Probe Before you Answer! Answerability Probing for Abstention in Cross-Lingual Context-Based Question Answering

Lea Krause[1], Stefan Schouten[1], Michiel van der Meer[2], Piek Vossen[1]

[1]Vrije Universiteit Amsterdam, [2]Leiden University

Abstract

We separate answering from abstention in a context-based question answering task, where a language model answers a question about a document in a possibly different language or abstains when the document contains no answer. A LoRA adapter on Qwen3.5-9B, trained only on answerable questions, generates the answer, while a linear probe on the frozen base model's activations decides answerability independently and returns the required refusal string when it predicts none exists. This isolates the model's answerability signal from its instruction-following. The probe's ranking of answerable against unanswerable questions transfers across languages, corpora, and negative constructions it was not trained on, but its decision threshold does not; training on in-domain multilingual data, rather than recalibrating the threshold, closes most of that gap. For the adapter, how the answer target is constructed trades off automatic correctness metrics against well-formedness and answer length.

1 Introduction

Language models produce different answers depending on the language they are prompted in (Manev, 2026), which raises the question of what else varies with language, including whether a model can tell that a question has no answer at all. We report our submission to the Multilingual Instruction Shared Task (Schmidtová et al., 2026), Sub-task 1: context-based question answering. Given a document in language X, the task asks a question about that document, in X or in a different language Y, and requires either an answer grounded in the document or a fixed refusal string when the document does not contain one. This setup couples two abilities: that of refusing to answer (abstaining) and of generating an answer. Most work on abstention teaches the decision through explicit prompting, calibrated rejection scores, or fine-tuning the answering model itself (Jiang et al., 2021; Yang et al., 2024; Madhusudhan et al., 2025), which ties the decision to the model's instruction-following. The studies that leverage model's internal representations for answerability directly have so far stayed monolingual (Wen et al., 2025). However, in cross-lingual settings, where the question, the passage, or both can be in a language the model handles poorly, it is unclear whether refusing and answering draw on the same underlying capability, or whether a model can know a question is unanswerable without being able to say so.

We separate the two into (1) a LoRA adapter, trained only on answerable questions, that generates answers and (2) a linear probe on the frozen base model's whose activations decides answerability. If the probe decides to abstain, the adapter is not consulted, and the system returns a fixed refusal string. Neither component's accuracy depends on the model following an instruction to refuse, so combining the two estimates an upper bound for a model this size, separate from whether an instruction-tuned model would make the same decisions unprompted. We show that the probe's ranking of answerable against unanswerable questions transfers to unseen languages, corpora, and negative constructions, but the threshold it applies to that ranking does not, and that training on in-domain multilingual data, rather than recalibrating a fixed threshold, closes most of that gap. We find that how the answer target is written trades off automatic correctness metrics against well-formedness and target length. We submit two adapters built on these results, both paired with the same probe.

2 Related Work

2.1 Answerability & Abstention

Teaching models to withhold answers under uncertain predictions or for unanswerable questions

Proceedings of the Eleventh Conference on Machine Translation, pages 1933–1948
October 28-29, 2026 ©2026 Association for Computational Linguistics

has its roots in uncertainty prediction (Ulmer et al., 2022; Shorinwa et al., 2025). Recent approaches teach explicit abstention via calibrated uncertainty scores (Jiang et al., 2021; Madras et al., 2018), model collaboration (Feng et al., 2024b), or fine-tuning on rejections (Yang et al., 2024). On Abstain-QA, explicit prompting and CoT boost abstention performance in larger models (Madhusudhan et al., 2025), a finding corroborated by AbstentionBench scores across frontier LLMs (Kirichenko et al., 2025). However, smaller models which benefit less from explicit prompting remain understudied (Wen et al., 2025). We build on recent work combining cross-lingual abstention with model collaboration (Sun et al., 2025; Feng et al., 2024a), noting that prior multilingual uncertainty and abstention studies report considerable cross-lingual variation (Krause et al., 2023; Wen et al., 2026).

2.2 Probing

Probing trains a simple classifier on frozen activations to measure the task-relevant information they encode (Alain and Bengio, 2018; Belinkov, 2022; Tenney et al., 2019), and has been applied extensively to multilingual models (e.g. Şahin et al., 2020; Yin et al., 2022). Probing multilingual models' internal activations reveals a largely shared, English-centred representation space: non-English inputs are processed through an English-like "concept space" in middle layers before mapping back to the target language in final layers (Wendler et al., 2024; Schut et al., 2025; Zhao et al., 2024), with even independently trained monolingual models converging on the same circuit for certain tasks (Zhang et al., 2025). Other studies paint a more nuanced picture: models do not rely on a fully-aligned semantic space (Lu et al., 2025), and models retain language-specific features (Deng et al., 2025) and language-specific decoding mechanisms in later layers (Harrasse et al., 2026).

Similarly, probing has been applied to answerability, where models have been shown to encode the answerability of an input strongly at the very first decoded token (Kadavath et al., 2022; Slobodkin et al., 2023). Sparse autoencoders (SAEs) can recover good answerability features across domains, though struggle to generalise (Heindrich et al., 2025). Models also tend to internally represent unanswerability even when they fail to verbalise this in their output (Liu et al., 2026). For parametric answerability, where no passage is given, probes trained on low-resource languages achieve better performance on high-resource-projected representations of those languages than on the low-resource representations themselves (Xiao et al., 2025). Crucially, probing studies of answerability have largely remained monolingual (Wen et al., 2025). These two strands of research combined warrants the investigation of multilingual model representations for answerability.

3 Data

Two of our sources are corpora provided in the task, used as train and development set: TyDiQA (Clark et al., 2020) and BELEBELE (Bandarkar et al., 2024). We identified the Universal Declaration of Human Rights (UDHR) dataset as part of the test set, and built training data from the sections the shared task does not use. Our two trained components use the data differently. The answerability probe is trained on TyDiQA and UDHR. Of the two answer-generation adapters we submit (§4.1), one is trained on TyDiQA alone and one on a mix of all three datasets. The sections below describe how we extended each source for training and evaluation. Table 1 gives an overview of the resulting sets.

