Universiteit Leiden

Probing cross-lingual differences in how LLMs represent Natural Language Inference

Nicolas Ramos Fernandez (s3917886)

Supervisors:
Michiel van der Meer & Gijs Wijnholds

Bachelor Thesis– Data Science and Artificial Intelligence

Leiden Institute of Advanced Computer Science (LIACS)
liacs.leidenuniv.nl

June 26, 2026

Abstract

LLMs have been shown to be able to accurately detect inference relations between sentences. However, it is not clear to what extent this performance is due to a genuine understanding of Natural Language Inference (NLI) concepts. It is also not clear whether models encode these concepts differently depending on the language. We address these questions by looking at the internal representations of two LLMs: Olmo 3 7B Instruct, which was trained almost exclusively on English, and Tiny Aya Global, which was explicitly trained to be multilingual. We probe these models on the SICK NLI dataset and its sentence-aligned translations into Spanish (SICK-ES), Japanese (JSICK), and Dutch (SICK-NL).

We ask (Q.1) which layers encode NLI most strongly, whether these layers are consistent across languages, (Q.2) how similar the resulting internal representations are across languages, and (Q.3) whether multilingual pretraining changes the way NLI is encoded in the models. At every layer, we train linear (multinomial logistic regression) probes on the model activations, where the probing task is to determine the entailment relation between two sentences. We then report the performance compared to a control probe (selectivity), and corroborate these layerwise results with a second probe type (mass-mean probes).

We observe a clear difference between what the models encode and what they actually output. Direct prompting is strongly language-dependent for Olmo, being best in English but collapsing in Japanese, and more uniform for Aya. However, linear probes recover high selectivity in the middle and later layers for all four languages in both models, including Japanese in Olmo, despite the low performance when directly prompted. Both models, therefore, encode entailment more reliably than they express it when prompted.

Cross-lingual probe transfer and direct comparison of probe weights show that NLI representations are most alignable in the middle layers: probes transfer best there, and probes trained independently on different languages converge to similar weight vectors, peaking mid-network. English, Spanish, and Dutch probes transfer better and are more similar to each other, while Japanese is a consistent outlier, even in Aya, despite its Japanese training. This suggests that script and typology shape representational alignment beyond training-data coverage alone.

Together, these results give layerwise, cross-lingual evidence that NLI is encoded as a linearly decodable structure most concentrated in the later layers and most language-agnostic in the middle layers, and they quantify a measurable gap between a model's internal NLI competence and its surface task performance.

Contents

1 Introduction

Large Language Models (LLMs) are used across a great number of languages, yet many of them are trained predominantly on English [Qin et al., 2024, Blasi et al., 2022], and the interpretability research that studies their internal workings has historically focused largely on English [Xiao et al., 2025], with cross-lingual interpretability only recently receiving sustained attention [Wen-Yi and Mimno, 2023, Schut et al., 2025, Zhao et al., 2024]. This means that we do not yet have a sufficiently clear picture on LLM internal representations for different languages. This raises the question: when a model handles the same concept in two different languages, does it represent that concept internally in the same way, or does it instead develop separate, language-specific mechanisms for each? The answer has direct implications for how well a model is able to generalise across languages, for how knowledge might transfer between them, and for how reliably the model can be expected to behave in languages that were less well represented during its training. Understanding whether and how internal representations are shared across languages is the main question that this thesis sets out to investigate.

Studying this question requires two components. The first is a way of looking inside the model in order to inspect its internal representations rather than only its final outputs, for which we use probing, an interpretability technique [Belinkov et al., 2017, Alain and Bengio, 2018]. The second is a task where the same content and concepts are encoded in different languages; for this, we use Natural Language Inference (NLI).

NLI is the task of determining whether one sentence (the hypothesis) follows from, contradicts, or is unrelated to another sentence (the premise); Example 1 in Section 3.1 gives an illustration. We chose it for two main reasons: firstly, NLI has a long history of study within natural language processing, which means that established methods, datasets, and expectations already exist, keeping our work comparable to a large body of previous research. Secondly, there exist sentence-aligned multilingual versions of the NLI dataset SICK [Marelli et al., 2014b] together with its translations into Spanish [Huertas-Tato et al., 2022], Japanese [Yanaka and Mineshima], and Dutch [Wijnholds and Moortgat, 2021], all of which contain the same sentence pairs. Thanks to the alignment, direct comparisons can be made between the model's behaviour on different languages.

We carry out the comparison using two contrasting models: Olmo 3 7B Instruct [Olmo et al., 2025], a model trained almost exclusively on English, serving as a near-monolingual baseline, which we will refer to as *Olmo*, and Tiny Aya Global [Salamanca et al., 2026], which was explicitly trained to be multilingual, including all four languages studied in this thesis, which we will refer to as *Aya*. Comparing a monolingual model with a multilingual one allows us to test whether multilingual training yields representations that are more uniform or more strongly aligned across languages. In order to determine where in the network entailment information is encoded, we train a separate probe at each layer and compare it against a control task following Hewitt and Liang [2019].

Our results indicate that both models do encode entailment information in their middle and later layers, and that this information is most alignable across languages in the middle layers, where probes transfer best between languages and where probes trained independently on different languages converge to the most similar weights. English, Spanish, and Dutch group closely together, while Japanese remains a consistent outlier, even in Aya, despite its explicitly multilingual training that included Japanese. This suggests that the script and typology of a language shape the alignment of its representations beyond what can be explained by training-data coverage alone. Alongside this finding, we also observe a clear gap between what the models encode internally

and what they can express when they are prompted: the direct performance of Olmo collapses in Japanese, yet a probe is still able to recover the entailment relation from its activations, which indicates that internal representations and surface task performance can differ substantially.

The remainder of this thesis is organised as follows. Section 2 positions the work within previous research on NLI, cross-lingual LLM capabilities, and probing. Section 3 describes the datasets, the two models, the probing setup, and the similarity metrics used throughout. Section 4 presents the experiments and their results, Section 5 discusses the limitations and directions for future work, and Section 6 offers a conclusion.

1.1 Research questions

From the background above, we formulate the research question: **To what extent is there a difference in LLM understanding of logical inferences in different languages?** This question can be divided into the following:

Q.1. What layers in an LLM present the highest NLI capabilities? Are these layers the same across different languages?

Q.2. How similar are internal representations of NLI concepts across different languages?

Q.3. Does multilingual pretraining change the way NLI is encoded in the models?

We study four languages that were chosen to span a range of typological distances from English: English, Spanish, Dutch, and Japanese [Chiswick and Miller, 2005]. Dutch is closely related (West Germanic, the same family as English), Spanish is moderately related (Romance, which is also Indo-European and shares the Latin script as well as many broad European syntactic patterns), and Japanese is highly dissimilar (a non-Indo-European Japonic language with a different writing system, word order, and agglutinative morphology). It has been observed that languages with larger typological distances experience higher degradation in zero-shot language transfer NLI tasks [Lauscher et al., 2020], so this choice of languages lets us observe if there is also a higher difference in internal representations between these languages.

2 Related work

This work draws on three strands of research: the study of natural language inference, the analysis of how LLMs represent and process multiple languages, and probing as an interpretability method. We review each and point out the gap that our work addresses.

2.1 Natural language inference

NLI has long served as a benchmark for natural language understanding. The release of large annotated corpora such as SNLI [Bowman et al., 2015] and SICK [Marelli et al., 2014b] established it as a standard task, and the current generation of LLMs has renewed interest in how well these models perform it [Noble et al., 2025, Jie et al., 2025]. The task has also been extended beyond English, both through purpose-built multilingual datasets such as XNLI [Conneau et al., 2018] and

through sentence-aligned translations of SICK into other languages. These resources have shown that NLI performance is uneven across languages: models struggle with tense and aspect in Chinese and Japanese [Jie et al., 2025], with word-order freedom in Dutch [Wijnholds and Moortgat, 2021], and with case particles in Japanese [Yanaka and Mineshima]. However, these studies assess the model's outputs, measuring whether the predicted label is correct; they do not look inside the model to ask how the entailment relation is represented, or whether that representation is shared across languages.

2.2 Cross-lingual capabilities and representations

Another body of work studies how LLMs handle multiple languages. Since most models are trained on data heavily skewed toward English [Qin et al., 2024, Blasi et al., 2022], their performance tends to be best in English and to degrade in lower-resource languages, with the degree of degradation depending both on the amount of pretraining data in a language [Li et al., 2024] and on its typological distance from the training languages [Lauscher et al., 2020]. Interpretability research has begun to ask how this behaviour arises internally. Several studies report that multilingual models rely on a largely shared, English-centred representation space, processing non-English inputs through an English-like "concept space" in the middle layers before mapping back to the target language in the final layers [Wendler et al., 2024, Schut et al., 2025, Zhao et al., 2024]; notably, even independently trained monolingual models can converge on the same circuit for certain tasks [Zhang et al., 2024]. Other studies paint a more nuanced picture, finding that models do not rely on a fully-aligned semantic space [Lim et al., 2025], that token embeddings can encode language identity as strongly as cross-lingual meaning depending on the model family [Wen-Yi and Mimno, 2023], and that models retain language-specific features [Deng et al., 2025] and language-specific decoding mechanisms in the later layers [Harrasse et al., 2026]. Closest to our setting, Xiao et al. [2025] probe internal representations across languages and find that a model's perception of its own knowledge boundaries is encoded in the middle-to-upper layers, with the cross-lingual differences following a linear structure. However, this line of work targets general semantic representations or specific syntactic and factual phenomena; the cross-lingual encoding of a reasoning task such as NLI has not been examined.

2.3 Probing

Probing investigates these internal representations by training a simple classifier on a model's frozen activations and taking its performance as a measure of how much task-relevant information those activations carry [Alain and Bengio, 2018, Belinkov et al., 2017] (a more in-depth explanation of probing can be found in Section 3.3). Early studies probed for linguistic structure such as part-of-speech and semantic tags, observing that lower layers favour part-of-speech information while higher layers favour semantics [Belinkov et al., 2017]. More recent studies have found the later layers of a network encode more complex and context-dependent information [Ju et al., 2024, Jin et al., 2025]. Probing has occasionally been applied to logical reasoning and inference: Pirozelli et al. [2023] find that the validity of a logical argument is decoded mostly in the higher layers of encoder-only models, and Manigrasso et al. [2024] localise logical deductions to the middle-to-late layers. The two models we study have barely been examined this way: to our knowledge only a single probing study exists for Olmo 3 [Reblitz-Richardson, 2026] and none for Tiny Aya Global.

However, the few probing studies of inference are confined to a single language, leaving open the question of whether NLI is encoded similarly across languages.

Taken together, NLI has been studied largely at the level of model outputs, cross-lingual interpretability has concentrated on general or syntactic representations rather than reasoning tasks, and the probing of inference has remained monolingual. We bring these strands together by probing how the entailment relation is encoded across four typologically varied languages in both a near-monolingual model (Olmo 3) and an explicitly multilingual one (Tiny Aya Global), and by directly comparing the resulting representations across languages.

3 Methods

This section goes into detail about the methods that will be used in this work.

3.1 Data

All experiments in this thesis make use of the SICK (Sentences Involving Compositional Knowledge) dataset [Marelli et al., 2014b], along with three translated versions of it. SICK is a commonly used benchmark in NLI research. SICK sentences feature a wide array of lexical, syntactic, and semantic phenomena, rather than focusing only on logical relations. Consider the pair in Example 1.

Example 1.

Premise:	A woman is wearing an Egyptian hat on her head
Hypothesis:	A woman is wearing an Egyptian headdress
Label:	*entailment*

It is not possible to correctly classify this pair as *entailment* through a simple logical rule; instead, it requires the model to understand that a headdress is a hat that is worn on the head. The same pair, but with the word "headdress" substituted for another noun such as "tunic", would get a different label. This makes SICK well-suited to probing whether a model represents the entailment relation itself rather than simply relying on surface cues.

Beyond this, the main reason for choosing SICK is that it has been translated into all four languages studied in this thesis. The datasets are SICK for English [Marelli et al., 2014b], SICK-ES for Spanish [Huertas-Tato et al., 2022], JSICK for Japanese [Yanaka and Mineshima], and SICK-NL for Dutch [Wijnholds and Moortgat, 2021]. Since the same sentence pairs are present in every language, we are able to directly compare the behaviour of a model on identical content across different languages. This is precisely what our research questions require, and it is something that most other NLI datasets do not allow.

The four versions differ in how they were produced and labelled. SICK-ES and SICK-NL were both created similarly, through machine translation followed by a manual revision of the sentences, and both keep the original SICK labels. JSICK, on the other hand, was translated entirely by hand and then re-annotated, so its labels are not identical to the English ones. For this reason, we use the JSICK labels for the Japanese task rather than the original ones, although the majority (86.6%) of pairs still keep the same label as in English. Table 1 shows relevant statistics about the four versions of SICK.

These datasets also have a number of limitations. They inherit the weaknesses of SICK, such as a limited range of topics and vocabulary, some cases of coreference ambiguity, and certain

Table 1: Descriptive statistics for the four versions of the SICK dataset. Lexical overlap is the average Jaccard similarity ($|A \cap B|/|A \cup B|$) between the lowercased token sets of each premise-hypothesis pair.

Statistic	**SICK (EN)**	**SICK-ES**	**SICK-NL**	**JSICK**
Total Size	9840	9840	9840	9840
Train Set Size	4439	4439	4439	4439
Test Set Size	4906	4906	4906	4906
Validation Set Size	495	495	495	495
Label Ratios				
Entailment	28.7%	28.7%	28.7%	21.8%
Neutral	56.9%	56.9%	56.9%	62.1%
Contradiction	14.5%	14.5%	14.5%	16.1%
Label Alignment	100%	100%	100%	86.6%
(vs. English labels)				
Lexical Overlap				
Avg. Lexical Overlap	47.4%	40.5%	39.7%	38.5%
Entailment	61.0%	58.2%	53.2%	60.5%
Neutral	35.7%	28.4%	28.6%	26.5%
Contradiction	66.3%	53.3%	56.9%	55.1%
Token Counts				
Avg. Tokens (Premise)	10.3	12.0	12.3	12.1
Entailment	10.5	12.3	12.5	12.2
Neutral	10.3	12.1	12.3	12.2
Contradiction	9.6	11.1	11.7	11.5
Avg. Tokens (Hypothesis)	10.0	11.7	12.0	11.8
Entailment	9.6	11.0	11.3	10.9
Neutral	10.4	12.1	12.4	12.2
Contradiction	9.6	11.1	11.8	11.5

annotation artefacts; the artefacts are expanded upon in Section 5.1. The translated versions additionally introduce occasional translation errors, especially in the case of SICK-ES. However, we do not consider these limitations to be severe enough to discard the datasets due to the benefit of being able to carry out a more controlled cross-lingual comparison, and because they are, to the best of our knowledge, the only aligned versions of SICK available for these four languages. We discuss the impact of these limitations in more detail in Section 5.1.

3.2 Models

We investigate two different models: Olmo 3 7B Instruct and Tiny Aya Global. These two models were deliberately chosen to contrast a model trained to be monolingual with one trained to be explicitly multilingual. If multilingual training produces NLI representations that are more uniform

Language	Aya	Olmo
English	13.8%	100%*
Spanish	1.7%	0%*
Dutch	1.6%	0%*
Japanese	1.8%	0%*

Table 2: Proportion of Aya and Olmo's training data accounted for by each of the four languages studied in this thesis, according to [Salamanca et al., 2026]. The four languages together make up 18.9% of the overall mix, with the remainder distributed across other languages.
*The Olmo paper does not specify the per-language proportion of training data, but it assumes that the vast majority of its data is in English, as they implemented filters to filter out non-English texts

or more strongly aligned across languages, we would expect this difference to surface most clearly when comparing these two models.

Olmo 3 7B Instruct [Olmo et al., 2025] is a fully-open LLM with 7 billion parameters, meaning that its weights, training data, and training pipeline are all publicly documented. It was trained as a monolingual English model. Its base model is pretrained on the Dolma 3 Mix corpus [Soldaini et al., 2024], whose web-cleaning pipeline removes, among other things, documents with too many symbols or too few alphabetic characters, applies a language filter that keeps only documents identified as English, and then applies sentence-level cleaning heuristics. The instruction-tuning stages that produce the Instruct variant build on a predominantly English data mixture inherited from the earlier OLMo 2 instruction data.[1]

Although this language filtering is designed to target all non-English texts, it may have removed Japanese more thoroughly than Spanish or Dutch, since Spanish and Dutch share the Latin script and have many cognates with English, while Japanese uses a completely different script and has fewer cognates. Olmo therefore serves as a near-monolingual baseline in which we would expect strong English performance, weaker performance in Spanish and Dutch, and the weakest performance in Japanese.

Tiny Aya Global [Salamanca et al., 2026] is an open-weight model with 3.35 billion parameters that, unlike Olmo, was explicitly trained to be multilingual, including all four of the languages studied in this thesis. The share of the training data devoted to each is given in Table 2.

This deliberate multilingual training makes Aya especially relevant to our research questions: since the model has been exposed to all four languages, we would expect it to display more uniform performance across languages, and potentially more strongly aligned internal representations, than a model such as Olmo. Whether this expectation holds is not obvious a priori. Some studies have found that multilingual models develop representations that are well aligned across languages [Zhang et al., 2024, Harrasse et al., 2026], commonly using an English-aligned latent space [Schut et al., 2025, Wendler et al., 2024], while others report that such alignment is weaker or more superficial than it first appears [Lim et al., 2025]. Comparing Aya against Olmo lets us examine which of these pictures better describes the encoding of NLI specifically.

[1]A further filter that drops responses with a high proportion of Chinese characters is applied only to the Think variant of Olmo 3 and does not concern the Instruct model used here.

