# Representational Analysis of Binding in Language Models

**Qin Dai**[1], **Benjamin Heinzerling**[2,1], **Kentaro Inui**[3,1,2]
[1]Tohoku University [2]RIKEN AIP [3]MBZUAI
qin.dai.b8@tohoku.ac.jp, benjamin.heinzerling@riken.jp
kentaro.inui@mbzuai.ac.ae

## Abstract

Entity tracking is essential for complex reasoning. To perform in-context entity tracking, language models (LMs) must bind an entity to its attribute (e.g., bind a container to its content) to recall attribute for a given entity. For example, given a context mentioning "The coffee is in Box Z, the stone is in Box M, the map is in Box H", to infer "Box Z contains the coffee" later, LMs must bind "Box Z" to "coffee". To explain the binding behaviour of LMs, Feng and Steinhardt (2023) introduce a Binding ID mechanism and state that LMs use a abstract concept called Binding ID (BI) to internally mark entity-attribute pairs. However, they have not captured the Ordering ID (OI) from entity activations that directly determines the binding behaviour. In this work, we provide a novel view of the BI mechanism by localizing OI and proving the causality between OI and binding behaviour. Specifically, by leveraging dimension reduction methods (e.g., PCA), we discover that there exists a low-rank subspace in the activations of LMs, that primarily encodes the order (i.e., OI) of entity and attribute. Moreover, we also discover the causal effect of OI on binding that when editing representations along the OI encoding direction, LMs tend to bind a given entity to other attributes accordingly. For example, by patching activations along the OI encoding direction we can make the LM to infer "Box Z contains the stone" and "Box Z contains the map". The code and datasets used in this paper are available at https://github.com/cl-tohoku/OI-Subspace.

## 1 Introduction

The ability of a model to track and maintain information associated with an entity in a context is essential for complex reasoning (Karttunen, 1976; Heim, 1983; Nieuwland and Van Berkum, 2006; Barzilay and Lapata, 2008; Kamp et al., 2010). To recall attribute information for a given entity in a context, the model must bind entities to their attributes (Feng and Steinhardt, 2023). For example, given Sample 1 and 2, a model must bind the entities (e.g., "Box Z", "Box M", "Box H", "Alex", "John" and "Carl") to their corresponding attributes (e.g., "coffee", "stone", "map", "bean", "pie" and "fruit") so as to recall (or answer) such as what is in "Box Z" or what is sold by "Alex" without confusion. Binding has also been studied as a fundamental problem in Psychology (Treisman, 1996).

To uncover how Language Models (LMs) realize binding in term of internal representation, Feng and Steinhardt (2023) introduce a Binding ID mechanism and state that LMs apply a abstract concept called Binding ID (BI) to bind and mark Entity-Attribute (EA) pairs (e.g., "Box Z" and "coffee" in Sample 1, where BI is denoted as a numbered square). They also claim that the BI is represented as a vector to be added on the representation (or activation) of a EA pair so that the common vector is used as a key clue to search attribute for a given entity. However, they have not captured the Ordering ID (OI) information from the entity (or attribute) activations that causally affects binding behaviour and thus BI information as well. Here, OI is defined as the input order (or ordering index) of entities and attributes, no matter they are bound by a relation (e.g., "is_in" in Sample 1) or not, such as the indexing number in Sample 1 and Sample 3. We can observe that in a 1E-to-1A bound context, such as in Sample 1, BI and OI are interchangeable.

(1) **Context**: The coffee$_{\boxed{0}}$ is in Box Z$_{\boxed{0}}$, the stone$_{\boxed{1}}$ is in Box M$_{\boxed{1}}$, the map$_{\boxed{2}}$ is in Box H$_{\boxed{2}}$.
**Query**: Box Z$_{\boxed{0}}$ contains the

(2) **Context**: The bean$_{\boxed{0}}$ is sold by Person Alex$_{\boxed{0}}$, the pie$_{\boxed{1}}$ is sold by Person John$_{\boxed{1}}$, the fruit$_{\boxed{2}}$ is sold by Person Carl$_{\boxed{2}}$.
**Query**: Person Alex$_{\boxed{0}}$ sells the

(3) **Non-related Context**: The coffee$_{\boxed{0}}$ and Box Z$_{\boxed{0}}$ are scattered around, the stone$_{\boxed{1}}$



*Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17468–17493
November 12-16, 2024 

