How Do Llamas Process Multilingual Text? A Latent Exploration through Patchscopes

Anonymous ACL submission

Abstract

A central question in multilingual language modeling is whether large language models (LLMs) develop a universal concept representation, disentangled from specific languages. In this paper, we address this question by analyzing the Llama-2's forward pass during a word translation task. We strategically extract latents from a source translation prompt and insert them into the forward pass on a target translation prompt. By doing so, we find that the output language is encoded in the latent at an earlier layer than the concept to be translated. Utilizing this insight we show that both, target concept in source language and source concept in target language, are achievable via patching alone. Furthermore, we show that patching in the mean of multiple source language latents does not impair our ability to decode source concept in target language, indicating that concept representations are language-agnostic.

1 Introduction

The emergence of the field of mechanistic interpretability has led to the conception of powerful tools (Carter et al., 2019; Nostalgebraist, 2020; Schubert et al., 2020; Belrose et al., 2023; Cunningham et al., 2023; Kramár et al., 2024; Marks et al., 2024; O'Neill and Bui, 2024; Tufanov et al., 2024) for the investigation of the inner workings of deep neural networks such as large language models (LLMs) (Vaswani et al., 2017; Radford et al., 2019; Touvron et al., 2023) with the ultimate goal of reverse engineering the algorithms encoded in their weights. As a result, researchers today are often able to open up a "black box" neural network, and with near surgical precision pinpoint where a certain input-output behaviour comes from (Wang et al., 2022; Conmy et al., 2023; Nanda et al., 2023; Zhong et al., 2024; Furuta et al., 2024).

One such recent approach has been to use patchscopes (Ghandeharioun et al., 2024). The key idea of a patchscope is to repurpose a LLM[1] to unpack information contained in its own intermediate results. This can be achieved by patching a latent from one forward pass into another one while observing the output (cf. Fig. 1).

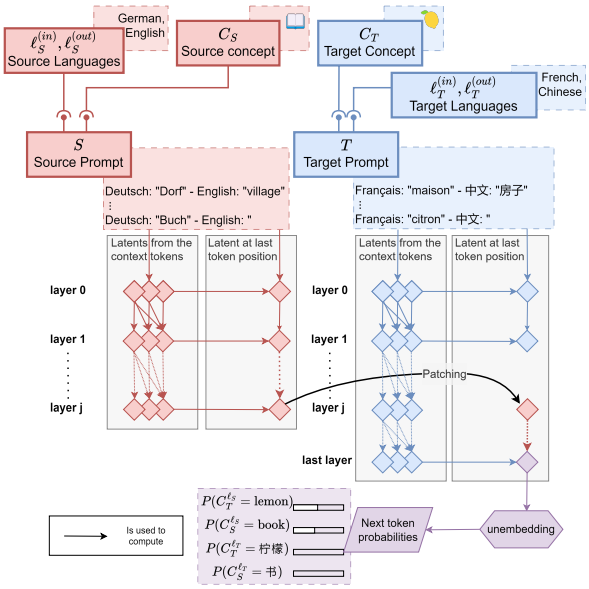

Figure 1: For two given concepts, e.g., BOOK and LEMON, we construct a source prompt for translating from German to English, and a target prompt for translating from French to Chinese. Then we extract the latent of the last token after some layer j from the source prompt and insert it at the corresponding position in the forward pass of the target prompt. The resulting next token probabilities will concentrate on the *target concept in target language* (LEMON$^{\text{ZH}}$), i.e. 柠檬) when patching at layers 0-11, on the *target concept in source language* ("lemon") for layers 12-16, and on the *source concept in source language* ("book") for layers 17-31.

Summary of contributions. In this work, we leverage patchscopes to understand how Llama-2 (Touvron et al., 2023) processes multilingual text. In particular, we investigate whether it uses a language-agnostic concept space as theorized

[1]Note that we use LLM and transformer (Vaswani et al., 2017) interchangeably.

by Wendler et al. (2024). In such a space, concepts would be represented independently of the language used to express them. In order to do so, we design multiple patchscope experiments leveraging pairs of translation prompts with differing expected predicted concept and language.