3.3 Probing

Probing is an Explainable AI method that has been used to assess models' NLI capabilities [Manigrasso et al., 2024, Pirozelli et al., 2023]. This method sheds light on the internal representations of LLMs. A simple model (probe) is inserted on top of one of the layers of an LLM, all the other parameters are frozen, and the probe is then trained on the embeddings of the layer to perform a particular task. It is taken as an indication that the LLM internal representations may encode some information about the task at that layer if the probe can make predictions with good performance, usually compared to a control task [Hewitt and Liang, 2019].

In this work, the probes are used to gauge the degree to which different layers encode NLI knowledge in their activations. A different probe is trained with the output of each full transformer decoder block (we expand on this in Section 3.3.1), which we will henceforth refer to as a layer. The input to the probe is a vector containing the activations at this particular layer; the output is a number (0, 1, or 2) representing one of the following three classes: entailment, neutral, or contradiction. These predictions are then compared with the gold-standard labels of SICK. This task of determining the entailment relation present in the sentence pair will henceforth be referred to as the *standard task*.

A control probe uses the same inputs and architecture as the standard probe but is trained on arbitrary labels. We assign control labels by extracting the first noun of each pair (premise first), randomly mapping every distinct noun to one of the three labels with equal probability, and labelling each pair by its first noun. A fully random labelling would give the control nothing to learn, so this grouping instead provides learnable but meaningless structure: if the control matches the standard probe, the standard probe too is only exploiting incidental structure rather than encoded NLI. Because different random groupings vary in difficulty, we generate the control labels once on English and reuse them across all languages.

Before describing the individual probe types, we fix the notation used for probes throughout the rest of this thesis. We refer to the four languages by the two-letter codes en (English), es (Spanish), nl (Dutch), and jp (Japanese), and write probes as follows:

- $P_L^T(l)$ denotes a single probe, where L is the language on whose activations it was trained, l is the layer those activations were taken from, and the superscript T denotes the probing task.
- $T = S$ indicates the standard task, in which the probe is trained on the gold SICK entailment labels, and $T = C$ the control task, in which it is trained on the control labels defined above.
- P_L^T drops the layer argument and refers to a probe across all layers rather than at one specific layer. We drop the layer argument in the same way for all the variants below.
- $P_{A\to B}^T(l)$, used in the cross-lingual experiments of Section 4.3, is a probe trained on a source language A and tested on a target language B (in some of these experiments after briefly retraining it on B).
- $P_{A\to B\to A}^T(l)$ is such a transfer probe after retraining on B but evaluated back on its original language A.
- $P_{A\to x\vee y}^T(l) = \{P_{A\to x}^T(l),\ P_{A\to y}^T(l)\}$ uses an *or* symbol in the subscript as shorthand for the set of transfer probes that share a source language and differ only in their target.

- $P^T_{x\vee y\to B}(l) = \{P^T_{x\to B}(l),\ P^T_{y\to B}(l)\}$ is the symmetrical counterpart of the previous notation. It refers to the set of probes that share a target language and differ only in their source.

These transfer variants are defined only for the logistic regression probes, since the cross-lingual experiments that rely on retraining are only carried out with logistic regression; the mass-mean probes are optimisation-free and are therefore not retrained.

3.3.1 Recording activations

The first step for probing is to save the model activations for each layer as the sentence pairs are processed. For every sentence pair, both models received the same prompt, although it was adapted to the models' respective standard chat templates. The templates can be found in the appendix (A).

The activations are saved for each layer; more precisely, at the output of each full transformer decoder block, after both the self-attention sublayer and the MLP sublayer have been applied, and their contributions have been added to the residual stream. For each prompt, 4 new tokens are generated. This is because in the tokenisation used for Spanish and Japanese, a minimum of 4 tokens was required to allow the model to generate the correct labels. However, we only record the activations of the last token of the prompt, which is the position at which the model has integrated the full premise–hypothesis pair and must commit to an answer and is therefore the most interesting token position to investigate.

We went through the SICK dataset in all four languages for both models. For each sentence pair, we prompted the model and saved the activations at each of the layers, as well as the model's answer. Inference was run deterministically.

3.3.2 Logistic regression probes

The first type of probe is multinomial logistic regression, using the *LogisticRegression* model from the Python *SKlearn* package. Each probe takes the flattened activation vector $\vec{x}$ of size n, equal to the model's activation size, and computes a score per class,

$$\hat{s}_i(\vec{x}) = \vec{w}_i \cdot \vec{x} + c_i \tag{1}$$

where $\vec{w}_i$ and c_i are the weight vector and intercept for class i. These scores are turned into probabilities by the softmax (Equation 2),

$$p_i(\vec{x}) = \frac{\exp(\hat{s}_i(\vec{x}))}{\sum_j \exp(\hat{s}^j(\vec{x}))} \tag{2}$$

and the weights are fit by minimising the L2-regularised cross-entropy loss with the `lbfgs` solver, with classes weighted inversely to their frequency to counteract the label imbalance in SICK. Because we use `lbfgs` with three classes, the probe is fit as a single multinomial model rather than as independent one-vs-rest classifiers, so the three class vectors are estimated jointly. The predicted class is the one with the highest probability, which is always the one with the highest score (Equation 3), so the decision function is linear in the activations, making this a linear probe in the standard sense.

$$y(\vec{x}) = \arg\max_i \hat{s}_i(\vec{x}) \tag{3}$$

We use logistic regression for three reasons. First, it is low-capacity and purely linear, so strong performance cannot be credited to the probe learning the task and must instead reflect entailment information already encoded as a linearly separable structure in the activations [Alain and Bengio, 2018], which is in turn usable by the model's own largely linear downstream computation. The simple-versus-powerful-probe trade-off is debated [Pimentel et al., 2020], but there is a second advantage to using simple linear probes: a linear probe identifies the concept with a direction in activation space, letting us compare probe weights across layers and languages (Section 4.4); this cannot be easily done with complex probes. Third, logistic regression has been previously used in probing studies [Marks and Tegmark, 2024].

The probe hyperparameters, namely the regularisation strength and whether an intercept is included, are fixed to a single setting shared by all probes, chosen as the configuration most frequently selected when each layer and language was tuned separately. We hold them fixed because Section 4.4 compares the weights of probes trained at different layers and on different languages directly, and that comparison is only meaningful if every probe is the same kind of model: were each probe given its own hyperparameters, any difference in their weights could reflect differing regularisation rather than a genuine difference in what was learned from the activations.

3.3.3 Mass-mean probes

As a second, methodologically different probe, we use mass-mean probes [Marks and Tegmark, 2024]. A mass-mean probe is optimisation-free: rather than fitting weights by minimising a loss, its direction is the difference between the class means, corrected by the inverse of the within-class covariance so that high-variance but non-discriminative directions are down-weighted. A point is classified by which side of the midpoint between the class means it falls on along this direction. We refer to Marks and Tegmark [2024] for the full derivation.

Mass-mean probing was introduced for binary classification and, to our knowledge, has not previously been applied to a multi-class task. We adapt it using the fact that the binary mass-mean probe is equivalent to linear discriminant analysis between the two classes [Marks and Tegmark, 2024], which generalises naturally to more than two classes. We compute a mean $\vec{\mu}_i$ for each class and a single within-class covariance Σ shared across all classes, and give each class the score

$$\hat{s}_i(\vec{x}) = \vec{x}^\top \Sigma^{-1} \vec{\mu}_i - \tfrac{1}{2}\, \vec{\mu}_i^\top \Sigma^{-1} \vec{\mu}_i, \tag{4}$$

predicting the class with the highest score via Equation 3. Intuitively, $\hat{s}_i(x)$ measures how close x is to the mean of class i, using the same covariance correction as the binary probe to discount high-variance directions; a point is assigned to the nearest class mean in this corrected geometry.

We include this probe alongside logistic regression as an independent check: their simplicity makes these probes easy to interpret with our tools. Additionally, Marks and Tegmark [2024] report that difference-in-means directions of this kind are more causally implicated in model behaviour than those found by logistic regression.

3.4 Metrics

Throughout the experiment results, we report F1 and selectivity as the main indicators of probe performance; whenever F1 is mentioned, it is macro-F1. No significance values are reported. We performed significance tests, but with 4906 sentence pairs per language, the test set is large enough that paired bootstrap tests flag almost every comparison as significant regardless of how small the underlying difference is, so significance does little to separate substantial differences from negligible ones; effect sizes are more informative for our questions, and therefore we report these.

In Section 4.4, we report similarity metrics rather than probe performance. For this, we mainly use cosine similarity, with Euclidean distance reported as a complementary measure.

Cosine similarity Cosine similarity measures the angle between two weight vectors. It is described by Equation 5.

$$\cos(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{\|\vec{x}\| \, \|\vec{y}\|} \tag{5}$$

It is invariant to vector magnitude, so two probes that respond to the same feature with different scales are still judged similar. Comparing learned directions by cosine similarity is standard in representation-space analyses [Park et al., 2024, Arditi et al., 2024]; it is justified under the linear representation hypothesis, although Steck et al. [2024] warns about simply using cosine similarity, as it may produce arbitrary results in some circumstances.

Euclidean distance Euclidean distance complements cosine similarity by being sensitive to magnitude: two probes can be near-parallel yet differ substantially in scale, which cosine similarity alone would not reveal. It is described by Equation 6.

$$d(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^{n} |x_i - y_i|^2} \tag{6}$$

Mahalanobis cosine similarity is a variant of cosine similarity created to be more resistant to situations where variance is concentrated along a small number of directions [Ying et al., 2026]. We considered adding it as a measure, but ultimately chose to keep it out of comparisons, as the type of loss used for logistic regression probes means that the probe weights are regularised and therefore the probe weights are not heavily concentrated along a small number of directions, therefore making Mahalanobis cosine similarity not as useful.

4 Experiments and results

In this section, we present the results of our experiments, structured to address the research questions defined in Section 1.1. We begin by establishing a baseline for the models' surface-level task competence through direct prompting (Section 4.1), which addresses the practical performance gap across languages (Q.1, Q.3). We then transition to internal analysis using within-language probing (Section 4.2) to identify which layers most strongly encode NLI information (Q.1). To address the similarity of these representations across languages (Q.2) and the influence of multilingual

Olmo:

	f1	f1 for entailment	f1 for neutral	f1 for contradiction	unk count	baseline f1
en	0.70	0.89	0.67	0.54	0	0.24
es	0.55	0.71	0.47	0.46	46	0.24
jp	0.17	0.17	0.05	0.29	125	0.26
nl	0.57	0.41	0.72	0.59	0	0.24

Aya:

	f1	f1 for entailment	f1 for neutral	f1 for contradiction	unk count	baseline f1
en	0.66	0.77	0.83	0.36	0	0.24
es	0.77	0.85	0.79	0.67	0	0.24
jp	0.62	0.69	0.59	0.57	0	0.26
nl	0.58	0.80	0.80	0.14	0	0.24

Table 3: Direct-response F1 of Olmo (top) and Aya (bottom) on the SICK test set in English (en), Spanish (es), Japanese (jp), and Dutch (nl). *f1* is the macro F1, *f1 for class* is the per-class F1s for each particular class, *unk count* is the number of responses that could not be mapped to a label and are therefore excluded from the F1, and *baseline f1* is the F1 of the majority-class baseline. Olmo is strongly language-dependent and drops below the baseline in Japanese, whereas Aya is far more uniform.

pretraining (Q.3), we perform cross-lingual probe transfer (Section 4.3) and a direct comparison of probe weights (Section 4.4).

4.1 Direct prompting performance

In this experiment, we check the responses generated by the models when prompted. For each sentence pair, the models were instructed to reply with one single word: entailment, neutral, or contradiction. There were cases where the models generated a response that did not adhere to the instructions to varying degrees. In cases where the response contained a long enough substring that resembled one of the three labels, it was classified as such. For instance, "contradicts" was classified as a contradiction, but "contra" was not. The specific accepted forms can be found in Appendix B. In cases where the model response contained two different labels, was empty, or was not similar enough to any of the labels, it was marked as unknown. Unknown labels are not counted toward F1 calculations.

Table 3 (top) shows the F1 of Olmo responses across the test set. The performance was highly dependent on the language: the highest performance was for English, achieving 0.70. This is followed by Dutch and Spanish, with 0.57 and 0.55, respectively. Finally, Japanese has a significantly lower score of 0.17. The F1s for English, Spanish and Dutch all lie above the F1 that would be obtained by always guessing neutral, which is the most common label (0.24 for English, Spanish, and Dutch; 0.26 for Japanese), but the F1 for Japanese lies below this score. It is therefore difficult to claim that Olmo's responses in Japanese capture any NLI capabilities.

The model was able to generate responses that could be mapped to one of the three labels for all sentence pairs in English and Dutch. For Spanish, it only failed to do so for 46 pairs (0.94% of all test set pairs), and for Japanese, it failed on 125 (2.55% of all pairs).

While the scores for Spanish and Dutch are not as high as for English, it is remarkable that Olmo achieves these even though the creators explicitly tried to avoid non-English data for its training set. There are two possibilities for why Olmo is not performing as badly for Spanish and Dutch: because these languages share the Latin script and are more typologically similar to English, the model may pick up enough words to perform the tasks. The other possibility is that enough Spanish and Dutch data leaked into the training dataset to allow the model to generate valid responses. Japanese may have leaked less into the training set, leading to much worse performance.

Aya's responses (Table 3, bottom) were far more uniform across languages than Olmo's, which is consistent with its explicitly multilingual training. The highest F1 was for Spanish (0.77), followed by English (0.66), Japanese (0.62), and Dutch (0.58); all four lie well above the majority-class baseline. Unlike Olmo, Aya produced a mappable label for every pair in all four languages, with no unmappable responses; nevertheless, some responses did not fully adhere to the prompt: in English, it responded "contradition" instead of "contradiction" to six sentence pairs; in Dutch, it commonly answered "neutral" instead of "neutraal", which would be the correct Dutch spelling, and consistently answered "tegensp" for contradiction, while the correct word is "tegenspraak". In these cases, we still assigned a label, since the focus of this experiment is to test whether the model can answer with the correct entailment, not whether it can fully adhere to the prompt structure. It is, however, an indicator that Aya may struggle to adhere to instructions in Dutch. Notably, the per-class contradiction F1 is low for English (0.36) and especially low for Dutch (0.14), suggesting that most of Aya's errors come from failing to recognise contradictions rather than from a general inability to follow the task. The absence of the large Japanese collapse seen in Olmo indicates that Aya's multilingual training gives it a more even, if still imperfect, command of the task across languages.

4.2 Within-language probing

In this experiment, we train a different probe at each layer of both models. Each probe is trained exclusively on the training set of model activations in one language. Then, the probe is tested on the test split of that same language. The purpose of this experiment is to ascertain the degree to which activations at each of the layers encode information regarding the entailment relation between the two sentences: if the probe at a particular layer is capable of consistently predicting the correct entailment label, this is taken as evidence that the entailment information is linearly retrievable at this particular layer.

- In the first part, we look at the F1 obtained by the standard task probes and compare it to the results observed in Experiment 4.1.
- In the second part, following [Hewitt and Liang, 2019], we summarise each layer with its *selectivity*. Selectivity is defined as the difference between the standard-task F1 and the control-task F1 at that layer. High selectivity means the activations carry NLI information that a low-capacity probe can extract beyond what it could achieve simply from incidental structure.
- In the third part, we corroborate the results using a mass-mean probe.

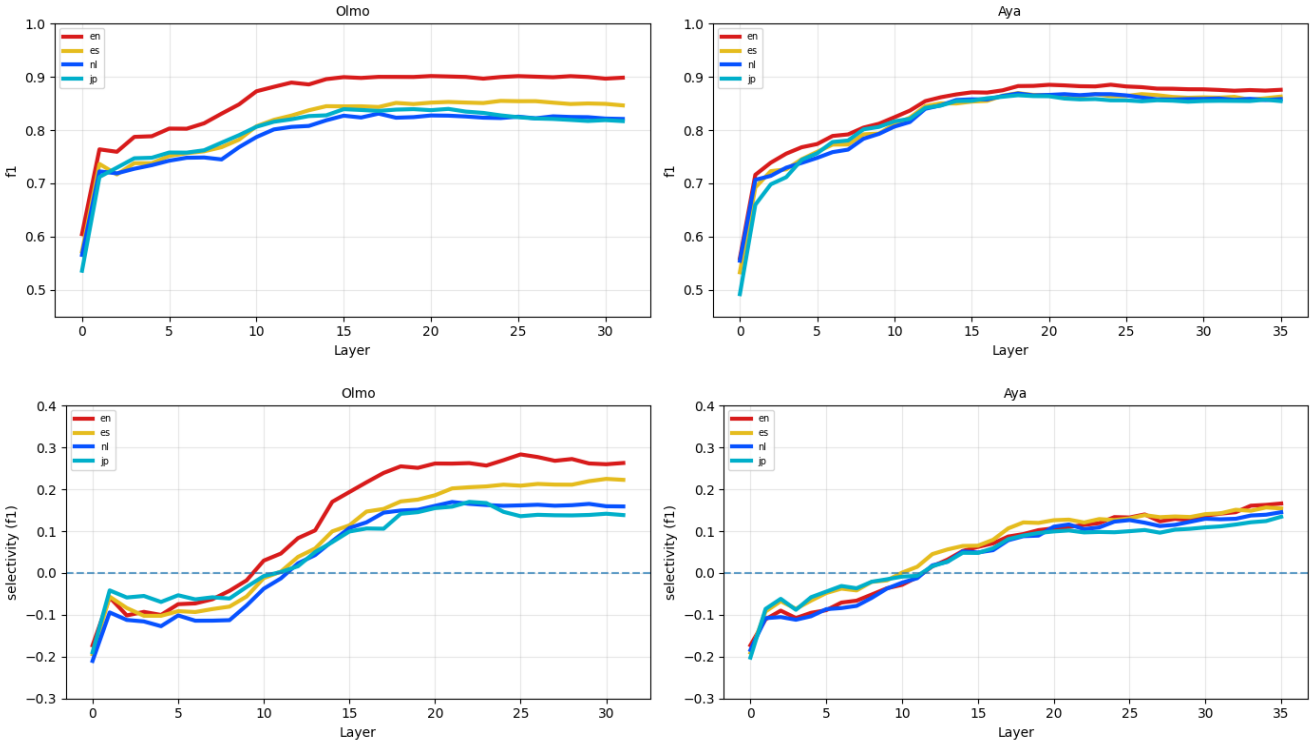

Figure 1: Logistic regression probes trained and tested within the same language at each layer for Olmo (left) and Aya (right); each line corresponds to one language. (Top) Standard-task F1: low in the earliest layers, rising sharply before levelling off through the middle and later layers, with the language curves much closer together than the direct-prompting scores. (Bottom) Selectivity (standard-task minus control-task F1): negative in the early layers, crossing zero around layer 10 before rising to a positive plateau in the middle-to-late layers, more tightly grouped for Aya than for Olmo and weakest for Japanese in Olmo.