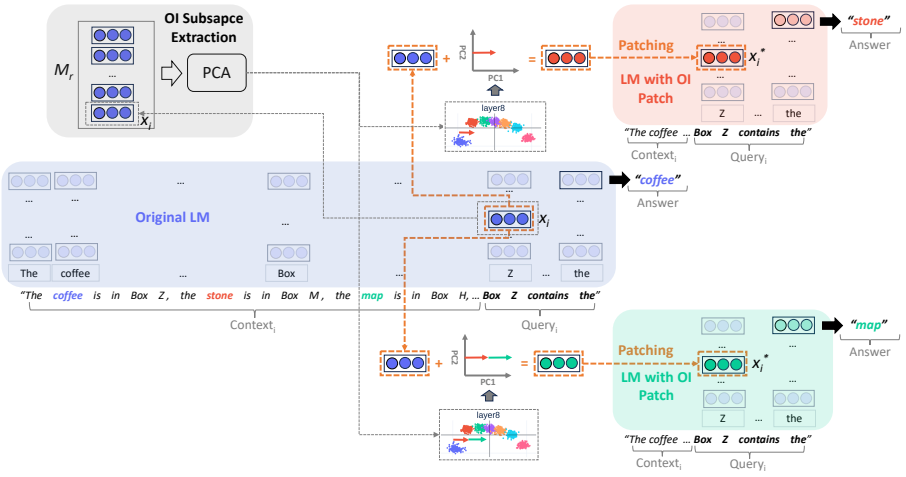



Figure 1: Our main finding on Ordering ID (OI) subspace intervention. Patching entity (e.g., "Z") representations along OI direction (i.e., PC1) in activation space yields corresponding changes in model output.

is here and Box $M_{\boxed{1}}$ is there, the map$_{\boxed{2}}$ and Box $H_{\boxed{2}}$ are in different place. **Query**: Box $Z_{\boxed{0}}$ ...

Since binding is the foundational skill that underlies entity tracking (Feng and Steinhardt, 2023), in this work, we take the entity tracking task (Kim and Schuster, 2023; Prakash et al., 2024) as a benchmark to analyze the LM's binding behaviour. Based on the analysis of internal representation on this task, we localize the OI information from the activiations and provide a novel view of the BI mechanism. Specifically, we apply Principle Component Analysis (PCA) as well as other dimension reduction methods such as Independent Component Analysis (ICA) [1] to analyze the activations of LMs, and which are empirically proven to be effective. We discover that LMs encode (or store) the OI information into a low-rank subspace (called OI subspace hereafter), and the discovered OI subspace can causally affect binding behaviour and thus BI information as well. That is, we find that by causally intervening along the OI encoding Principle Component (PC), LMs swap the binding and infer a new attribute for a given entity accordingly. For example, as shown in Figure 1, by patching activations along the direction (i.e., PC1), we can make the LMs to infer "Box Z contains the stone" and "Box Z contains the map" instead of "Box Z contains the coffee". Therefore, our findings extend the previous BI based understanding of binding in LMs (Feng and Steinhardt, 2023) by revealing the causality between OI and binding.

Overall, our findings suggest that LMs encode OI information into a subspace of LMs' activations that primarily encodes the order index of entities and attributes in a given context. What is more, the discovered OI subspace plays a crucial role in the in-context binding computation. In addition, we find that such OI subspace that determines binding is prevalent across multiple LM families such as Llama2 (Touvron et al., 2023) (and Llama3 (AI@Meta, 2024)), Qwen1.5 (Bai et al., 2023) and Pythia (Biderman et al., 2023), and the code fine-tuned LM Float-7B (Prakash et al., 2024).

## 2 Finding OI Subspace

In this section we describe our Principle Component Analysis (PCA) based method to localize the OI subspace in activations of LMs. As shown in Figure 1, we firstly extract entity activation from LMs. Given a LM (e.g., Llama2), and a collection of texts which describes a set of EA pairs related by a relation such as "is_in" in Sample 1, we extract the activation of entity token (e.g., "Z") in query (denoted as $\mathbf{x}_i$) from certain layer [2] and construct a activation matrix $M_r \in R^{n \times d}$ for a relation $r$, where $n$ denotes the number of entities and $d$ denotes the dimension of the activation. The row $i$ of $M_r$ is the activation of an entity token (i.e., $\mathbf{x}_i$).