TyDiQA/cross TyDiQA provides question-passage-answer triples in a single language. We extend it to the cross-lingual setting by translating the questions and answers, leaving the passages original. We used Tencent's `Hy-MT2-7B`[1] (Zheng et al., 2026) for 18 languages (TyDiQA/cross18) and `MADLAD-400-10B`[2] (Kudugunta et al., 2023) to supplement five it does not cover: Bhojpuri, Central Kurdish, Finnish, Haitian Creole and Slovak (TyDiQA/cross23). Previous qualitative analysis had shown that MADLAD translations were frequently not of sufficient quality. We therefore used back-translation to filter the translations as well as checks for repetition loops, copy-through, length-ratio outliers, and over-frequent answer strings. Retention is 33.6% for Bhojpuri, 55.9% for Central Kurdish, 57.2% for Finnish, 62.7% for Haitian Creole, and 62.7% for Slovak. Yoruba was dropped due to very low back-translation scores and high occurrence of repetition loops. These five languages contribute 4,081 samples, or 4.1% of the set, at 504 to 941 samples each against roughly 5,355 for each Hy-MT2 language. The result is 100,466 examples, with questions in

[1] https://huggingface.co/tencent/Hy-MT2-7B
[2] https://huggingface.co/google/madlad400-10b-mt

Dataset	Register	Unanswerable	Langs (psg.×q.)	Items	Samples	Use
Generation data — adapter fine-tuning						
TyDiQA/cross23	Wikipedia	excluded	11 × 23	64,257	100,466	train (both)
BELEBELE	FLORES news/wiki	none	27 × 23	900	54,924	train (mix only)
UDHR	treaty / statutory	none	24 (parallel)	72	13,776	train (mix only)
Probe data — answerability classification						
TyDiQA	Wikipedia	existing	11 (mono.)	64,257	116,067	train + eval
TyDiQA/cross18	Wikipedia	existing	11 × 18	64,257	232,230	eval
BELEBELE/unans.	FLORES news/wiki	constructed	26 × 27	373	22,464	eval (held out)
UDHR	treaty / statutory	generated	24 (parallel)	164	17,280	train + eval
Evaluation						
Test/silver	legal + cultural	designed	25 × 24	360	8,640	eval only

Table 1: QA datasets. We list datasets twice, once for fine-tuning and once for probing, since they use different renderings: the probe sets carry unanswerable samples and the generation sets deliberately do not. "items" is the pool of distinct passage–question pairs a dataset is drawn from, before language expansion. The mixed adapter samples 24,000 / 30,000 / 6,000 from TyDiQA/cross, BELEBELE and UDHR respectively, while the TyDiQA-only adapter uses all 100,466 TyDiQA/cross samples.

23 languages and passages in the 11 languages TyDiQA itself covers, over 245 distinct language pairs. Appendix B shows how this extends TyDiQA's language coverage.

BELEBELE/unanswerable BELEBELE is already fully cross-lingual, but contains only answerable questions. We additionally construct unanswerable questions by removing the sentences that contain the gold label span, which removes the information the question asks for.

UDHR The test data for the contextual question answering subtask consists mostly of legal documents, and we established that one source of the test data was the Universal Declaration of Human Rights, which is available in all languages the task targets. The test set only uses specific sections of it, leaving the rest available for training. We sectioned the 24 unused articles into 18 passages of 224–714 characters, matching the length of the test set's own passages, and had Claude Opus 5 generate five questions per passage, reviewed by an independent pass. Eighteen of the 90 items are authored unanswerable, a 20.0% share taken from the test set's own rate. The adapter is never shown a question it should refuse, so its training set uses the 72 answerable items alone. The probe set uses all 90, plus 74 transplants (a question moved onto a passage that does not answer it) for 164 items.

Test/silver We used Claude Opus 5 to generate silver labels for the provided test set. This data was used purely for evaluation (Section 5), not for training or the selection of parameters of our system.

4 System

Our QA system has two components[3]:

1. A pre-trained language model (Qwen3.5-9B; Qwen Team, 2026), using the `unsloth/Qwen3.5-9B`[4] checkpoint, complemented with LoRA adapters optimised for question answering, and
2. a probe operating on the same base model, deciding on the answerability of the question.

If the probe detects that a question is unanswerable, the LoRA weights remain unused, and instead of generating an answer from the model, we simply return the fixed 'not answerable' phrase mandated in the instructions.

With this approach, we mean to estimate the empirical upper limit, using the probe to directly assess how well the model can decide on answerability in principle, regardless of its ability to follow instructions. This then allows us to train the adapters purely for answering the question correctly, independent of whether the model should attempt to answer it. By providing such a system, we aim to give insight into the best performance that may be expected from a model this size. [5]

4.1 Adapter Training

We submit two answer-generation adapters: one trained on TyDiQA/cross23 only, and one on a mix of all three sources.

[3] Code at https://github.com/lkra/answerability_probing

[4] https://huggingface.co/unsloth/Qwen3.5-9B

[5] Another system trained for Subtask 2 Summarization is described in Appendix C.

Adapter 1: TyDiQA/cross Our first adapter is trained on the 100,466 TyDiQA/cross23 examples of §3, with questions in 23 languages, passages in 11, and every example cross-lingual by construction. Based on the task instructions, targets should be short one-sentence answers rather than passage quotations. We therefore used TyDiQA's answer sentence, in cases it is brief enough, or the answer span otherwise, at a median of 41 characters. Answer spans are normalised to be capitalised and terminated. The terminator for each language is the majority sentence-final character over existing target sentences. The result is *danda* for Bengali and Hindi, the ideographic full stop for Japanese and Chinese, and the period elsewhere. It assigns the period to Bhojpuri, where our generated silver references use *danda*. We follow the training data, so Bhojpuri surface form remains open. The set contains answerable questions only, so the model is never shown a question it should refuse, and the loss is computed on completions alone. Training is a single epoch of LoRA ($r = 16$, $\alpha = 16$, dropout $= 0$, all linear layers, learning rate $= 1\mathrm{e}-4$, batch size $= 8$, sequence length $= 4096$). We keep the checkpoint with the lowest held-out loss of 62 evaluations, at step 12,400 of 12,534 and an eval loss of 0.3905.

Adapter 2: Mixed data We submit a second adapter trained on a mix of TyDi, BELEBELE, and UDHR, using identical hyperparameters and selection rules as Adapter 1. The mix is 40% TyDi, 50% BELEBELE, and 10% UDHR over 60,000 samples. We cap each distinct completion at four occurrences per language, so a language with fewer acceptable translations contributes fewer samples rather than being topped up with repeated ones.

The two adapters differ in data composition and in target construction, so the comparison in §5.1 is not a strict ablation of either. The mixed set is more repetitive, at 62.6% distinct completions against 90.9%. It covers one additional language, Yoruba, through the two additional data sources. It contains 1,495 monolingual samples, or 2.5%, which the TyDi-only set does not and the test set does.

4.2 Probe training

The probe decides, for a given input, whether the question can be answered from the information in the provided passage. Throughout, the probe is a linear classifier on frozen layer 19 of 33 of Qwen3.5-9B (Qwen Team, 2026), trained once and applied unchanged. The probe incorporated into the system was trained on the original, monolingual TyDiQA and our UDHR development data set. The layer was chosen by a sweep over all 33 layers (see Appendix E). During both train- and test-time the probe is presented with samples wrapped in the model's chat template, but without any kind of instruction. The probe is applied to the final input token, which is the newline after the start of the assistant response, i.e. the token immediately prior to the model's generated response.

5 Evaluation

We evaluate our system through various ablations for both the adapter fine-tuning data and the probes.

5.1 Answer Generation Adapter

The adapter modifies the base model so that it performs well at answering questions as accurately as possible, regardless of whether they are answerable. We currently evaluate against the silver references of Section 3, on the 6,325 answerable test samples and report scores against gold references for the two submitted adapters (Table 3). Central Kurdish is excluded throughout, since its silver references are in the wrong Kurdish variety. Each approach uses identical inference settings to compare the effect of training data alone.