4.2.1 Standard-task probe performance

The top panel of Figure 1 shows the F1 of Olmo and Aya logistic regression standard task probes trained at each layer. Each line represents a different set of probes that were trained and tested in one particular language. In all cases, a low initial F1 can be observed, which rises sharply before slowly levelling off in the middle layers and maintaining a stable score until the final layer.

Olmo At the final layer, P^S_{en}, P^S_{es}, P^S_{nl}, and P^S_{jp} received F1s of 0.90, 0.85, 0.82, and 0.82. These are all higher than the F1s obtained in Experiment 4.1 by directly looking at the model predictions, which were 0.70, 0.55, 0.57, and 0.17 respectively, and they are also much more similar across languages. English probes exhibited the best performance, followed by Spanish, and finally Dutch and Japanese tied with each other. The model predictions were also best for English, with Spanish and Dutch achieving worse performance, but Japanese was drastically worse than any other, which is not observed when looking at the probes.

Aya Aya exhibits the same difference in performance between the final layer probes and the actual model predictions, having F1s of 0.88, 0.86, 0.86, and 0.85 for the former and F1s of 0.66, 0.77, 0.58, and 0.62 for the latter. The probe F1s in the final layer are even more similar between the four languages than in Olmo. Aya was particularly good at responding in Spanish, which is not reflected in the probe performance.

Interpretation of results The high F1 of probes in the final layer suggests that the information necessary to correctly determine the sentence entailment is present in the final layer of both models, but that the models do not make full use of this information, leading them to produce incorrect responses. There are, however, alternative explanations for this gap. The probe reads the last-token activations directly and is trained to extract the entailment label, whereas the model must decode that information through its unembedding and commit to a single output word; the gap may therefore reflect the probe being a more favourable readout than next-token prediction, rather than the model failing to use the information. Some of it is also attributable to format and instruction-following failures that penalise the direct responses but not the probe, as seen in Section 4.1. The much smaller difference in F1s between languages, compared to the model predictions, is also notable. Olmo may have incidentally developed enough Japanese capabilities during its training to become able to encode entailment relations in its activations, while still not being proficient enough to properly understand the task it was given by the prompt or to actually generate the right words that express entailment relations in Japanese. In the case of Aya, the even greater similarity between languages may be due to the model having been purposefully trained in all four languages.

4.2.2 Selectivity against the control task

The bottom panel of Figure 1 shows the selectivity of Olmo and Aya logistic regression probes trained at each layer.

Olmo Olmo probes start off with negative selectivities of around -0.2; this is due to the control probes being especially good in the early layers, but then decreasing in the middle layers. The selectivity rises over the layers, becoming positive at around layer 10–12 depending on the language. After this point, the selectivity keeps increasing before levelling off in the middle to late layers, finally reaching 0.26, 0.22, 0.16, and 0.14 for English, Spanish, Dutch, and Japanese, respectively.

Aya Aya follows a similar pattern but reaches a lower selectivity, starting off in the negatives but crossing the $y = 0$ line at around layer 10–12 and then maintaining itself at around 0.14–0.16. Notably, the probes for different languages were closer together than they were for Olmo. English got the highest selectivity in the final layer, but only by a small margin, and Spanish had the lead in the middle layers. Aya probes reach an overall lower selectivity than Olmo probes, but it is due to higher control F1 and not due to lower standard F1.

Interpretation of results These results indicate that the early layers do not possess any inherent NLI capabilities, but that in the middle-to-late layers, information regarding entailment is stored in a way that is more linearly extractable than an arbitrary pattern. In Olmo, this information is

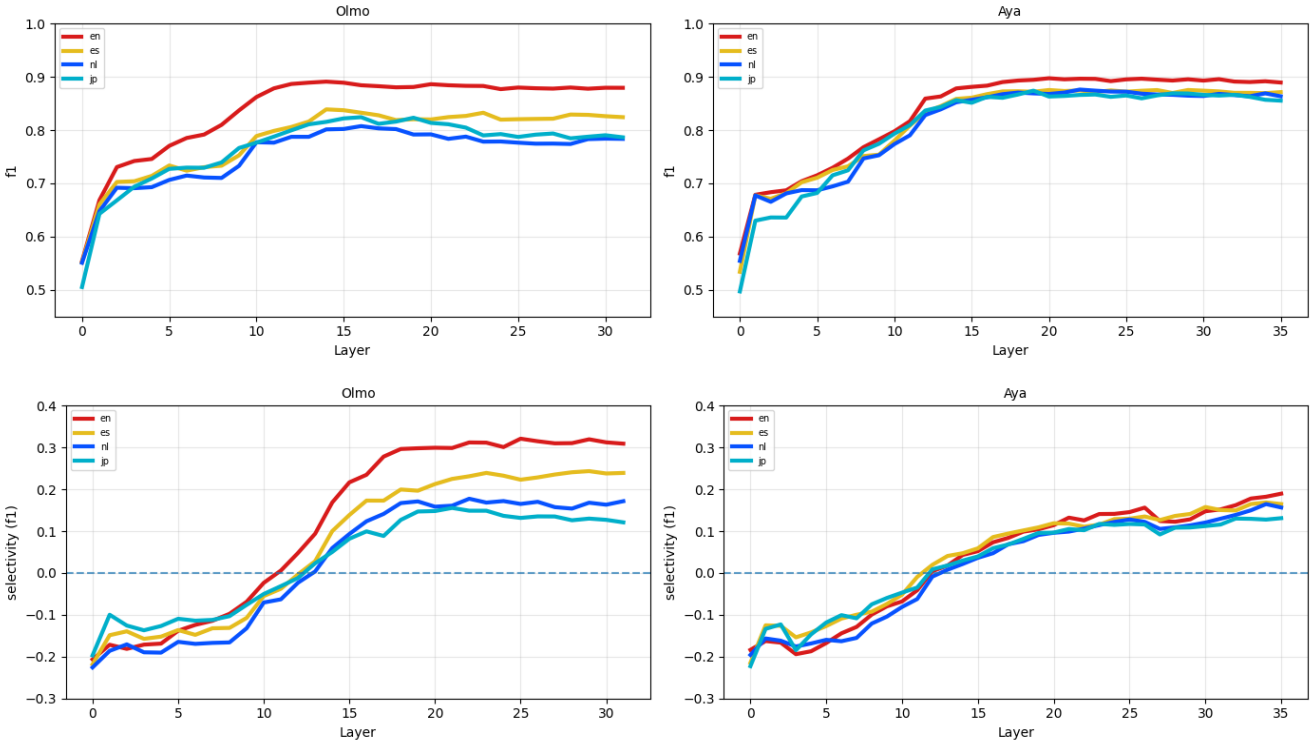

Figure 2: Mass-mean probes trained and tested within the same language at each layer for Olmo (left) and Aya (right); standard-task F1 (top) and selectivity (bottom), laid out as in Figure 1. Both panels closely follow the logistic regression results: F1 rises from the early layers to a plateau, and selectivity turns positive in the middle-to-late layers, remaining weakest for Japanese in Olmo and tightly grouped across languages in Aya.

stored most strongly in English and most weakly in Japanese. In Aya, by contrast, the closeness of the language curves indicates that the model stores NLI concepts with roughly equal strength for all four studied languages. That Aya's lower selectivity stems from a higher control F1 rather than a lower standard F1 indicates that NLI concepts are still extractable from its activations more than arbitrary concepts, but to a smaller degree than for Olmo.

4.2.3 Corroboration using mass-mean probes

We also experimented with using mass-mean probes instead of logistic regression. The F1 results can be observed in the top panel of Figure 2 and the selectivity results in the bottom panel. In both models, the F1 is low in the early layers but steadily rises before levelling off in the middle-late layers, very similarly to logistic regression probes.

Olmo At the final layer, the F1s for English, Spanish, Dutch, and Japanese were 0.88, 0.82, 0.78, and 0.79, respectively; as in logistic regression, English obtained the best score, while the rest were behind. The selectivity is negative in the early layers, crosses the $y = 0$ line at layers 11–13, and reaches the highest final value for English.

Aya The F1 results for Aya are highly similar to those obtained by logistic regression probes. The selectivity is likewise negative in the early layers, crosses the $y = 0$ line at layers 12–13, and reaches nearly identical values for all languages.

Interpretation of results As in logistic regression, the languages other than English were behind in Olmo, but not to the extent that was observed in direct probing. Regardless of which type of probe tells a more accurate story of how entailment is stored in activations, both point towards the same idea that the middle and late layers in both models are indeed capable of storing NLI concepts for all languages.

4.3 Cross-lingual probe transfer

In this experiment, rather than training and testing a probe on the same language, the probe is trained on the activations of one language, the source language A, and then tested on the activations of a different one, the target language B, following the transfer notation $P^T_{A\to B}$ introduced in Section 3.3. The previous experiment showed that both Olmo and Aya encode NLI knowledge in their layers to varying extents. The idea behind this experiment is to determine the similarity in these representations across different languages: if a probe at a particular layer trained on A is able to perform well on B, it means that A contains NLI knowledge representations that are sufficiently similar to those in B such that a probe that was exclusively trained on A would be able to extract them.

Setup We will split this experiment into three sub-experiments: in the first one, we will directly take the probe trained in A and test it in B; in the second one, we will retrain the probe on B for 2 iterations before testing it; in the third one, we will train the probe on B for 1000 iterations or until convergence before testing it. An iteration is a full pass through the training set of activations.

- The first sub-experiment, directly transferring the probe, is the strictest test of cross-lingual alignment: the probe is given no opportunity to adapt to B, so it can only perform well if the NLI representations of A and B are already similar enough that the exact same weights extract the entailment relation in both. Strong direct transfer at a given layer therefore indicates that the model encodes entailment in a closely aligned, and thus largely language-agnostic, way at that layer, whereas poor transfer indicates that the representations differ between the two languages.

- The second sub-experiment, retraining for 2 iterations on B, is motivated by two considerations. Firstly, the models may encode NLI in the same directions in the activation space for different languages while still differing enough that a probe needs to slightly readjust its weights before it is able to capture them. If this is the case, even a small tweak to the probe weights should drastically improve performance on B; if, on the contrary, the encoding is too different between languages, the probe will need large changes to its weights to work on B. Secondly, after retraining on B we can test the probe again on A: if its performance on A does not significantly degrade, this indicates that the model's NLI encodings are sufficiently similar across languages to be linearly decoded with the same weights, providing further evidence that the models store NLI concepts similarly across languages. We settled on 2 iterations by

retraining the probes for different numbers of iterations and observing their performance: 2 provides a good balance, keeping the probe highly similar to its original version while still allowing it enough retraining to adapt to the new language.

- The third sub-experiment, retraining for up to 1000 iterations on B (or until convergence), uses the same maximum number of iterations for which the probe was originally trained on A, which should be enough for the probe to adapt completely to B. Because the probe is not trained on the activations of A at any point during this process, it has no incentive to maintain good performance on A, so a degree of degradation on A can be expected, which would be especially large if determining the entailment relation in B required significant changes compared to doing so in A. The objective is to use the extent of this degradation as a measure of the similarity between the NLI probing tasks in the two languages: if the degradation is minor, the model encodes NLI concepts similarly enough across the languages that a single probe can extract the relation in both; if the degradation is large, it indicates that the probe was relying on features of the activations that were highly language-specific and were therefore lost during retraining, when there was no longer any incentive to preserve them.

In every case, we will compare the transferred probe F1s to the score that would be obtained by a simple majority-class baseline classifier. The reason for not comparing the probe to the control task probes is that the idea of selectivity as a strong metric weakens when the probe is transferred to another language: a low selectivity would not just depend on the real difference in capabilities between the standard and control probes, but also on the degree to which the control probes were able to adapt to the new language, therefore making selectivity potentially no longer as indicative of a concept being stored in the activations. To our knowledge, there has not yet been an in-depth investigation of the validity of selectivity in this kind of scenario.

In Section 4.2, the results were shown layerwise; however, to support a clearer visualisation, the models were split into *early*, *middle*, and *late* layers, defined as the ranges $[0, 10]$, $[11, 21]$, and $[22, 31]$ for Olmo, and $[0, 11]$, $[12, 23]$, and $[24, 35]$ for Aya. Within each layer group, the F1s of all layers were averaged.

We focus on logistic regression probes for this experiment, as mass-mean probes do not go through a training process and therefore do not fit into this experiment's structure.

4.3.1 Transfer without retraining

In this sub-experiment, the probe trained in A is directly tested on B.

Olmo Figure 3 shows the results of the experiment when performed in Olmo. In each subplot, all the probes were originally trained in one language A, and then tested in all other languages. We also include a bar showing the F1 in the original language, and a line showing the baseline F1. Only standard task probes are shown.

In all cases, early layers exhibit the lowest cross-lingual transfer capabilities. These probes do not rise far above the baseline and some perform even worse than it. The middle layers perform the best overall, but this is not the case for all probes. The late layers perform similarly well to the middle layers in probes trained in Spanish or Dutch, but perform worse than them in probes trained in English or Japanese.

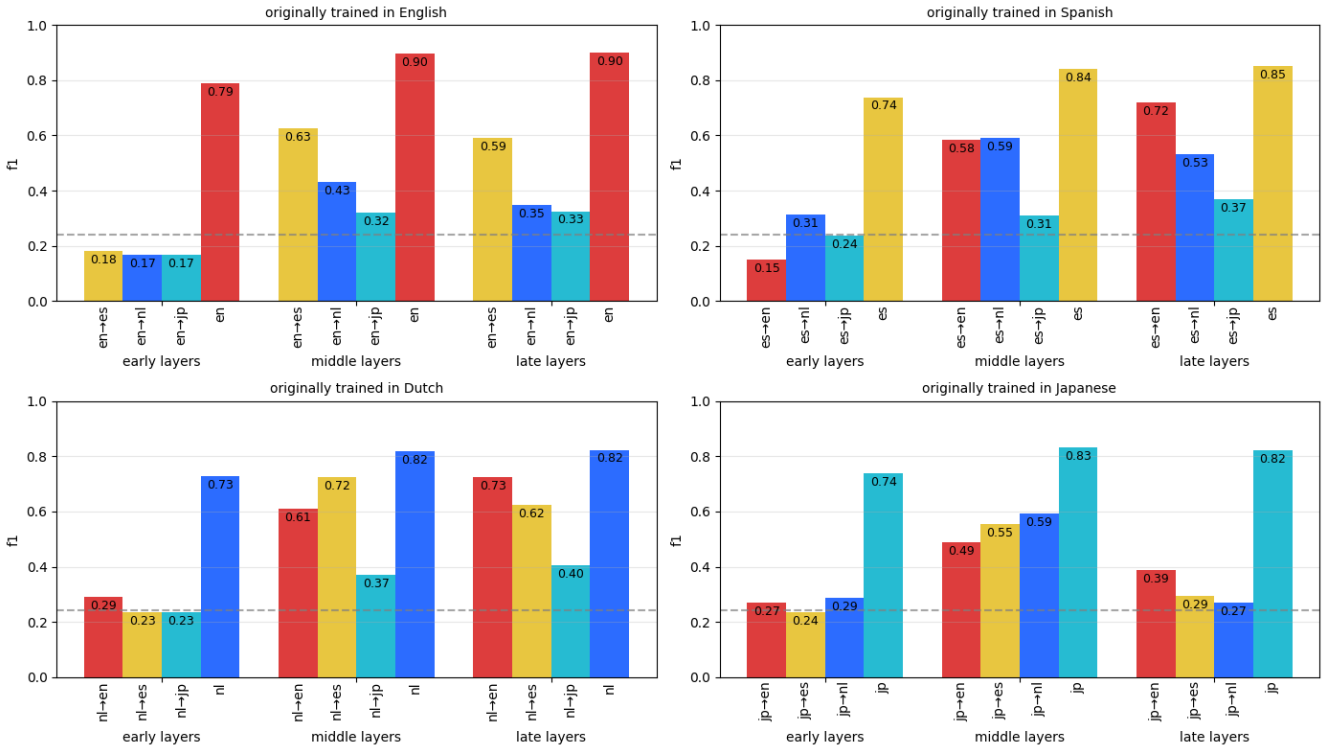

Figure 3: Direct cross-lingual transfer of the logistic regression probes in Olmo. Each subplot contains the probes trained on a single source language and tested on every language, as well as the probe trained and tested in that source language; the majority-class baseline is shown as a line. Transfer is weakest in the early layers and strongest in the middle layers, with Japanese as the clear outlier: probes trained on Japanese transfer well to the other languages in the middle layers, but probes trained on the other languages transfer poorly into Japanese.

