Do Llamas Work in English?
On the Latent Language of Multilingual Transformers

Chris Wendler*, Veniamin Veselovsky*, Giovanni Monea*, Robert West*
EPFL
{chris.wendler, veniamin.veselovsky, giovanni.monea, robert.west}@epfl.ch

Abstract

We ask whether multilingual language models trained on unbalanced, English-dominated corpora use English as an internal pivot language—a question of key importance for understanding how language models function and the origins of linguistic bias. Focusing on the Llama-2 family of transformer models, our study uses carefully constructed non-English prompts with a unique correct single-token continuation. From layer to layer, transformers gradually map an input embedding of the final prompt token to an output embedding from which next-token probabilities are computed. Tracking intermediate embeddings through their high-dimensional space reveals three distinct phases, whereby intermediate embeddings (1) start far away from output token embeddings; (2) already allow for decoding a semantically correct next token in middle layers, but give higher probability to its version in English than in the input language; (3) finally move into an input-language-specific region of the embedding space. We cast these results into a conceptual model where the three phases operate in "input space", "concept space", and "output space", respectively. Crucially, our evidence suggests that the abstract "concept space" lies closer to English than to other languages, which may have important consequences regarding the biases held by multilingual language models. Code and data is made available here: `https://github.com/epfl-dlab/llm-latent-language`.

1 Introduction

Most modern large language models (LLMs) are trained on massive corpora of mostly English text (Touvron et al., 2023; OpenAI, 2023). Despite this, they achieve strong performance on a broad range of downstream tasks, even in non-English languages (Shi et al., 2022). This raises a compelling question: How are LLMs able to generalize so well from their mainly English training data to other languages?

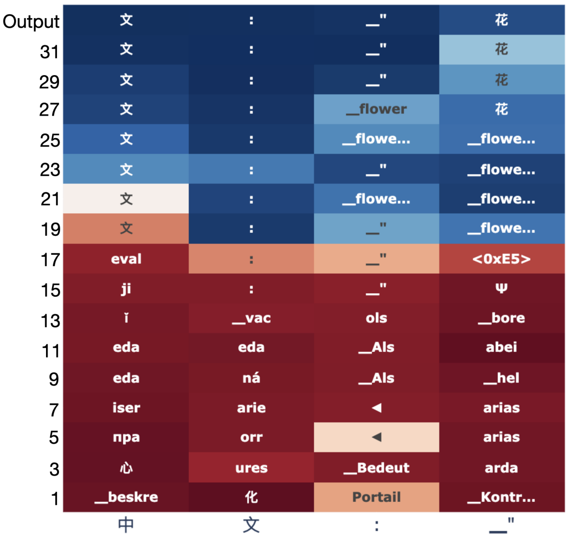


Figure 1: **Illustration of logit lens,** which applies language modeling head (here, Llama-2-7B) prematurely to latent embeddings in intermediate layers, yielding one next-token distribution per position (x-axis) and layer (y-axis). We show final tokens of translation prompt (cf. Sec. 3.3) ending with "Français: "fleur" - 中文: "" (where "中文" means "Chinese"). Final layer correctly ranks "花" (translation of "fleur") on top, whereas intermediate layers decode English "flower". Color indicates entropy of next-token distributions from low (blue) to high (red). (Plotting tool: Belrose et al. (2023).)

Intuitively, one way to achieve strong performance on non-English data in a data-efficient manner is to use English as a pivot language, by first translating input to English, processing it in English, and then translating the answer back to the input language. This method has been shown to lead to high performance when implemented explicitly (Shi et al., 2022; Ahuja et al., 2023; Huang et al., 2023). Our guiding inquiry in this work is whether pivoting to English also occurs implicitly when LLMs are prompted in non-English.

In the research community as well as the popular press, many seem to assume that the answer is yes,

[*] Equal contribution.

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 15366–15394
August 11-16, 2024 ©2024 Association for Computational Linguistics

epitomized by claims such as, “The machine, so to say, thinks in English and translates the conversation at the last moment into Estonian” (Piir, 2023). In this work, we set out to move beyond such speculation and investigate the question empirically.

The question is of major importance. On the one hand, implicitly using English as an internal pivot could bias LLMs toward Anglocentric patterns that could predispose the model to certain linguistic elements (lexicon, grammar, metaphors, etc.), while also shaping more profound behaviors related to, e.g., emotional stance (Boroditsky et al., 2003) or temporal reasoning (Núñez and Sweetser, 2006). On the other hand, if LLMs do not use English as a pivot, it raises questions of how else they manage to work so remarkably well even in low-resource languages. Overall, the quest for an internal pivot language holds promise to advance our understanding of how LLMs function no matter if we succeed.

Investigating the existence of an internal LLM language is complicated by the scale and notoriously inscrutable nature of the neural networks behind LLMs, which after the input layer do not operate on discrete tokens, but on high-dimensional floating-point vectors. How to understand if those vectors correspond to English, Estonian, Chinese, etc.—or to no language at all—is an open problem, and the question of whether LLMs use an internal pivot language has therefore, to the best of our knowledge, not been addressed empirically before.

Summary of contributions. To overcome these hurdles, we draw on, and contribute to, the nascent field of mechanistic interpretability (cf. Sec. 2). In a transformer, each input token’s embedding vector is gradually transformed layer by layer without changing its shape. After the final layer, an “unembedding” operation turns the vector into a next-token distribution. Focusing on the Llama-2 family of models (Touvron et al., 2023)—among today’s largest open-source LLMs—we find that applying the “unembedding” operation prematurely in intermediate, non-final layers—a technique called *logit lens* (Nostalgebraist, 2020)—already decodes a contextually appropriate token early on (Fig. 1), giving us a (limited) glimpse at the model’s otherwise hard-to-interpret numerical internal state.

Exploiting this fact, we carefully devise prompts that allow us to determine whether a logit-lens-decoded token is semantically correct and to what language it belongs (e.g., a prompt asking the model to translate French “fleur” [“flower”] to Chinese “花”; cf. Fig. 1). Tracking language probabilities across layers, we observe that no contextually appropriate tokens are decoded in the first half of layers, followed by a sudden shift of probability mass onto the English version (“flower”) of the correct next token, and finally a shift to the correct next token in the target language (“花”).

Expanding on this first evidence of English as an internal pivot language, we analyze latent embeddings directly as high-dimensional Euclidean points, rather than via the logit lens. This allows us to draw a more nuanced picture of the anatomy of Llama-2’s forward pass, suggesting that, in middle layers, the transformer operates in an abstract “concept space” that is partially orthogonal to a language-specific “token space”, which is reached only in the final layers. In this interpretation, the latent embeddings’ proximity to English tokens observed through the logit lens follows from an English bias in concept space, rather than from the model first translating to English and then “restarting” its forward pass from there.

We conclude by discussing implications and future directions for studying latent biases and their effects—a crucial step toward trustworthy AI.

2 Related work

Multilingual language models. Multilingual language models (LMs) are trained to simultaneously handle multiple input languages. Examples include mBERT (Devlin et al., 2018), mBART (Liu et al., 2020), XLM-R (Conneau et al., 2020a), mT5 (Xue et al., 2021), XGLM (Lin et al., 2022), mGPT (Shliazhko et al., 2022), BLOOM (Scao et al., 2022), and PolyLM (Wei et al., 2023). Current frontier models such as GPT-4, PaLM, and Llama-2, despite performing better in English due to their Anglocentric training data (Huang et al., 2023; Bang et al., 2023; Zhang et al., 2023), still do well across languages (Shi et al., 2022).

Researchers have devised numerous methods for efficiently transferring LM capabilities across languages, e.g., by aligning contextual embeddings (Schuster et al., 2019; Cao et al., 2020), relearning embedding matrices during finetuning on a new language (Artetxe et al., 2020), or repeatedly doing so during pretraining (Chen et al., 2023).

Several approaches leverage English as a pivot language. For instance, Zhu et al. (2023) show that Llama can be efficiently augmented with multilingual instruction-following capabilities thanks to its English representations. Likewise, Zhu et al. (2024) demonstrate the feasibility of leveraging language models' proficiency in English for non-English contexts by fine-tuning them on translation data and English-only instructional data. They successfully employ this approach to enhance the multilingual reasoning capabilities of Llama-2. Regarding non-Latin low-resource languages, Husain et al. (2024) illustrate that leveraging both romanized and English data proves to be an effective strategy for efficiently improving multilingual task performance. Prompting strategies, too, can improve multilingual performance by leveraging English as a pivot language, e.g., by simply first translating prompts to English (Shi et al., 2022; Ahuja et al., 2023; Etxaniz et al., 2023) or by instructing LMs to perform chain-of-thought reasoning (Wei et al., 2022) in English (Huang et al., 2023).

Although employing high-resource languages can enhance performance on low-resource languages, it might also bias output generation in low-resource languages, e.g., in terms of grammar (Papadimitriou et al., 2022).

Researchers have also investigated how latent representations differ across languages within multilingual models. In the case of encoder-only models such as mBERT, converging evidence suggests the existence of a language-agnostic space in later layers following language-specific early layers (Libovický et al., 2020; Conneau et al., 2020b; Muller et al., 2021; Choenni and Shutova, 2020).

Mechanistic interpretability. The nascent field of mechanistic interpretability (MI) aims to reverse-engineer and thereby understand neural networks, using techniques such as circuit discovery (Nanda et al., 2023; Conmy et al., 2023), controlled task-specific training (Li et al., 2022; Marks and Tegmark, 2023), and causal tracing (Meng et al., 2022; Monea et al., 2023).

For smaller models, e.g., GPT-2 (Radford et al., 2019) and Pythia (Biderman et al., 2023), MI approaches such as sparse probing (Gurnee et al., 2023) have revealed monosemantic French (Gurnee et al., 2023) and German (Quirke et al., 2023) language neurons and context-dependent German n-gram circuits (subnetworks for boosting the probability of German n-grams when the monosemantic German context neuron is active) (Quirke et al., 2023).

The most relevant tools from the MI repertoire in the context of this work are the *logit lens* (Nostalgebraist, 2020), *tuned lens* (Belrose et al., 2023), and *direct logit attribution* (Elhage et al., 2021), which decode intermediate token representations from transformer models in different ways. The logit lens does so by using the language modeling head, which is usually only applied in the final layer, prematurely in earlier layers, without any additional training. The more sophisticated tuned lens additionally trains an affine mapping for transforming an intermediate latent state such that it mimics the token predictions made by the final latent state. Finally, direct logit attribution generalizes the logit lens by considering the logit contribution of each individual attention head.

In this work, we heavily rely on the logit lens, described further in Sec. 3.2, as opposed to the tuned lens. The latter would defeat our purpose of understanding whether Llama-2, when prompted in non-English, takes a detour via English internal states before outputting non-English text. As the tuned lens is specifically trained to map internal states—even if corresponding to English—to the final, non-English next-token prediction, the optimization criterion would "optimize away" our signal of interest.

3 Materials and methods

3.1 Language models: Llama-2

We focus on the Llama-2 family of language models (Touvron et al., 2023), some of the largest and most widely used open-source models. The models were trained on a multilingual corpus that is largely dominated by English, which comprises 89.70% of the corpus. However, given the size of the training data (two trillion tokens), even a small percentage of non-English training data still constitutes a large number of tokens in absolute terms (e.g., 0.17% = 3.4B German tokens, 0.13% = 2.6B Chinese tokens). Consequently, Llama-2 is, despite its English bias, considered a multilingual model.

Versions. Llama-2 comes in three model sizes, with 7B/13B/70B parameters, 32/40/80 layers, and embedding dimension $d = 4096/5120/8192$, respectively. Across all model sizes, the vocabulary V contains $v = 32{,}000$ tokens. Here we study all model sizes, using 8-bit quantization (Dettmers et al., 2022) in our experiments.

Architecture. Llama-2 is an autoregressive, decoder-only, residual-based transformer. Such models maintain the shape of the input data throughout the computation process during a forward pass: one embedding vector, a so-called *latent,* per input token $x_1, \ldots, x_n \in V$, where n is the input sequence length. The initial latents $h_1^{(0)}, \ldots, h_n^{(0)} \in \mathbb{R}^d$ are obtained from a learned embedding dictionary that contains one fixed vector per vocabulary token. Each of these latents is incrementally updated layer by layer by adding a residual. The residual added to the latent at position i in layer j is a function f_j of all preceding tokens' latents $h_1^{(j-1)}, \ldots, h_i^{(j-1)}$:

$$h_i^{(j)} = h_i^{(j-1)} + f_j\left(h_1^{(j-1)}, \ldots, h_i^{(j-1)}\right), \tag{1}$$

where the resulting vector $h_i^{(j)}$ is still of dimension d. The function f_j itself, called a transformer block, is composed of a masked self-attention layer followed by a feed-forward layer with a residual connection and root mean square (RMS) normalization in between (Vaswani et al., 2017; Touvron et al., 2023). Due to RMS normalization, all latents lie on a d-dimensional hypersphere of radius $\sqrt{d}$.

In pretraining, all transformer blocks $f_1, \ldots, f_m$ (with m the number of layers) are tuned such that the final latent $h_i^{(m)}$ for position i is well-suited for predicting the token at position $i+1$. For prediction, the final embedding vector is multiplied with a so-called *unembedding matrix* $U \in \mathbb{R}^{v \times d}$, which yields a real vector $z_i = Uh_i^{(m)} \in \mathbb{R}^v$ containing a so-called *logit* score z_{it} for each vocabulary token $t \in V$. These scores are then transformed into probabilities $P(x_{i+1} = t \,|\, x_1, \ldots, x_i) \propto e^{z_{it}}$ via the softmax operation.

3.2 Interpreting latent embeddings: Logit lens

When transformers are deployed in practice, only the final latent vectors after the last transformer block are turned into token distributions by multiplying them with U and taking a softmax. However, since latents have the same shape in all layers, any latent can in principle be turned into a token distribution, by treating it as though it were a final-layer latent. Prematurely decoding tokens from latents this way, a method called the *logit lens* (cf. Sec. 2), can facilitate the inspection and interpretation of the internal state of transformers. Using the logit lens, we obtain one next-token distribution $P(x_{i+1} \,|\, h_i^{(j)})$ per position i and layer j.

We illustrate the logit lens in Fig. 1, where every cell shows the most likely next token when applying the logit lens to the latent in that position and layer. As seen, the logit lens decodes contextually appropriate tokens already in intermediate layers.

3.3 Data: Tasks for eliciting latent language

Our goal is to explore whether Llama-2's internal, latent states correspond to specific natural languages. Although the logit lens allows us to map latent vectors to token distributions, we still require a mapping from token distributions to languages.

Doing so in general is difficult as many tokens are ambiguous with respect to language; e.g., the token "an" is commonly used in English, French, and German, among others. To circumvent this issue, we construct prompts $x_1 \ldots x_n$ where the correct next token x_{n+1} is (1) obvious and (2) can be unambiguously attributed to one language.

Prompt design. To ensure that the next token is obvious (criterion 1), we design three text completion tasks where the next token x_{n+1} can be easily inferred from the prompt $x_1 \ldots x_n$. In describing the tasks, we use Chinese as an example language.