PCA has been applied for identifying various subspace (or direction) such as the subspace encoding language bias (Yang et al., 2021), truth value of assertions (Marks and Tegmark, 2023) and sentiment (Tigges et al., 2023). Inspired by these studies, we choose PCA as our first attempt to localize OI subspace. In addition, we also apply other di-

[1] See Appendix (§A.2) and Appendix (§A.3) for details.

[2] See Appendix (§A.4) for the layer selection.

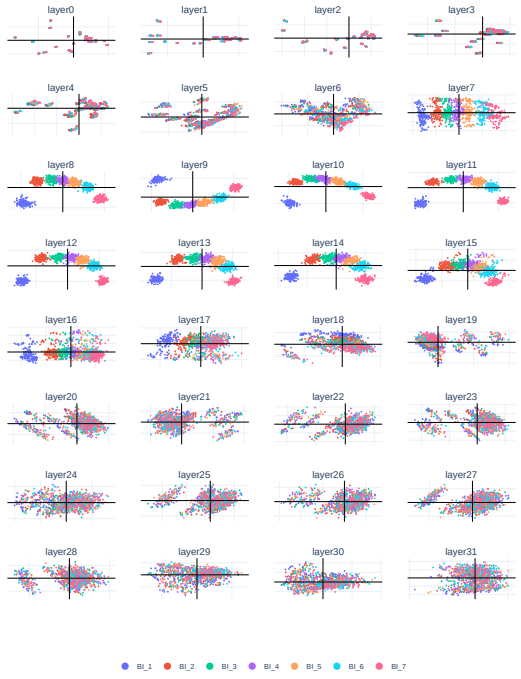



Figure 2: Layer-wise OI subspace visualization on Llama2-7B, where "BI" primarily denotes OI.

mension reduction methods such as Independent Component Analysis (ICA) to capture OI subspace, and which are empirically proven to be effective. See Appendix (§A.2) and Appendix (§A.3) for details.

We hypothesize that in a activation subspace, entities with the same OI tend to cluster together (w.r.t the ones with different OIs), even though these entities have different semantic meaning, and the OIs are encoded as directions (or a PC) in the subspace. For convenience, we number OIs in left-to-right order, and the leftmost OI$= 0$.

Since PCA is applied to identify the principle directions from a multidimensional space, we leverage PCA to capture OI subspace (or direction) from activations of LMs. Specifically, the PCA of a activation matrix is $M_r = U_r \Sigma_r V_r^T$, where the columns of $V_r \in R^{d \times d}$ are principle directions of $M_r$. We takes first $c$ columns of $V_r$ as the OI direction, denoted as $B_r \in R^{d \times c}$.

## 3 OI Subspace Visualization

We adopt a subset of the entity tracking dataset (Kim and Schuster, 2023; Prakash et al., 2024), which contains $n = 1000$ samples, to create layer ($l$) wise activation matrix $M_r^l$. We then uses the $M_r^l$ to extract the layer-wise OI subspace projection matrix $B_r^l \in R^{d \times 2}$ to visualize the activations. Figure 2 shows the embedding visualization on Llama2-7B, where each point represents the activation of an entity projected via the $B_r^l$, and the colors represent OIs. From which, we can observe that middle layers, such as layer 8, have a clearly visible direction along which OI increases, while the others have tangled distribution.

We also observe similar pattern of distribution on Llama3-8B, Float-7B (§A.5) and other LM families such as Qwen1.5 and Pythia (§A.6). This indicates that LMs use the middle layers to encode OI information, and the finding is prevalent across multiple LM families. This finding is also consistent with the "stages of inference hypothesis" (Lad et al., 2024) stating that the function of early layers is to perform detokenization, middle layers do feature engineering, and late layers map the representations from the middle layers into the output embedding space for next-token prediction. According to the hypothesis, we would expect to find the ordering feature most prominently represented in middle layers, which is exactly what the visualization shows. We call this dimension that represents OI as OI Principle Component (OI-PC). In the following section, we apply causal intervention on the OI-PC to analyze how OI-PC affect the model output.

## 4 Causal Interventions on OI Subspace

By projecting the activation matrix $M_r$ into the OI subspace, we have found a correlative evidence for the existence of the direction (i.e., OI-PC) that encodes OI information. However, it is possible that the OI information is encoded in the OI subspace but has no effect on LMs' binding behaviour.