1. We start by patching at the last token (as in Fig. 1). As a result, we find that first the model resolves the output language and, in later layers, the concept to be translated.
2. Next, we come up with two hypotheses about how Llama-2's forward pass might have solved the task. **H1** in which language and concept are represented in a disentangled way and **H2** in which they are always entangled.
3. Finally, we perform targeted experiments to gather more evidence for either **H1** or **H2** and find **H1** is better supported by our results.

Therefore, our results agree with the theory outlined by Wendler et al. (2024). In contrast to their analysis which is purely observational with the logit lens, ours is based on interventions by virtue of on the patchscope. Additionally, by using patchscopes we circumvent the potential pitfalls of cosine similarity (Steck et al., 2024) inherent in the logit lens analysis and instead utilize Llama-2's full power to draw coclusions about the computations performed and representations used.

2 Llama-2's forward pass

Because we need full model access for our analysis, we focus on Llama-2 (Touvron et al., 2023). Llama-2 is an autoregressive, decoder-only, residual-based transformer (Vaswani et al., 2017) that was trained to map a sequence of input tokens $x_1, \ldots, x_n \in V$, where n is the sequence length, to a sequence of latents in $\mathbb{R}^d$ that is refined layer by layer such that the final latents are well-suited for predicting the next tokens $x_2, \ldots, x_{n+1} \in V$.

On a technical level, this is achieved using transformer blocks, consisting of a causally 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), that are used to update the latent at position i in layer j:

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

where $h_1^{(j-1)}, \ldots, h_i^{(j-1)}$ and $h_i^{(j)} \in \mathbb{R}^d$.

The initial latents $h_1^{(0)}, \ldots, h_n^{(0)} \in \mathbb{R}^d$ are learnt token embeddings. Finally, for a m-layer transformer, the next-token probabilities are obtained via a learnt linear layer followed by a softmax operation mapping $h_i^{(m)}$ to $P(x_{i+1}|h_i^{(m)})$.

3 Exploratory analysis with patchscopes

Notation. Let C denote an abstract concept and C^ℓ its language-specific version. Further, let $w(C^\ell)$ denote the set of words[2] expressing the abstract concept C in language ℓ. For example using capitalization to denote the abstract concepts, let $C = \text{CAT}$. Then for $\ell = \text{EN}$ we have $w(C^{\text{EN}}) = \{\text{“cat”}\}$ and similarly $w(C^{\text{DE}}) = \{\text{“Katze”}, \text{“Kater”}\}$.

Problem statement. We aim to understand whether language- and concept information can vary independently during Llama-2's forward pass when processing a multilingual prompt. For example, a representation of C^ℓ of the form $z_{C^\ell} = z_C + z_\ell$, in which $z_C \in U$, $z_\ell \in U^\perp$ and $U \oplus U^\perp = \mathbb{R}^d$ is a decomposition of $\mathbb{R}^d$ into a subspace U and its orthogonal complement $U^\perp$, would allow for language- and concept information to vary independently: language can be varied by changing $z_\ell \in U^\perp$ and concept by changing $z_C \in U$.[3] Conversely, if language- and concept information were entangled a decomposition like this should not exist: varying the language would mean varying the concept and vice-versa.

3.1 Experimental design

We start our analysis with an exploratory experiment in which we utilize simple few-shot translation prompts from Wendler et al. (2024) to create paired source- and target prompt datasets with different input language $\ell_S^{(in)} \neq \ell_T^{(in)}$, concept $C_S \neq C_T$, output language $\ell_S^{(out)} \neq \ell_T^{(out)}$.

If not mentioned otherwise ℓ_S and ℓ_T refer to the output language of the source and target prompt.