Translation task. Here the task is to translate the preceding non-English (e.g., French) word to Chinese. We show the model four words with their correct translations, followed by a fifth word without its translation, and let the model predict the next token ("中文" means "Chinese" below):

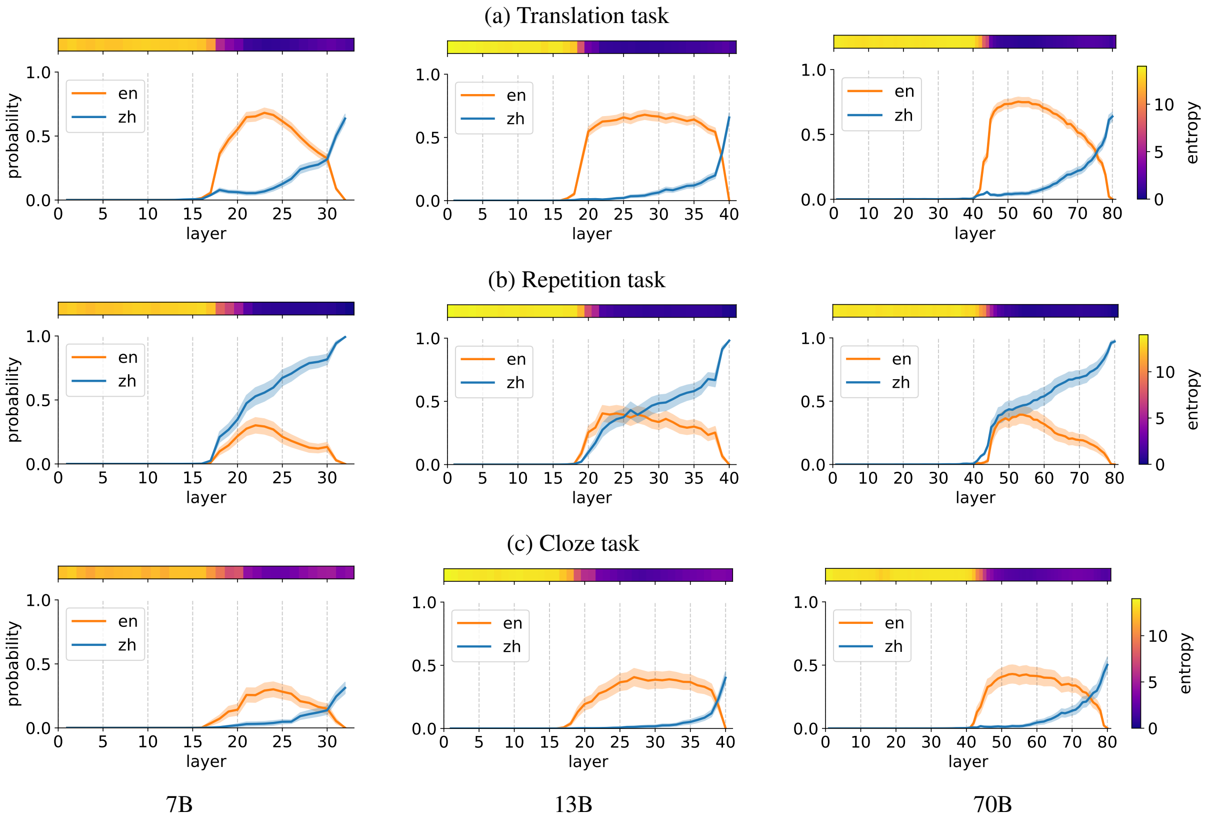
```
Français: "vertu" - 中文: "德"
Français: "siège" - 中文: "座"
Français: "neige" - 中文: "雪"
Français: "montagne" - 中文: "山"
Français: "fleur" - 中文: "
```

With such a prompt, Llama-2 can readily infer that it should translate the fifth French word. We carefully select words as described below and construct one prompt per word by randomly sampling demonstrations from the remaining words.

Repetition task. Similarly, we task the model to simply repeat the last word, instead of translating it, by prompting as follows:

```
中文: "德" - 中文: "德"
中文: "座" - 中文: "座"
中文: "雪" - 中文: "雪"
中文: "山" - 中文: "山"
中文: "花" - 中文: "
```

Cloze task. As a slightly harder task, we consider a cloze test, where the model must predict a masked word in a sentence. Given a target word, we construct an English sentence starting with the word by prompting GPT-4, mask the target word, and translate the sentence to the other languages. To construct prompts, we sample two demonstrations


(a) Translation task

(b) Repetition task

(c) Cloze task

Figure 2: **Language probabilities for latents during Llama-2 forward pass,** for **(a)** translation task from union of German/French/Russian to Chinese, **(b)** Chinese repetition task, **(c)** Chinese cloze task. Each task evaluated for model sizes (columns) 7B, 13B, 70B. On x-axes, layer index; on y-axes, probability (according to logit lens) of correct Chinese next token (blue) or English analog (orange). Error bars show 95% Gaussian confidence intervals over input texts (353 for translation, 139 for repetition and cloze).

from the remaining words. An English example before translation to the other languages follows:

```
A "___" is used to play sports like soccer and basketball. Answer: "ball".
A "___" is a solid mineral material forming part of the surface of the earth. Answer: "rock".
A "___" is often given as a gift and can be found in gardens. Answer: "
```

Word selection. To enable unambiguous language attribution (criterion 2), we construct a closed set of words per language. As a particularly clean case, we focus on Chinese, which has many single-token words and does not use spaces. We scan Llama-2's vocabulary for single-token Chinese words (mostly nouns) that have a single-token English translation. This way, Llama-2's probabilities for the correct next Chinese word and for its English analog can be directly read off the next-token probabilities.

For robustness, we also run all experiments on German, French, and Russian. For this, we translate the selected Chinese/English words and, for each language, discard words that share a token prefix with the English version, as this would render language detection (cf. Sec. 3.4) ambiguous.

We work with 139 Chinese, 104 German, 56 French, and 115 Russian words (cf. Appendix A.1).

3.4 Measuring latent language probabilities

To investigate a hypothetical pivot language inside Llama-2, we apply the logit lens to the latents $h_n^{(j)}$ corresponding to the last input token x_n for each layer j, obtaining one next-token distribution $P(x_{n+1} \mid h_n^{(j)})$ per layer. Our prompts (cf. Sec. 3.3) are specifically designed such that an intermediate next-*token* distribution lets us estimate the probability of the correct next *word* in the input language as well as English. Since we specifically select single-token words in Chinese (ZH) as well as English (EN), we can simply define the probability of language $\ell \in \{\text{ZH}, \text{EN}\}$ as the probability of the next token being ℓ's version t_ℓ of the correct single-token word: $P(\text{lang} = \ell \mid h_n^{(j)}) := P(x_{n+1} = t_\ell \mid h_n^{(j)})$. (For readability we also simply write $P(\text{lang} = \ell)$.)

Note that this does not define a distribution over languages, as generally $\sum_\ell P(\text{lang} = \ell) < 1$.

In other languages (and in corner cases in Chinese and English), we must account for multiple tokenizations and whitespaces (cf. Appendix A.2).

4 Results

When presenting results, we first (Sec. 4.1) take a probabilistic view via the logit lens (Sec. 3.2), for all tasks and all model sizes. (Since the results are consistent across languages, we focus on Chinese here and refer to Appendix B for French, German, and Russian.) Then (Sec. 4.2) we drill deeper by taking a geometric view of how token embeddings drift as the transformer computes layer by layer.

4.1 Probabilistic view: Logit lens

The logit lens gives us one set of language probabilities (cf. Sec. 3.4) per input prompt and layer. Fig. 2 tracks the evolution of language probabilities from layer to layer, with one plot per combination of model size (columns) and task[1] (rows). The x-axes show layer indices, and the y-axis the language probabilities $P(\text{lang} = \text{ZH})$ and $P(\text{lang} = \text{EN})$ averaged over input prompts.

On the translation and cloze tasks a consistent picture emerges across model sizes. Neither the correct Chinese token nor its English analog garner any noticeable probability mass during the first half of layers. Then, around the middle layer, English begins a sharp rise followed by a decline, while Chinese slowly grows and, after a crossover with English, spikes on the last five layers. On the repetition task, Chinese already rises alongside English (discussed in Sec. 6). This is in contrast to all other languages, where English rises first (Appendix B).

On top of the language probabilities (Sec. 3.4), the entropy of the full next-token distribution is shown as a heatmap above the plots. We again observe a consistent pattern across tasks and model sizes: high entropy in the first half of layers, while both $P(\text{lang} = \text{ZH})$ and $P(\text{lang} = \text{EN})$ are close to zero, followed by a sharp drop at the same time that $P(\text{lang} = \text{EN})$ rises. From there on, entropy remains low, with a slight rebound as probability mass shifts from English to Chinese.

With $32{,}000 \approx 2^{15}$ tokens in the vocabulary, the early entropy of around 14 bits implies a close-to-uniform next-token distribution (around 15 bits).

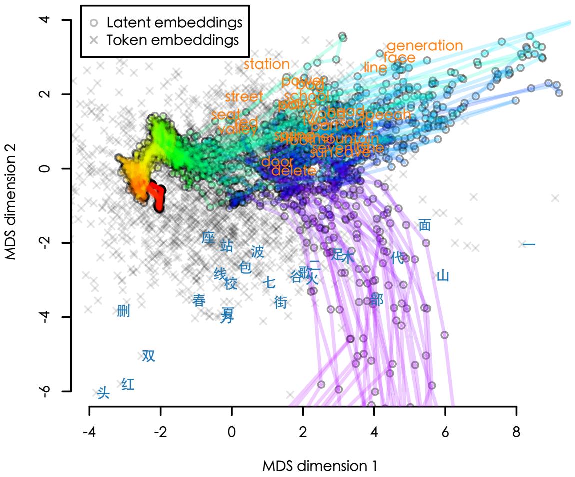


Figure 3: **Latent trajectories through transformer layers.** 2D embedding of latents (○) and output tokens (×) found via multidimensional scaling. Latents for same prompt connected by rainbow-colored path, proceeding from layer 1 (red) to 80 (violet). Labels for correct Chinese next tokens (one per prompt) in blue, for English analogs in orange. Takeaway: latents reach correct Chinese token after detour through English.

Path visualization. The plots of Fig. 2 only consider the probability of the correct Chinese next token and its English analog, without speaking to the remaining tokens. To form an intuition of the entire distribution, we use dimensionality reduction to visualize the data. First, we define the distance between a latent h_n at position n and a token t via the negative log-likelihood of t given h_n, as computed by the logit lens (cf. Sec. 3.4): $d(h_n, t) = -\log P(x_{n+1} = t \mid h_n)$. Then, we use classical multidimensional scaling to embed tokens and latents in an approximately distance-preserving joint 2D space. (Intra-token and intra-latent distances are set to $\max_{h,t} d(h,t)$, which serves as a “spring force” pushing the 2D points apart.)

A transformer’s forward computation for a given final input token x_n can now be visualized by connecting the 2D embeddings of the latents $h_n^{(j)}$ in subsequent layers j, as presented and explained in Fig. 3 (German-to-Chinese translation, 70B). We make two observations: (1) An English and a Chinese token cluster emerges, suggesting that the same latent also gives high probability to an entire language, in addition to the language-specific version of the correct next token. (2) Paths first pass through the English cluster, and only later reach the Chinese cluster. Taken together, the emerging picture is that, when translating a German word

[1] In Fig. 2, translation task uses union of German, French, and Russian as source languages. For individual source languages, as well as all target languages, cf. Appendix B.

to Chinese, Llama-2 takes a "detour" through an English subspace.

So far, we have characterized the transformer's intermediate latent states from a probabilistic perspective, by studying the next-token distributions obtained via the logit lens. For a deeper understanding, we next take a geometric perspective and analyze latents directly as points in Euclidean space, i.e., before mapping them to token probabilities.

4.2 Geometric view: An 8192D space Odyssey

Simplistically, the task solved by an autoregressive transformer is to map the input embedding of the current token to the output embedding of the next token. The task is solved incrementally, each layer modifying (by adding a residual) the latent vector produced by the previous layer, a process that, geometrically, describes a path through d-dimensional Euclidean space. We now set out to characterize this path. Since the probabilistic view (Fig. 2) gave consistent results across tasks and model sizes, we focus on one task (translation) and one model size (70B, i.e., $d = 8192$).

Embedding spheres. Output token embeddings (rows of the unembedding matrix U) and latents h cohabitate the same d-dimensional Euclidean space. In fact, due to RMS-normalization (Sec. 3.1), latents by construction live on a hypersphere of radius $\sqrt{d} \approx 90.1$. Additionally, by analyzing the 2-norm of output token embeddings (mean 1.52, SD 0.23), we find that the latter also approximately lie on a sphere, of radius 1.52.

Token energy. Importantly, token embeddings occupy their sphere unevenly; e.g., the first 25% of the principal components account for 50% of the total variance, and the first 54% for 80%.[2] To build intuition, first consider a hypothetical extreme case where tokens lie in a proper subspace ("token subspace") of the full d-dimensional space (even though, empirically, U has rank d, so the tokens' output embeddings span all of $\mathbb{R}^d$). If a latent h has a component orthogonal to the token subspace, it includes information that is irrelevant for predicting the next token based on h alone (since logits are scalar products of latent and token vectors). The orthogonal component can still be important for the computations carried out by later layers and for predicting the next token in those layers. But the logit lens, which decodes latents into tokens prematurely in intermediate layers, will be blind to the orthogonal component.

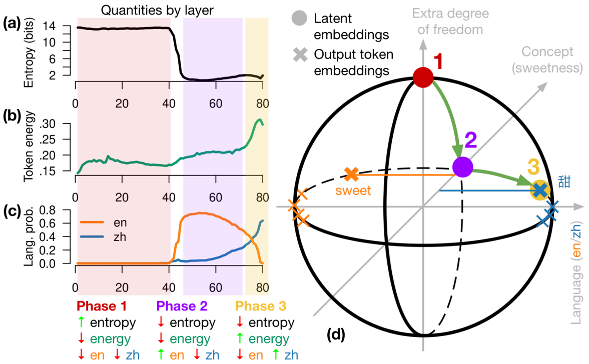


Figure 4: **Anatomy of transformer forward pass** when translating to Chinese (cf. Sec. 3.3). Layer-by-layer evolution of **(a)** entropy of next-token distribution, **(b)** token energy, **(c)** language probabilities. As latents are transformed layer by layer, they go through three phases (Sec. 4.2), **(d)** traveling on a hypersphere, here in 3D instead of actual 8192D (Sec. 5). "甜" means "sweet".

A latent h's angle with the "token subspace" thus measures how much of h is irrelevant for immediately predicting the next token. Concretely, we consider the mean squared cosine between h and the token embeddings (rows of U) to capture how much of h's "energy" translates into logit scores. For interpretability, we normalize by the mean squared cosine among token embeddings themselves,[3] obtaining what we call h's squared *token energy*

$$E(h)^2 = \frac{\frac{1}{v}\|\hat{U}h\|_2^2 \,/\, \|h\|_2^2}{\frac{1}{v^2}\|\hat{U}\hat{U}^\top\|_F^2} = \frac{v}{d}\,\frac{\|\hat{U}h\|_2^2}{\|\hat{U}\hat{U}^\top\|_F^2} \qquad (2)$$

($\hat{U}$ being U with 2-normalized rows), which captures h's proximity to "token subspace", compared to a random token's proximity to "token subspace".