As can be observed, the performance of the transferred probe varies depending on the target language. $P^S_{en \to es \vee nl \vee jp}$ exhibit low initial F1, lower than the baseline, but in the middle layers, $P^S_{en \to es}$ rises considerably and remains high in the late layers, while $P^S_{en \to nl \vee jp}$ experience a less substantial increase, suggesting that probes trained on English transfer more effectively to Spanish than to Dutch or Japanese. Meanwhile, $P^S_{es \to en \vee nl}$ and $P^S_{nl \to en \vee es}$ appear to transfer roughly equally well in the middle and late layers.

As can be observed, the performance of the transferred probe varies depending on the original training language. In the middle and late layers, probes trained on English, Spanish or Dutch transfer well to each other (except for $P^S_{en \to nl}$). In all cases, Japanese appears to follow a different pattern than all other languages. $P^S_{jp \to en \vee es \vee nl}$ transfer badly in early and late layers, but they transfer well in the middle layers. Meanwhile, $P^S_{en \vee es \vee nl \to jp}$ transfer badly in all layers, even though they correspond to the same language combinations as the other probes. This indicates that even though probes originally trained in Japanese middle layer can work in other languages, the opposite is not true; probes trained in other languages cannot directly transfer to Japanese.

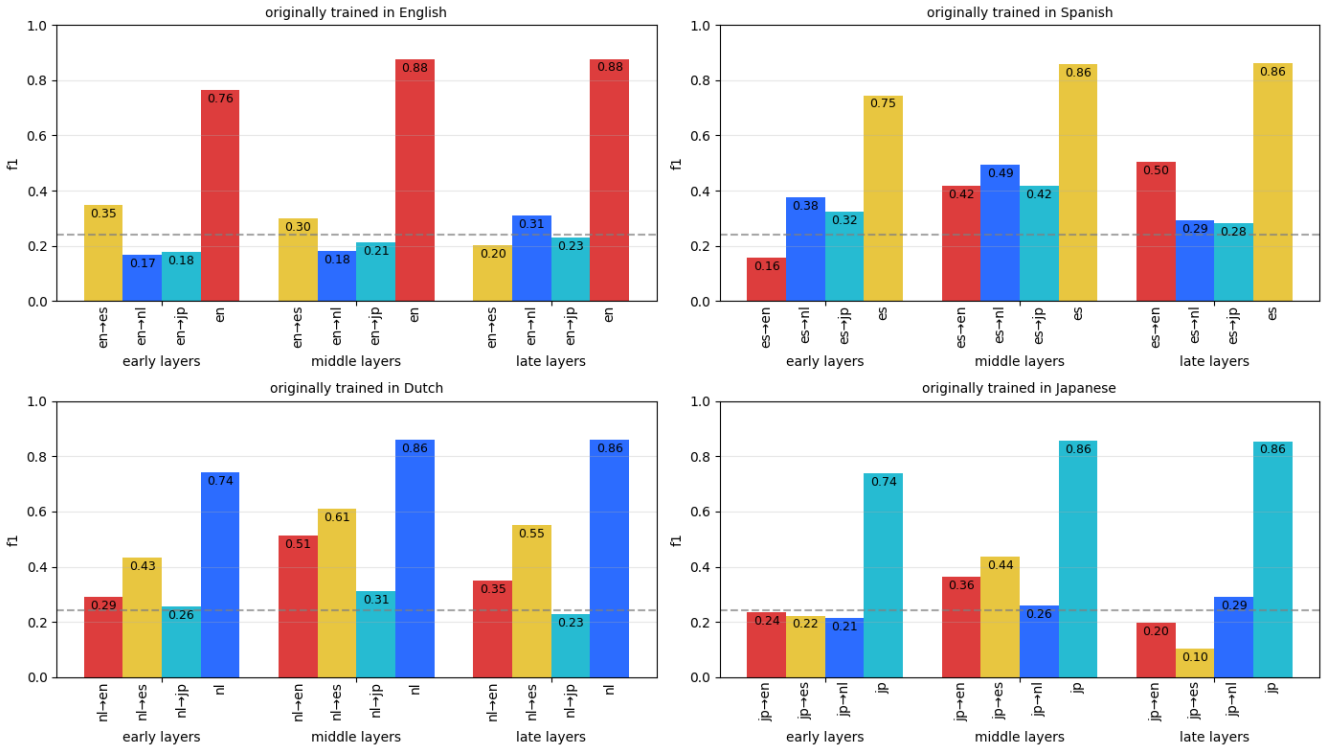

Figure 4: Direct cross-lingual transfer of the logistic regression probes in Aya, laid out as in Figure 3. The transfer curves are markedly more jagged than Olmo's and frequently fall below the majority-class baseline, making a consistent layerwise pattern harder to discern.

Aya Figure 4 shows the results of the experiment for Aya. Unlike in Figure 3, the results for Aya are less clear. While it is not appreciable in the bar plot, the performance in the bar plot between the layers was heavily inconsistent, with some layers performing significantly better than others despite being in the same layer region. By averaging over all layers in each region and creating the bar plot, it can be observed that there was overall less cross-lingual transfer in Aya than in Olmo, especially in probes trained in English and Japanese, and in probes from all languages in the late layers. However, in the following section we show that the poor performance observed here quickly disappears after just two iterations.

Interpretation of results We can identify multiple findings from our observations of Olmo. First, they show that probes exhibit different cross-lingual transfer capabilities depending on the language they were originally trained on, with Japanese being the most distinct case. This points towards **models having more language-agnostic representations of NLI in their activations for some languages like Japanese, while they have more language-specific representations for other languages like English**. This would explain the observed behaviour, as Japanese probes would learn to exploit these language-agnostic features and perform well in other languages, while English probes would have learnt some English-specific indicators of NLI, which would not serve if the probe is transferred to Japanese. The consistently high transferability between Spanish and Dutch suggests that their activations encode NLI in mostly the same dimensions. Second, the

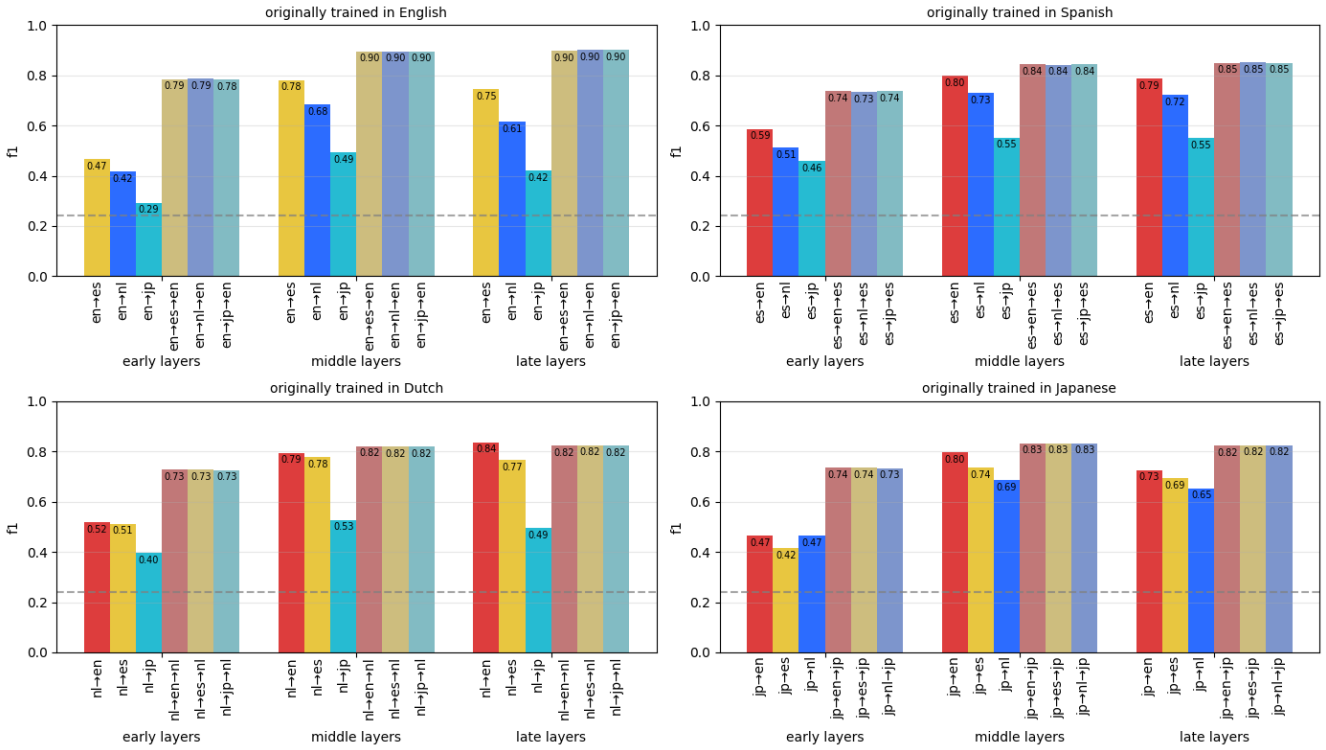

Figure 5: Cross-lingual transfer in Olmo after brief retraining. Each subplot corresponds to a source language A; three bars show the probes that were trained in A, retrained for two iterations on every other language B, and then tested in B. Three other bars show the same retrained probes, but tested on A. The majority-class baseline is also included for reference. The retrained weights remain almost identical to the originals.

cross-lingual transfer is especially strong in the middle and late layers, and very weak in the early layers.

4.3.2 Transfer after brief retraining

In this sub-experiment, we retrain the probes for 2 iterations on B before testing them. The retraining process involves taking the existing probe and adjusting its weights on the train set of B, with each iteration being a full pass. We calculated the cosine similarity between each retrained probe and their original version, and the lowest cosine similarity was of 0.998, although the majority stayed above 0.999, indicating that the change in weights was minimal.

Olmo Figure 5 shows the performance of probes originally trained on A after being retrained on B. The first three bars in each cluster are the F1 of the probes when tested on B, and the last three bars corresponds to the F1 of the same probes tested back on A. Compared to Figure 3, the bars are higher overall, with some probes showing substantial improvements. The early layers still suffer from lower performance, but the difference with the other layer groups is less intense. In the middle and late layers, $P^S_{es \vee nl \to en}$, $P^S_{en \vee nl \to es}$, and $P^S_{en \vee es \to nl}$ all achieve high scores, some even

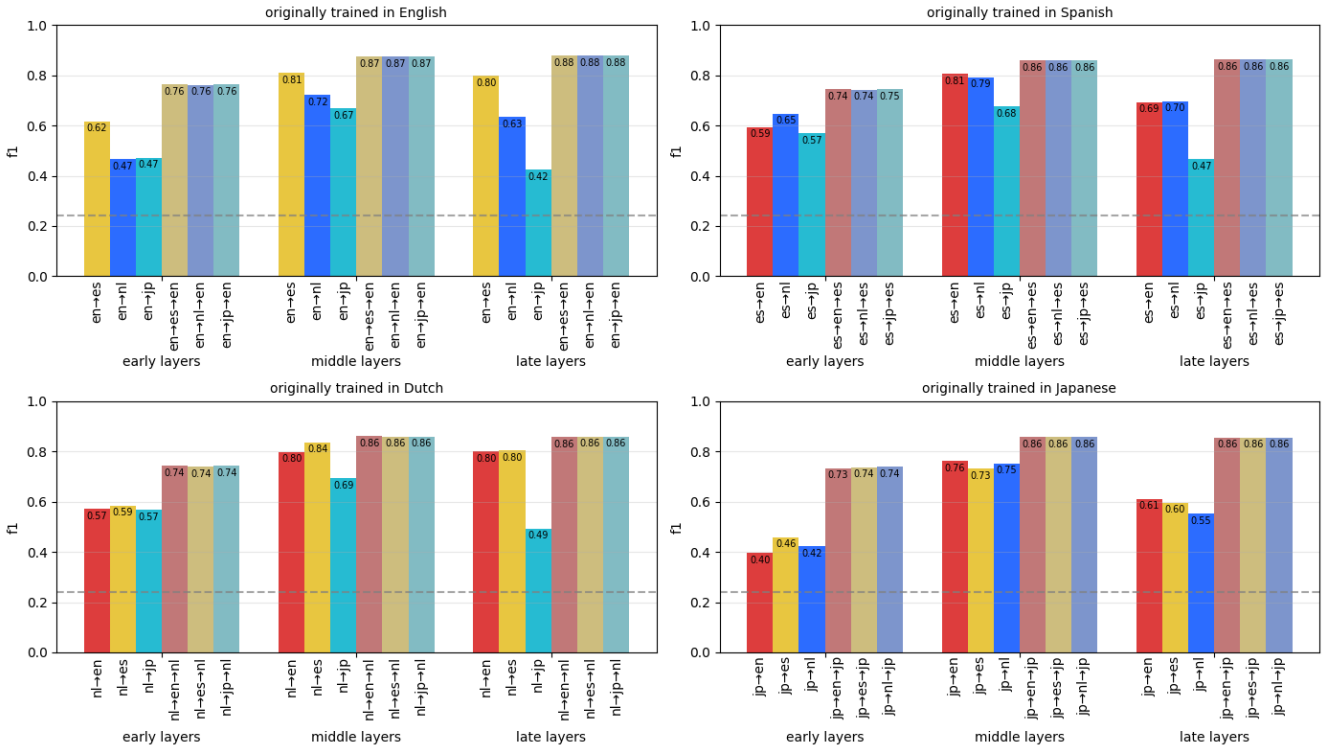

Figure 6: The brief retraining analysis of Figure 5, applied to Aya. As with Olmo, two iterations of retraining on a second language barely move the probe weights; however, the performance improves substantially.

matching their performance on A.

Japanese still maintains the same pattern that was noted in Section 4.3.1, although there is a smaller difference between probes tested in Japanese and probes tested in other languages than before.

The probes all receive very similar scores on A that they originally received before having been retrained, indicating that there was close to no degradation in A despite the large improvement in B.

Aya Figure 6 shows the results on the Aya model. Compared to the previous Aya experiment, all the probes have higher scores across the board, even in layers where the probes previously performed worse than the baseline, in some cases being very close to the $A \to B \to A$ probes. Similar to what was observed in Figure 5, probes that were retrained in a language and then tested in their original language experienced no important degradation in performance.

While probes in the late layers still perform worse when tested in Japanese compared to other languages, it can be observed that the difference in performance is much less present in the early and middle layers, and has even disappeared in some cases, as in $P^{S}_{nl \to en \vee jp}$, both probes achieving the same F1.

Finally, the lines are smoother than in the previous Aya experiment, but large drops and increases still occur across the layers. Retraining these probes for more iterations led to the lines

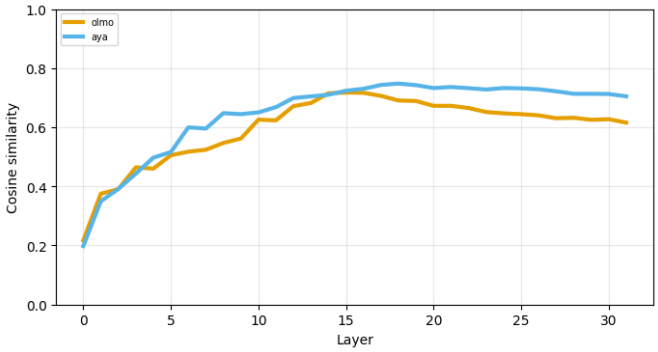

Figure 7: Cosine similarity between each logistic regression probe retrained for up to 1000 iterations and its original pre-retraining version, averaged over all source–target language pairs at each layer. Early-layer probes change the most under full retraining, with similarity increasing toward the later layers. The lines of Olmo and Aya are both shown

smoothing out, but we decided to keep the iterations at 2, since retraining for longer leads to a slow decrease in similarity with the original probes. A plot showing the results of this experiment with lines instead of bars where this effect is visible can be found in Appendix D.

Interpretation of results From all these observations, it can be concluded that models do not store entailment information in exactly the same parts of the activations for all languages, but that in the case of English, Spanish, and Dutch, it is stored similarly in a way that a small tweak in the probe weights can lead to strong performance increases, particularly for Aya. This is, however, not the case for all layers, as probes at early layers still struggle to be transferred; this is likely due to the model not meaningfully storing entailment information in these early layers, as was shown by Experiment 4.2. The difficulty of probes to adapt to Japanese suggests that both Olmo and Aya encode entailment information more differently for Japanese than they do for other languages. Since Aya received similar amounts of training for Spanish, Japanese, and Dutch, the difference cannot simply be explained by the model not having had the opportunity to develop cross-lingual representations in Japanese; a more likely reason is the typological distance between Japanese and the other languages. This idea is corroborated by other studies, such as Jie et al. [2025], Lauscher et al. [2020].

4.3.3 Transfer after extensive retraining

This sub-experiment follows the same procedure as Section 4.3.2, but in this case, rather than retraining for 2 iterations, we are doing so for a maximum of 1000 iterations.

Figure 7 shows the cosine similarities between probes that were retrained for 1000 iterations, and their original versions before retraining. For conciseness, the cosine similarities of all the $P^S_{A\to B}, A, B \in \{en, es, nl, jp\}$ at each layer are averaged together into a single similarity value. Unlike probes that were retrained for only two iterations, these probes have experienced significant changes in their weights. As can be seen, probes in the early layers suffer more considerable changes than probes in the later layers.