Based on the shared task, we evaluate four aspects: (1) following the format instructions, (2) giving the correct answer, (3) writing a well-formed sentence, and (4) answering in the correct language. We measure answer correctness with chrF (Popović, 2015) and a rescaled BERTScore (Zhang et al., 2020) over XLM-R large (Conneau et al., 2020). Instruction following is measured through the median answer length in characters, currently against 48 for the silver reference. Well-formed sentences are computed as the share of answers that are both capitalised and sentence-terminated. Finally, we observe the share of responses detected in a language other than the one the question was asked in using the fastText `lid.176` classifier (Joulin et al., 2017). The official evaluation uses GlotLID over all 24 question languages and reports 95.9% and 95.4% of answers in the requested language for the two submitted adapters.

BERTScore is reported with the baseline rescaling of Zhang et al. (2020), $(F_1 - b)/(1 - b)$, but with the baseline b estimated per variant. We esti-

Target construction	chrF	BERT	med. len	well-f.	off-t.
Full sentence, 23 lang.	36.2	23.8	159	79.8	**0.7**
Adaptive	**39.4**	30.0	59	42.8	2.1
Adaptive+normalised	38.9	33.3	56	**97.7**	2.3
Mixed sources	37.1	**37.5**	34	97.2	1.7
silver reference	—	—	*48*	*100*	*0.7*

Table 2: Adapter variants, on 6,325 answerable samples with Central Kurdish excluded. BERT is BERTScore rescaled against a per-variant baseline, as a percentage of the distance from that baseline to a perfect score. Off-target is restricted to the 13 languages where the detector is reliable. The last two are the submitted adapters.

	judge	chrF++	R-L	COMET	C-QE	NLI	lang.
Adapter 1	**67.5**	**36.3**	**28.3**	71.2	62.7	24.4	**95.9**
Adapter 2	62.0	34.8	27.7	**72.6**	**63.3**	**26.1**	95.4

Table 3: Official scores of the submitted adapters against gold references (Schmidtová et al., 2026). *judge*: majority vote of three LLM judges, the primary criterion; R-L: ROUGE-L; C-QE: reference-free COMET; NLI: mutual entailment with the reference; *lang.*: share of answers in the requested language. Adapters 1 and 2 rank 5th and 6th of 14.

mate it by re-pairing each of a variant's predictions with another sample's reference from the same language, which holds language and register fixed and changes only the answer content. Measured this way, the baseline lies between .855 and .875 and raw F_1 between .889 and .922, so we report the margin above the baseline.

The results can be seen in Table 2. The adapters differ in the target format and which sources were used in training. *Full sentence* takes the whole sentence containing the answer as the target, over the 23 languages (see Section 3). *Adaptive* takes the answer sentence where it is already brief and the answer span otherwise, and + *normalised* additionally capitalises and terminates it. *Mixed sources* uses a subset of all three corpora rather than TyDiQA alone (see Section 4.1). The last two rows are the two adapters we submit.

Correctness The two automatic metrics rank the two submitted adapters in opposite orders, and the ranking seems to be based on answer length: character overlap favours the longer variant, embedding similarity the shorter one. Scores against gold references (Table 3) show the same split. On the primary criterion the difference is 67.5% against 62.0%, and Adapter 1 is ahead in 22 of 24 question languages. The gap is largest in Korean (+16.9) and Chinese (+16.3) and within a point in English, German and Spanish.

Target length Shortening the target from the full answer sentence to a short answer raises chrF by 3.2 and rescaled BERTScore by 6.2 and moves the median answer from 159 characters to 59. No hyperparameter we varied produced a comparable effect. It is also the only change that brings answer length near the length of the silver references.

Well-formedness The span fallback costs well-formedness, which falls from 79.8% to 42.8%. A span extracted from a passage is a bare noun phrase with no sentence punctuation, so only 54.8% of the adaptive targets are terminated, and the model reproduces that rate in its own answers. The outputs are lowercase fragments such as *"the language"* and *"must be brought to justice"*, with correct content and length not exceeding one sentence. ChrF places this variant as strongest. Normalization of the targets restores the well-formedness to 97.7% at a cost of 0.5 chrF.

5.2 Answerability Probe

We construct two probes for answerability detection, differing only in what they are trained on: **TyDi**, which uses only the natural, multilingual answerability data (not the cross-lingual derivative we create), and **TyDi+UDHR**, which adds the cross-lingual UDHR data in the legal domain. The second probe is used in combination with both adapters in our submission.

We evaluate probes on four datasets, each of which removes a different assumption: TyDi (validation), monolingual with natural negatives; UDHR, cross-lingual and in the test set's own legal domain; BELEBELE/cross, cross-lingual with constructed negatives and passages from a corpus no probe here was trained on; and the shared task's own test set (with silver labels).

The rest of this section asks four questions of that probe, in order: (1) Does it transfer to *languages* it has never been trained on? (2) Does it transfer to the *cross-lingual* setting, where the question is not in the passage's language? (3) Does it transfer between *distributions* such as another register, another way of constructing negatives, another class balance? And (4) where transfer fails, how much of the failure is due to the threshold and how much to the probe direction? The first three of those are measured with AUROC, which is a ranking metric: it measures only whether answerable samples

evaluation set	samples	lexical	TyDi	**TyDi+UDHR**
TyDi (validation)	9,813	.755	.981	**.981**
UDHR (validation)	17,280	.795	.888	**.959**
BELEBELE	780	.700	.883	**.917**
BELEBELE/cross	20,280	.701	.823	**.869**
passage lg. above the baseline			22/26	**25/26**

Table 4: Probe AUROC, a threshold-free ranking measure, against the lexical baseline. The two BELEBELE samples rest on 15 items and differ only in whether the question is in the passage's language, so the third row is a 780-sample diagonal rather than an independent evaluation; UDHR is 164 items and TyDi validation is sampled.

outscore unanswerable ones and is unchanged by any monotonic rescaling of the score. It says nothing about whether the probe would decide correctly at any particular threshold. This is what allows us to distinguish between the language-independence of a direction, which ranks, and a threshold, which decides.

Chance is not 0.5 Because every corpus here builds its negatives from text closely related to the passage (Section 3), word overlap between question and passage is by itself a strong predictor. Hence, on TyDi, a lexical baseline reaches .755 for its validation set, rather than 0.5. On the cross-lingual datasets, where we have human translations into the passage's language, a similar baseline is appropriate. Specifically, we look at the overlap computed for human-translated questions, i.e., what a lexical system with a perfect translator would achieve, which gives .795 on UDHR and .701 on BELEBELE.[6]

Overall probe generalisation

Table 4 shows generalisation across languages, domains, and negative constructions for both probes.[7] The TyDi+UDHR probe beats the lexical baseline in 25 of 26 passage languages. The exception is Bengali, where the probe ties the baseline, .830 against .831. It has the second-highest lexical baseline of the 26 and the fourth-lowest probe score. The three languages with the highest lexical baselines in the set are Vietnamese, Bengali, and Hindi, and those are three of the four languages the TyDi-only probe also fails to beat (the fourth is Marathi, where the baseline is unremarkable and the probe is weak).