We visualize token energy and its relation to other key quantities in Fig. 4. As a function of layer (Fig. 4(b)), root mean squared token energy is low (around 20%) and mostly flat before layer 70, when it suddenly spikes—just when next-token predictions switch from English to Chinese (Fig. 4(c)). In sum, Fig. 4(a–c) reveals three phases:

1. **Phase 1** (layers 1–40): High entropy (14 bits, nearly uniform), low token energy, no language dominates.
2. **Phase 2** (layers 41–70): Low entropy (1–2 bits), low token energy, English dominates.

[2] Moreover, Cancedda (2024) showed that a significant fraction of the principal components can be omitted as long as attention sinking are preserved.

[3] In practice, we use $\hat{U}^\top\hat{U}$ instead of $\hat{U}\hat{U}^\top$ in (2), which has equal Frobenius norm but is more efficient to compute.

3. **Phase 3** (layers 71–80): Low entropy, high token energy (up from 20% to 30%), Chinese dominates.

5 Conceptual model

Next, we formulate a conceptual model that is consistent with the above observations.

In order to predict the next token, the transformer's job essentially consists in mapping the input embedding of the current token to the output embedding of the next token. **Phase 1** is focused on building up a better feature representation for the current token from its input embedding, by dealing with tokenization issues (e.g., integrating preceding tokens belonging to the same word), integrating words into larger semantic units, etc. This phase is not yet directly concerned with predicting the next token, with latents remaining largely orthogonal to output token space (low token energy), leading to small dot products between latents and output token embeddings, and thus to high entropy.

In **Phase 2,** latents live in an abstract "concept space", which, unlike in Phase 1, is no more orthogonal to the output token space. Rather, latent "concept embeddings" are closer to those output token embeddings that can express the respective concept (across languages, synonyms, etc.), leading to low entropy. Among the concept-relevant tokens, English variants lie closer to the concept embedding than non-English variants (due to the model's overwhelming exposure to English during training), leading to higher probabilities for English than Chinese tokens. Despite the correlation between concept and token embeddings, concept embeddings also carry much information that goes beyond output tokens (including input-specific contextual information and information about the target language), leading to a still-low token energy.

In **Phase 3,** the model maps abstract concepts to concrete words/tokens in the target language. Information that is irrelevant for next-token prediction is discarded, leading to a spike in token energy.

Sketch. This model is illustrated—with a strongly simplified toy-like sketch—in Fig. 4(d). In this picture, the model operates in 3D (rather than the actual 8192D) space. All embeddings (output tokens and latents) lie on a sphere around the origin. Token embeddings lie on the equator and are mostly spread out along the x-axis (left/right), which captures language (English left, Chinese right). The y-axis (front/back) captures concepts, in this toy picture along a 1D "sweetness" scale. The z-axis (bottom/top) provides an extra degree of freedom that can be used to store information about context, language, etc. A transformer forward pass moves along the surface of the sphere. In Phase 1, the latent starts out at the north pole, orthogonal to both output token and concept embeddings. Phase 2 rotates the latent into concept space; English tokens are more likely because their embeddings have a stronger concept component y. Finally, Phase 3 rotates the latent along the equator into the target language's hemisphere, onto the output token that best captures the active concept in that language.

6 Discussion

In our attempt to answer whether Llama-2 models internally use English as a pivot language, we found that latent embeddings indeed lie further from the correct next token in the input language than from its English analog, leading to overwhelmingly English internal representations as seen through the logit lens. It might thus be tempting to conclude that, yes, Llama-2 uses English as an implicit pivot, similar to researchers' prior use of English as an explicit pivot (Shi et al., 2022; Ahuja et al., 2023; Huang et al., 2023). But our answer must be more nuanced, as much of the latents' "energy" points in directions that are largely orthogonal to output token embeddings and thus do not matter for next-token prediction. The model can use these directions as extra degrees of freedom for building rich feature representations from its raw inputs (Yosinski et al., 2014, 2015; Geva et al., 2022), which could be seen as forming an abstract "concept space". In this interpretation, the model's internal lingua franca is not English but concepts—concepts that are biased toward English. Hence, English could still be seen as a pivot language, but in a semantic, rather than a purely lexical, sense.

Our experiments involve three text completion tasks. The translation and cloze tasks operate at a semantic level, whereas the word repetition task is purely syntactic. Yet, in most languages (Fig. 7) the pattern is similar to that for the two other tasks, with tokens first going through an "English phase"—possibly because recognizing that the task is to simply copy a token requires semantic understanding, which is achieved only in concept space, which in turn is closer to English token embeddings.

This said, note that the English-first pattern is less pronounced on the repetition task (Fig. 7), where the input language rises earlier than on the other tasks or, for Chinese (Fig. 7(e)) even simultaneously with, or faster than, English. This might be due to tokenization: for Chinese we explicitly chose 100% single-token words, as opposed to only 13% for Russian, 43% for German, and 55% for French (Table 1). Where language-specific tokens are available, the detour through English seems less pronounced. This supports prior concerns about the importance of tokenization, which not only burdens minority languages with more tokens per word (Artetxe et al., 2020), but, as we show, also forces latents through an English-biased semantic space.

Future work should investigate in what ways an English bias in latent space could be problematic, e.g., by biasing downstream model behavior. We see promise in designing experiments building on work from psycholinguistics, which has shown that concepts may carry different emotional values in different languages (Boroditsky et al., 2003) and that using one word for two concepts (colexification) may affect cognition (Di Natale et al., 2021). Future work should also study how English bias changes when decreasing the dominance of English during training, e.g., by applying our method to Llama-2 derivatives with a different language mix (Goddard, 2023; Plüster, 2023; Huang, 2023; Kim, 2023), or by using less Anglocentric tokenizers.

Such work will give important clues for decreasing English bias and enabling more equitable AI.

Limitations

In this paper, we focus on the Llama-2 family of language models, which limits the claims we can make about other English-dominated models (but see Appendix B.2 for initial evidence that Mistral-7B behaves identically). Moreover, since the proposed method relies on model parameters, little can be said about the more widely used closed-source models. Nonetheless, the methods outlined in this paper can be straightforwardly applied to other autoregressive transformers and generalized to non-autoregressive ones (given their parameters are available), a direction that warrants future exploration.

Additionally, the tasks outlined in the paper are simple and provide a highly controlled, yet toy-like, context for studying the internal language of LLMs. This is essential as a first step to illustrate existence, but future work should extend to a wider range of tasks; these may include more culturally sensitive problems, popular use-cases (cf. Sec. 6), and technical analyses that go beyond single tokens.

While we find evidence of a “concept space” in our interpretation (Sec. 5), we have limited understanding of the structure of this space in its original high-dimensional form. We believe that better understanding and mapping out this concept space is an important future direction and will result in a stronger basis for the presented conceptual model.

Finally, while the logit lens grants us approximate access to the internal beliefs about what should be the output at a given sequence position, everything else contained in the intermediate representations (e.g., information to construct keys, queries, values, or to perform intermediate calculations that do not directly contribute to the output beliefs) remains hidden and only enters the logit lens–based part of our analysis as noise.

Acknowledgements

We thank Nina Rimsky (2023) for sharing her Llama-2 wrapper and logit lens implementation;[4] Lucia Quirke for inputs on mechanistic interpretability, on our experimental setup, and for a fruitful discussion; Saibo Geng for helping us with the Chinese dataset; Nicola Cancedda, David Garcia, Eric Horvitz, Manoel Horta Ribeiro, Maxime Peyrard, Saibo Geng, Tim Davidson, Valentin Hartmann, and Zachary Horvitz for insightful discussions and feedback; and Meta for open-sourcing Llama-2 and thereby helping democratize LLM research. Finally, we thank our anonymous peer reviewers for their productive input, which has led, among others, to Appendices B.1 and B.2. West’s lab is partly supported by grants from Swiss National Science Foundation (200021_185043, TMSGI2_211379), Swiss Data Science Center (P22_08), H2020 (952215), and by generous gifts from Meta, Google, and Microsoft.

References

Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Maxamed Axmed, Kalika Bali, and Sunayana Sitaram. 2023. Mega: Multilingual evaluation of generative ai.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, et al. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman,

[4] https://github.com/nrimsky/LM-exp/blob/main/intermediate_decoding/intermediate_decoding.ipynb

and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR.

Lera Boroditsky, Lauren A. Schmidt, and Webb Phillips. 2003. Sex, syntax, and semantics. In Dedre Gentner and Susan Goldin-Meadow, editors, *Language in Mind: Advances in the Study of Language and Thought*, pages 61–79. MIT Press, Cambridge, MA.

Nicola Cancedda. 2024. Spectral filters, dark signals, and attention sinks. *arXiv preprint arXiv:2402.09221*.

Steven Cao, Nikita Kitaev, and Dan Klein. 2020. Multilingual alignment of contextual word representations.

Yihong Chen, Kelly Marchisio, Roberta Raileanu, David Ifeoluwa Adelani, Pontus Stenetorp, Sebastian Riedel, and Mikel Artetxe. 2023. Improving language plasticity via pretraining with active forgetting.

Rochelle Choenni and Ekaterina Shutova. 2020. What does it mean to be language-agnostic? probing multilingual sentence encoders for typological properties. *arXiv preprint arXiv:2009.12862*.

Arthur Conmy, Augustine N Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. 2023. Towards automated circuit discovery for mechanistic interpretability. *arXiv preprint arXiv:2304.14997*.

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

Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2020b. Emerging cross-lingual structure in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6022–6034.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. 2022. LLM.int8(): 8-bit matrix multiplication for transformers at scale. *arXiv preprint arXiv:2208.07339*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Anna Di Natale, Max Pellert, and David Garcia. 2021. Colexification networks encode affective meaning. *Affective Science*, 2(2):99–111.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. 2021. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 1.

Julen Etxaniz, Gorka Azkune, Aitor Soroa, Oier Lopez de Lacalle, and Mikel Artetxe. 2023. Do multilingual language models think better in english?

Mor Geva, Avi Caciularu, Kevin Ro Wang, and Yoav Goldberg. 2022. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space.

Charles Goddard. 2023. Llama-polyglot-13b. `https://huggingface.co/chargoddard/llama-polyglot-13b`. Accessed: 2024-01-22.

Wes Gurnee, Neel Nanda, Matthew Pauly, Katherine Harvey, Dmitrii Troitskii, and Dimitris Bertsimas. 2023. Finding neurons in a haystack: Case studies with sparse probing. *arXiv preprint arXiv:2305.01610*.

Bofeng Huang. 2023. vigogne-2-13b-instruct. `https://huggingface.co/bofenghuang/vigogne-2-13b-instruct`. Accessed: 2024-01-22.

Haoyang Huang, Tianyi Tang, Dongdong Zhang, Wayne Xin Zhao, Ting Song, Yan Xia, and Furu Wei. 2023. Not all languages are created equal in llms: Improving multilingual capability by cross-lingual-thought prompting.

Jaavid Aktar Husain, Raj Dabre, Aswanth Kumar, Ratish Puduppully, and Anoop Kunchukuttan. 2024. Romansetu: Efficiently unlocking multilingual capabilities of large language models models via romanization.

Daekeun Kim. 2023. Llama-2-ko-dpo-13b. `https://huggingface.co/daekeun-ml/Llama-2-ko-DPO-13B`. Accessed: 2024-01-22.

Kenneth Li, Aspen K Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2022. Emergent world representations: Exploring a sequence model trained on a synthetic task. *arXiv preprint arXiv:2210.13382*.

Jindřich Libovický, Rudolf Rosa, and Alexander Fraser. 2020. On the language neutrality of pre-trained multilingual representations. *arXiv preprint arXiv:2004.05160*.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. 2022. Few-shot learning with multilingual generative language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Samuel Marks and Max Tegmark. 2023. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *arXiv preprint arXiv:2310.06824*.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. *Advances in Neural Information Processing Systems*, 35:17359–17372.

Giovanni Monea, Maxime Peyrard, Martin Josifoski, Vishrav Chaudhary, Jason Eisner, Emre Kıcıman, Hamid Palangi, Barun Patra, and Robert West. 2023. A glitch in the matrix? locating and detecting language model grounding with fakepedia. *arXiv preprint arXiv:2312.02073*.

Benjamin Muller, Yanai Elazar, Benoît Sagot, and Djamé Seddah. 2021. First align, then predict: Understanding the cross-lingual ability of multilingual bert. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2214–2231.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. 2023. Progress measures for grokking via mechanistic interpretability. *arXiv preprint arXiv:2301.05217*.

Nostalgebraist. 2020. Interpreting gpt: The logit lens. LessWrong.

Rafael E. Núñez and Eve Sweetser. 2006. With the future behind them: Convergent evidence from aymara language and gesture in the crosslinguistic comparison of spatial construals of time. *Cognitive Science*, 30(3):401–450.

OpenAI. 2023. Gpt-4 technical report.

Isabel Papadimitriou, Kezia Lopez, and Dan Jurafsky. 2022. Multilingual bert has an accent: Evaluating english influences on fluency in multilingual models.

Rait Piir. 2023. Finland's chatgpt equivalent begins to think in estonian as well. ERR News.

Björn Plüster. 2023. LeoLM: Ein Impuls für Deutschsprachige LLM-Forschung. https://laion.ai/blog-de/leo-lm/. Accessed: 2024-01-22.

Lucia Quirke, Lovis Heindrich, Wes Gurnee, and Neel Nanda. 2023. Training dynamics of contextual n-grams in language models. *arXiv preprint arXiv:2311.00863*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Nina Rimsky. 2023. Decoding intermediate activations in Llama-2-7b. LessWrong.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Tal Schuster, Ori Ram, Regina Barzilay, and Amir Globerson. 2019. Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1599–1613, Minneapolis, Minnesota. Association for Computational Linguistics.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. 2022. Language models are multilingual chain-of-thought reasoners.

Oleh Shliazhko, Alena Fenogenova, Maria Tikhonova, Vladislav Mikhailov, Anastasia Kozlova, and Tatiana Shavrina. 2022. mgpt: Few-shot learners go multilingual.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Xiangpeng Wei, Haoran Wei, Huan Lin, Tianhao Li, Pei Zhang, Xingzhang Ren, Mei Li, Yu Wan, Zhiwei Cao, Binbin Xie, Tianxiang Hu, Shangjie Li, Binyuan Hui, Bowen Yu, Dayiheng Liu, Baosong Yang, Fei Huang, and Jun Xie. 2023. Polylm: An open source polyglot large language model.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mt5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks? *Advances in neural information processing systems*, 27.

Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. 2015. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*.

Xiang Zhang, Senyu Li, Bradley Hauer, Ning Shi, and Grzegorz Kondrak. 2023. Don't trust ChatGPT when your question is not in English: A study of multilingual abilities and types of LLMs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7915–7927, Singapore. Association for Computational Linguistics.

Wenhao Zhu, Shujian Huang, Fei Yuan, Shuaijie She, Jiajun Chen, and Alexandra Birch. 2024. Question translation training for better multilingual reasoning.

Wenhao Zhu, Yunzhe Lv, Qingxiu Dong, Fei Yuan, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2023. Extrapolating large language models to non-english by aligning languages.

A Additional methodological details

A.1 Word translation

A detail that we omitted in the main paper for brevity is how we translate the English words resulting from the procedure outlined in Sec. 3.3 to French, German, and Russian. During these translations we translated both the individual words alongside their cloze sentences using DeepL.[5] For each word translation, we include the context of the cloze task to disambiguate homonyms. We then filter the translations to remove words that have the same prefix token across English and the

[5] https://www.deepl.com/translator

target language. For example, the French translation of the word "photograph", "photographier", shares the "photo" prefix token. Additionally, we parse through the translations and filter any cloze translations where the target word doesn't align with the expected word from the individual word translation, which was due to failures in the DeepL translation. These filterings result in a different number of final words across the different languages.