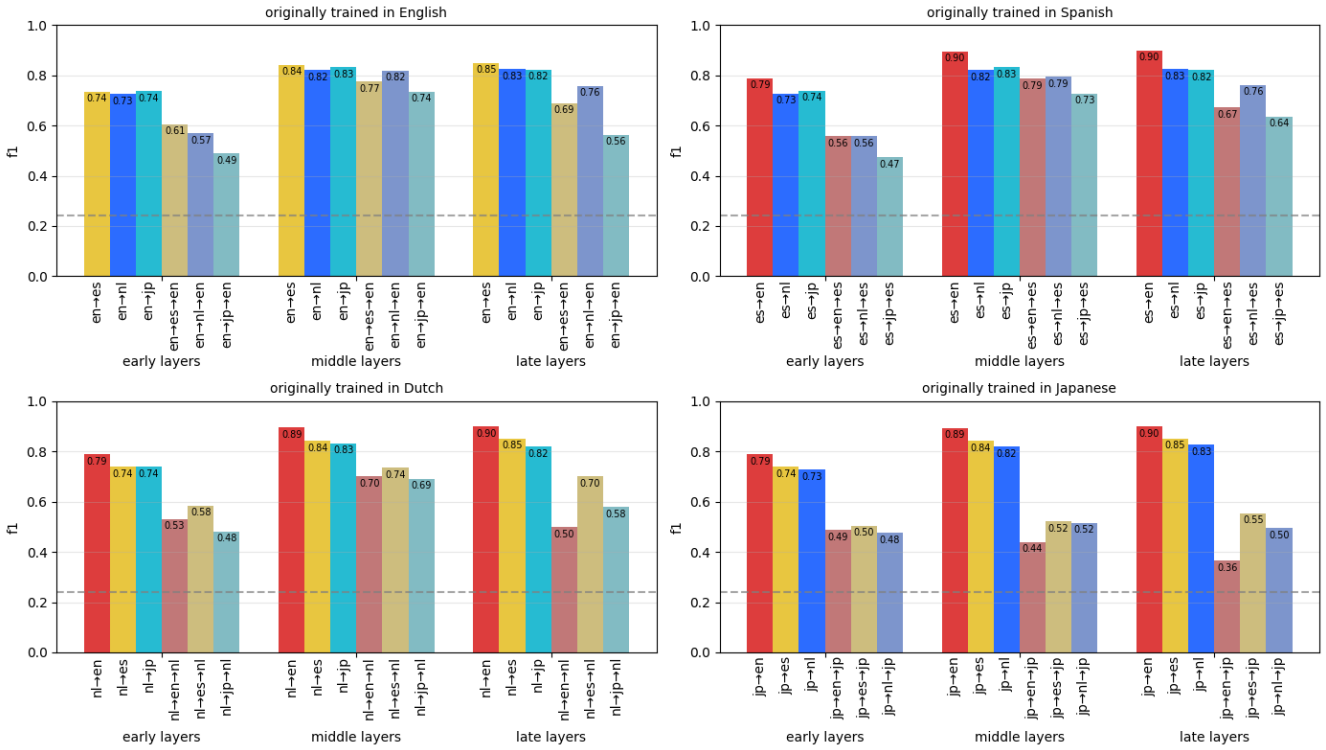

Figure 8: Cross-lingual transfer in Olmo after full retraining. Each subplot shows probes retrained for up to 1000 iterations (or until convergence)

Olmo Figure 8 shows the results of Olmo probes on this experiment. First of all, the performance of $A \to B \to A$ probes declined compared to the last experiment, especially in the early and late layers, and at the same time, the performance of the $A \to B$ probes increased across the board. There were, however, some probes that suffered only a very small performance degradation, like $P^S_{en \to nl}$ or $P^S_{es \to nl}$ in the middle layers. Secondly, $jp \to B \to jp$ probes experienced the largest degradation, and did so in all layers. In all cases, the $A \to B \to A$ probes still performed better than the baseline.

Aya Figure 9 shows the results of the experiment on Aya. The results are similar to those of Olmo: there is a degree of degradation, especially in the early and late layers. However, the degradation appears to be less intense across the board.

The very high performance of $A \to B$ probes in both models is as expected, since these probes were trained in the new language for 1000 iterations, giving the probe ample time to adapt to the new language. In both models, middle layers were the ones that suffered the smallest degradations. However, early and late layers both achieved comparable overall levels of degradation.

Interpretation of results The cosine similarities in Figure 7 are far lower than the near-identical weights obtained after only two iterations in Section 4.3.2, where the lowest similarity was still above 0.998. Together with the fact that the $A \to B$ probes nonetheless reach high F1 in their new language, this suggests that the activations of A and B are similar enough that a single probe can

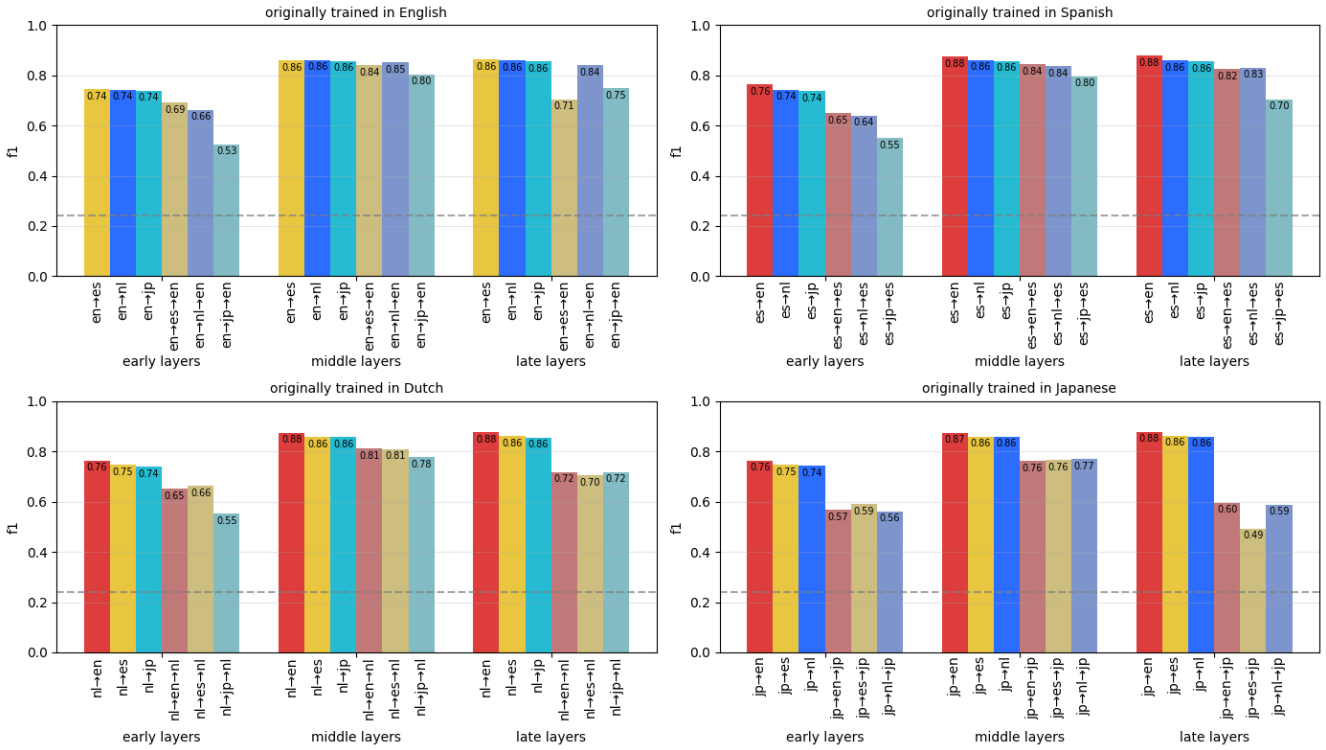

Figure 9: The full retraining analysis of Figure 8, applied to Aya.

operate in both representational spaces, but not so similar that it can do so with exactly the same weights: fully adapting to B requires the probe to move appreciably away from the configuration that worked on A, which is why its performance on the original language degrades once there is no longer any incentive to preserve it.

The amount of degradation can therefore be read as a rough measure of how differently the two languages encode entailment. Seen this way, the strong degradation of the $jp \to B \to jp$ probes reinforces the conclusion of the previous experiments: Japanese encodes NLI in a way that is the most distinct from the other three languages, so a probe that has been fully refitted to a different language loses most of what allowed it to work on Japanese. Conversely, the comparatively small degradation of probes originally trained on English and Spanish indicates that these languages encode entailment in ways close enough to the other Latin-script languages that adapting a probe between them demands only modest changes to its weights.

The less pronounced degradation in Aya than in Olmo again points toward Aya encoding NLI more uniformly across the four languages, in line with what was observed throughout Experiment 4.3. This is consistent with the idea that an explicitly multilingual model represents the entailment relation in more similar ways across languages than a near-monolingual one, lending further support to our answer to Q.3.

Finally, as already noted, the middle layers suffered the smallest degradations in both models, which agrees with the rest of our results in indicating that NLI is encoded most cross-lingually in the middle of the network. This experiment does not, however, reveal a clear difference between the early and late layers, which degraded to a comparable extent. This is somewhat at odds with the

interpretation we offered in Section 4.3.2, where we attributed the poor transfer of the early layers to their barely encoding any entailment information in the first place, as found in Experiment 4.2. If that were the whole explanation, we would expect the late layers, which do encode NLI strongly, to degrade differently from the early ones; instead, both lose a comparable amount of performance once the probe is refitted. This indicates that the degradation under full retraining is not governed solely by whether a layer encodes entailment, and that the early and late layers, despite differing greatly in how strongly they represent NLI, are similarly difficult to align across languages.

4.4 Similarity between probe parameters across languages

4.4.1 Setup

In this experiment, we investigate the extent to which the probes trained in Experiment 4.2 develop similar weights to one another. Specifically, we compare a probe trained at a given layer in one language with the probe trained at the same layer but on a different language, doing this for all layers and, at each layer, comparing the six possible language pairs. The purpose is to determine the degree to which probes trained on different languages converge to a similar structure despite being trained independently: if they are very similar, this would mean that the model activations contain patterns that are common across languages, which the probes for each language independently learn to exploit, leading them to develop similar weights. This would help answer Q.2: "How similar are internal representations of NLI concepts across different languages?"

This experiment focuses on the logistic regression probes. As explained in Section 3.3.2, these have different weights $\vec{w}_i$ and intercept values c_i for each class i. There are three classes in our task, so we can interpret the weights of the probe as a matrix W of dimensions $3 \times (n+1)$, where the $+1$ comes from the intercept. In order to determine the similarity between two probes, we flatten this matrix into a single vector $\vec{p}$ of length $3 \cdot (n+1)$ and compare the resulting vectors directly, rather than matching the three class vectors one by one. We do this for two reasons. The first is conciseness. The second, and more important, is interpretability: as noted in Section 3.3.2, the three class vectors are not fit independently but jointly, under a shared softmax normalisation, so that only the contrasts $\vec{w}_i - \vec{w}^j$ determine the probe's predictions, while the absolute value of any single $\vec{w}_i$ is fixed only by the L2 penalty. An individual class vector therefore has no clean interpretation as "the direction for class i".

We compare the flattened probe vectors using two metrics: cosine similarity and Euclidean distance, with cosine similarity as the primary metric. Euclidean distance is normalised across the layers, since probes at different layers may work at different scales. High cosine similarity coupled with low Euclidean distance are taken as an indication that the probe weights are similar. We compare the similarities with those of the control probes.

We do not include the mass-mean probes in this comparison. Unlike logistic regression, the mass-mean probe is not fit by a discriminative optimisation that concentrates the parameters on the dimensions separating the classes; its per-class scores are built from the class means $\vec{\mu}_i$ and a single within-class covariance Σ estimated separately for each language, so they may absorb whatever incidental features happen to separate the class groups in a given language. A cross-lingual comparison of these parameters could therefore confound genuine NLI alignment with language-specific covariance structure and incidental features, making it unsuitable as a measure of representational alignment.

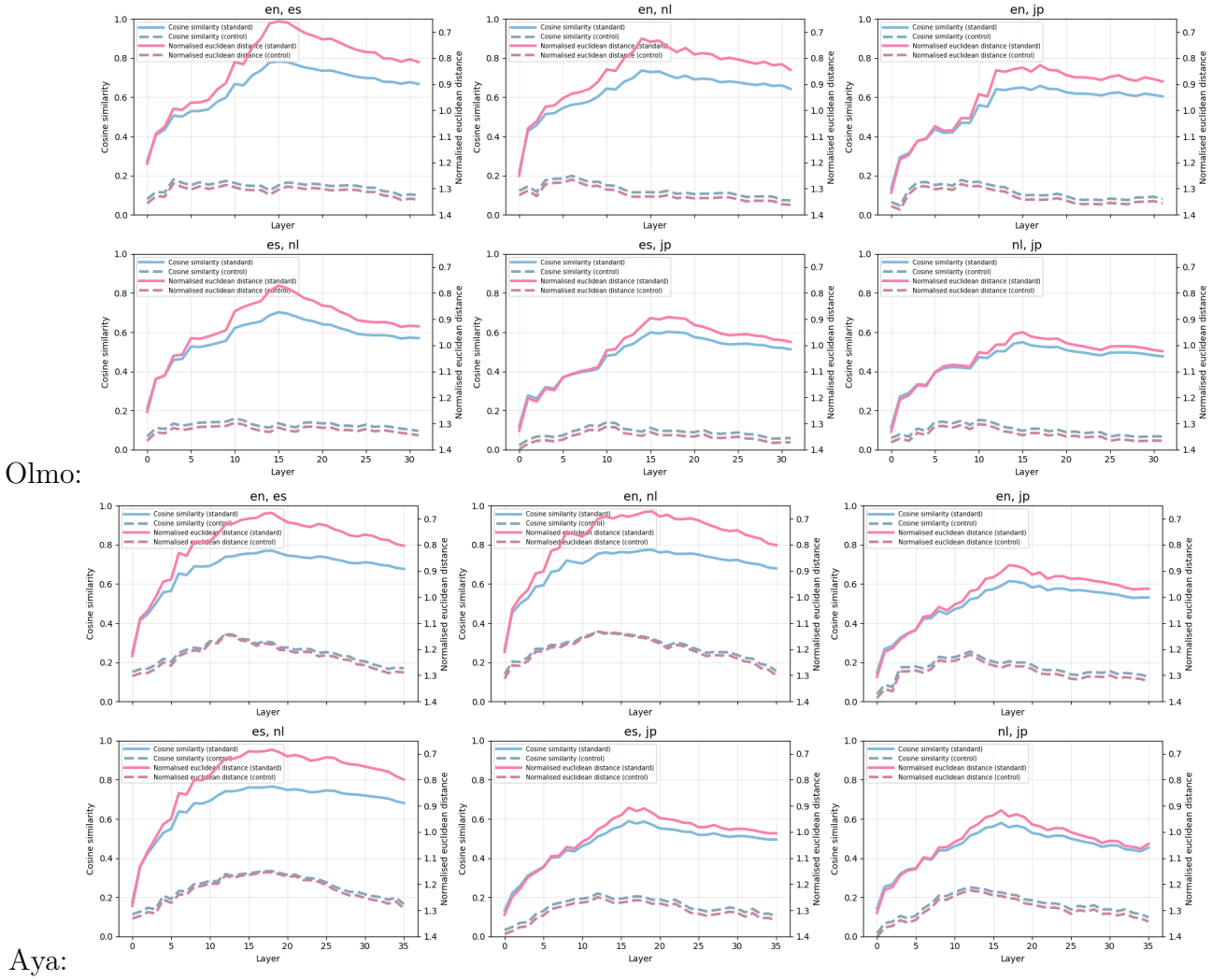

Figure 10: Layerwise similarity between pairs of logistic regression probes trained independently on different languages in Olmo (top) and Aya (bottom). Each of the six language pairs is compared at every layer using cosine similarity and Euclidean distance.

4.4.2 Result

Figure 10 shows the results for Olmo (top) and Aya (bottom). Cosine similarity (blue) is plotted in the $[0, 1]$ range, while Euclidean distance (pink) uses a range of $[0.65, 1.4]$. The Euclidean distance lines are vertically flipped to fit the intuition that up corresponds to higher similarity in both metrics. Standard-probe similarities are shown as solid lines and control-probe similarities as dotted lines.

Both models exhibit the same overall pattern for the standard probes. Cosine similarity is lowest in the early layers, rises gradually to a peak in the middle layers, and then slowly declines; Euclidean distance mirrors this, starting high, reaching a minimum in the middle layers, and growing again toward the later layers, though this late rise varies in size across pairs and is absent for en-jp and jp-nl. The pairs among English, Spanish, and Dutch reach the highest cosine similarities (around 0.7–0.8), whereas pairs involving Japanese peak lower (around 0.6) in both models.

In Olmo, the cosine similarity of the control probes stays below 0.2 at every layer with no clear pattern, and their Euclidean distance does something similar, remaining higher than that of the standard probes (keeping in mind that higher in the case of Euclidean distance means lower similarity). Control probes are highly distinct across languages at every layer, implying that Olmo's activations differ sharply between languages in the features relevant to this basic task. In Aya, however, the control probe similarities are still low across the board, but there is an observable pattern, with the similarity being highest in the early-to-middle layers, in terms of both higher cosine similarity and lower Euclidean distance. This may hint at the features relevant for the control task (namely the first noun in the sentence pair) being encoded in a mostly distinct manner across languages, but the early-to-middle layer activations being slightly more cross-lingual in this respect but still far below the actual NLI task.