In order to test if OIs are not only encoded in the OI subspace, but that these representations can be steered so as to swap the binding and change LM's output, in this section, we perform interventions to analyze the causality. That is, we want to find out if making interventions along OI-PC leads to a change in LM's binding computation.

### 4.1 Activation Patching

Activation Patching (AP) (Vig et al., 2020; Geiger et al., 2020, 2021; Wang et al., 2022; Stolfo et al., 2023; Heinzerling and Inui, 2024; Engels et al., 2024; Hanna et al., 2024) has been recently proposed to causally intervene computational graph of a LM so as to interpret the function of a target

| Context | Query | Answer for # Step | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| The coffee is in Box Z, the stone is in Box M, the map is in Box H, the coat is in Box L, the string is in Box T, the watch is in Box E, the meat is in Box F. | Box Z contains the | stone | map | map | string | watch | meat |
| The letter is in Box Q, the boot is in Box C, the fan is in Box N, the crown is in Box R, the guitar is in Box E, the bag is in Box D, the watch is in Box K. | Box Q contains the | boot | fan | crown | guitar | watch | watch |
| The cross is in Box Z, the ice is in Box D, the ring is in Box F, the plane is in Box Q, the clock is in Box X, the paper is in Box I, the engine is in Box K. | Box Z contains the | ice | ring | ring | clock | paper | engine |

Table 1: Attributes inferred by Llama2-7B as a result of directed activation patching along OI-PC in the OI subspace on the dataset of "r: is_in", where color denotes the BI.

computational node (or edge). The process of AP usually involves preparing corrupted input, using its activations to replace the corresponding ones obtained from original input and analyzing the effect on model output. Different with the common AP setup, we realize AP by directly editing activations along a particular direction (i.e., along OI-PC), similar to the activation editing method of (Matsumoto et al., 2023; Heinzerling and Inui, 2024; Engels et al., 2024).

### 4.2 Setting

**Dataset** To explore the internal representation that enables binding, we adopt the entity tracking dataset (Kim and Schuster, 2023; Prakash et al., 2024). The dataset consists of English sentence describing a set of objects (here called attributes) located in a set of boxes with difference labels (here called entities), and the task is to infer what is contained by a given box. For instance, when a LM is presented with "The coffee is in Box Z, the stone is in Box M, the map is in Box H, ... Box Z contains the", the LM should infer the next token as "coffee". Each sample involves 7 EA pairs. To evaluate the binding in various context, we also apply the templates shown in Table 2 to generate other 5 datasets with different relation, where $a_i$ and $e_i$ denotes the attribute and entity, and they are sampled from a fixed pool of 224 one-token objects (e.g., "dog", "corn" and "cookie") and 523 of one-token names (e.g., "Alex", "Juli" and "Dan") respectively. We sample $n = 1000$ context from each dataset to run the following analysis.

**Metrics** We apply two evaluation metrics: logit difference and logit flip. The logit difference metric is introduced in Wang et al. (2022), which calculates difference in logits of a target token between original and intervened setting. The "logit flip" accuracy metric is introduced by Geiger et al. (2022), which represents the proportion of candidate tokens in model output after a causal intervention.