We provide the numbers for the aggregated translation task (Table 1), repetition task (Table 2), cloze-task (Table 3), and individual translation tasks (Table 4).

	Total	Single Token
de	287	126
fr	162	88
ru	324	45
zh	353	353

Table 1: Aggregated translation task dataset sizes.

	Total	Single Token
de	104	45
en	132	132
fr	56	31
ru	115	15
zh	139	139

Table 2: Repetition task dataset sizes.

	Total	Single Token
de	104	45
en	132	132
fr	56	31
ru	115	15
zh	139	139

Table 3: Cloze task dataset sizes.

A.2 Computing language probabilities

In order to compute language probabilities, we search Llama-2's vocabulary for all tokens that could be the first token of the correct word in the respective language. In particular, we search Llama-2's vocabulary for all prefixes of the word without and with leading space.[6] For Chinese and Russian we also consider tokenizations based on the UTF-8 encodings of their unicode characters. For a language ℓ and its corresponding target word w, we define

$$P(\text{lang} = \ell) := \sum_{t_\ell \in \text{Start}(w)} P(x_{n+1} = t_\ell), \tag{3}$$

where $\text{Start}(w)$ denotes the set of starting tokens of the word w.

For example, if the correct next Chinese word is "花" ("flower"), which can be tokenized either using the single token "花" or via its UTF-8 encoding "<0xE8>·<0x8A>·<0xB1>", we have $P(\text{lang} = \textsc{zh}) = P(x_{n+1} = \text{“花”}) + P(x_{n+1} = \text{“<0xE8>”})$ and $P(\text{lang} = \textsc{en}) = P(x_{n+1} = \text{“f”}) + P(x_{n+1} = \text{“fl”}) + P(x_{n+1} = \text{“flow”}) + P(x_{n+1} = \text{“\_f”}) + P(x_{n+1} = \text{“\_fl”}) + P(x_{n+1} = \text{“\_flo”}) + P(x_{n+1} = \text{“\_flow”}) + P(x_{n+1} = \text{“\_flower”})$ (all the token-level prefixes of "flower" and "_flower").

	de	en	fr	ru	zh
de	–	120 (120)	56 (31)	105 (15)	120 (120)
en	104 (45)	–	57 (31)	114 (15)	132 (132)
fr	93 (40)	118 (118)	–	104 (15)	118 (118)
ru	90 (41)	114 (114)	49 (26)	–	115 (115)
zh	104 (45)	132 (132)	57 (31)	115 (15)	–

Table 4: Translation statistics between languages, including total numbers and single-token translations (in brackets).

B Additional results

Here we provide the results for all languages: Chinese, English, French, German, and Russian.

Language probability. Language probability plots (with entropy heatmaps) for the aggregated translation task are in Fig. 5, for the repetition task in Fig. 7, and, for the cloze task in Fig. 9. Additionally, we provide the translation task results for individual language pairs in Fig. 11, Fig. 13, Fig. 15, Fig. 17, Fig. 19.

We observe the same pattern—noise in the early layers, English in the middle, target language in the end—across almost all languages and model sizes. The only exception is the Chinese repetition task.

Energy. Energy (Sec. 4.2) plots for the aggregated translation task are in Fig. 6, for the repetition task in Fig. 8, and, for the cloze task in Fig. 10. Additionally, we provide the translation task results for individual language pairs in Fig. 12, Fig. 14, Fig. 16, Fig. 18, Fig. 20.

Energy plots are consistent with the theory outlined in Sec. 5.

B.1 Low-resource language Estonian

We also performed our analysis with Llama-2-7B on Estonian, a low-resource language, in Fig. 21. The fact that Estonian is a low-resource language is already evident in the number of single-token words: only one out of our 99 Estonian words can be represented with a single token.

Copy task. In the copy task, Estonian behaves the most similarly to Chinese, with the Estonian probability exceeding the English probability already in the intermediate layers.

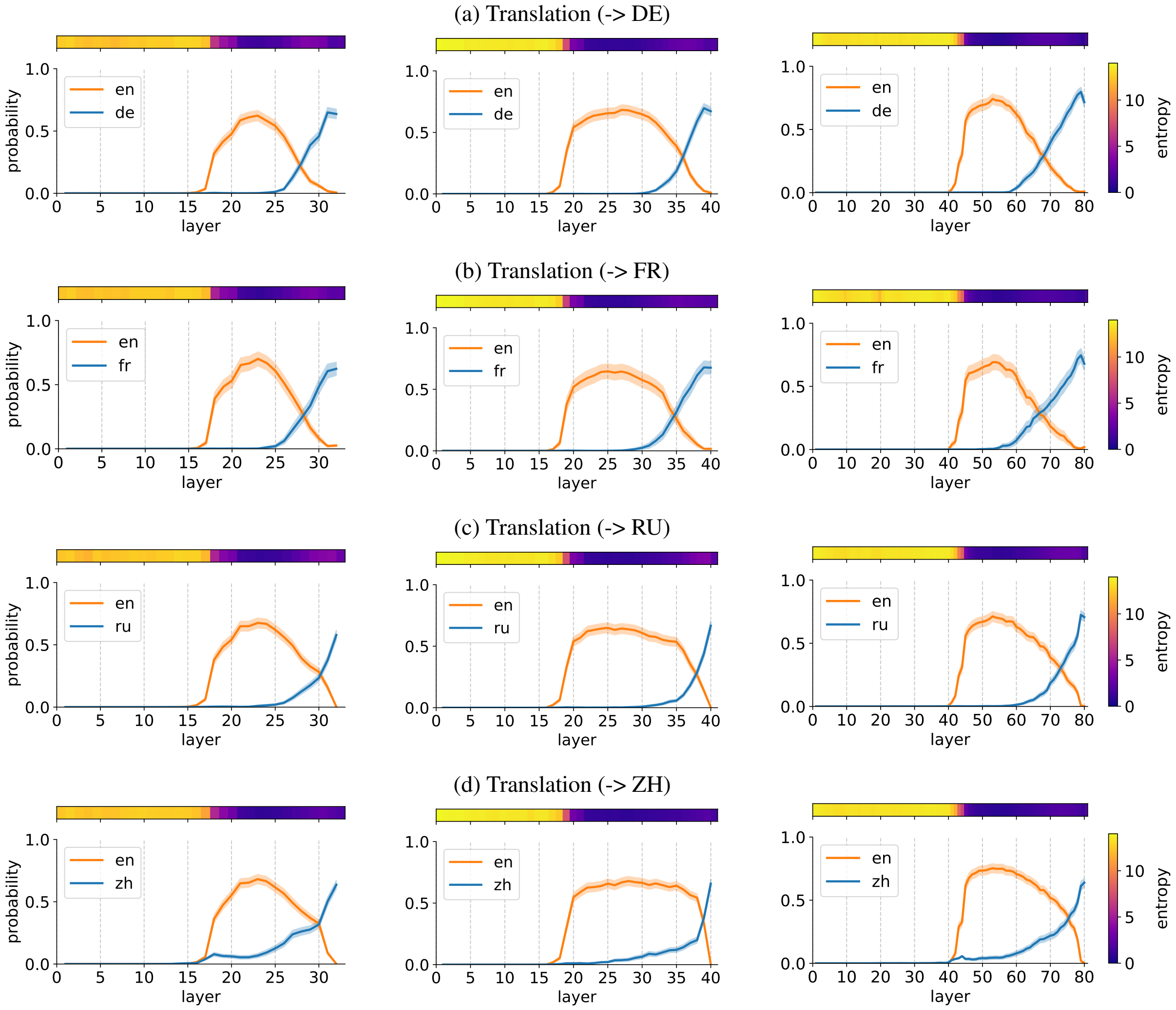
Translation task. While the success probability on the translation task after the final layer is significantly smaller than in the languages studied in the main paper, we still observe the same effect as for the other languages: the intermediate next-token distributions decoded via the logit lens concentrate their probability mass on the correct English tokens and only in the final layers transition to Estonian.

Cloze task. The Estonian cloze task seems too hard, possibly due to the extremely low resources of Estonian in the Llama-2 training data: Llama-2-7B has a 0% success probability after the last layer. Interestingly, the Estonian success probability is slightly greater than 0% in the intermediate layers, when the logit lens decodes to English. The success probability might increase if we included synonyms of the translated words or used human experts for the creation of the cloze examples instead of GPT-4.

B.2 Other models: Mistral

We also performed our analysis on Mistral-7B, a model from outside the Llama model family. The results, shown in Fig. 22, are consistent with those for Llama-2, pointing at the universality of our findings.

[6] Represented by "_".



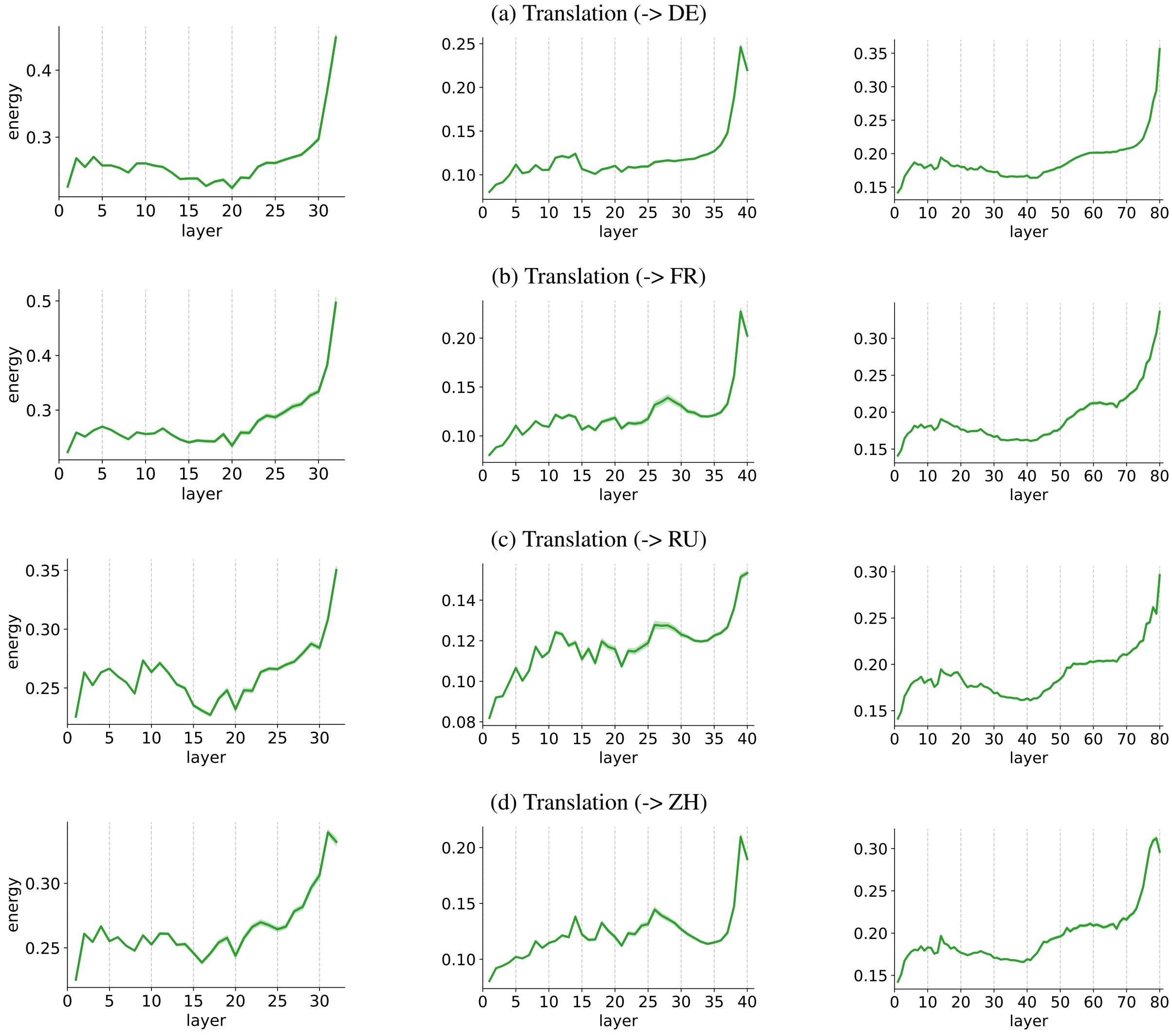
(a) Translation (-> DE)

(b) Translation (-> FR)

(c) Translation (-> RU)

(d) Translation (-> ZH)

Figure 5: Figures illustrate the translation task where Llama-2 7B, 13B, and 70B are tasked with translating a word from all non-English input languages to output language. There is one column per model size. The x-axis shows the layer number of the model, and the y-axis the total probability mass falling on the correct token across languages. The orange line illustrates the probability of the correct target word in English and the blue line shows it for the non-English output language. We do not include the probability the input language since it is zero throughout. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.



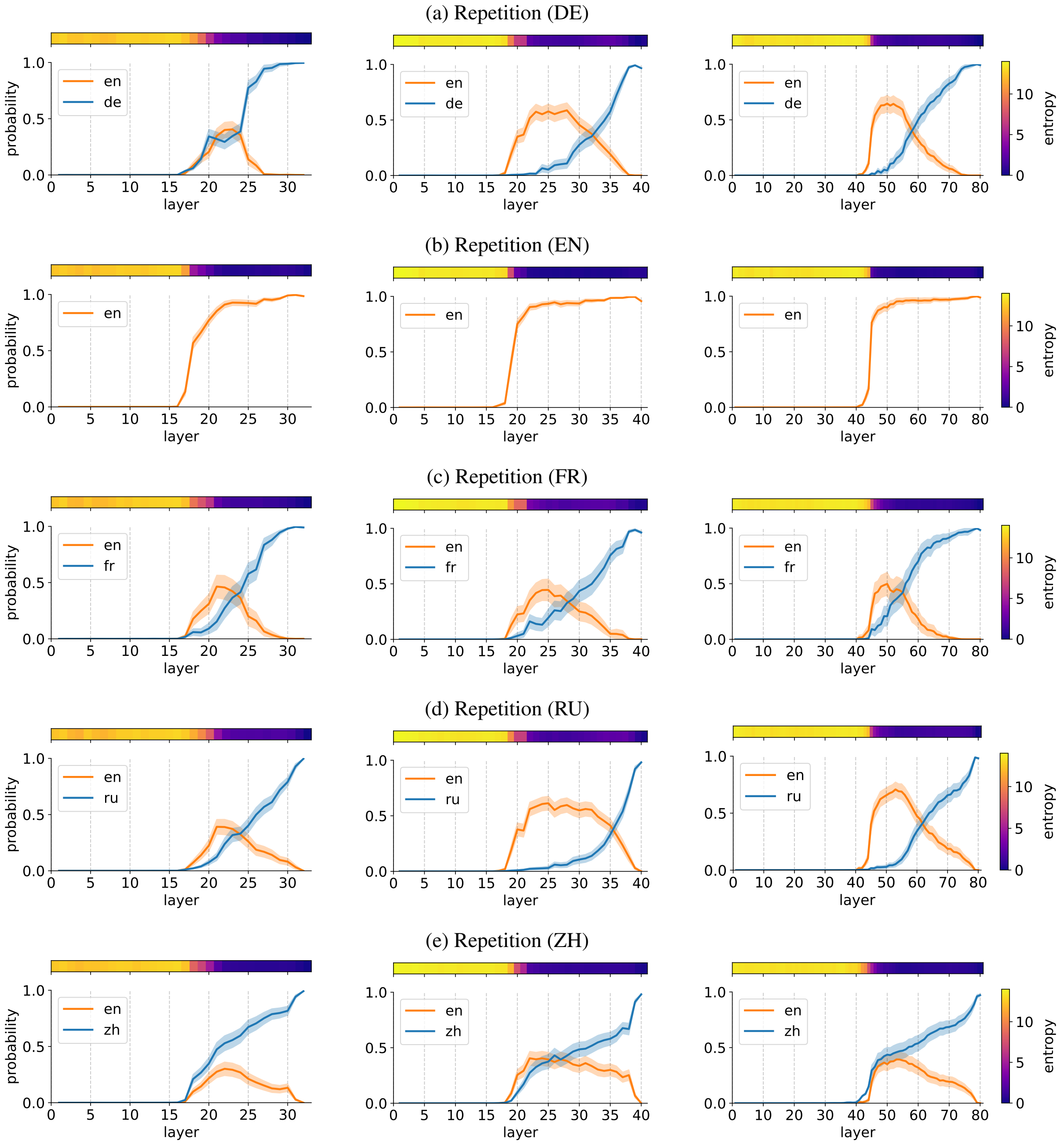
(a) Translation (-> DE)

(b) Translation (-> FR)

(c) Translation (-> RU)

(d) Translation (-> ZH)

Figure 6: Figures illustrate the translation task where Llama-2 7B, 13B, and 70B are tasked with translating a word from all non-English input languages to output language. There is one column per model size. The x-axis shows the layer number of the model, and the y-axis the energy. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.