Transfer to unseen languages

To test whether the probe relies on features specific to a language, we train probes on data without that language. Every sample whose question or passage is in that language is dropped from every training source.[8] and the probe is then scored on that language alone. We hold the number of training samples fixed, so that the removal changes which languages are seen and not how much data. Relative to the probe fitted on all languages, removal costs $-.007$ AUROC for the TyDi probe over its 11 languages and $+.000$ for TyDi+UDHR over its 24. The ranking is almost unchanged: the probe does not need to have seen a language to read answerability in it. For this leave-one-language-out experiment, the held-out language still has the same register, negative construction, and class balance the probe was fitted on. We investigate whether the direction is impacted by a change in those below.

Transfer to the cross-lingual setting

On BELEBELE/cross we reach an AUROC of .869 against a lexical baseline of .701. The cost of asking the question in a different language from the passage is –.048 AUROC on that data set. The penalty depends on how the negatives were built. Varying only the question language, the penalty is –.014 on TyDi's natural negatives against –.048 where the answer sentence is deleted from the true passage.

We also checked if the question sharing a script with the passage helps, but on TyDi/cross18 it only adds .0047 AUROC, computed within passage language. Having seen the question language during training also helps little, on either cross-lingual set: the TyDi probe scores .9635 on the 11 TyDi/cross18 question languages it never trained on, against .9646 on the 7 it did, and .823 on BELEBELE's 16 unseen against .824 on its 11 seen.[9]

Transfer across distributions

The third question is about the data-generating process. This includes: (1) how the negatives were built (Appendix F), (2), the class balance, and (3) the register, i.e. the kind of text the passages are

[6] Overlap on the cross-lingual pair actually presented to the probe is near chance at .519.

[7] Every split is item-disjoint rather than sample-disjoint, since each UDHR item appears in many language renderings (also see Appendix D).

[8] TyDi is monolingual, so there the two removals coincide; the distinction bites only on the cross-lingual sources.

[9] Neither split can be run informatively for TyDi+UDHR, which has seen 26 of BELEBELE's 27 question languages and all 18 of translated TyDi's.

drawn from, encyclopaedic in TyDi, news prose in BELEBELE, and legal in UDHR and the test set.

A probe trained on UDHR alone reads its own register very well (AUROC .944 on held-out UDHR items) but falls to .886 on TyDi validation, against .981 for a probe that saw TyDi. So a probe fitted on one register does not generalise outside of it. Training on both allows the probe to handle both registers. TyDi+UDHR loses .0003 AUROC on TyDi validation relative to the TyDi-only. It also improves transfer performance on BELEBELE by .032 AUROC, a corpus with a different register, a different negative construction and a different class balance from either of its training sets.

The negative construction can be varied on its own, holding the register, the items, the languages and even the answerable samples fixed, by substituting a different passage instead of deleting the answer sentence from the true one. Doing so is worth about $+.10$ AUROC, the submitted probe's performance rises from .863 to .960, but nothing in margin, because it raises a lexical baseline by the same amount (.696 to .792). The harder construction is therefore harder for reasons that do not apply uniquely to a probing strategy (see Appendix F for details).

Class balance is equally important, since on the provided test set, according to our silver labels, only 20% of the samples are unanswerable, compared to an even split on TyDi. While this only influences the validity of a threshold, not a ranking, it matters for the final classification.

How much of the gap is the threshold?

So far, we have looked at ranking performance as measured by AUROC, but in the system, we need to actually classify samples as answerable or not. This requires a threshold. Since we are hoping for a probe that can classify different combinations of question and passage languages, we want a threshold that transfers as well as the direction did, i.e. one that does not depend on the languages in a sample. To investigate this, we introduce a per-language log-odds offset, which is a constant added to the probe output for every sample in that language. This is equivalent to a per-language threshold. We look at the spread of log-odd offsets across languages, since we ideally want all languages to require similar log-odd adjustments (similar thresholds).

On the test set, the per-language log-odds offset that the TyDi-only probe needs is $+1.1$ larger for languages absent from its training data than for languages present in it. A probe's direction therefore transfers to unseen data and languages more easily than its threshold does.

offsets fitted on	TyDi	**TyDi+UDHR**
UDHR (held-out items)	1.011	**0.231**
BELEBELE	0.908	**0.427**
Test/silver	0.959	**0.451**
noise floor	*0.215*	

Table 5: Spread (sd across question languages) of a free per-language log-odds offset, refitted on each probe's own held-out scores. The noise floor is the same estimator on TyDi validation, which is monolingual, in-domain and 50/50 and so has no cross-language bias to find. The spread reaches the noise floor on UDHR and roughly halves on the two corpora neither probe was trained on.

Per-language calibration When dealing with thresholds that do not transfer, one remedy is to keep the probe's direction and fit only the threshold i.e. fit one log-odds offset per question language on held-out data, and add it to the score. Figure 1 shows how far apart those per-language offsets have to be. On test/silver the spread is 0.959 against a measured noise floor of 0.215, a per-language bias of up to 3.5 log-odds.

More probe training data Another possible remedy is to use more data for the probe's training. In Figure 1, we can see that the probe trained with in-domain data needs less per-language correction: the spread falls to 0.451. On held-out UDHR data it falls further still, to 0.231, i.e. $1.1\times$ the noise floor. Appendix G plots the full per-language score distributions behind these spreads.

Table 5 shows that this is not only an artifact of measuring the offsets on the same or similar distributions of data that the probe was trained on. The same estimator run on BELEBELE, which neither probe trains on and differs in source text, register, and negative construction, contracts the spread from 0.908 to 0.427. This is why the submitted systems use the TyDi+UDHR probe.

The probe can absorb the bias itself Language identity is linearly decodable at the operating layer. A one-vs-rest linear probe on the same frozen vectors identifies the question language at AUROC 1.000 across all 24 of them so a linear probe given the 24 languages in-domain can absorb per-language bias into its weight vector, so no separate offset is needed.

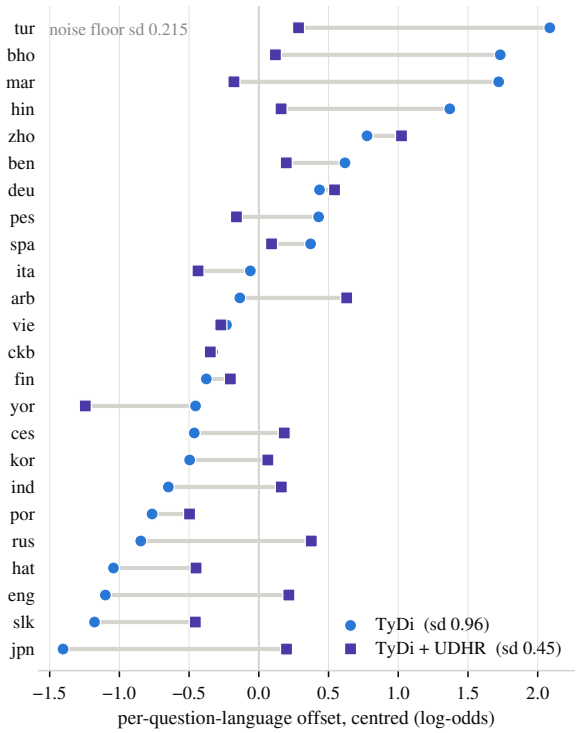

Figure 1: The per-language calibration gap is a property of the training distribution. Each sample is one of the 24 test question languages; the marks are the log-odds offset a free per-language correction would apply to each probe, fitted on the test set. Training on the 24 languages in-domain contracts the spread (sd) in offsets from 0.959 to 0.451, against a noise floor of 0.215 measured on data where no per-language shift exists. This is not an artifact of the set chosen: BELEBELE, which neither probe trains on also contracts from 0.908 to 0.427.