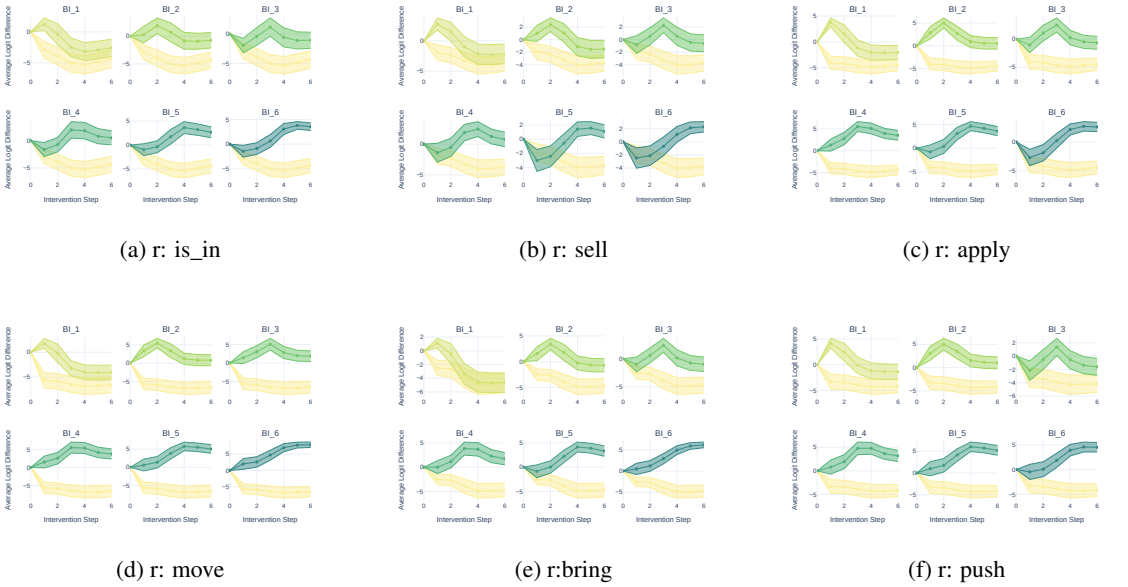
| | Template |
|---|---|
| 1 | The $a_0$ is sold by person $e_0$, ..., the $a_i$ is ..., $a_7$ is sold by person $e_7$. Person $e_i$ is selling the |
| 2 | The $a_0$ is applied by person $e_0$, ..., the $a_i$ is ..., $a_7$ is applied by person $e_7$. Person $e_i$ applies the |
| 3 | The $a_0$ is moved by person $e_0$, ..., the $a_i$ is ..., $a_7$ is moved by person $e_7$. Person $e_i$ moved the |
| 4 | The $a_0$ is brought by person $e_0$, ..., the $a_i$ is ..., $a_7$ is moved by person $e_7$. Person $e_i$ brings the |
| 5 | The $a_0$ is pushed by person $e_0$, ..., the $a_i$ is ..., $a_7$ is pushed by person $e_7$. Person $e_i$ pushes the |

Table 2: Templates of Dataset.

### 4.3 Results: Direct Editing OI Subspace

We hypothesize that LMs encode OI information into a low-rank subspace that causally affect BI information and thus binding behaviour as well. Therefore, we wonder if a LM changes the binding behavior, when adding a particular value $v$ (called step hereafter) along the OI-PC mentioned in Section (§2) [3]. For example, if we add one unit of $v$ on the OI-PC of $e_0$, the LM will reset its BI as 1 and bind attribute $a_1$ to the entity based on the similarity of BI so that infers the $a_1$ as the attribute of $e_0$ instead of the original $a_0$. Similarly, adding two units of $v$ will make the LM infer $a_2$ for $e_0$, and so on. We intervene via the Equation 1, where

[3] where PC2 is assigned a fixed value, which is a hyperparameter.



(a) r: is_in (b) r: sell (c) r: apply

(d) r: move (e) r:bring (f) r: push

Figure 3: Logit Difference (LD) for OI-PC based intervention across datasets on Llama2-7B, where x axis denotes the number of intervention steps on $e_0$, y axis does the LD, BI_i represents each target attribute and the light yellow bottom line indicates the LD of original attribute (i.e., $a_0$). Here, $l = 8$, $v = 2.5$, and $\alpha = 3.0$.

$\mathbf{x}_{0,l}$ is the original activation of $e_0$ (i.e., the leftmost entity) in layer $l$, $\mathbf{x}^*_{0,l}$ is the intervened activation, $B_r$ is the OI subspace projection matrix mentioned in Section (§2), $\alpha$ is a hyper-parameter to scale the effect of intervention and $\beta$ ($0 \leqslant \beta \leqslant 6$) denotes the number of steps.

$$\mathbf{x}^*_{0,l} = \mathbf{x}_{0,l} + \alpha B_r^T (B_r \mathbf{x}_{0,l} + \beta v) \quad (1)$$

Table 1 lists several examples under the OI subspace intervention on the entity tracking dataset (Kim and Schuster, 2023; Prakash et al., 2024). We also list the examples from other datasets in Appendix (§A.7). We can see that when adding 1 step along OI-PC, the model selects "stone" for entity "Z" instead of its original attribute "coffee". Similarly, when the step is doubled, the model will select attribute "map" for the entity, and so on. This indicates that changing the value along OI-PC can induce the swap of attribute. In addition, we notice that some attributes are repeated or skipped after the intervention. The reason is that, as shown in Figure 2, OI is represented as a continuous range (e.g., $[a_0, b_0], [a_1, b_1], [a_2, b_2], ...$) on OI-PC, implying the points in the same range share OI, and if a sample (e.g., its OI-PC is $s_i$) is still in the range (e.g., $a_0 < (s_i + v) < b_0$) or skip its neighbouring range (e.g., $a_2 < (s_i + v) < b_2$) after the intervention (e.g., $s_i + v$), its OI will be the same or added by 2, and thus the binding information will change accordingly.