(a) Repetition (DE)

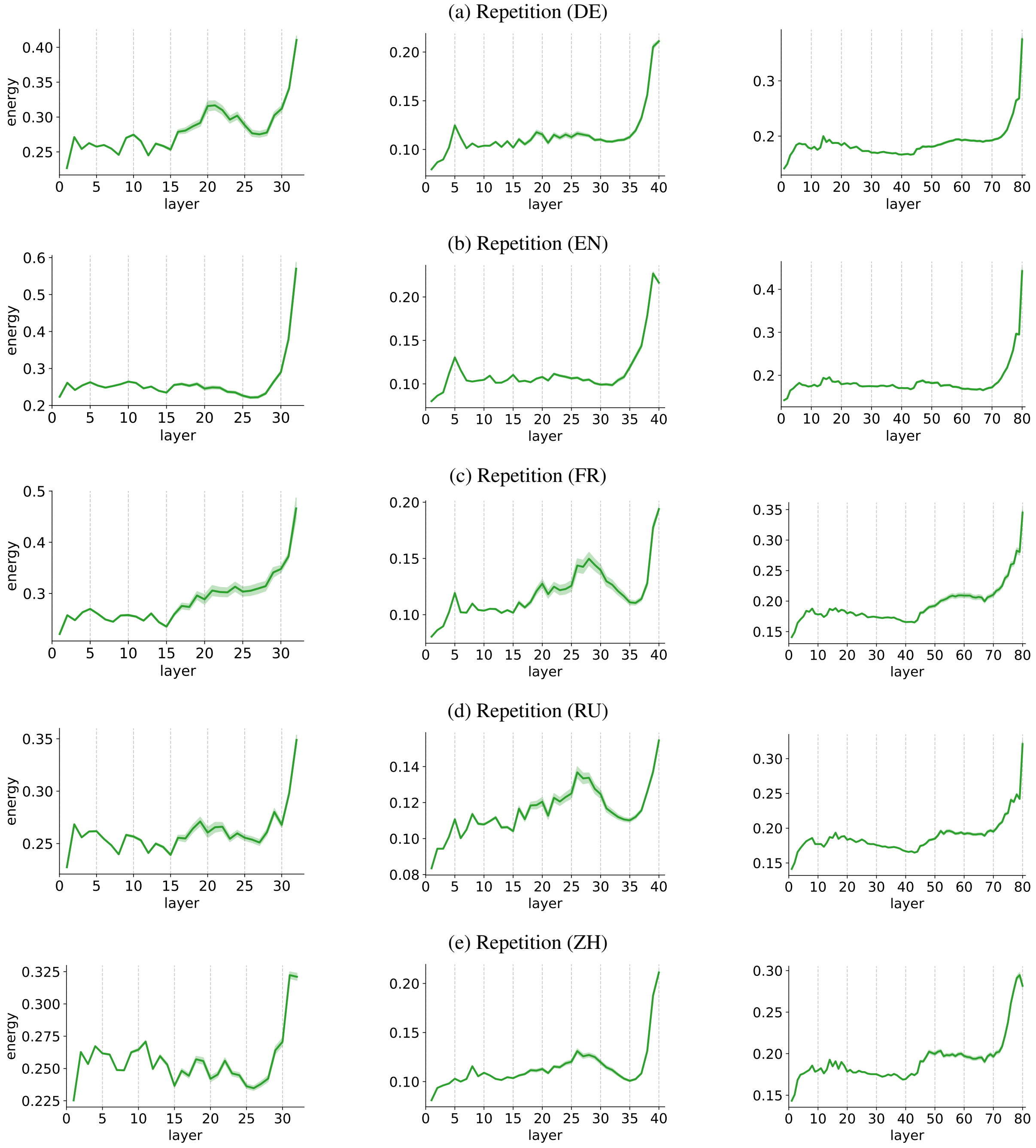
(b) Repetition (EN)

(c) Repetition (FR)

(d) Repetition (RU)

(e) Repetition (ZH)

Figure 7: Figures illustrate the repetition task where Llama-2 7B, 13B, and 70B are tasked with copying a non-English word. There is one column per model size. The x-axis shows the layer number of the model, and the y-axis the total probability mass falling on the correct token across languages. The orange line illustrates the probability of the correct target word in English and the blue line shows it for the non-English output language. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.


(a) Repetition (DE)

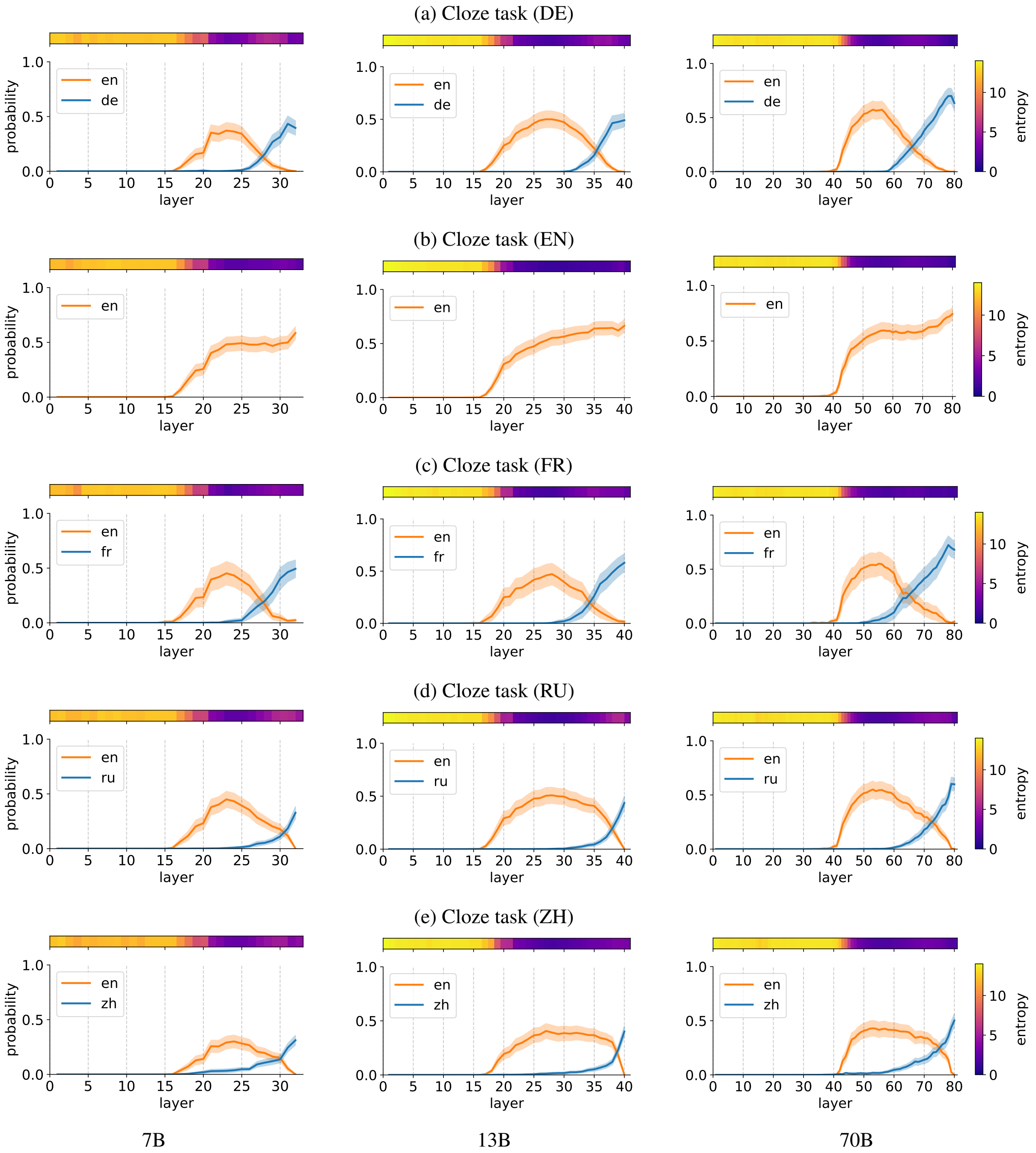
(b) Repetition (EN)

(c) Repetition (FR)

(d) Repetition (RU)

(e) Repetition (ZH)

Figure 8: Figures illustrate the energy plots for the repetition task where Llama-2 7B, 13B, and 70B are tasked with copying a non-English word. There is one column per model size. The x-axis shows the layer number of the model, and the y-axis the energy. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.



(a) Cloze task (DE)

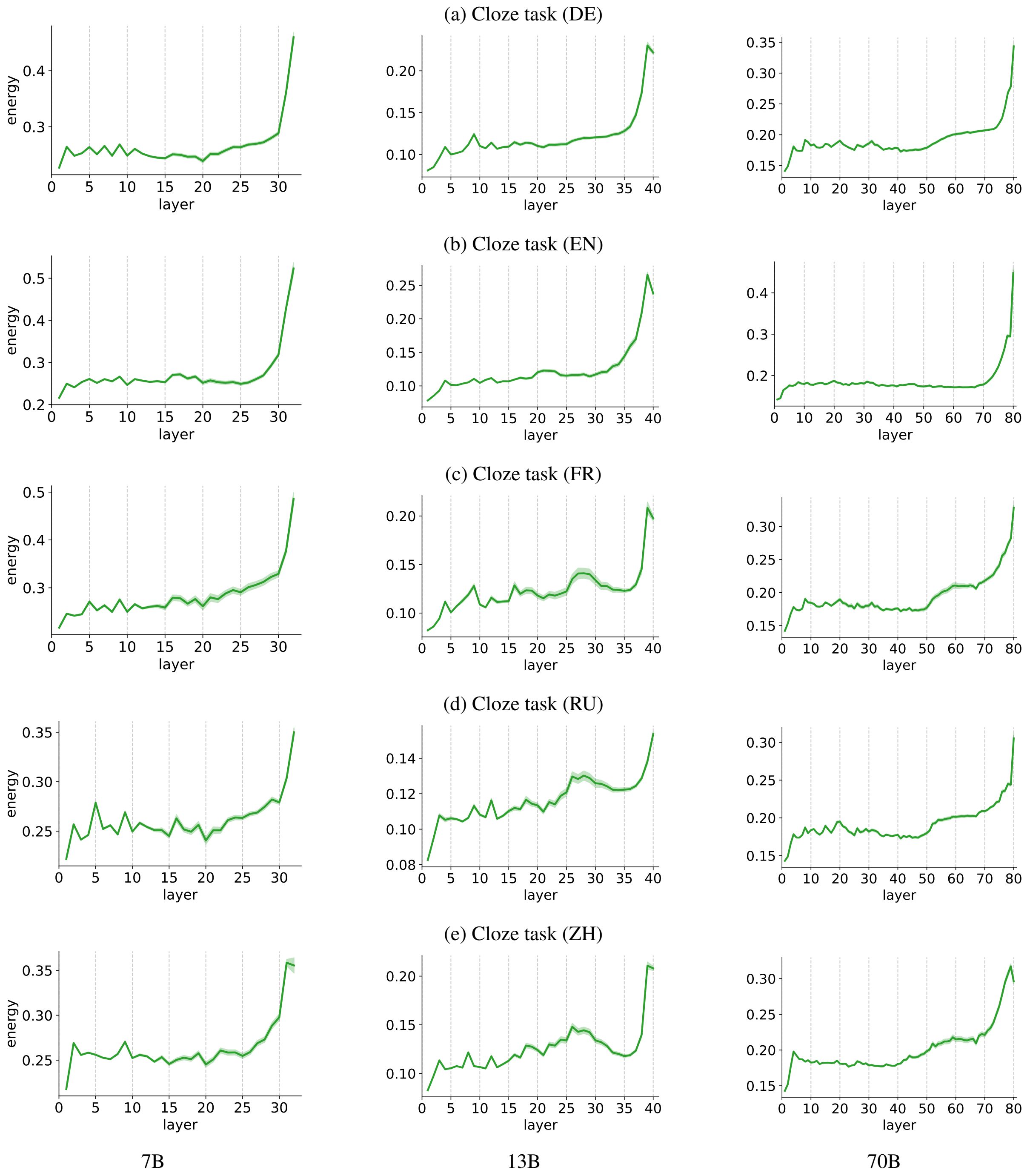
(b) Cloze task (EN)

(c) Cloze task (FR)

(d) Cloze task (RU)

(e) Cloze task (ZH)

Figure 9: Figures show the same plots only for the cloze task where the correct token is defined in a fill-in-the-blank setting. In the plots, we illustrate the results for German. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.