The remaining gap The difference between the TyDi and TyDi+UDHR probes is not only a matter of thresholds. We evaluated every decision rule we can fit on the probes, from a single global constant to per-language thresholds fitted on test/silver itself. The latter is an oracle bound on any thresholding scheme: for the TyDi probe it reaches .783 macro-F1, .113 short of the TyDi+UDHR probe's .896.

Once the probe is trained in-domain, thresholds are not language-specific, and what remains is ranking (not calibration) error, as shown by the macro AUROC over the 24 test languages rising from .937 to .978. Table 6 shows the same pattern. Macro-F1 rises .697 → .896 and false abstentions fall 1,260 → 185 at near-unchanged recall, and all 24 languages improve, from +0.034 ('yor') to +0.303 ('tur'). Per-language correction of the log-odds spread is a redistribution, it moves macro-F1 by +0.003 and makes 10 of 24 languages worse.

probe	abst.	false	rec.	prec.	macro F1
TyDi, per-lg. offsets	2,859	1,260	.913	.559	.697
TyDi+UDHR, global	1,768	**185**	.904	**.895**	**.896**

Table 6: Abstention decisions on all 8,640 test samples, against silver labels (agreement, not accuracy). The baseline is the TyDi probe with a fitted per-language log-odds offset, TyDi+UDHR uses a single global threshold and no per-language term. Against gold labels, the non-answerable F1 of the shipped rule is 86.4% and 86.5% for the two adapter combinations.

Adding training data improves every language. The resulting abstention rate is 20.5%, against the test set's designed 20.3% unanswerable rate, which was not used in calibration, the probe uses a single global bias.

Official abstention scores In the official ranking abstention is scored as detection F1 of the refusal string on unanswerable questions, against gold labels. Our two submissions reach 86.4% and 86.5%, second and third of 14 systems, 0.6 points behind the best (87.1%) and ahead of the two systems that lead the answer-accuracy ranking (82.0% and 81.8%). The two submissions share the probe and differ by 0.1 F1, so the adapter does not affect the abstention decision. Other systems show the same separation. The system with the best refusal F1 ranks eighth on accuracy, and the most accurate system ranks fourth on refusal. Our submissions are fifth and sixth on accuracy, at 67.5% and 62.0% against 81.6% for the best system, meaning the gap to the leading systems is in the adapter, not the probe.

6 Conclusion

We propose a system that decouples answerability from answer generation, and evaluate both against silver labels. On these labels, the probe's ranking of answerable against unanswerable questions transfers to unseen languages and corpora, its threshold does not, and training on in-domain multilingual data closes most of that gap. The adapter's target construction trades correctness metrics against well-formedness. In the official evaluation, the two submissions rank second and third of 14 systems on refusing unanswerable questions (86.4% and 86.5% F1) and fifth and sixth on answer accuracy (67.5% and 62.0%).

Limitations

Language variance Only the question's language varies in most of our cross-lingual evaluation. The one set in which the *passage* language also varies is BELEBELE, whose negatives are constructed by sentence deletion rather than occurring naturally. Whether the answerability direction survives a translated passage with natural negatives is a separate and probably harder question for which no data currently exists.

Lowest resource languages The lowest-resource languages remain unresolved. Yoruba was excluded from fine-tuning of one of our adapters due to data quality issues. Yoruba and Marathi sit at the floor of every evaluation for answerability and are not moved by either intervention we report.

Acknowledgments

This work was partially funded by the Netherlands Organization for Scientific Research (NWO NWA # 1518.22.105, HAICu), and partially by Huawei Finland through the Dreamslab project. All content represented the opinions of the authors, which were not necessarily shared or endorsed by their respective employers and/or sponsors.

References

Guillaume Alain and Yoshua Bengio. 2018. Understanding intermediate layers using linear classifier probes. *arXiv preprint*. ArXiv:1610.01644 [stat.ML].

Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. 2024. The Belebele Benchmark: A Parallel Reading Comprehension Dataset in 122 Language Variants. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 749–775, Bangkok, Thailand. Association for Computational Linguistics.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. TyDi QA: A Benchmark for Information-Seeking Question Answering in Typologically Diverse Languages. *Transactions of the Association for Computational Linguistics*, 8:454–470.

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 the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Boyi Deng, Yu Wan, Baosong Yang, Yidan Zhang, and Fuli Feng. 2025. Unveiling language-specific features in large language models via sparse autoencoders. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4563–4608, Vienna, Austria. Association for Computational Linguistics.

Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Orevaoghene Ahia, Shuyue Stella Li, Vidhisha Balachandran, Sunayana Sitaram, and Yulia Tsvetkov. 2024a. Teaching LLMs to abstain across languages via multilingual feedback. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4125–4150, Miami, Florida, USA. Association for Computational Linguistics.

Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. 2024b. Don't hallucinate, abstain: Identifying LLM knowledge gaps via multi-LLM collaboration. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14664–14690, Bangkok, Thailand. Association for Computational Linguistics.

Abir Harrasse, Florent Draye, Punya Syon Pandey, Zhijing Jin, and Bernhard Schölkopf. 2026. Tracing multilingual representations in llms with cross-layer transcoders. *Preprint*, arXiv:2511.10840.

Lovis Heindrich, Philip Torr, Fazl Barez, and Veronika Thost. 2025. Do sparse autoencoders generalize? a case study of answerability. In *ICML 2025 Workshop on Reliable and Responsible Foundation Models*.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, and

17 others. 2022. Language models (mostly) know what they know. *Preprint*, arXiv:2207.05221.

Amir Hossein Kargaran, Ayyoob Imani, François Yvon, and Hinrich Schuetze. 2023. GlotLID: Language identification for low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6155–6218, Singapore. Association for Computational Linguistics.

Polina Kirichenko, Mark Ibrahim, Kamalika Chaudhuri, and Samuel J. Bell. 2025. Abstentionbench: Reasoning llms fail on unanswerable questions. In *Advances in Neural Information Processing Systems*, volume 38. Curran Associates, Inc.

Lea Krause, Wondimagegnhue Tufa, Selene Baez Santamaria, Angel Daza, Urja Khurana, and Piek Vossen. 2023. Confidently wrong: Exploring the calibration and expression of (un)certainty of large language models in a multilingual setting. In *Proceedings of the Workshop on Multimodal, Multilingual Natural Language Generation and Multilingual WebNLG Challenge (MM-NLG 2023)*, pages 1–9, Prague, Czech Republic. Association for Computational Linguistics.

Sneha Kudugunta, Isaac Caswell, Biao Zhang, Xavier Garcia, Derrick Xin, Aditya Kusupati, Romi Stella, Ankur Bapna, and Orhan Firat. 2023. Madlad-400: A multilingual and document-level large audited dataset. In *Advances in Neural Information Processing Systems*, volume 36, pages 67284–67296. Curran Associates, Inc.