In both models, the pairs involving Japanese are consistently less similar than the pairs among English, Spanish, and Dutch, which cluster closely together. One possible explanation, which would also account for the transfer results of Experiment 4.3, is that Japanese's different script and greater typological distance cause its activations to lack the script, cognate, and lexical-overlap-based features shared by the three Latin-script languages. If that were the case, the English, Spanish, and Dutch probes could partly rely on these shared surface features, making them similar to one another but leaving them with components that carry no signal in Japanese. This would explain both the low similarity of Japanese probes with all others observed here and the poor transfer of other-language probes into Japanese seen in Experiment 4.3. A Japanese probe, lacking such features to exploit, would instead have to rely on the deeper, language-agnostic entailment signal present in every language, which could in turn explain why Japanese-trained probes transferred comparatively well to the other languages. Such an explanation would also be consistent with the effect persisting in Aya despite its explicit Japanese training, since it would attribute the divergence of Japanese to its script and typology rather than to the amount of training data it received.

Both models point to the same conclusion: probes trained independently on different languages develop very similar weight vectors, with similarity highest in the middle layers, slightly lower in the late layers, and lowest in the early layers. This high cross-lingual similarity reinforces the interpretation of Experiment 4.3: if independently trained probes already converge to nearly the same weights, then only a small adjustment should be needed for a probe trained on one language to work on another, which is precisely what we observed there, where two iterations of retraining sufficed to recover strong transfer. It also helps explain why Aya achieved comparatively poor performance in the direct probe transfer experiment but drastically improved after only two iterations. Finally, the similarity peaking in the middle layers mirrors Experiment 4.3, where those layers showed the strongest cross-lingual transfer without further training, and agrees with Zhao et al. [2024], who find cross-lingual representations to be strongest in the middle layers and more language-specific in the early and late layers.

5 Discussion

5.1 Limitations

5.1.1 Using datasets that are direct translations of each other

Because SICK-ES, JSICK, and SICK-NL are translations of the original SICK, the four datasets are parallel: each sentence pair corresponds to roughly the same underlying semantic content in every language, apart from any differences brought by translating the sentences. This parallelism is what makes a controlled comparison possible, but it also introduces a subtle confound into the cross-lingual comparisons of Section 4.3 and the comparison of probe weights across languages: when we train a probe independently on each language and then compare them, the probes are being fit to discriminate the same data points, carrying the same labels, from input features that are themselves similar across languages if the model maps content into a shared cross-lingual semantic space.

As a consequence, a high cross-lingual probe similarity might reflect that the model has cross-lingually aligned meaning representations in general, which is well documented [Schut et al., 2025, Harrasse et al., 2026, Wen-Yi and Mimno, 2023], rather than aligned NLI-specific representations. A linear probe will exploit whatever directions separate the three classes; if those directions happen to be aligned across languages for reasons unrelated to entailment, the probes will end up similar regardless of how entailment itself is encoded. This does not invalidate our findings, since recovering any cross-lingual structure is informative, but it does mean that we are not able to claim with absolute certainty that the similar representations encountered between languages exclusively correspond to NLI knowledge but also they may partially correspond to general semantic concepts. A stronger test would train the probes on non-parallel NLI data and check whether the cross-lingual similarity persists. We expect it would, but we leave this as future work.

5.1.2 SICK in LLM training data

A further concern is data contamination. When prompted directly, both models were able to describe what SICK is and who created it, and could even generate sentences in the general style and format of the dataset. If the labelled SICK pairs were present in the models' pretraining data, this could give them an unfair advantage, since the models might in part be recalling memorised labels rather than computing the entailment relation. However, when we asked the models to reproduce actual SICK sentences, they produced sentences that matched the expected format but did not correspond to genuine SICK items. This suggests that the models have some metadata-level knowledge of SICK, likely acquired from descriptions, papers, or repositories discussing it, but that they most probably did not train directly on the labelled pairs.

We did not carry out a rigorous contamination analysis. The inability to reproduce SICK items verbatim is not conclusive, since models do not always regurgitate training data even when it is present [Carlini et al., 2023], so we cannot fully exclude partial exposure. It is worth noting, however, that the models' final answers obtained far from perfect scores, unlike what we would expect had there been dataset contamination.

5.1.3 Differences in JSICK labels

As noted in Section 3.1, JSICK was translated by hand and then re-annotated, so a minority of its pairs carry labels that differ from the original English SICK labels. Because the Japanese probes are trained and evaluated against the JSICK labels, while the probes for the other three languages all share the original SICK labels, the Japanese task is not defined over strictly identical targets. This is one plausible contributor to the lower Japanese probe performance and the weaker cross-lingual alignment of Japanese observed in some experiments, alongside the typological and orthographic factors discussed earlier.

To gauge how much the relabelling contributes, we run a small ablation by restricting the dataset to only the pairs where the JSICK label coincides with the original English label. The results are in Appendix C.

5.1.4 Artefacts in SICK

A well-known issue with NLI datasets is that shallow surface features are correlated with the gold labels. The clearest example is negation: the presence of a negation marker is strongly associated with the contradiction label. This has been documented as an annotation artefact in large NLI datasets such as SNLI and MultiNLI, where a hypothesis-only classifier can recover the label far above chance, and the same negation-to-contradiction pattern has been observed in SICK itself [Marelli et al., 2014a]. Negation is among the strongest cues for contradiction [Gururangan et al., 2018, Poliak et al., 2018]. Other documented shortcuts include lexical overlap between premise and hypothesis predicting entailment [McCoy et al., 2019], as well as more incidental features such as sentence length or the presence of particular words. Table 1 shows that there is a substantial difference in overlap between premise and hypothesis for neutral pairs and others. Additionally, sentence pairs have different average premise and hypothesis lengths depending on the label (in terms of tokens). Therefore, it is plausible that the probes could be exploiting these artefacts to some extent.

This is relevant to the strength of our probing claims. A probe, and, more fundamentally, the model representation it reads from, could achieve moderate entailment-classification performance partly by exploiting such surface cues rather than by encoding the entailment relation itself. Our control task and the resulting selectivity guard against the probe trivially memorising examples or exploiting incidental structure, but they do not fully eliminate this concern. If more robust NLI datasets are made in the future, it may be worth trying to replicate the results observed in this thesis on them.

5.2 Future work

The probing methodology used in this thesis is fundamentally correlational: it establishes that information about the entailment relation is linearly decodable from the activations at a given layer, but it does not, on its own, establish that the model draws on this information when producing its predictions. This is a known limitation of probing [Elazar et al., 2020]. The gap we observed between what the models encode and what they express makes this distinction particularly relevant, since it shows that the presence of decodable information does not guarantee that the model makes use of it. A natural next step would therefore be to complement probing with a causal interpretability method such as path patching, in which specific internal activations are intervened

upon and the effect on the model's output is measured [Ameisen* et al.]. This would make it possible to test whether the representations identified here are causally responsible for the model's entailment judgements, and whether the same causal pathways are shared across languages or differ between them in the same way that the probe-based representations do.

Secondly, we examined four languages, only one of which (Japanese) used a non-Latin script, and it was precisely this language that behaved as a consistent outlier in our cross-lingual alignment analyses. Future work could include a larger and more varied set of languages, spanning additional scripts and a finer range of typological distances from English in order to test whether the outlier behaviour observed for Japanese generalises to other typologically distant or differently scripted languages. In a similar vein, the analysis could be extended to more challenging NLI datasets featuring more difficult inference than SICK, whose range of phenomena is relatively limited. This would help clarify whether the layerwise and cross-lingual patterns we observed persist when the inference task is harder, or whether they are in part a consequence of the simplicity of SICK. The main obstacle here is the scarcity of sentence-aligned multilingual versions of such datasets, which was the reason for choosing SICK in the first place. Relatedly, as noted in Section 5.1, the parallel structure of SICK and its translations means that a high cross-lingual probe similarity could in part reflect the model having generally aligned meaning representations across languages, rather than aligned NLI-specific representations. A way to address this would be to train the probes on data such as XNLI [Conneau et al., 2018], in which the sentence pairs differ across languages.

Thirdly, we decided not to use selectivity for the language transfer experiments, as it is not clear whether selectivity is an interpretable metric in the situation where the standard and control probes are transferred to another language, as explained in Section 4.3. It would, however, be interesting to research whether and how selectivity could be adapted to be used in a cross-lingual transfer scenario.

Finally, the cross-lingual probing methodology developed here is not specific to NLI, and could be applied to other phenomena in order to ask the same questions of where in the network they are encoded and how aligned that encoding is across languages. One promising candidate would be Theory of Mind, which has also been the subject of recent research in the field of interpretability [Prakash et al., 2025]. Applying the methodology to such phenomena would help to establish whether the patterns found in this thesis are specific to NLI or are instead general properties of how multilingual models organise their internal representations.

6 Conclusion

This thesis set out to determine whether and how large language models encode the entailment relation similarly across languages, where in the network this happens, and whether multilingual pretraining changes it. We probed Olmo 3 7B Instruct, a near-monolingual model, and Tiny Aya Global, an explicitly multilingual one, at every layer on SICK and its sentence-aligned translations into Spanish, Japanese, and Dutch, training linear probes, measuring their selectivity against a control task, and corroborating the results with a second probe type, before transferring probes across languages and comparing their weights. We take each research question in turn.

Regarding the first research question, NLI information is barely present in the earliest layers, where control probes frequently outperform the standard ones, but becomes linearly decodable from the middle layers onwards and remains so through the later layers. This held across all four

languages, both models, and both probe types. The strength of the encoding varied, being weakest for Japanese in Olmo, but **the layers that encode NLI most strongly are largely shared across languages**.

Regarding the second research question, probe transfer and weight comparison agreed that representations are **most alignable in the middle layers**. English, Spanish, and Dutch grouped together, while Japanese was the clearest outlier. Transfer was also asymmetric, clearest in Olmo: Japanese-trained probes transferred reasonably to the other languages, but other-language probes transferred poorly into Japanese. We read this as the three Latin-script languages sharing surface features, namely script, cognates, and lexical overlap, that carry no signal in Japanese, leaving Japanese probes to rely on the deeper language-agnostic entailment signal common to all four.

Regarding the third research question, Aya generally encoded NLI more uniformly than Olmo, having closer selectivity curves, less degradation when refitted to another language, and no Japanese collapse when prompted. This supports the view that **multilingual training yields more consistent entailment representations across languages**; however, the difference with Olmo is less pronounced than was originally expected. Furthermore, Japanese remained the outlier in Aya too, and since Aya received comparable data for Spanish, Dutch, and Japanese, Japanese stays apart as a result of how distant it is, not how little data it received. Finally, the cross-lingual probe transfer was only effective after a small modification of the probes' weights.

A separate result concerns Olmo, supposedly trained almost exclusively on English. When prompted, it performed surprisingly well in Spanish and Dutch, with only Japanese collapsing. More surprising still, when probed it encoded entailment strongly in all four languages, Japanese included, despite its collapsed prompted performance.

This all points to a gap between what the models encode internally and what they express when prompted. Probes recovered entailment far more reliably, and more uniformly across languages, than the models' own answers. This was starkest for Olmo on Japanese, whose direct responses fell below the majority-class baseline while a probe recovered the relation with high selectivity. Some of this gap is expected, since the probe reads activations directly whereas the model must decode them and commit to a word, but its size indicates that decodable information is no guarantee the model uses it.

Taken together, **the cross-lingual difference is far smaller at the level of internal representation than at the level of prompted output**. Both models encode entailment as a linearly decodable structure present across the middle and later layers and most language-agnostic in the middle, with multilingual pretraining narrowing the cross-lingual gap without closing it, and Japanese the clearest point of divergence throughout.

References

G. Alain and Y. Bengio. Understanding intermediate layers using linear classifier probes, Nov. 2018. URL http://arxiv.org/abs/1610.01644. arXiv:1610.01644 [stat.ML].

A. E. Ameisen*, J. Lindsey*, A. Pearce*, W. Gurnee*, N. L. Turner*, B. Chen*, C. Citro*, D. Abrahams, S. Carter, B. Hosmer, J. Marcus, M. Sklar, A. Templeton, T. Bricken, C. McDougall, H. Cunningham, T. Henighan, A. Jermyn, A. Jones, A. Persic, Z. Qi, T. B. Thompson, S. Zimmerman, K. Rivoire, T. Conerly, C. Olah, and J. B. A. A. P.

March 27. Circuit Tracing: Revealing Computational Graphs in Language Models. URL https://transformer-circuits.pub/2025/attribution-graphs/methods.html.

A. Arditi, O. Obeso, A. Syed, D. Paleka, N. Panickssery, W. Gurnee, and N. Nanda. Refusal in Language Models Is Mediated by a Single Direction, Oct. 2024. URL http://arxiv.org/abs/2406.11717. arXiv:2406.11717 [cs.LG].

Y. Belinkov, L. Màrquez, H. Sajjad, N. Durrani, F. Dalvi, and J. Glass. Evaluating Layers of Representation in Neural Machine Translation on Part-of-Speech and Semantic Tagging Tasks. In G. Kondrak and T. Watanabe, editors, *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1–10, Taipei, Taiwan, Nov. 2017. Asian Federation of Natural Language Processing. URL https://aclanthology.org/I17-1001/.

D. Blasi, A. Anastasopoulos, and G. Neubig. Systematic Inequalities in Language Technology Performance across the World's Languages. *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 1:5486–5505, 2022. ISSN 9781955917216. doi: 10.18653/V1/2022.ACL-LONG.376.

S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning. A large annotated corpus for learning natural language inference. In L. Màrquez, C. Callison-Burch, and J. Su, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, Sept. 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1075. URL https://aclanthology.org/D15-1075/. Read_Status: Read Read_Status_Date: 2026-05-10T16:47:11.321Z.

N. Carlini, D. Ippolito, M. Jagielski, K. Lee, F. Tramer, and C. Zhang. Quantifying Memorization Across Neural Language Models, Mar. 2023. URL http://arxiv.org/abs/2202.07646. arXiv:2202.07646 [cs.LG].

B. R. Chiswick and P. W. Miller. Linguistic Distance: A Quantitative Measure of the Distance Between English and Other Languages. *Journal of Multilingual and Multicultural Development*, 26(1):1–11, Jan. 2005. ISSN 0143-4632. doi: 10.1080/14790710508668395. URL https://doi.org/10.1080/14790710508668395. _eprint: https://doi.org/10.1080/14790710508668395.

A. Conneau, R. Rinott, G. Lample, H. Schwenk, V. Stoyanov, A. Williams, and S. R. Bowman. XNLI: Evaluating Cross-lingual Sentence Representations. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018*, pages 2475–2485, 2018. ISSN 9781948087841. doi: 10.18653/V1/D18-1269.

B. Deng, Y. Wan, Y. Zhang, B. Yang, and F. Feng. Unveiling Language-Specific Features in Large Language Models via Sparse Autoencoders, May 2025. URL http://arxiv.org/abs/2505.05111. arXiv:2505.05111 [cs] version: 1.

Y. Elazar, S. Ravfogel, A. Jacovi, and Y. Goldberg. Amnesic Probing: Behavioral Explanation with Amnesic Counterfactuals. *Transactions of the Association for Computational Linguistics*, 9:160–175, 2020. doi: 10.1162/tacl_a_00359.

S. Gururangan, S. Swayamdipta, O. Levy, R. Schwartz, S. R. Bowman, and N. A. Smith. Annotation Artifacts in Natural Language Inference Data. In M. Walker, H. Ji, and A. Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2017. URL https://aclanthology.org/N18-2017/.

A. Harrasse, F. Draye, P. S. Pandey, Z. Jin, and B. Schölkopf. Tracing Multilingual Representations in LLMs with Cross-Layer Transcoders, Feb. 2026. URL http://arxiv.org/abs/2511.10840. arXiv:2511.10840 [cs].

J. Hewitt and P. Liang. Designing and Interpreting Probes with Control Tasks. *EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, pages 2733–2743, 2019. ISSN 9781950737901. doi: 10.18653/v1/D19-1275.

J. Huertas-Tato, A. Martín, and D. Camacho. SILT: Efficient transformer training for inter-lingual inference. *Expert Systems with Applications*, 200:116923, 2022. doi: 10.1016/J.ESWA.2022.116923.

L. Jie, D. Jin, and H. Yanaka. LLMs Struggle with NLI for Perfect Aspect: A Cross-Linguistic Study in Chinese and Japanese. In K. Evang, L. Kallmeyer, and S. Pogodalla, editors, *Proceedings of the 16th International Conference on Computational Semantics*, pages 89–97, Düsseldorf, Germany, Sept. 2025. Association for Computational Linguistics. ISBN 979-8-89176-316-6. URL https://aclanthology.org/2025.iwcs-main.8/.

M. Jin, Q. Yu, J. Huang, Q. Zeng, Z. Wang, W. Hua, H. Zhao, K. Mei, Y. Meng, K. Ding, F. Yang, M. Du, and Y. Zhang. Exploring Concept Depth: How Large Language Models Acquire Knowledge and Concept at Different Layers? In O. Rambow, L. Wanner, M. Apidianaki, H. Al-Khalifa, B. D. Eugenio, and S. Schockaert, editors, *Proceedings of the 31st International Conference on Computational Linguistics*, pages 558–573, Abu Dhabi, UAE, Jan. 2025. Association for Computational Linguistics. URL https://aclanthology.org/2025.coling-main.37/.

T. Ju, W. Sun, W. Du, X. Yuan, Z. Ren, and G. Liu. How Large Language Models Encode Context Knowledge? A Layer-Wise Probing Study. *2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC-COLING 2024 - Main Conference Proceedings*, pages 8235–8246, 2024. ISSN 9782493814104.