(a) Cloze task (DE)

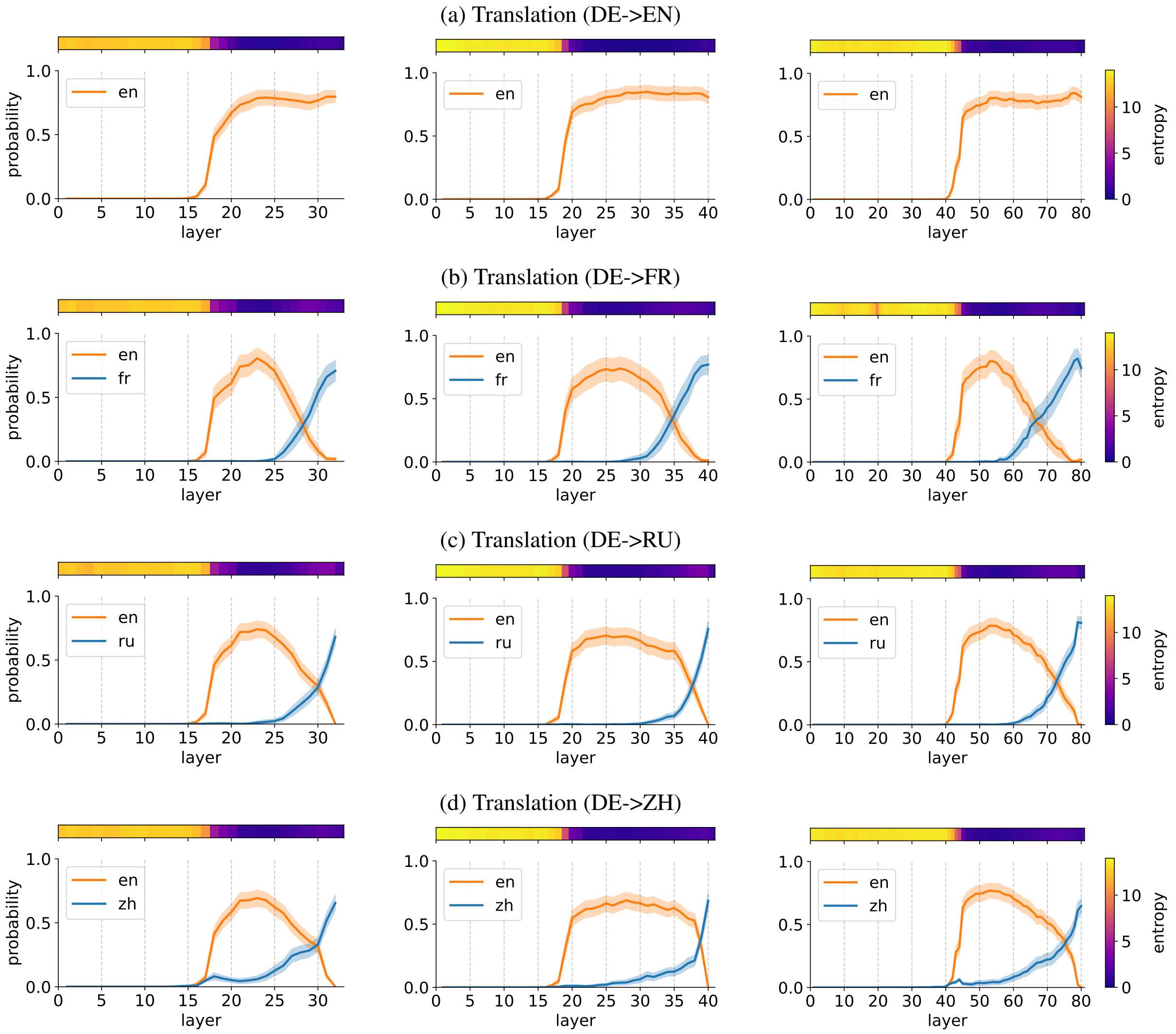
(b) Cloze task (EN)

(c) Cloze task (FR)

(d) Cloze task (RU)

(e) Cloze task (ZH)

Figure 10: Figures show the same plots only for the cloze task where the correct token is defined in a fill-in-the-blank setting. In the plots, we illustrate the results for German. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.

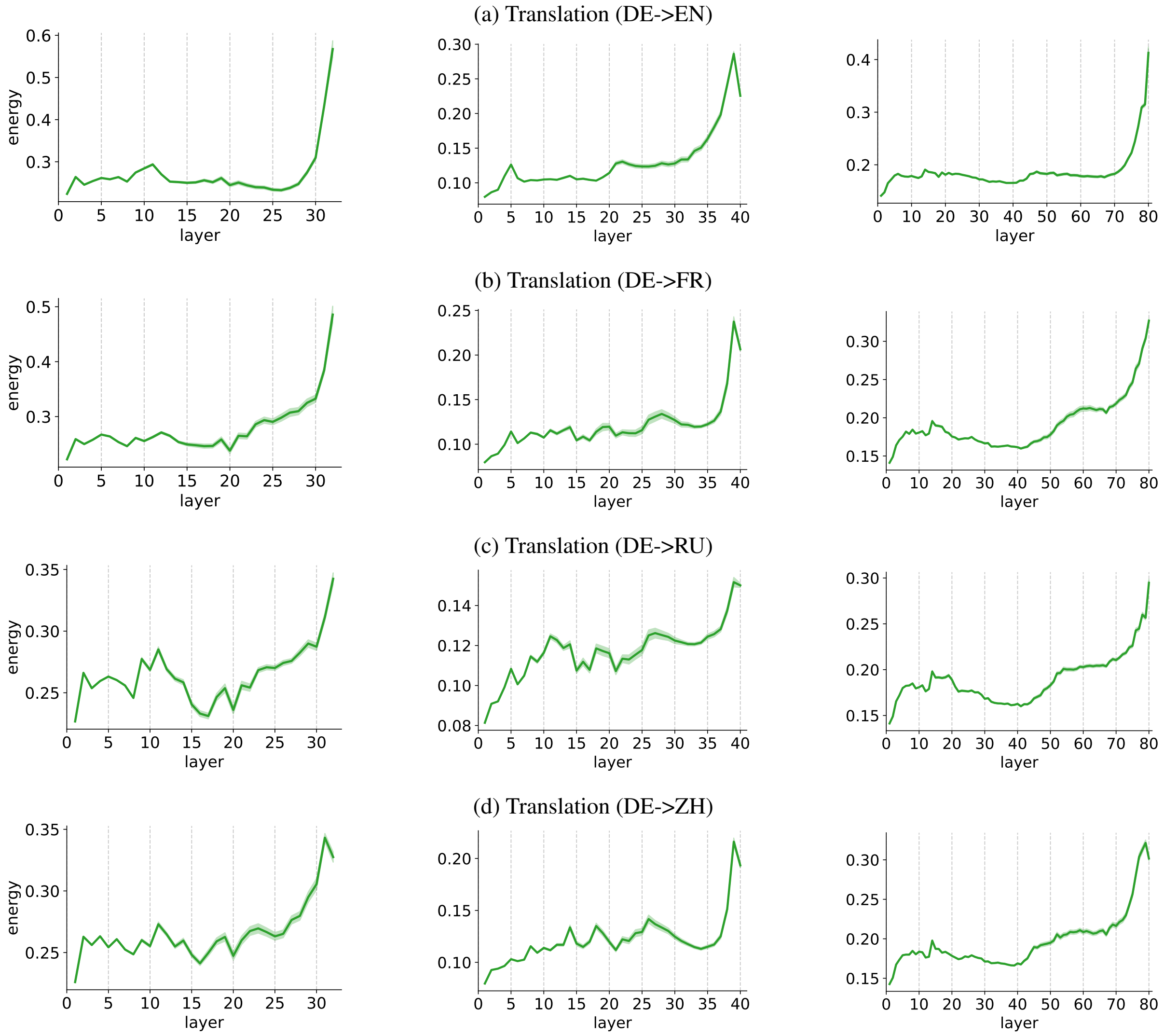

(a) Translation (DE->EN)

(b) Translation (DE->FR)

(c) Translation (DE->RU)

(d) Translation (DE->ZH)

Figure 11: Figures illustrate the translation task where Llama-2 7B, 13B, and 70B are tasked with translating a word from non-English input language to output language. There is one column per model size. The x-axis shows the layer number of the model, and the y-axis the total probability mass falling on the correct token across languages. The orange line illustrates the probability of the correct target word in English and the blue line shows it for the non-English output language. We do not include the probability the input language since it is zero throughout. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.



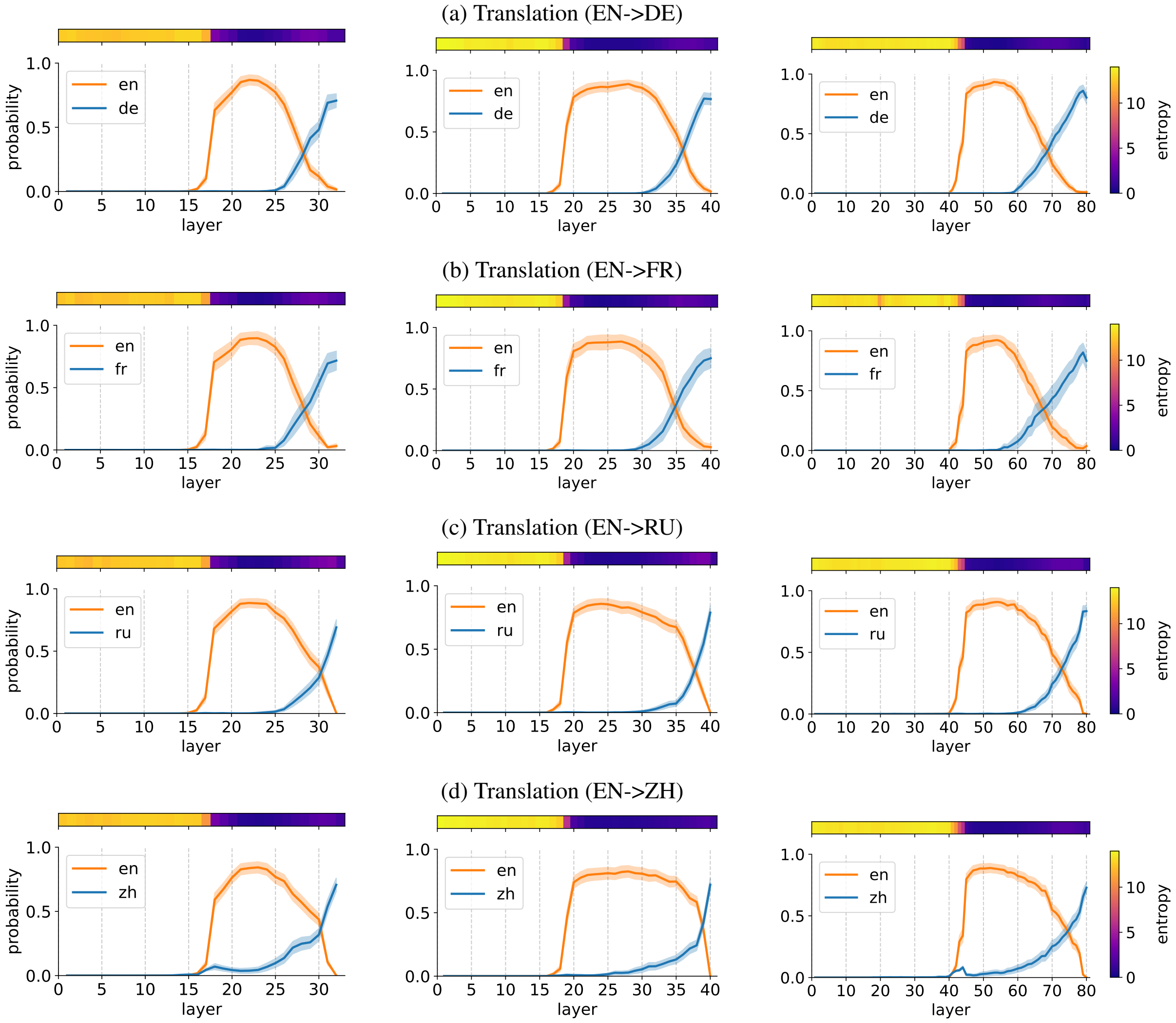
(a) Translation (DE->EN)

(b) Translation (DE->FR)

(c) Translation (DE->RU)

(d) Translation (DE->ZH)

Figure 12: Figures illustrate the translation task where Llama-2 7B, 13B, and 70B are tasked with translating a word from non-English input language to output language. There is one column per model size. The x-axis shows the layer number of the model, and the y-axis the energy. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.



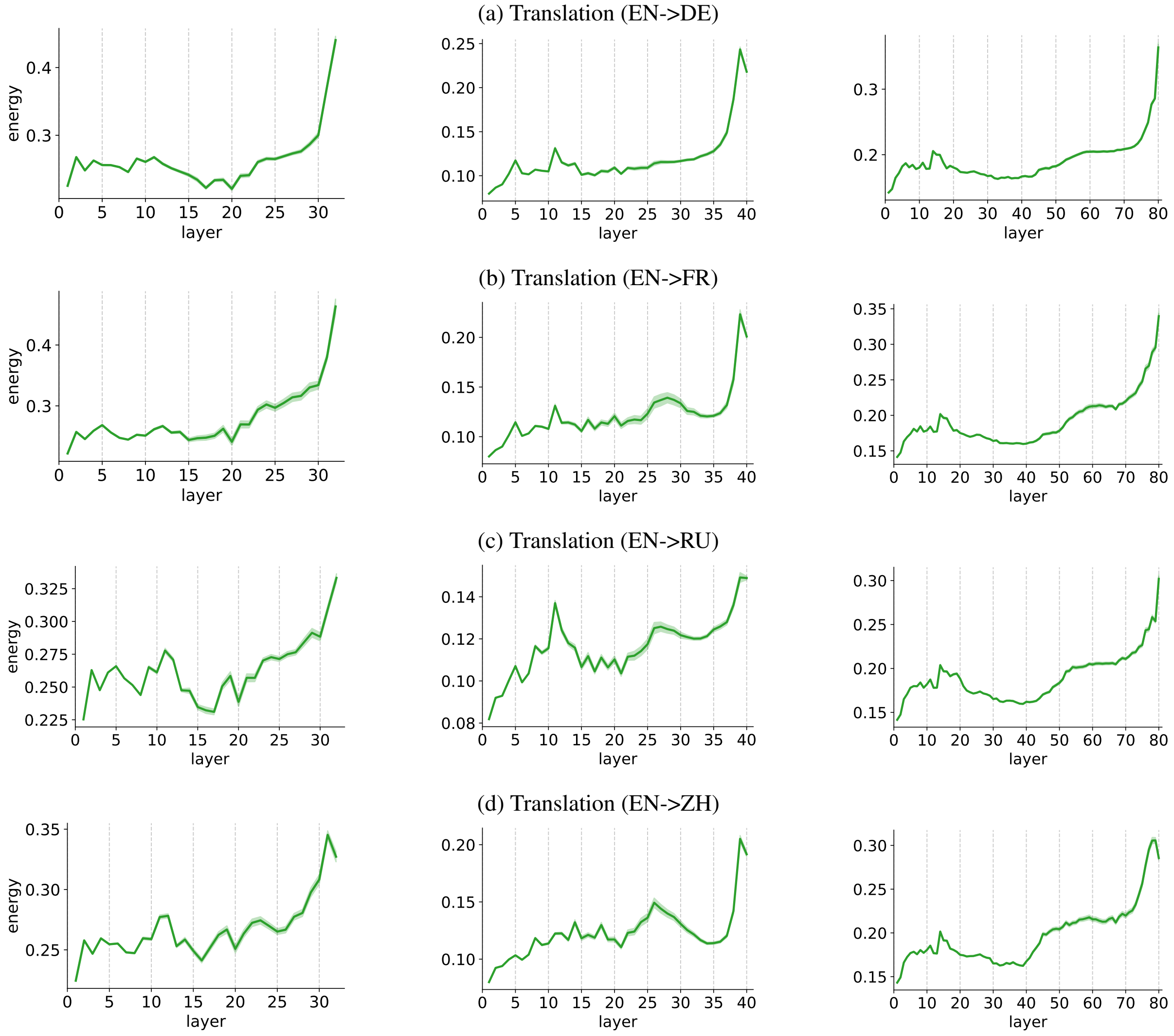
(a) Translation (EN->DE)

(b) Translation (EN->FR)

(c) Translation (EN->RU)

(d) Translation (EN->ZH)

Figure 13: Figures illustrate the translation task where Llama-2 7B, 13B, and 70B are tasked with translating a word from English input language to output language. There is one column per model size. The x-axis shows the layer number of the model, and the y-axis the total probability mass falling on the correct token across languages. The orange line illustrates the probability of the correct target word in English and the blue line shows it for the non-English output language. We do not include the probability the input language since it is zero throughout. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.