Prompt design. An example translation prompt:

> English: “lake” - Français: “lac"
> English: “south" - Français: “sud"
> English: “mother" - Français: “mère"
> English: “seat" - Français: “siège"
> English: “cloud" - Français: “

Here the task is to translate $w(\text{CLOUD}^{\text{EN}}) = \{\text{“cloud"}\}$ into $w(\text{CLOUD}^{\text{FR}}) = \{\text{“nuage"}\}$.

[2] We talk about words for the sake of simplicity. However, on a technical level w refers to the set of starting tokens of these words.

[3] Note that we are not trying to claim that this indeed is the form of the representations, instead it is a cartoon to help us think about the phenomenon of interest.

Importantly, whether the model correctly answers the prompt is determined by its next token prediction. For example above, the next token predicted should be “nu", the first token of “nuage". Thus, we can track $P(C^\ell)$, i.e., the probability of the concept C occurring in language ℓ, by simply summing up the probabilities of all starting tokens of $w(C^\ell)$ in the next-token distribution.

We improve upon the construction of Wendler et al. (2024) by considering all the possible expressions of C in ℓ using BabelNet (Navigli et al., 2021), instead of GPT-4 translations, when computing $P(C^\ell)$. This allows us to capture many possible translations, instead of one. For example, “commerce”, “magasin” and “boutique” are all valid words for SHOP^{FR}.

Patchscope setup. We would like to infer at which layers the output language and the concept enter the latent $h^{(j)}_{n_T}(T)$ respectively and whether they can vary independently. In order to investigate this question, we perform the experiment depicted in Fig. 1. For each transformer block f_j we create two parallel forward passes, one processing the source prompt $S = (s_1, \ldots, s_{n_S})$ and one processing the target prompt $T = (t_1, \ldots, t_{n_T})$. While doing so, we extract the latent of the last token of the source prompt at layer j, $h^{(j)}_{n_S}(S)$, and insert it at the same layer at position n_T in the forward pass of the target prompt, i.e., by setting $h^{(j)}_{n_T}(T) = h^{(j)}_{n_S}(S)$ and subsequently completing the altered forward pass. From the resulting next token distribution, we compute $P(C_S^{\ell_S}), P(C_S^{\ell_T}), P(C_T^{\ell_S})$, and $P(C_T^{\ell_T})$.

3.2 Results

In this experiment, we use differing concepts and $\ell_S^{(in)} = \text{DE}, \ell_S^{(out)} = \text{EN}$ for the source- and $\ell_T^{(in)} = \text{FR}, \ell_T^{(out)} = \text{ZH}$ for the target prompt. We perform the patching at one layer at a time and report the probability that is assigned to $P(C_S^{\ell_S}), P(C_S^{\ell_T}), P(C_T^{\ell_S})$, and $P(C_T^{\ell_T})$. As a result we obtain Fig. 2 in which we report means and 95% confidence interval over 200 examples. As model we use Llama-2-7B-base.

Interpretation. We observe the following pattern while patching at different layers (see Fig. 2):

- Layers 0-11: Target concept decoded in target language, resulting in large $P(C_T^{\text{ZH}})$.
- Layers 12-16: Target concept decoded in source language, resulting in large $P(C_T^{\text{EN}})$.
- Layers 16-31: Source concept in source language, resulting in large $P(C_S^{\text{EN}})$.

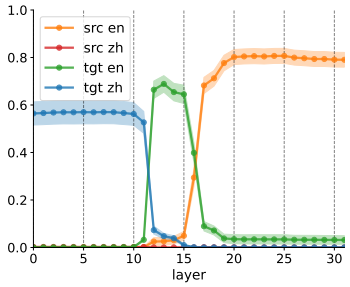

Figure 2: Our first patchscope experiment with a DE to EN source prompt and a FR to ZH target prompt with different concepts. We patch at the last token respectively. For each of the plots the x-axis shows at which layer the patching was performed during the forward pass on the target prompt and the y-axis shows the probability of predicting the correct concept in language ℓ (see legend). In the legend the prefix "src" stands for source and "tgt" for target concept. We report means and 95% Gaussian confidence intervals computed over 200 soure-, target prompt pairs.