A. Lauscher, V. Ravishankar, I. Vulić, and G. Glavaš. From Zero to Hero: On the Limitations of Zero-Shot Language Transfer with Multilingual Transformers. In B. Webber, T. Cohn, Y. He, and Y. Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4483–4499, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.363. URL https://aclanthology.org/2020.emnlp-main.363/.

Z. Li, Y. Shi, Z. Liu, F. Yang, A. Payani, N. Liu, and M. Du. Language Ranker: A Metric for Quantifying LLM Performance Across High and Low-Resource Languages. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(27):28186–28194, 2024. ISSN 157735897X. doi: 10.1609/aaai.v39i27.35038.

Z. W. Lim, A. F. Aji, and T. Cohn. Understanding Cross-Lingual Inconsistency in Large Language Models, May 2025. URL http://arxiv.org/abs/2505.13141. arXiv:2505.13141 [cs] version: 1.

F. Manigrasso, S. Schouten, L. Morra, and P. Bloem. Probing LLMs for Logical Reasoning. In T. R. Besold, A. d'Avila Garcez, E. Jimenez-Ruiz, R. Confalonieri, P. Madhyastha, and B. Wagner, editors, *Neural-Symbolic Learning and Reasoning*, pages 257–278, Cham, 2024. Springer Nature Switzerland. ISBN 978-3-031-71167-1. doi: 10.1007/978-3-031-71167-1_14.

M. Marelli, L. Bentivogli, M. Baroni, R. Bernardi, S. Menini, and R. Zamparelli. SemEval-2014 Task 1: Evaluation of Compositional Distributional Semantic Models on Full Sentences through Semantic Relatedness and Textual Entailment. In P. Nakov and T. Zesch, editors, *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 1–8, Dublin, Ireland, Aug. 2014a. Association for Computational Linguistics. doi: 10.3115/v1/S14-2001. URL https://aclanthology.org/S14-2001/.

M. Marelli, S. Menini, M. Baroni, L. Bentivogli, R. Bernardi, and R. Zamparelli. A SICK cure for the evaluation of compositional distributional semantic models. pages 216–223, 2014b.

S. Marks and M. Tegmark. The Geometry of Truth: Emergent Linear Structure in Large Language Model Representations of True/False Datasets, Aug. 2024. URL http://arxiv.org/abs/2310.06824. arXiv:2310.06824 [cs.AI].

R. T. McCoy, E. Pavlick, and T. Linzen. Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference. In A. Korhonen, D. Traum, and L. Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1334. URL https://aclanthology.org/P19-1334/.

B. Noble, R. Blanck, and G. Wijnholds. In the Mood for Inference: Logic-Based Natural Language Inference with Large Language Models. In L. Abzianidze and V. de Paiva, editors, *Proceedings of the 5th Workshop on Natural Logic Meets Machine Learning (NALOMA)*, pages 33–47, Bochum, Germany, Aug. 2025. Association for Computational Linguistics. ISBN 979-8-89176-287-9. URL https://aclanthology.org/2025.naloma-1.4/.

T. Olmo, A. Ettinger, A. Bertsch, B. Kuehl, D. Graham, D. Heineman, D. Groeneveld, F. Brahman, F. Timbers, H. Ivison, J. Morrison, J. Poznanski, K. Lo, L. Soldaini, M. Jordan, M. Chen, M. Noukhovitch, N. Lambert, P. Walsh, P. Dasigi, R. Berry, S. Malik, S. Shah, S. Geng, S. Arora, S. Gupta, T. Anderson, T. Xiao, T. Murray, T. Romero, V. Graf, A. Asai, A. Bhagia, A. Wettig, A. Liu, A. Rangapur, C. Anastasiades, C. Huang, D. Schwenk, H. Trivedi, I. Magnusson, J. Lochner, J. Liu, L. J. V. Miranda, M. Sap, M. Morgan, M. Schmitz, M. Guerquin, M. Wilson, R. Huff, R. L. Bras, R. Xin, R. Shao, S. Skjonsberg, S. Z. Shen, S. S. Li, T. Wilde, V. Pyatkin, W. Merrill, Y. Chang, Y. Gu, Z. Zeng, A. Sabharwal, L. Zettlemoyer, P. W. Koh, A. Farhadi, N. A. Smith, and H. Hajishirzi. Olmo 3, Dec. 2025. URL http://arxiv.org/abs/2512.13961. arXiv:2512.13961 [cs].

K. Park, Y. J. Choe, and V. Veitch. The Linear Representation Hypothesis and the Geometry of Large Language Models, July 2024. URL http://arxiv.org/abs/2311.03658. arXiv:2311.03658 [cs.CL].

T. Pimentel, J. Valvoda, R. H. Maudslay, R. Zmigrod, A. Williams, and R. Cotterell. Information-Theoretic Probing for Linguistic Structure. In D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4609–4622, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.420. URL https://aclanthology.org/2020.acl-main.420/.

P. Pirozelli, M. M. José, P. d. T. P. Filho, A. A. F. Brandão, and F. G. Cozman. Assessing Logical Reasoning Capabilities of Encoder-Only Transformer Models, Dec. 2023. URL https://arxiv.org/abs/2312.11720v2.

A. Poliak, J. Naradowsky, A. Haldar, R. Rudinger, and B. Van Durme. Hypothesis Only Baselines in Natural Language Inference. In M. Nissim, J. Berant, and A. Lenci, editors, *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/S18-2023. URL https://aclanthology.org/S18-2023/.

N. Prakash, N. Shapira, A. S. Sharma, C. Riedl, Y. Belinkov, T. R. Shaham, D. Bau, and A. Geiger. Language Models use Lookbacks to Track Beliefs, Oct. 2025. URL http://arxiv.org/abs/2505.14685. arXiv:2505.14685 [cs].

L. Qin, Q. Chen, Y. Zhou, Z. Chen, Y. Li, L. Liao, M. Li, W. Che, and P. S. Yu. Multilingual Large Language Model: A Survey of Resources, Taxonomy and Frontiers. 2024.

O. Reblitz-Richardson. When Probing Accuracy Saturates, Fragility Resolves: A Complementary Metric for LLM Pre-Training Analysis, June 2026. URL http://arxiv.org/abs/2606.11375. arXiv:2606.11375 [cs.CL] version: 1.

A. Salamanca, A. Diana, D. D'souza, A. Khairi, D. Mora, S. Dash, V. Aryabumi, S. Rajaee, M. Mofakhami, A. Sahu, T. Euyang, B. Prince, M. Smith, H. Lin, A. Locatelli, S. Hooker, T. Kocmi, A. Gomez, I. Zhang, P. Blunsom, N. Frosst, J. Pineau, B. Ermis, A. Üstün, J. Kreutzer, and M. Fadaee. Tiny Aya: Bridging Scale and Multilingual Depth, Feb. 2026. URL https://github.com/Cohere-Labs/tiny-aya-tech-report/blob/main/tiny_aya_tech_report.pdf.

L. Schut, Y. Gal, and S. Farquhar. Do Multilingual LLMs Think In English?, Feb. 2025. URL http://arxiv.org/abs/2502.15603. arXiv:2502.15603 [cs].

L. Soldaini, R. Kinney, A. Bhagia, D. Schwenk, D. Atkinson, R. Authur, B. Bogin, K. Chandu, J. Dumas, Y. Elazar, V. Hofmann, A. Jha, S. Kumar, L. Lucy, X. Lyu, N. Lambert, I. Magnusson, J. Morrison, N. Muennighoff, A. Naik, C. Nam, M. Peters, A. Ravichander, K. Richardson, Z. Shen, E. Strubell, N. Subramani, O. Tafjord, E. Walsh, L. Zettlemoyer, N. Smith, H. Hajishirzi, I. Beltagy, D. Groeneveld, J. Dodge, and K. Lo. Dolma: an Open Corpus of Three Trillion Tokens for Language Model Pretraining Research. In L.-W. Ku, A. Martins, and V. Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15725–15788, Bangkok, Thailand, Aug. 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.840. URL https://aclanthology.org/2024.acl-long.840/.

H. Steck, C. Ekanadham, and N. Kallus. Is Cosine-Similarity of Embeddings Really About Similarity? In *Companion Proceedings of the ACM Web Conference 2024*, pages 887–890, May 2024. doi: 10.1145/3589335.3651526. URL http://arxiv.org/abs/2403.05440. arXiv:2403.05440 [cs.IR].

A. W. Wen-Yi and D. Mimno. Hyperpolyglot LLMs: Cross-Lingual Interpretability in Token Embeddings. *EMNLP 2023 - 2023 Conference on Empirical Methods in Natural Language Processing, Proceedings*, pages 1124–1131, 2023. doi: 10.18653/v1/2023.emnlp-main.71.

C. Wendler, V. Veselovsky, G. Monea, and R. West. Do Llamas Work in English? On the Latent Language of Multilingual Transformers. In L.-W. Ku, A. Martins, and V. Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15366–15394, Bangkok, Thailand, Aug. 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.820. URL https://aclanthology.org/2024.acl-long.820/.

G. Wijnholds and M. Moortgat. SICK-NL: A Dataset for Dutch Natural Language Inference. *EACL 2021 - 16th Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference*, pages 1474–1479, 2021. ISSN 9781954085022. doi: 10.18653/V1/2021.EACL-MAIN.126.

C. Xiao, H. P. Chan, H. Zhang, M. Aljunied, L. Bing, N. Al Moubayed, and Y. Rong. Analyzing LLMs' Knowledge Boundary Cognition Across Languages Through the Lens of Internal Representations. *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 1:24099–24115, 2025. ISSN 9798891762510. doi: 10.18653/v1/2025.acl-long.1174.

H. Yanaka and K. Mineshima. Compositional Evaluation on Japanese Textual Entailment and Similarity. doi: 10.1162/tacl_a_00518.

Z. J. Ying, S. Ravfogel, N. Kriegeskorte, and P. Hase. The Truthfulness Spectrum Hypothesis, Feb. 2026. URL http://arxiv.org/abs/2602.20273. arXiv:2602.20273 [cs.LG] version: 1.

R. Zhang, Q. Yu, M. Zang, C. Eickhoff, and E. Pavlick. The Same But Different: Structural Similarities and Differences in Multilingual Language Modeling, Oct. 2024. URL http://arxiv.org/abs/2410.09223. arXiv:2410.09223 [cs].

Y. Zhao, W. Zhang, G. Chen, K. Kawaguchi, and L. Bing. How do Large Language Models Handle Multilingualism?, Nov. 2024. URL http://arxiv.org/abs/2402.18815. arXiv:2402.18815 [cs].

A Prompt structure

For every sentence pair, both models received a system message specifying the task and a user message containing the premise and hypothesis. The English prompt was written first; the Spanish, Japanese, and Dutch versions were produced by machine translation and then checked by native speakers, keeping the same structure throughout. The full set of templates is given in Table 4.

Role	Content
English	
System	You are a textual entailment classifier. Always respond with exactly one word: entailment, contradiction, or neutral.
User	Premise: {Sentence a} Hypothesis: {Sentence b} Classification:
Spanish	
System	Eres un clasificador de implicación textual. Responde siempre con una sola palabra: implicación, contradicción o neutral.
User	Premisa: {Sentence a} Hipótesis: {Sentence b} Clasificación:
Japanese	
System	あなたはテキスト含意分類器です。常に一言で答えてください：含意、矛盾、または中立。
User	前提：{Sentence a} 仮説：{Sentence b} 分類：
Dutch	
System	Jij bent een classificator van tekstuele implicaties. Reageer altijd met precies één woord: implicatie, tegenspraak of neutraal.
User	Premisse: {Sentence a} Hypothese: {Sentence b} Classificatie:

Table 4: Prompt templates used for each language, each consisting of a system message specifying the textual entailment classification task and a user message containing the premise and hypothesis. The slots {Sentence a} and {Sentence b} are filled with the corresponding sentence pair from the dataset.

B Accepted variants of labels

Label	Accepted Versions
English	
entailment	entail
neutral	neutral
contradiction	contradict, contradition
Spanish	
entailment	implica
neutral	neutral, neutro
contradiction	contradic
Japanese	
entailment	含意
neutral	中立
contradiction	矛盾
Dutch	
entailment	implicatie
neutral	neutral, neutraal
contradiction	tegensp, contradict

Table 5: Accepted versions and valid abbreviations for the entailment, neutral, and contradiction labels across English, Spanish, Dutch, and Japanese.

C JSICK label ablation

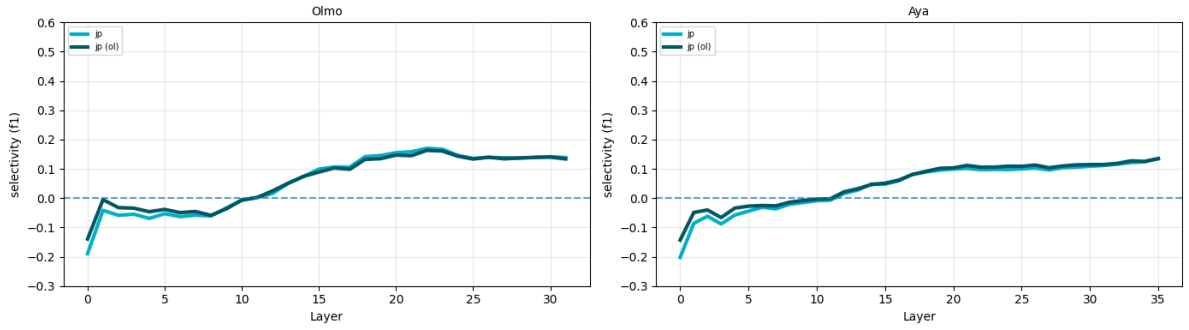

Figure 11: Repetition of the selectivity analysis of Experiment 4.2. This compares the Japanese probes trained and tested on the original SICK labels (*jp (ol)*) against the probes trained with JSICK labels (*jp*) for both Olmo and Aya models.

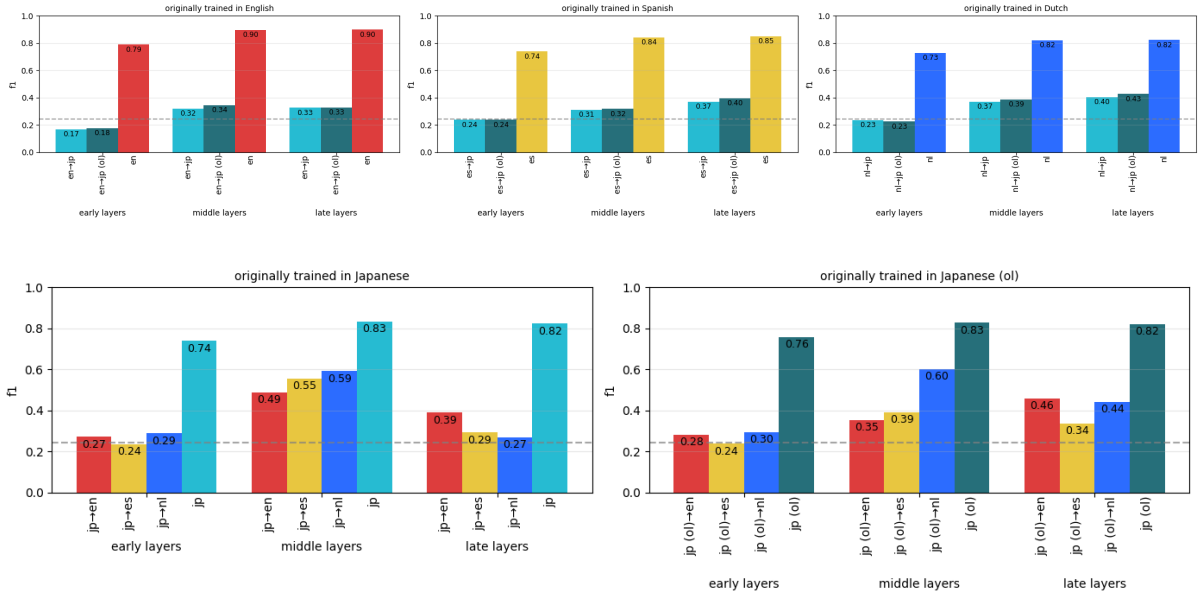

Figure 12: Direct cross-lingual transfer of Experiment 4.3 with no retraining for the **Olmo** model. Compares the original SICK labels (*jp (ol)*) and JSICK labels (*jp*).

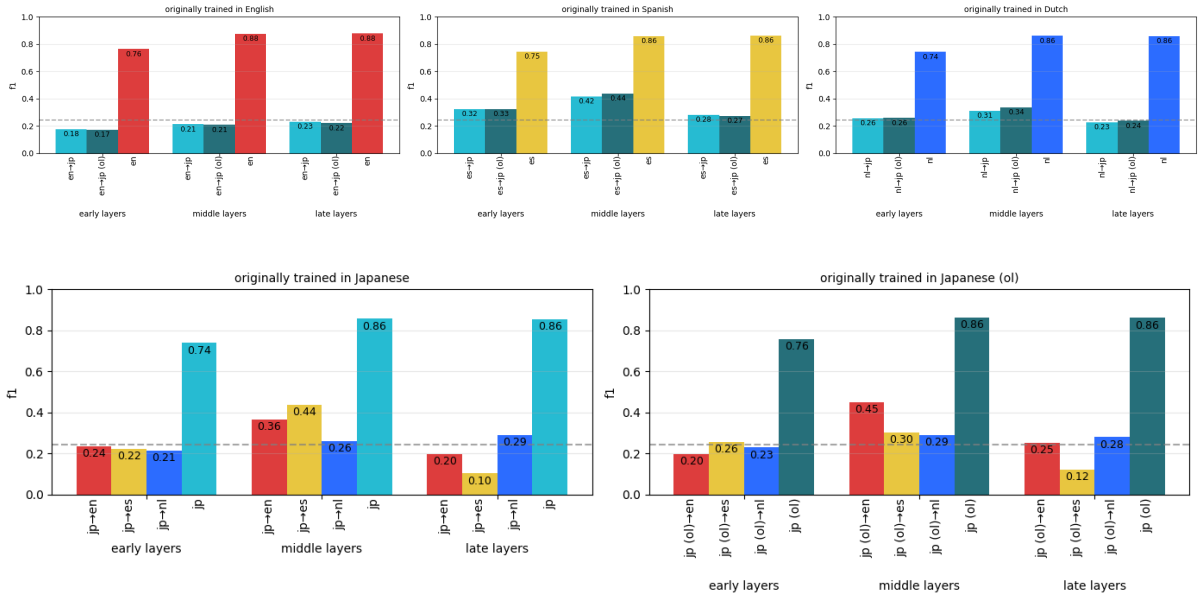

Figure 13: Direct cross-lingual transfer of Experiment 4.3 with no retraining for the **Aya** model. Compares the original SICK labels (*jp (ol)*) and JSICK labels (*jp*).