Yi Liu, Xiangyu Liu, Zequn Sun, and Wei Hu. 2026. Answering the unanswerable is to err knowingly: analyzing and mitigating abstention failures in large reasoning models. In *Proceedings of the Fortieth AAAI Conference on Artificial Intelligence and Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence and Sixteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'26/IAAI'26/EAAI'26. AAAI Press.

Meng Lu, Ruochen Zhang, Carsten Eickhoff, and Ellie Pavlick. 2025. Paths not taken: Understanding and mending the multilingual factual recall pipeline. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 15066–15096, Suzhou, China. Association for Computational Linguistics.

Nishanth Madhusudhan, Sathwik Tejaswi Madhusudhan, Vikas Yadav, and Masoud Hashemi. 2025. Do LLMs know when to NOT answer? investigating abstention abilities of large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 9329–9345, Abu Dhabi, UAE. Association for Computational Linguistics.

David Madras, Toni Pitassi, and Richard Zemel. 2018. Predict responsibly: Improving fairness and accuracy by learning to defer. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Alexander Manev. 2026. Inference-time steering for cross-lingual factual consistency in llms. *Preprint*, arXiv:2607.19243.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Qwen Team. 2026. Qwen3.5: Accelerating productivity with native multimodal agents.

Shaurya Rohatgi, Yanxia Qin, Benjamin Aw, Niranjana Unnithan, and Min-Yen Kan. 2023. The ACL OCL corpus: Advancing open science in computational linguistics. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10348–10361, Singapore. Association for Computational Linguistics.

Alejandro R. Salamanca, Diana Abagyan, Daniel D'souza, Ammar Khairi, David Mora, Saurabh Dash, Viraat Aryabumi, Sara Rajaee, Mehrnaz Mofakhami, Ananya Sahu, Thomas Euyang, Brittawnya Prince, Madeline Smith, Hangyu Lin, Acyr Locatelli, Sara Hooker, Tom Kocmi, Aidan Gomez, Ivan Zhang, and 7 others. 2026. Tiny aya: Bridging scale and multilingual depth. *Preprint*, arXiv:2603.11510.

Patrícia Schmidtová, Katia Artemova, Seth Aycock, Niyati Bafna, Tom Kocmi, Philipp Koehn, Danni Liu, Nam Luu, Sara Papi, Beatrice Savoldi, Mariya Shmatova, Vilém Zouhar, Maike Züfle, and Pinzhen Chen. 2026. Findings of the WMT26 multilingual instruction shared task: Small models are not yet polyglots. In *Proceedings of the Eleventh Conference on Machine Translation*. Association for Computational Linguistics.

Lisa Schut, Yarin Gal, and Sebastian Farquhar. 2025. Do multilingual LLMs think in english? In *ICLR 2025 Workshop on Building Trust in Language Models and Applications*.

Ola Shorinwa, Zhiting Mei, Justin Lidard, Allen Z. Ren, and Anirudha Majumdar. 2025. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *ACM Comput. Surv.*, 58(3).

Aviv Slobodkin, Omer Goldman, Avi Caciularu, Ido Dagan, and Shauli Ravfogel. 2023. The curious case of hallucinatory (un)answerability: Finding truths in the hidden states of over-confident large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3607–3625, Singapore. Association for Computational Linguistics.

Yuxi Sun, Aoqi Zuo, Wei Gao, and Jing Ma. 2025. CausalAbstain: Enhancing multilingual LLMs with

causal reasoning for trustworthy abstention. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 14060–14076, Vienna, Austria. Association for Computational Linguistics.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Dennis Ulmer, Jes Frellsen, and Christian Hardmeier. 2022. Exploring predictive uncertainty and calibration in NLP: A study on the impact of method & data scarcity. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2707–2735, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Bingbing Wen, Jihan Yao, Shangbin Feng, Chenjun Xu, Yulia Tsvetkov, Bill Howe, and Lucy Lu Wang. 2025. Know your limits: A survey of abstention in large language models. *Transactions of the Association for Computational Linguistics*, 13:529–556.

Chenfeiyu Wen, Ao Zhu, Runkun Long, Hejun Huang, Junjie Jiang, and Chi Shing Lee. 2026. Calibjudge: Calibrated llm-as-a-judge for multilingual rag with uncertainty-aware scoring. In *Proceedings of the 2026 International Conference on Artificial Intelligence and Control*, CAIC '26, page 661–667, New York, NY, USA. Association for Computing Machinery.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in English? on the latent language of multilingual transformers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15366–15394, Bangkok, Thailand. Association for Computational Linguistics.

Chenghao Xiao, Hou Pong Chan, Hao Zhang, Mahani Aljunied, Lidong Bing, Noura Al Moubayed, and Yu Rong. 2025. Analyzing LLMs' knowledge boundary cognition across languages through the lens of internal representations. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 24099–24115, Vienna, Austria. Association for Computational Linguistics.

Yuqing Yang, Ethan Chern, Xipeng Qiu, Graham Neubig, and Pengfei Liu. 2024. Alignment for honesty. In *Advances in Neural Information Processing Systems*, volume 37, pages 63565–63598. Curran Associates, Inc.

Da Yin, Hritik Bansal, Masoud Monajatipoor, Liunian Harold Li, and Kai-Wei Chang. 2022. GeoMLAMA: Geo-diverse commonsense probing on multilingual pre-trained language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2039–2055, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ruochen Zhang, Qinan Yu, Matianyu Zang, Carsten Eickhoff, and Ellie Pavlick. 2025. The same but different: Structural similarities and differences in multilingual language modeling. In *International Conference on Learning Representations*, volume 2025, pages 95372–95392.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

Yiran Zhao, Wenxuan Zhang, Guizhen Chen, Kenji Kawaguchi, and Lidong Bing. 2024. How do large language models handle multilingualism? In *Advances in Neural Information Processing Systems*, volume 37, pages 15296–15319. Curran Associates, Inc.

Mao Zheng, Zheng Li, Tao Chen, Bo Lv, Mingrui Sun, Mingyang Song, Jinlong Song, Hong Huang, Decheng Wu, Hai Wang, Yifan Song, Yanfeng Chen, and Guanwei Zhang. 2026. Hy-mt2: A family of fast, efficient and powerful multilingual translation models in the wild. *Preprint*, arXiv:2605.22064.

Gözde Gül Şahin, Clara Vania, Ilia Kuznetsov, and Iryna Gurevych. 2020. Linspector: Multilingual probing tasks for word representations. *Computational Linguistics*, 46(2):335–385.

A Submission Names

Adapter 1 (§4.1) was submitted as `TeamSpinoza-primary`, Adapter 2 (§4.1) as `TeamSpinoza-Variant_V4`, and the summarisation system of Appendix C as `TeamSpinoza-Variant_sum-verifier3`. Both adapters use the same TyDi+UDHR probe.

B TyDiQA/cross Language Coverage

Table 7 gives the full passage–question language matrix of TyDiQA/cross (Section 3). TyDiQA's question, passage, and answer are monolingual, so its own coverage is exactly the diagonal: 11 language pairs. Translating the questions and answers of every item into the shared task's evaluation languages fills in the rest of the matrix, adding 245 cross-lingual pairs. Rows and columns are ordered by translation model, and each model's cells form one block.