Besides the qualitative analysis, we also conduct quantitative analysis for the causality between the OI subspace based AP and the binding behaviour of LMs. We plot mean-aggregated effect of the OI-PC based AP across multiple datasets in Figure 3. Figure 3 indicates how the Logit Difference (LD) of each attribute changes as the step increases. We can observe that as the number of steps increases, LD of the original attribute decreases. In contrast, LD of other attributes gradually increase until a certain point and then gradually decrease. Given a candidate attribute, its LD peak roughly corresponds to the number of steps that is equal to its BI. For instance, when adding 3 steps, the points of BI_3 (i.e., attributes of BI= 3) achieve the highest LD score. This indicates that by adjusting the value along the OI-PC, we can adjust BI information and thus increase the logit score of the corresponding attribute.

Similarly, Figure 4 illustrates the relation between the number of steps and the logit flip, which gauges the percentage of the predicted attributes under an intervention. Figure 4 shows that as the step increases, the proportion bar becomes darker, it means that the model promotes the proportion of the corresponding attribute in its inference. For instance, when adding 3 step on the subspace, the $a_3$ (i.e., BI_3) becomes the major of the answers. This proves that the OI-PC based interventions can

causally affect BI information as well as the computation of Binding in a LM. See Appendix (§A.10) for the results on Llama3-8B, Appendix (§A.6) for the results on Qwen1.5-7B and Pythia-6.9B, and Appendix (§A.9) for the results on Float-7B.

### 4.4 Results: Activation Steering on OI Subspace

Inspired by the research on Activation Steering (AS) (Turner et al., 2023), we apply an AS method to verify the importance of the OI subspace on LM's binding behaviour. Specifically, we use the following Equation 2 to extract a subspace steering vector $\mathbf{s}_{0\to bi}$, which is proposed to swap BI from 0 to $bi$, where $n$ is the number of target entities, $\mathbf{x}^i_{bi,l}$ represents the activation of entity $e_i$ from layer $l$, and its BI is $bi$. We intervene via Equation 3 and assume that by adding $\mathbf{s}_{0\to bi}$ on the original activation $\mathbf{x}_{0,l}$, we can increase the LD and the proportion of the attribute $a_{bi}$. Figure 5 shows the results on the entity tracking dataset (Kim and Schuster, 2023; Prakash et al., 2024). (Appendix (§A.8) shows the results on other datasets) These results indicate that AS can achieve the similar tendency as the direct value intervention mentioned in Section (§4.3). For instance, adding $\mathbf{s}_{0\to 3}$, which is supposed to swap BI from 0 to 3, can increase the LD of $a_3$ and its proportion in the predicted answers. The consistent tendency with the results of the direct subspace editing, shown Figure 3 and Figure 4, further illustrates that the discovered OI subspace can causally affect BI information and binding behaviour. Therefore, OI subspace plays an important role when LMs perform in-context binding computation.

$$\mathbf{s}_{0\to bi} = \frac{1}{n}\sum_{i=1}^{n}(B_r\mathbf{x}^i_{bi,l} - B_r\mathbf{x}^i_{0,l}) \quad (2)$$

$$\mathbf{x}^*_{0,l} = \mathbf{x}_{0,l} + \alpha B_r^T\mathbf{s}_{0\to bi} \quad (3)$$

### 4.5 OI Subspace and Position

In this section, we discuss the relationship between the OI subspace and Positional Information (PI), which is namely the $postion\_ids$ of input tokens. As mentioned in Section (§4.3), the discovered subspace can causally determine BI information. Therefore, direct intervention on the subspace can swap the answer of a LM. However, one counter hypothesis is that the subspace is not used for storing OI information but the PI of attributes, and thus the direct intervention merely changes the PI of answer token so that the LM applies the new PI to generate corresponding token. In other words, the OI space (or OI-PC) might have high correlation with PI but not OI.

