base	67.84 $\pm$4.34	67.88 $\pm$3.20
RTE	LTZ-E1-mini	62.33 $\pm$4.05	63.10 $\pm$3.64
RTE	MMBERT-base	74.49 $\pm$1.81	75.44 $\pm$2.41
RTE	XLM-R-base	71.22 $\pm$0.70	70.82 $\pm$0.42

Table 13: **Dev and Test F_1 scores for Named Entity Recognition (NER), Topic Classification (TC), Intent Detection (ID) and Textual Entailment (RTE)**. Results are averaged over three runs, with standard deviations as subscripts.