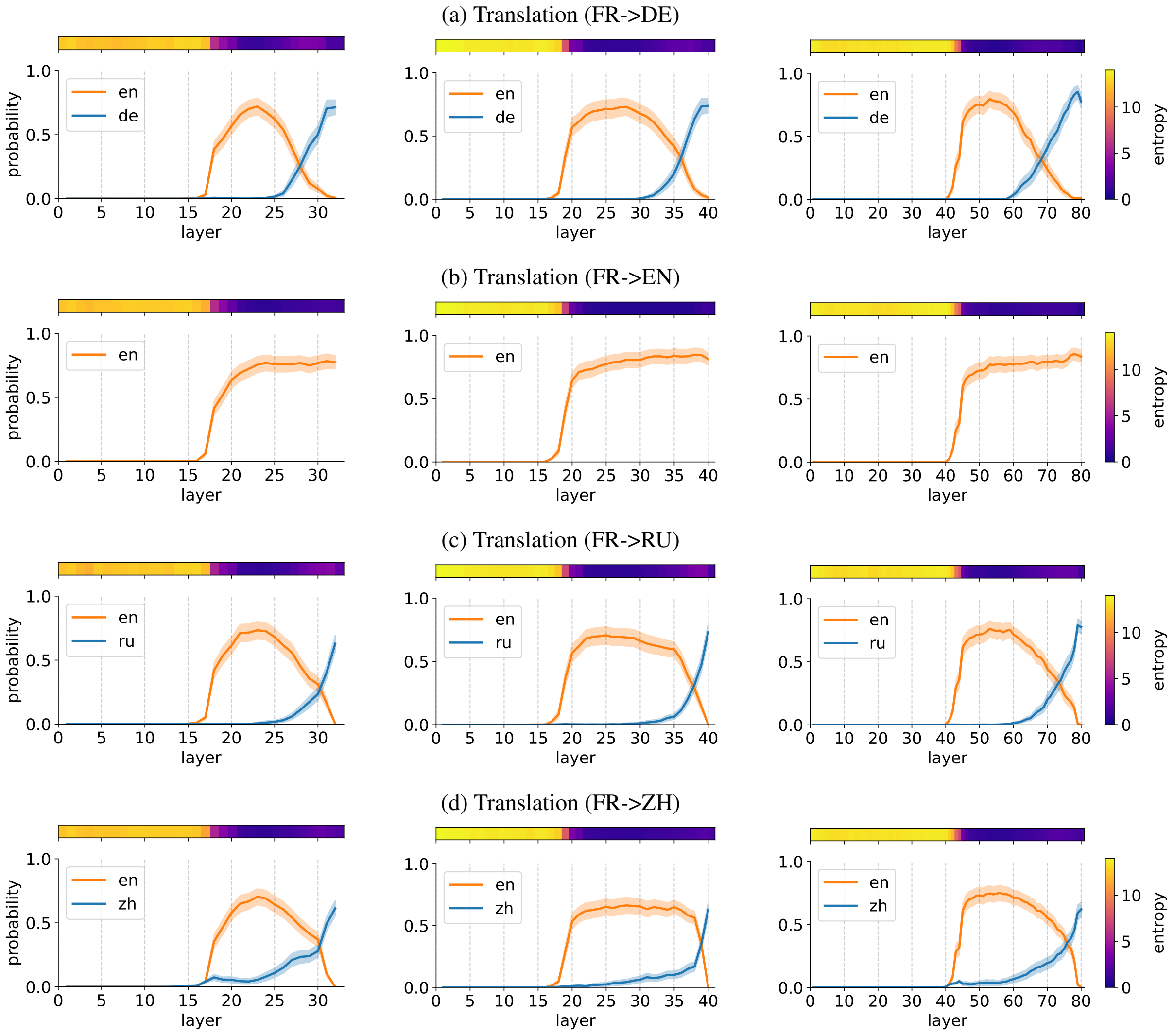
(a) Translation (EN->DE)

(b) Translation (EN->FR)

(c) Translation (EN->RU)

(d) Translation (EN->ZH)

Figure 14: Figures illustrate the translation task where Llama-2 7B, 13B, and 70B are tasked with translating a word from English input language to output language. There is one column per model size. The x-axis shows the layer number of the model, and the y-axis the energy. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.

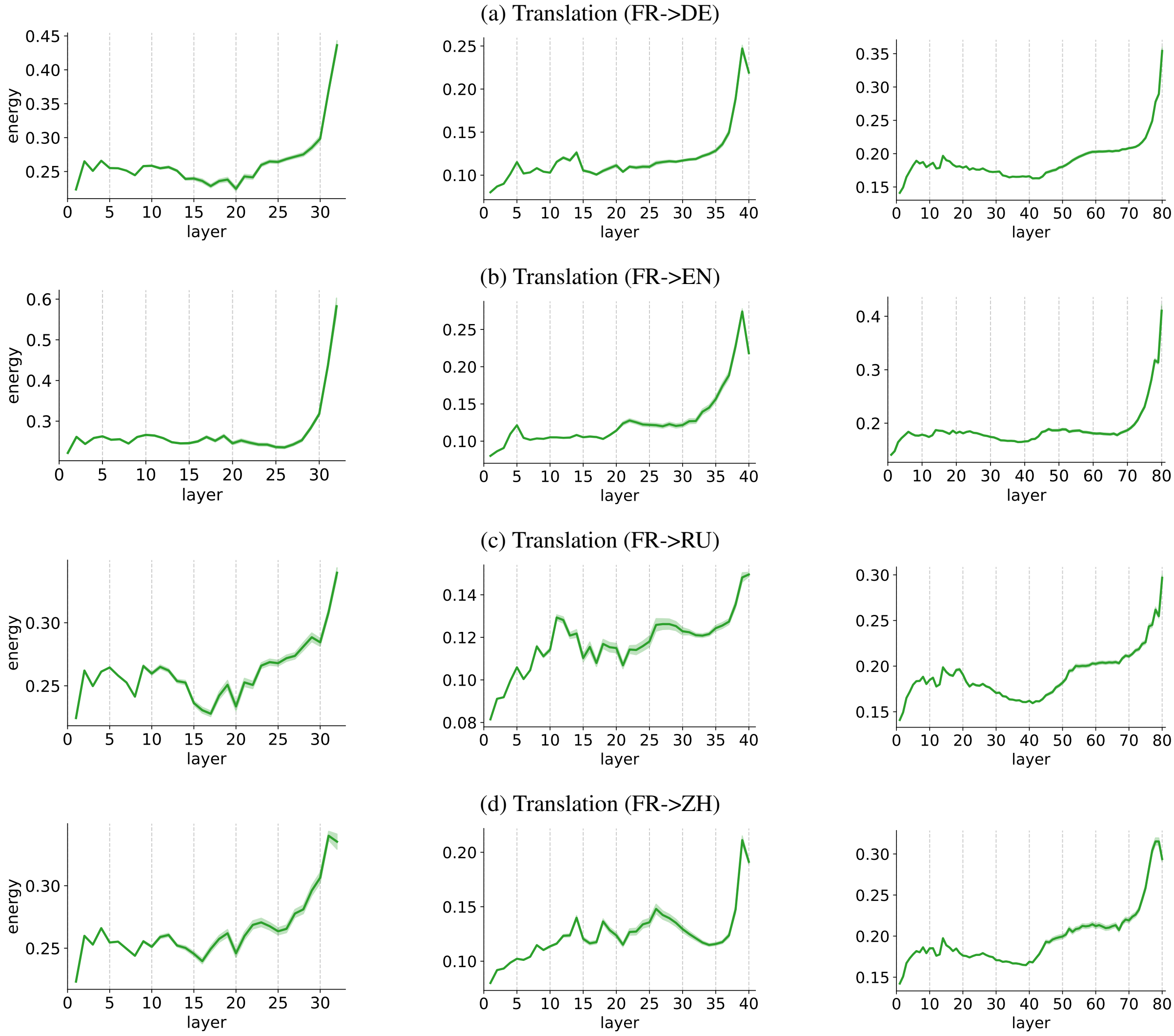

(a) Translation (FR->DE)

(b) Translation (FR->EN)

(c) Translation (FR->RU)

(d) Translation (FR->ZH)

Figure 15: Figures illustrate the translation task where Llama-2 7B, 13B, and 70B are tasked with translating a word from non-English input language to output language. There is one column per model size. The x-axis shows the layer number of the model, and the y-axis the total probability mass falling on the correct token across languages. The orange line illustrates the probability of the correct target word in English and the blue line shows it for the non-English output language. We do not include the probability the input language since it is zero throughout. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.



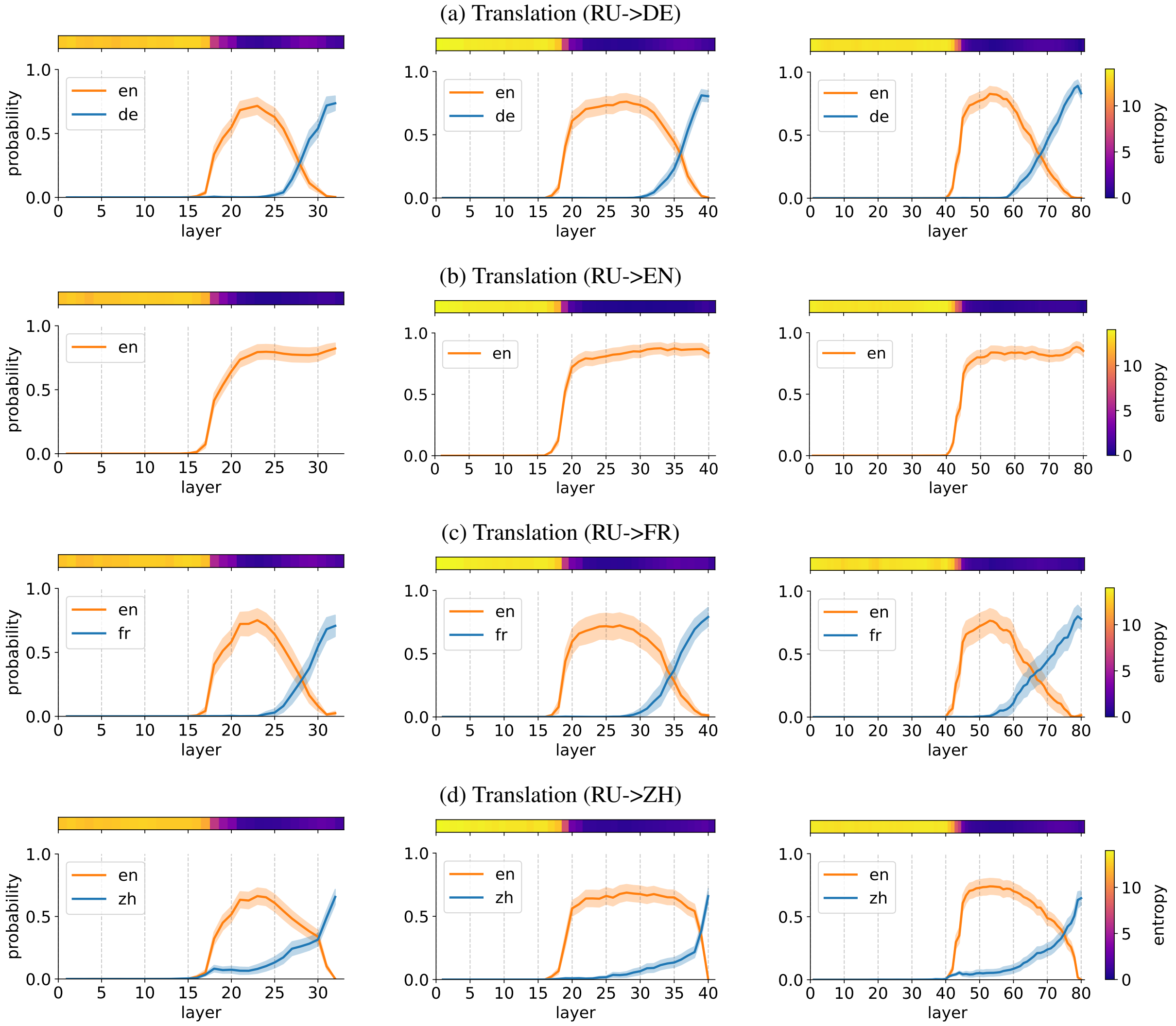
(a) Translation (FR->DE)

(b) Translation (FR->EN)

(c) Translation (FR->RU)

(d) Translation (FR->ZH)

Figure 16: Figures illustrate the translation task where Llama-2 7B, 13B, and 70B are tasked with translating a word from non-English input language to output language. There is one column per model size. The x-axis shows the layer number of the model, and the y-axis the energy.Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.



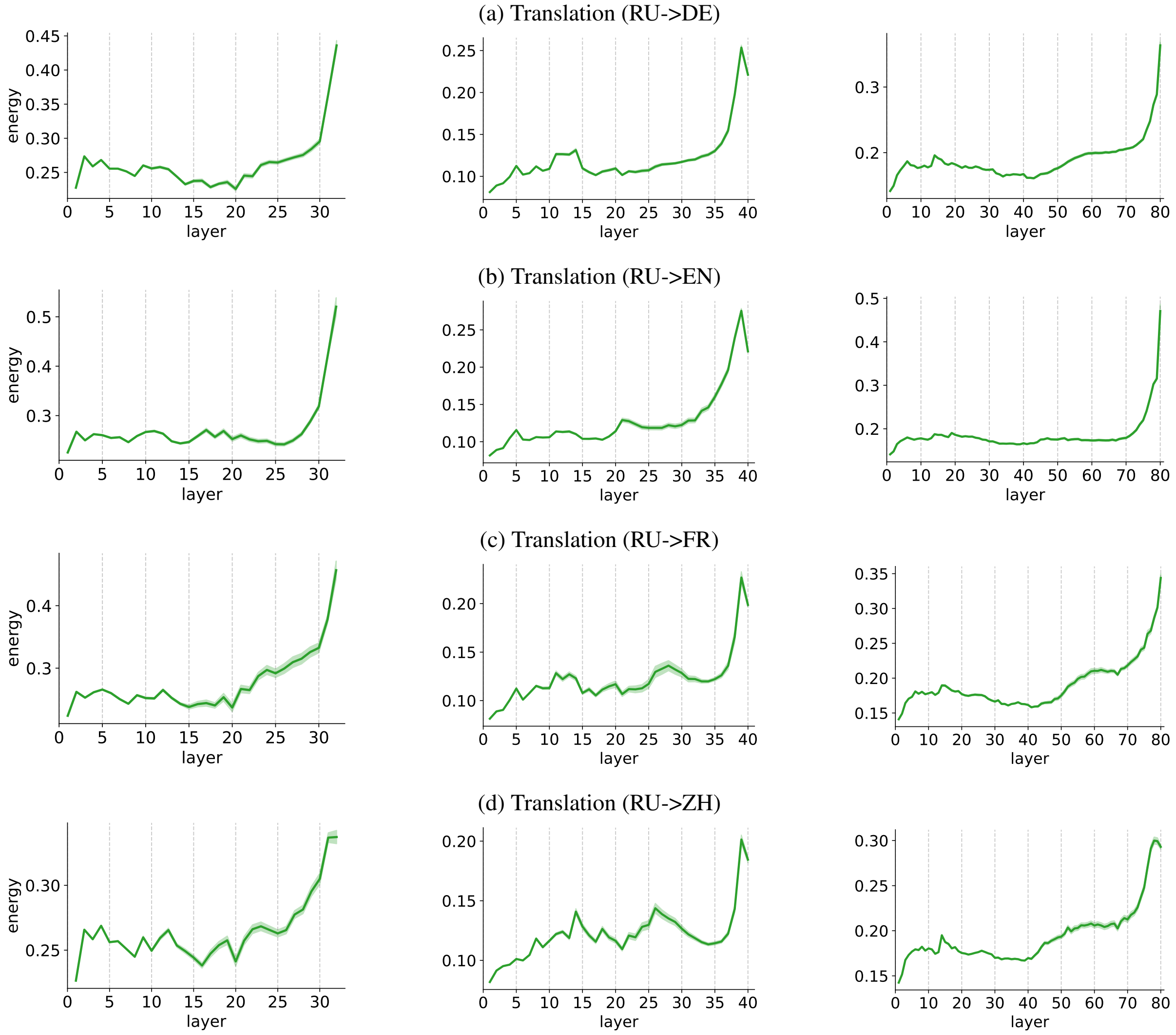
(a) Translation (RU->DE)

(b) Translation (RU->EN)

(c) Translation (RU->FR)

(d) Translation (RU->ZH)

Figure 17: Figures illustrate the translation task where Llama-2 7B, 13B, and 70B are tasked with translating a word from non-English input language to output language. There is one column per model size. The x-axis shows the layer number of the model, and the y-axis the total probability mass falling on the correct token across languages. The orange line illustrates the probability of the correct target word in English and the blue line shows it for the non-English output language. We do not include the probability the input language since it is zero throughout. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.