This pattern suggests that the model first computes the output language: from layer 12 onwards we decode in the source output language, indicating that $z_{\ell^{(out)}}$, a function vector (Todd et al., 2023) indicating the need to decode to $\ell^{(out)}$, is overwritten at layer 12. In later layers, it determines the concept: from layer 16 on the source concept is decoded, suggesting that $z_{C^{\ell^{(in)}}}$ enters in layer 16.

Hypotheses. We are left with two hypotheses compatible with these results depicted in App. Fig. 4:

- **H1**: Concept and language are represented independently. When doing the translation, the model first computes $\ell^{(out)}$ from context, and then identifies C. In the last layers, it then maps C to the first token of $w(C^{\ell^{(out)}})$.
- **H2**: The representation of a concept is always entangled with its language. When doing the translation, the model first computes $z_{\ell^{(out)}}$, then computes $\ell^{(in)}$ and $C^{\ell^{(in)}}$ from its context and solves the language-pair-specific translation task of mapping $C^{\ell^{(in)}}$ to $C^{\ell^{(out)}}$.

4 Ruling out hypotheses

Next, we run additional experiments to (1.) provide further evidence that we are either in **H1** or **H2** and (2.) to disambiguate whether we are in **H1** or **H2**.

Further evidence experiment. In the experiments in Sec. 3 we did not observe *source concept in target language*. However, both **H1** and **H2** would allow for that to happen via patching in the right

way. Therefore, in this experiment, instead of overwriting latents at the last token of the prompt, we overwrite them at the last token of the word to be translated. Let ρ_S and ρ_T denote the position of that token in source and target prompt respectively. Since the concept information seems to enter via multiple layers (16-20) into the latent of the last token, we overwrite the latent corresponding to the token at position ρ_T at layer j *and all subsequent ones*. By patching in this way in both **H1** and **H2** we'd expect to see large $P(C_S^{\ell_T})$.

Formally, we patch by setting $h^{(j)}_{\rho_T}(T) = h^{(j)}_{\rho_S}(S), \ldots, h^{(m)}_{\rho_T}(T) = h^{(m)}_{\rho_S}(S)$ (in Llama-2-7B with 0-indexing $m = 31$).

Disambiguation experiment. Both **H1** and **H2** compute $w(C_S^{\ell_T})$ but in different ways. In **H1** one decoding circuit per output language is required in order to compute the expression for the concept C in language ℓ_T. In contrast, in **H2** one translation circuit per input-output language pair is required to map the entangled $C^{\ell^{(in)}}$ to $C^{\ell^{(out)}}$. Therefore, in order to disambiguate the two we construct a patching experiment that should work under **H1** but fail under **H2**.

In order to do so, instead of patching the latent containing $C^{\ell_S^{(in)}}$ from a single source forward pass, we create multiple source prompts with the same concept but in different input languages $\ell^{(in)}_{S_1} \neq \ldots \neq \ell^{(in)}_{S_k}$ and patch by setting $h^{(j)}_{\rho_T}(T) = \frac{1}{k}\sum_{i=1}^{k} h^{(j)}_{\rho_{S_i}}(S_i), \ldots$ $h^{(m)}_{\rho_T}(T) = \frac{1}{k}\sum_{i=1}^{k} h^{(m)}_{\rho_{S_i}}(S_i)$. Under **H1** taking the mean of several language-specific concept representations should keep the concept information intact, since $\frac{1}{k}\sum_{i=1}^{k} z_{C_{S_i}^{\ell^{(in)}}} = z_C + \frac{1}{k}\sum_{i=1}^{k} z_{\ell^{(in)}_{S_i}}$. Therefore, we'd expect high $P(C_S^{\ell_T})$ in this case. However, under **H2** in which $z_{C_{S_i}^{\ell^{(in)}}}$ cannot be disentangled this mean is not a well-defined concept and additionally the interference inbetween multiple input languages should cause difficulties for the language-pair-specific translation, which should result in a drop in $P(C_S^{\ell_T})$.