| | |
|---|---|
| | **Input (original)** |
| $C_1$ | $a_0^{p0}\, r\, e_0^{p1}$, $a_1^{p2}\, r\, e_1^{p3}$, $a_2^{p4}\, r\, e_2^{p5}$. $\quad e_1^{p3}\, r^{-1}$ ? |
| $C_2$ | $a_0^{p0}\, r\, e_0^{p1}$, $a_1^{p2}\, r\, e_1^{p3}$, $a_2^{p4}\, r\, e_2^{p5}$. $\quad e_2^{p5}\, r^{-1}$ ? |
| | **Input (with pseudo)** |
| $C'_1$ | $a_{*0}^{p0}\, r\, e_{*0}^{p1}$, $a_{*0}^{p2}\, r\, e_{*0}^{p3}$, $a_1^{p4}\, r\, e_1^{p5}$. $\quad e_1^{p5}\, r^{-1}$ ? |
| $C'_2$ | $a_{*0}^{p0}\, r\, e_{*0}^{p1}$, $a_1^{p2}\, r\, e_1^{p3}$, $a_2^{p4}\, r\, e_2^{p5}$. $\quad e_2^{p5}\, r^{-1}$ ? |

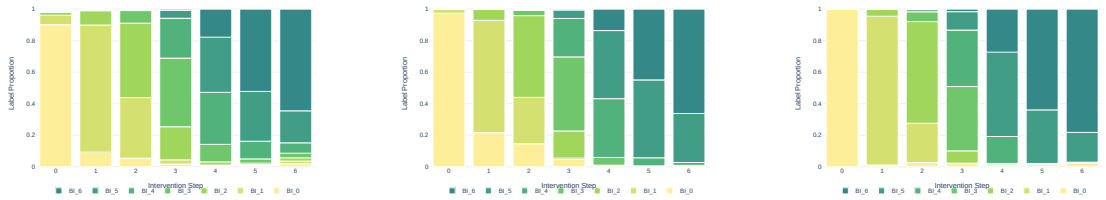
Table 3: Simplified expression of original inputs and the one modified with pseudo relation, which is proposed to equalize PI for PCA analysis, where $a_0^{p1}\, r\, e_0^{p2}$ represents a relation such as "the apple is in Box C", and $e_0^{p2}$ denotes an entity with OI of 0 and PI of $p2$, $e_2^{p5}\ r^{-1}$ ? denotes the query on entity $e_1$, such as "Box C contains the".

| **Input (original)** |
|---|
| "The apple is in Box E, the bell is in Box F, ..." |
| **Input (with filler words)** |
| "*I will find out that* the apple is in Box E, the bell is in Box F, ..." |

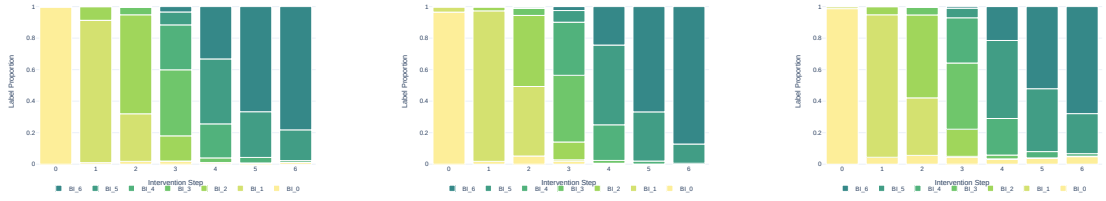
Table 4: An example of the dataset with filler words "*I will find out that*".

To prove the independence between OI subspace and PI, we create three datasets, the first one is by extending the original dataset with pseudo relation, as shown in Table 3, the second one is by prefixing the original dataset with filler words(e.g., "It can be seen that") and third one is by inserting a sequence of interjection(e.g., "ah, ah, ..."), as listed in Table 12.

**New Dataset with Pseudo Relation.** In Table 3, $a_{*0}^{p0}\, r\, e_{*0}^{p1}$ refers to a pseudo relation, which is a fixed expression, such as "the PC is in Box Z". The pseudo relation is applied to adjust the PI while keeping the OI. For instance, in Table 3, adding one or two $a_{*0}^{p0}\, r\, e_{*0}^{p1}$ before $a_1\, r\, e_1$ (i.e., $C'_2$ and $C'_1$) does not affect the OI of $e_1$ but its PI, because $e_1$ is still the second unique entity from the left (i.e., its ordering index 0I$=1$), but its PI is $p3$ and $p5$ respectively. Using the pseudo relation, we create the data in a manner that the target entity for activation analysis (e.g., $e_1^{p5}$ and $e_2^{p5}$) have the same PI but different OI, such as $C'_1$ and $C'_2$ in Table 3.