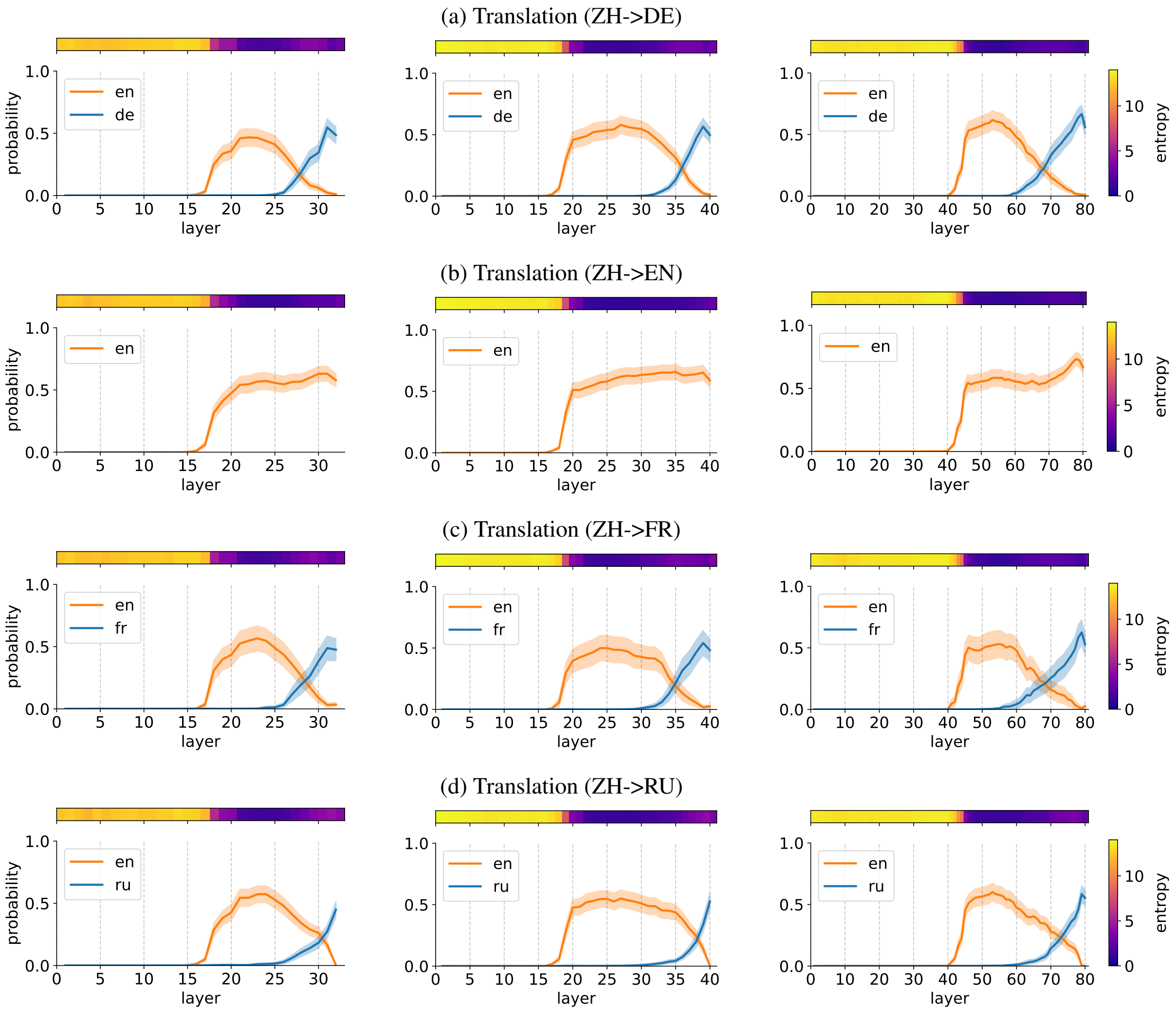
(a) Translation (RU->DE)

(b) Translation (RU->EN)

(c) Translation (RU->FR)

(d) Translation (RU->ZH)

Figure 18: Figures illustrate the translation task where Llama-2 7B, 13B, and 70B are tasked with translating a word from non-English input language to output language. There is one column per model size. The x-axis shows the layer number of the model, and the y-axis the energy. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.

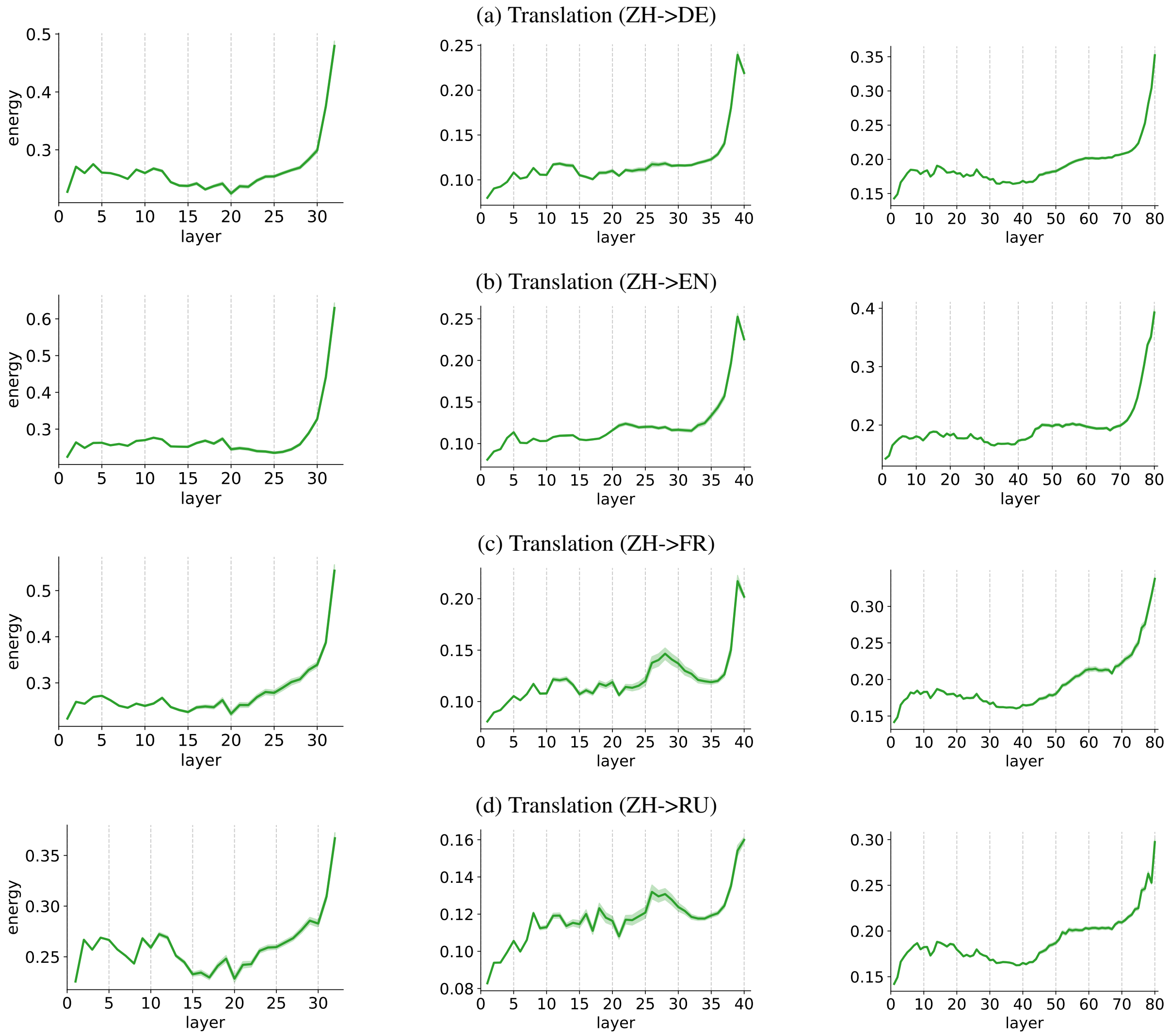

(a) Translation (ZH->DE)

(b) Translation (ZH->EN)

(c) Translation (ZH->FR)

(d) Translation (ZH->RU)

Figure 19: Figures illustrate the translation task where Llama-2 7B, 13B, and 70B are tasked with translating a word from non-English input language to output language. There is one column per model size. The x-axis shows the layer number of the model, and the y-axis the total probability mass falling on the correct token across languages. The orange line illustrates the probability of the correct target word in English and the blue line shows it for the non-English output language. We do not include the probability the input language since it is zero throughout. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.


(a) Translation (ZH->DE)

(b) Translation (ZH->EN)

(c) Translation (ZH->FR)

(d) Translation (ZH->RU)

Figure 20: Figures illustrate the translation task where Llama-2 7B, 13B, and 70B are tasked with translating a word from non-English input language to output language. There is one column per model size. The x-axis shows the layer number of the model, and the y-axis the energy. Means and 95% Gaussian confidence intervals have been computed over the input examples, numbers in Appendix A.

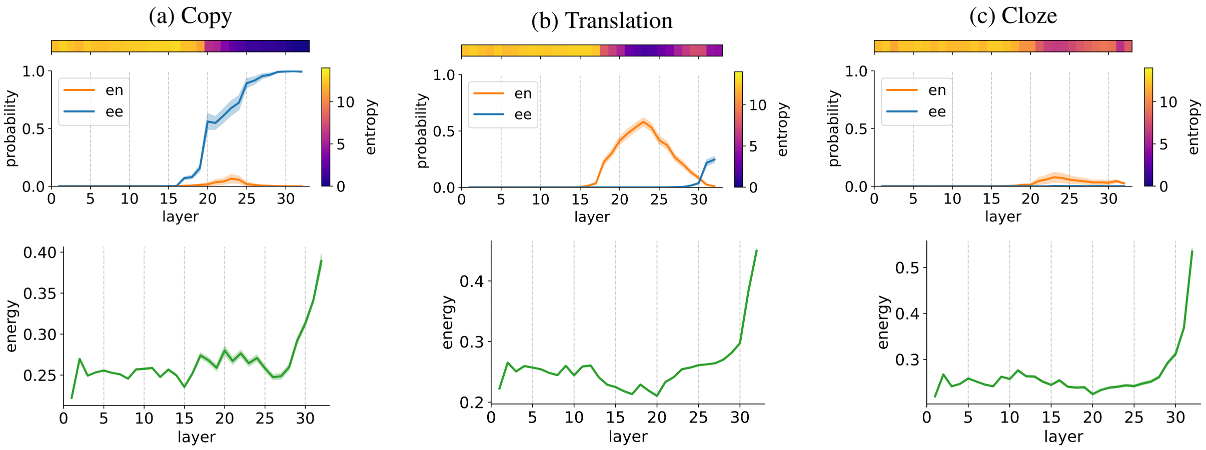


Figure 21: Figures illustrate our analysis of the copy-, translation-, and cloze task for the **Estonian** language on Llama-2-7B. In the first row, the x-axis shows the layer number of the model, and the y-axis the language probability. In the first row, the x-axis shows the layer number of the model, and the y-axis the token energy. Means and 95% Gaussian confidence intervals have been computed over the input examples.

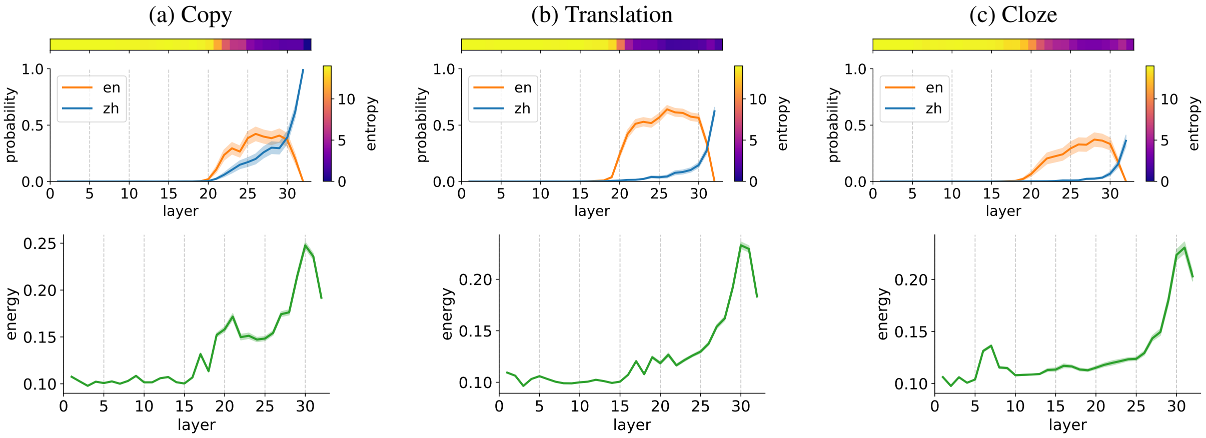


Figure 22: Figures illustrate our analysis of the copy-, translation-, and cloze task for Chinese on **Mistral-7B**. In the first row, the x-axis shows the layer number of the model, and the y-axis the language probability. In the first row, the x-axis shows the layer number of the model, and the y-axis the token energy. Means and 95% Gaussian confidence intervals have been computed over the input examples.