C Subtask 2: Summarisation

We observed the shared task summarization data included examples from academic papers on NLP,

	question / answer language																										
	Hy-MT2-7B																			MADLAD-400-10B							
passage	ces	deu	hin	ita	mar	pes	por	spa	tur	vie	zho	arb	ben	eng	ind	jpn	kor	rus	swh	tel	tha	fin	bho	ckb	hat	slk	yor†	fra
Arabic	•	•	•	•	•	•	•	•	•	•	•	■	•	•	•	•	•	•				○	○	○	○	○	○	
Bengali	•	•	•	•	•	•	•	•	•	•	•	•	■	•	•	•	•	•				○	○	○	○	○	○	
English	•	•	•	•	•	•	•	•	•	•	•	•	•	■	•	•	•	•				○	○	○	○	○	○	
Indonesian	•	•	•	•	•	•	•	•	•	•	•	•	•	•	■	•	•	•				○	○	○	○	○	○	
Japanese	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	■	•	•				○	○	○	○	○	○	
Korean	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	■	•				○	○	○	○	○	○	
Russian	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	■				○	○	○	○	○	○	
Swahili	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	■			○	○	○	○	○	○	
Telugu	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•		■		○	○	○	○	○	○	
Thai	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•			■	○	○	○	○	○	○	
Finnish	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•				■	○	○	○	○	○	

Table 7: Language coverage of TyDiQA/cross. Rows are the 11 passage languages TyDiQA provides, columns the 28 shared-task languages, ordered so that each translation model occupies one block. ■ = a pair TyDiQA itself has; as TyDiQA is monolingual, that is exactly the diagonal. • = question and answer translated with Hy-MT2-7B (18 languages), ○ = translated with MADLAD-400-10B (the five languages Hy-MT2 does not cover). The 245 marked off-diagonal cells outside the Yoruba column are what TyDiQA/cross adds; it contributes no diagonal cell, since no row has its question and its passage in the same language. The blank columns are the shared task's training-only languages: they are never evaluated, so we did not translate into them. Swahili, Telugu and Thai are still passage languages, which is why they sit between the two blocks; French is neither. †Yoruba was translated with MADLAD-400 but excluded after a round-trip check (Section 4).

likely sourced from the ACL Anthology. Following the intuition that showing similar examples to the model will probably help it in downstream translations, we supplement the training data with translated abstract/document pairs. In the official evaluation the system ranks 10th of 15 on the primary metric, language-gated ROUGE-L, at 12.33 against 17.55 for the best system; gated chrF++ is 28.26, COMET 73.58 and COMET-QE 58.46. 94.5% of summaries are in the requested language.

Training data The LoRA adapter is fine-tuned on abstracts from the ACL-OCL corpus[10] (Rohatgi et al., 2023). Abstracts are first reformatted from first person to third person, removing first-person pronouns (*I*, *we*, *our*, . . .), then translated with Hy-MT2-7B (Zheng et al., 2026).

Adapter training The adapter is trained from the `unsloth/Qwen3-4B` checkpoint, sequence length 4096. Training is a single epoch at learning rate 2e-4, with a batch size of 8.

System 3 generates with a two-pass cascade. All text is produced by the same merged bf16 Qwen3-4B checkpoint. The primary pass runs a generate–verify–revise loop (max 2 rounds, threshold 4.0). The judge is `CohereLabs/tiny-aya-global` (Salamanca et al., 2026), a separate multilingual instruction checkpoint decoded greedily against a written rubric with multiple criteria, covering faithfulness, coverage, compression, as well as fluency. Each criterion has anchors for scores ranging 1-–5, and returns its scores as JSON; In addition to scoring, the verifier model writes revision instructions in the answer language. Language fidelity is scored outside the judge by GlotLID (Kargaran et al., 2023), segment by segment, so wrong-language and romanized output are caught mechanically rather than guessed at. The pass decision is computed in Python as a weighted mean against the preset threshold. Faithfulness, coverage, and language fidelity can each cause a veto, in case the score is too low (1.5 or lower). Because the rubric does not contain a criterion for decoding degeneration, the outputs are screened by an intrinsic repetition detector and the samples that trigger it are replaced by a decode from the same checkpoint with repetition_penalty = 1.15 (1.35 for ckb/yor). The fallback fires on 16% of samples.

[10] https://huggingface.co/datasets/WINGNUS/ACL-OCL

D Probing Method

Here we document the full details of the probing methodology.

For every input sample we run one forward pass of a *frozen* decoder-only language model (LM) and keep the hidden state at a single token position from every layer. A probe is then an L2-regularised

logistic regression (LBFGS) fitted on those cached vectors, one probe per layer. The weights of the LM are not updated.

The question is whether answerability is *linearly decodable* from a representation formed by the model, and whether the direction is language agnostic. A non-linear probe that transfers tells us less, because it could be reconstructing the feature rather than reading it.

The language model used is Qwen3.5-9B (Qwen Team, 2026) with 32 decoder layers, 33 hidden states, width 4096 throughout, and thinking mode is disabled.

As the prompt, the model sees the data wrapped in the model's chat template, with *no task instruction*:

```
<|im_start|>user
{passage}

{question}<|im_end|>
<|im_start|>assistant
```

The probing uses the final token of the whole rendered prompt, i.e. the newline after `assistant`, and *not* the last token of the question.

E Choosing the Probing Layer

The probe reads the hidden state after a particular layer, and which one is a free parameter. This appendix reports the sweep that decided which layers are suitable, and then gives the per-language result at the layer we actually use.

Protocol For every hidden state 0..32 we fit the probe on ten of TyDi's eleven languages and score it on the eleventh, and average over the eleven folds. Passage and question are always in the same language, so this tests only whether the direction survives removing a language, with nothing else about the sample changing. The selection is therefore made on training-split languages held out from each other. Probes are fitted on a 3,000-sample-per-language subsample (32,899 samples), and the regularisation constant is fixed at $\lambda = 10^{-3}$ across layers so that the comparison is of layers rather than of tuning.

Layer sweep Figure 2 gives the Qwen3.5-9B curve. Layer 0 is at .500, chance, so nothing the probe reads at later layers is lexical leakage through the embedding. Decodability then climbs steeply, is within .01 of its maximum by layer 13, peaks in the middle layers, and loses .03 over the last third of the network. The peak is a plateau, which is why the choice is not obvious. The maximum is layer 20 at .9716, but every layer from 17 to 21 is within .003 of it.