Figures 11, 12, and 13 show the results of Experiments 4.2 and 4.3 if the Japanese probes had been trained and tested using the original SICK labels instead of the re-annotated JSICK labels. In the majority of plots, with the exception of the lower plots in Figures 12 and 13, the differences between probes trained with either label set are minimal. In those specific plots, there are some differences in the results; for instance, the way Olmo's probes trained in Japanese with JSICK

labels transfer better to other languages than the same probes trained with original SICK labels (Figure 12). However, this effect is not as clearly present in Aya (Figure 13) and disappears after training the probes for 2 iterations. We therefore conclude that if we had used the original SICK labels for Japanese, there would not have been a large difference in the overall results.

D Experiment 4.3 using line plots

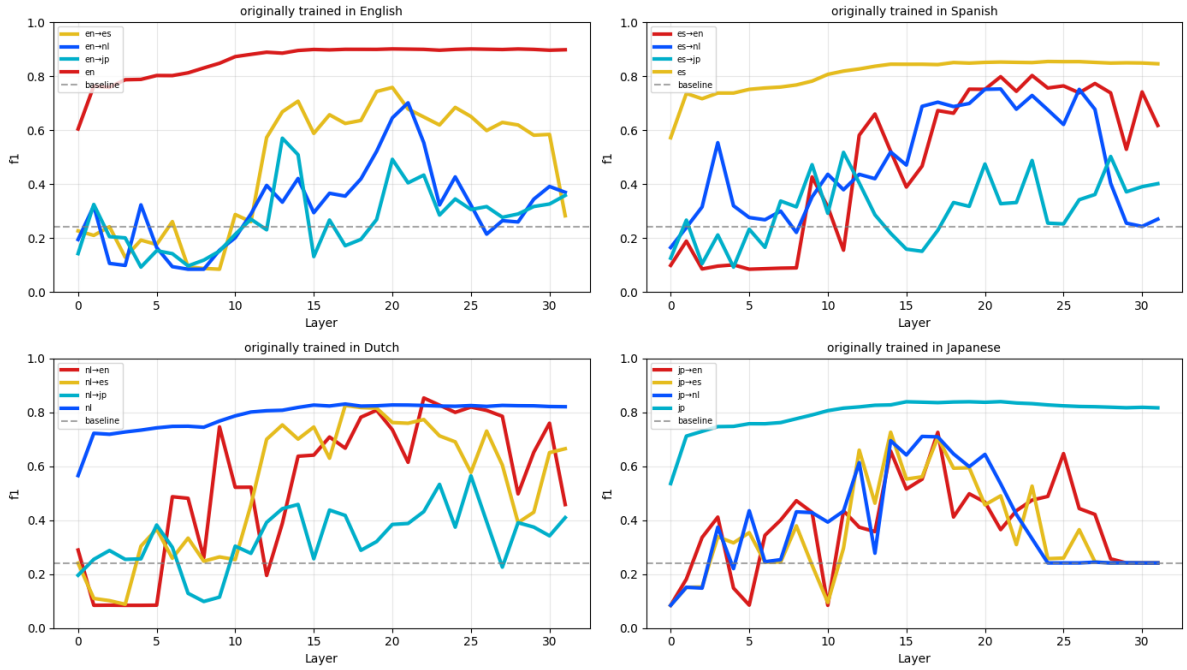

Figure 14: Line plot version of Figure 3 (Olmo transfer without retraining)

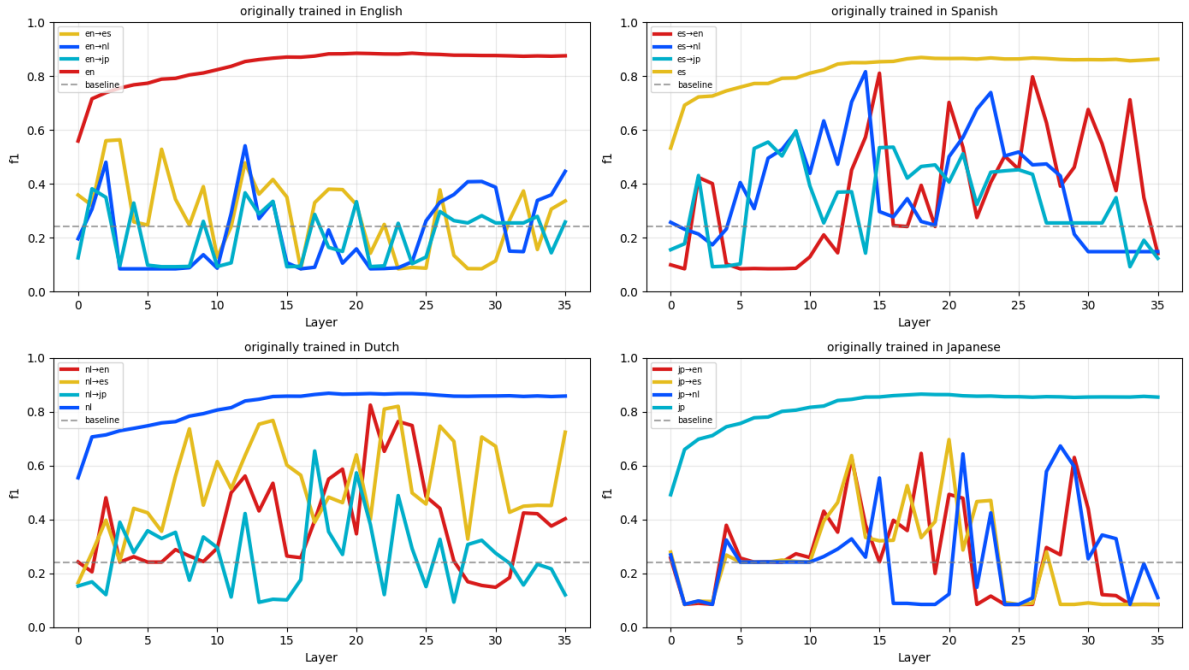

Figure 15: Line plot version of Figure 4 (Aya transfer without retraining)

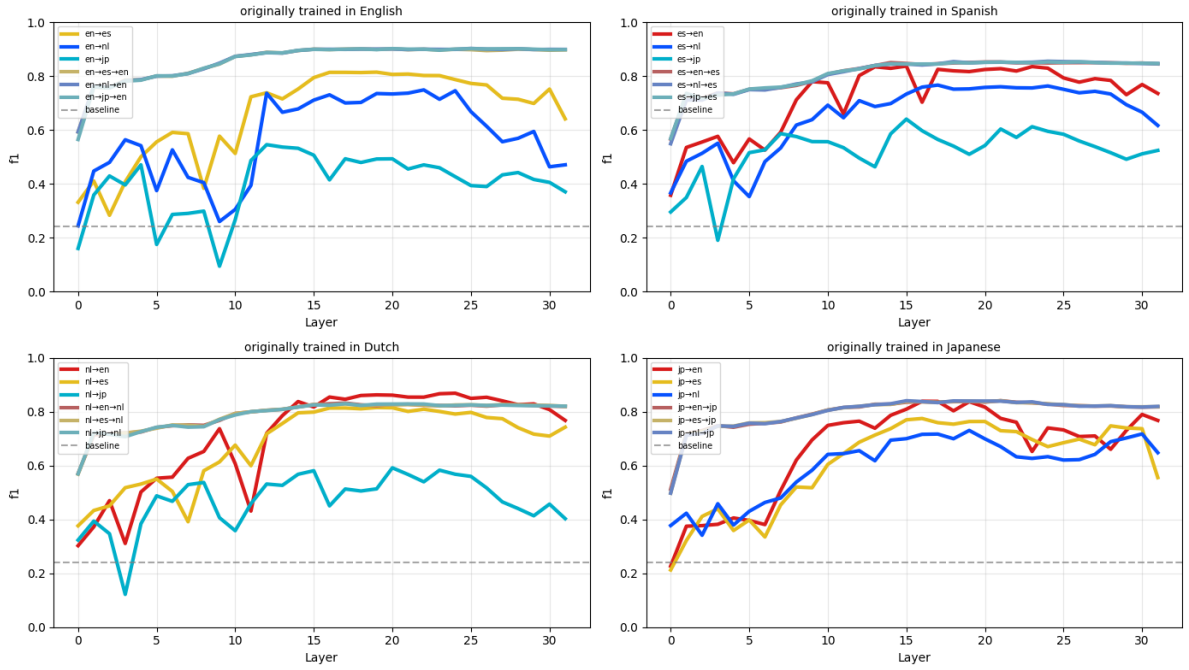

Figure 16: Line plot version of Figure 5 (Olmo transfer after 2 iterations of retraining)

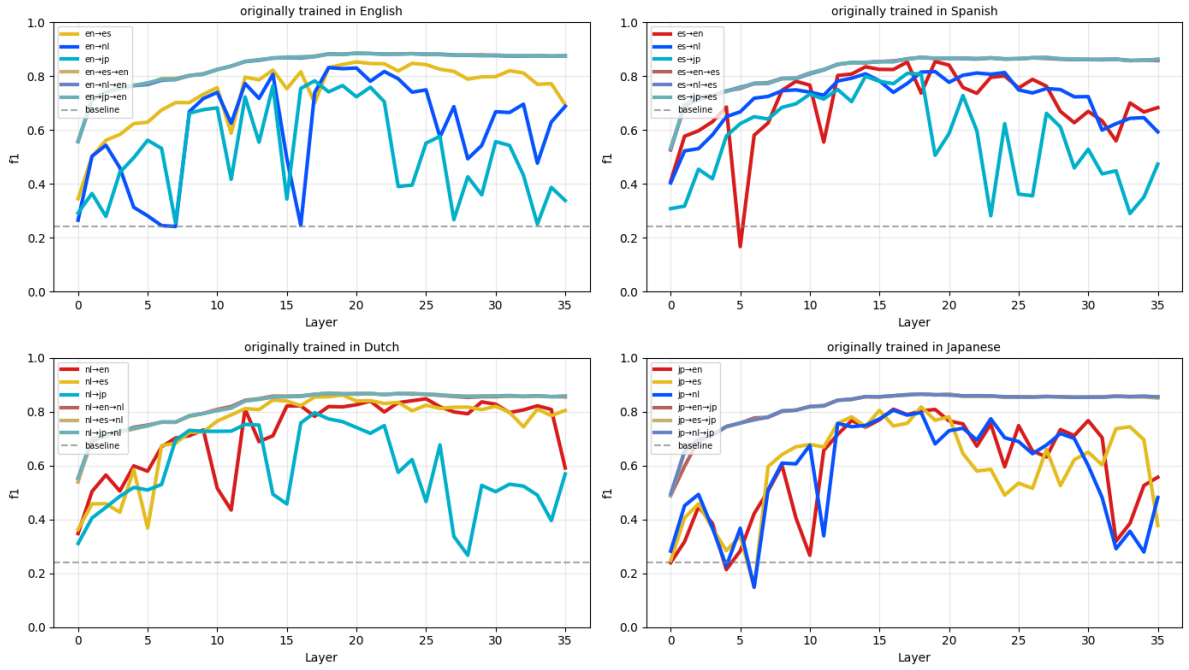

Figure 17: Line plot version of Figure 6 (Aya transfer after 2 iterations of retraining)

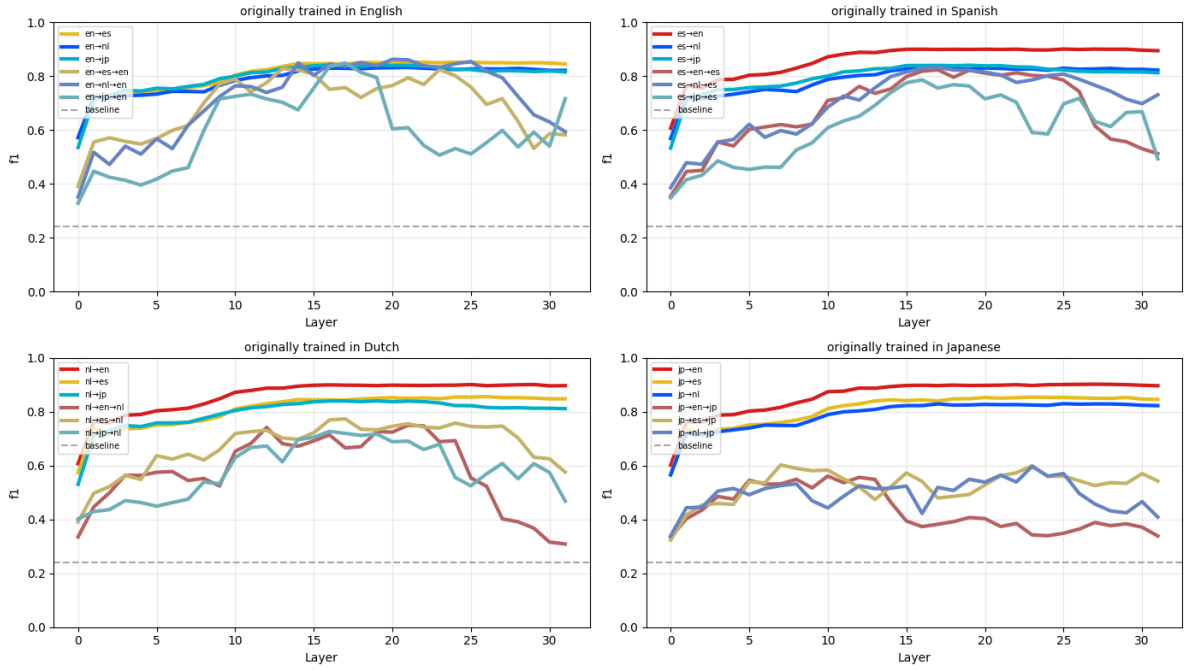

Figure 18: Line plot version of Figure 8 (Olmo transfer after 1000 iterations of retraining)

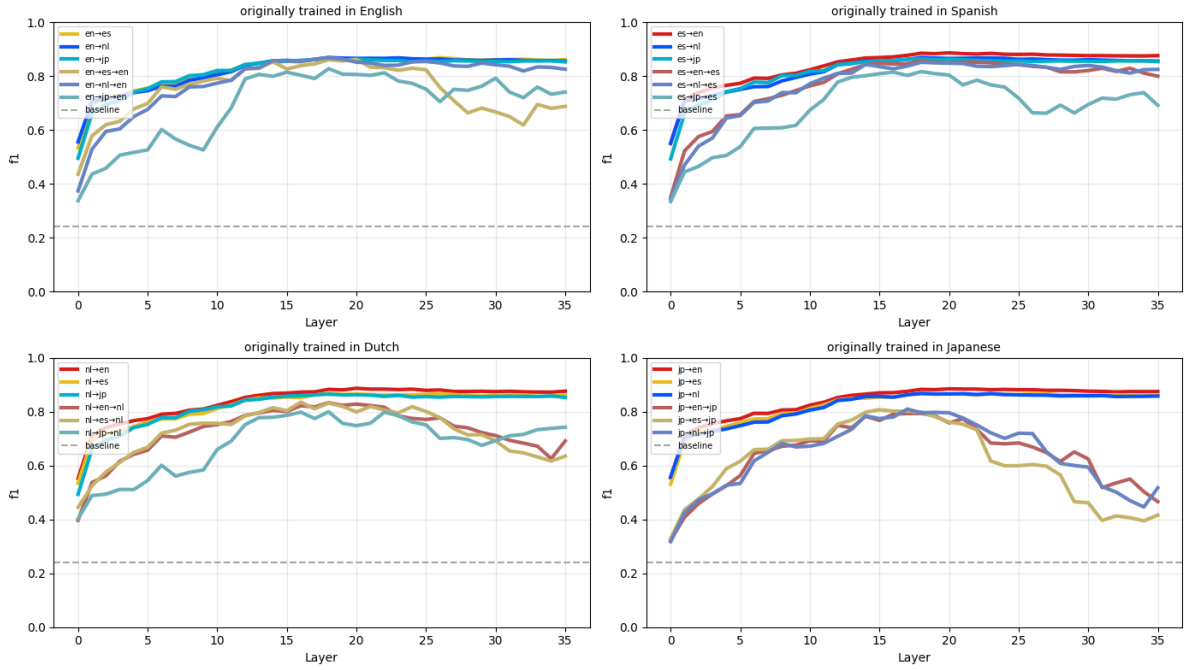

Figure 19: Line plot version of Figure 9 (Aya transfer after 1000 iterations of retraining)

As can be observed in these plots, Aya lines are highly inconsistent across the layers when directly transferred. This effect is reduced after 2 iterations.

E Code

All the code can be found in this GitHub repository