Results. In the first experiment we use the same languages as above and in the second one we used DE, NL, ZH, ES, RU as input- and FR, FI, ES, RU, KO as output languages for the source prompts, and, FR to ZH for the target prompt.

In Fig. 3 we observe, that in both experiments we obtain very high probability for the *source concept in the target language* $P(C_S^{ZH})$ from layers 0 to 15, i.e., exactly until the latents at the last token stop attending to the last concept-token.

Therefore, Fig. 3 (a) supports that we are indeed either in **H1** or **H2** since we successfully decode *source concept in the target language* $P(C_S^{ZH})$ from layers 0 to 15 and Fig. 3 (b) supports that we are in **H1** and not in **H2** because patching in the mean keeps the $P(C_S^{ZH})$ intact.

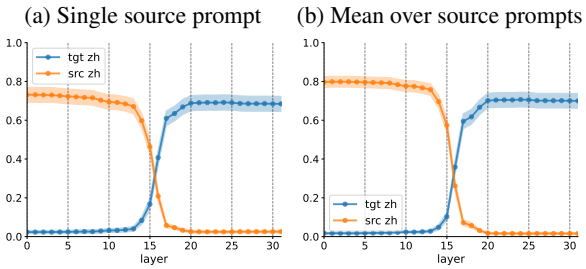

Figure 3: Here we use different input languages (DE, FR), different concepts, different output languages (EN, ZH) in (a). In (b) we use multiple source input languages DE, NL, ZH, ES, RU and output languages FR, FI, ES, RU, KO. We patch at the last token of the concept-word at all layers from j to 31. In (a) we patch latents from the single source prompt in (b) we patch the mean of the latents over the source prompts. For each of the plots, the x-axis shows at which layer the patching was performed during the forward pass on the target prompt and the y-axis shows the probability of predicting the correct concept in language ℓ (see legend). The prefix "src" stands for source and "tgt" for target concept. We report means and 95% Gaussian confidence intervals computed over a dataset of size 200.

5 Discussion

In this paper, we performed multiple experiments that indeed indicate that Llama-2 processes language and concept information independently in the few shot translation prompts used. This also speaks for language- and concept information being represented in a disentangled way. Our results are aligned with findings from prior work (Wendler et al., 2024) that indicate that Llama-2 represents concepts in a concept space independent of the language of the prompt. However, our analysis goes beyond the purely observational logit lens analysis performed by Wendler et al. (2024). Using patchscopes we circumvent potential pitfalls of cosine similarity (Steck et al., 2024) and instead utilize Llama-2's full power.

Limitations

In this work, we studied how Llama-2 represents concepts when processing multilingual text. However, we only considered very simple translation prompts and also probed only for the language-specific words describing the concept. While our results are clearly interesting, further experiments are needed to make claims about how Llama-2 and other language models process multilingual text in general settings. Furthermore, more finegrained probing will be required to determine to which degree Llama-2 is able to specialize a concept to a language and whether concepts and languages are entangled in more subtle ways.

References

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*.

Shan Carter, Zan Armstrong, Ludwig Schubert, Ian Johnson, and Chris Olah. 2019. Activation atlas. *Distill*. Https://distill.pub/2019/activation-atlas.

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*.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. 2023. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*.

Hiroki Furuta, Minegishi Gouki, Yusuke Iwasawa, and Yutaka Matsuo. 2024. Interpreting grokked transformers in complex modular arithmetic. *arXiv preprint arXiv:2402.16726*.

Asma Ghandeharioun, Avi Caciularu, Adam Pearce, Lucas Dixon, and Mor Geva. 2024. Patchscope: A unifying framework for inspecting hidden representations of language models. *arXiv preprint arXiv:2401.06102*.