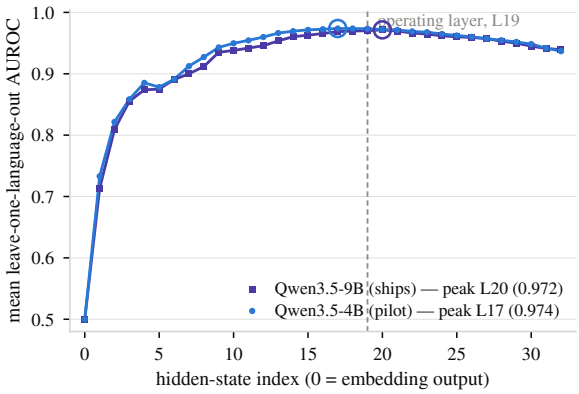

Figure 2: Answerability is most linearly decodable in the middle layers. Each point is a probe fitted at that hidden state and scored on the held-out language, meaned over the eleven TyDi folds; rings mark each language model's argmax and the dashed line the layer we use in the end. The lower curve is a Qwen3.5-4B.

held out	overlap	probe
arb	.810	.975
ben	.791	.966
eng	.775	.962
fin	.825	.985
ind	.805	.977
jpn	.757	.972
kor	.826	.964
rus	.743	.980
swh	.795	.967
tel	.806	.963
tha	.833	.961
mean	**.797**	**.970**

Table 8: Leave-one-language-out AUROC on TyDi at the shipped layer 19, **bare** prompt. Each row is a probe that trained on the other ten languages and has never seen this one. *overlap* is the per-language lexical baseline. The mean is the layer-19 point of Figure 2.

Layer selection Layer 19 is second on the sweep, .9702 against .9716. We use it because the cross-lingual hidden states were re-extracted, keeping only hidden states from ten layers rather than all 33, since the translation cache at every layer is 59 GB, and the ten retained are $\{0, 4, 8, 12, 16, 19, 22, 26, 29, 32\}$.

Per language at the shipped layer Table 8 gives a mean of .970 against a .797 lexical baseline, positive in every language, with the worst language, Thai, at .961. The signal is linearly decodable from the frozen language model in every language tested,

negatives	TyDi	**T+U**	lexical
deleted (as reported)	.833	**.863**	.696
swapped	.943	**.960**	.792
change	+.110	+.096	+.096
change in margin	+.014	+.000	—

Table 9: Negative construction with everything else held fixed: AUROC on the same 128 items, 8 passage languages and 2,048 monolingual samples, differing only in how the unanswerable sample was built. *lexical* is the overlap baseline on the identical samples. The two sets share their *answerable* samples exactly, so only the negative side moves.

including languages the probe never saw.

F Controlling Negative Construction

Deleting the answer sentence removes exactly the words the question asks about, so the construction chosen to defeat topic matching also gives the lexical baseline a strong cue. Table 9 compares deleting the answer sentence with substituting a topically similar passage whole.

G Per-Language Score Distributions

Every probe number reported elsewhere is a summary statistic: an AUROC, a macro-F1, the spread of a fitted offset. They hide whether a language is hard because its two classes overlap, or because they separate cleanly but sit on the wrong side of a global threshold.

Figure 3 shows violin plots for raw probe scores. The split is by *gold* label, so a panel is read as "how far apart does this probe hold the two classes, in this language, on this data". The addition of UDHR in the training data is visible as a *widening of the gap* rather than as a shift of a threshold. Averaged over languages, the median separation between the two classes grows from 4.54 to 7.99 log-odds on UDHR, 5.35 to 9.79 on mist-test and 3.71 to 5.03 on BELEBELE.

H Probe Confidence and Model Behaviour

Section 4 decouples answering from abstention: the probe decides, and the adapter only ever sees questions the probe has passed. Here we investigate whether the classifications made by our probes correlate with abstention by the pre-trained-only language model.

Protocol Our probes read a *bare* rendering of a test item. We show the model the organisers' `qa-context` prompt, complete with the per-language literal it is told to produce when a question cannot be answered. Its four segments are an intro, the passage, a question prefix, an instruction naming the abstention string, and the requested answer language:

```
I have a question about this passage:
{passage}
The question is: {question}
Could you please answer in one sentence,
using only what the passage says? If the
passage doesn't give the answer, just
write "not answerable". Please answer
in English.
```

The separator between segments is the two-character sequence `\n\n` rather than a single newline, which is what the organisers' own test prompts contain, so we keep it. Probes are applied as fixed directions at layer 19. Samples in the organisers' sample are excluded before generation rather than at the join, leaving 6,886 monolingual, 11,338 cross-lingual and 17,940 BELEBELE samples. Generation is greedy, at most 128 new tokens, thinking disabled. Probe scores and generations are joined per sample, restricted to the truly unanswerable samples, and bucketed by $p(\text{answerable})$. The reported quantity is how often the model produced an answer attempt rather than an abstention.

Model failure In all six arms of Table 10 the model answers more often as the probe's uncertainty grows, cross-lingually and on constructed negatives alike. On BELEBELE the rate under the TyDi probe rises from 26.0 to 54.8, a 29-point span over cells of a thousand samples each.

Figure 3: Raw probe score per question language, split by gold label, for both probes. Rows are probes, columns are evaluation sets; within a panel each language contributes a pair of half-violins, answerable above the row line and unanswerable below, with a white tick at the median. Language order is shared across all eight panels and is set once by the median answerable score under the TyDi probe on UDHR; the four languages UDHR does not cover are appended alphabetically, and an empty row means the set does not cover that language. The axis is log-odds, ticked in probability, because on a linear scale both classes collapse onto the edges. Each half is normalised to a common maximum *width*, so the shapes are comparable across languages but the figure carries no information about how many samples are behind a violin. **The right-hand two columns are not further test sets**: TyDi validation is the monolingual in-domain negative control whose spread is the estimator's noise floor, and BELEBELE's negatives are constructed, so levels compare within a column and not across. No threshold, offset or calibration is applied anywhere.

	monolingual		cross-lingual		BELEBELE	
p(answerable)	TyDi	TyDi+UDHR	TyDi	TyDi+UDHR	TyDi	TyDi+UDHR
$[0.00, 0.01)$	18.1 (2,680)	18.4 (2,564)	19.7 (5,294)	16.3 (3,735)	26.0 (2,278)	22.3 (777)
$[0.01, 0.05)$	23.6 (834)	22.4 (905)	28.2 (1,186)	21.1 (1,653)	30.2 (1,909)	21.1 (1,202)
$[0.05, 0.10)$	30.2 (281)	26.6 (331)	34.6 (407)	30.7 (596)	35.2 (860)	25.5 (764)
$[0.10, 0.25)$	31.9 (310)	30.1 (289)	39.7 (456)	33.4 (686)	43.9 (1,209)	25.6 (1,125)
$[0.25, 0.50)$	**40.5** (190)	**43.8** (210)	**45.4** (291)	**41.4** (526)	**54.8** (1,091)	**33.7** (1,106)
unanswerable samples	4,627		8,045		8,970	
probe detects	92.8	92.9	94.9	89.4	81.9	55.5
answers despite probe	22.0	21.9	24.0	22.1	35.4	25.8
probe AUROC	.981	.981	.963	.967	.831	.868

Table 10: How often Qwen3.5-9B answers a question that is genuinely unanswerable, as a function of the probe's score, for both candidate probes. Rows are the probe's confidence that the question *is* answerable; every row is a truly unanswerable item, so a low bucket means the probe is confidently right. Cells are the percentage of those items the model answered anyway, with n in grey. *probe detects* is the share of unanswerable samples scored below the decision threshold; *answers despite probe* is the aggregate over exactly those samples.