János Kramár, Tom Lieberum, Rohin Shah, and Neel Nanda. 2024. Atp*: An efficient and scalable method for localizing llm behaviour to components. *arXiv preprint arXiv:2403.00745*.

Samuel Marks, Can Rager, Eric J Michaud, Yonatan Belinkov, David Bau, and Aaron Mueller. 2024. Sparse feature circuits: Discovering and editing interpretable causal graphs in language models. *arXiv preprint arXiv:2403.19647*.

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

Roberto Navigli, Michele Bevilacqua, Simone Conia, Dario Montagnini, Francesco Cecconi, et al. 2021. Ten years of babelnet: A survey. In *IJCAI*, pages 4559–4567. International Joint Conferences on Artificial Intelligence Organization.

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

Charles O'Neill and Thang Bui. 2024. Sparse autoencoders enable scalable and reliable circuit identification in language models. *arXiv preprint arXiv:2405.12522*.

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.

L. Schubert, M. Petrov, S. Carter, N. Cammarata, G. Goh, and C. Olah. 2020. Openai microscope. microscope.openai.com/. [Online; accessed 28-May-2024].

Harald Steck, Chaitanya Ekanadham, and Nathan Kallus. 2024. Is cosine-similarity of embeddings really about similarity? In *Companion Proceedings of the ACM on Web Conference 2024*, pages 887–890.

Eric Todd, Millicent L Li, Arnab Sen Sharma, Aaron Mueller, Byron C Wallace, and David Bau. 2023. Function vectors in large language models. *arXiv preprint arXiv:2310.15213*.

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*.

Igor Tufanov, Karen Hambardzumyan, Javier Ferrando, and Elena Voita. 2024. Lm transparency tool: Interactive tool for analyzing transformer language models. *Arxiv*.

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.

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. 2022. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. *arXiv preprint arXiv:2211.00593*.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in english? on the latent language of multilingual transformers. *arXiv preprint arXiv:2402.10588*.

Ziqian Zhong, Ziming Liu, Max Tegmark, and Jacob Andreas. 2024. The clock and the pizza: Two stories in mechanistic explanation of neural networks. *Advances in Neural Information Processing Systems*, 36.

A Appendix

In Fig. 4 we provide a visualization of our hypotheses about Llama-2-7B's forward pass on the translation prompts after the explorative experiment in Sec. 3.

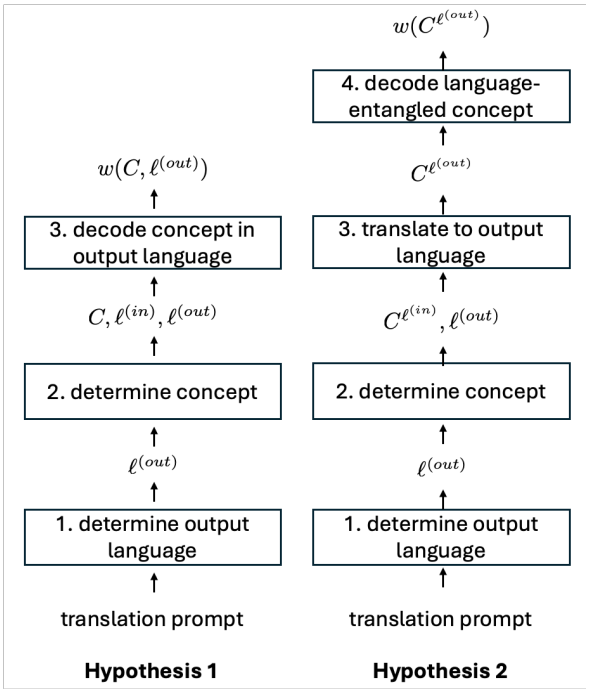

Figure 4: Our hypothesis about how the forward pass could look like on our translation prompts. Every block consists of multiple transformer blocks and inbetween the blocks we denote the relevant content contained in the latents (in the residual stream). Because in hypothesis 2 concept and language cannot be disentangled one input-output language specific translation circuit per language pair is required.