(a) r: is_in (b) r: sell (c) r: apply



(d) r: move (e) r: bring (f) r: push

Figure 4: Logit flip for OI-PC based intervention across datasets on Llama2-7B, where x axis denotes the number of intervention steps on $e_0$, y axis does the proportion of each inferred attribute in model output.

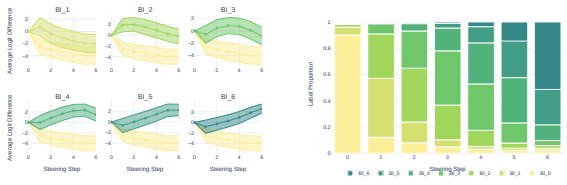


Figure 5: Logit Difference and Logit Flip for activation steering on the entity tracking dataset (i.e., r: is_in), where x axis represents the intervention of $\mathbf{s}_{0 \rightarrow bi}$.

We apply the method mentioned in Section (§2) on the set of activations $\{\overrightarrow{e_1^{p5}}, \overrightarrow{e_2^{p5}}, ...\}$, where $\overrightarrow{e_1^{p5}}$ denotes the activation of $e_1$ in query (i.e., $e_1^{p5}$ $r^{-1}$ ?), so as to capture the OI difference and exclude the PI difference, because they share the same PI (i.e., $P5$) but different OI (i.e., 1, 2, ...). Then we compare its OI subspace with the original one (e.g., $\{\overrightarrow{e_1^{p2}}, \overrightarrow{e_2^{p5}}, ...\}$) to analyze how the distribution of OI subspace changes after removing the PI variance. Figure 6 visualizes the OI subspace distribution, where the light colored points denote the original distribution, and the dark ones are from the new one with equalized PI. We can observe that after removing the PI difference, the distribution is still similar to the original one that there is a clearly visible direction along which OI increases. This illustrates that our PCA based method can capture OI information, that is, along the direction of OI-PC, and it does not causally depend on PI.

**New Dataset with Filler Words.** The dataset is created by adding Filler Words (FW) with various length, such as "OK", "I see that" and "There is no particular reason", in front of the entity tracking dataset (Kim and Schuster, 2023; Prakash et al., 2024), as shown in Table 4. Since the length (i.e., the number of tokens) of FW directly changes the PI of its following entities and attributes without affecting their OIs, we take the length as the measure of intervention on PI and apply Spearman's rank correlation $\rho$ to calculate the correlation between the length (denoted as PI) and the OI-PC value. Figure 7 shows $\rho$ between PI and OI-PC as well as between OI and OI-PC. We can observe that OI-PC has high $\rho$ with OI but almost zero $\rho$ with PI, indicating that the discovered OI-PC is highly correlated with OI information but independent on PI. Therefore, the OI-PC does not simply encode absolute token position. See Appendix (§A.11) for the analysis on the **New Dataset with Interjections**.

### 4.6 Consistency of OI Subspace

The BI mechanism (Feng and Steinhardt, 2023) mentions that a LM represents a related EA pair through the consistency of their BI information. This naturally raises a research question that if there is the consistency between OI-PC of entities (e.g., OI-PC of "Box Z" in Sample 1) and of their

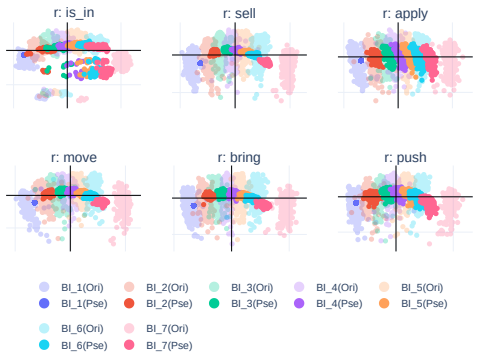



Figure 6: Embedding visualization for activation with equalized PI, where "Ori" denotes the distribution of original dataset, while "Pse" denotes the distribution of the new dataset with pseudo relation.


Figure 7: Spearman's rank correlation between OI-PC and PI (or OI), where "PC$i$" denotes the $i$-th PC of the OI subspace and "PI" is the length of FW.

corresponding attributes [4] (e.g., OI-PC of"coffee").

To analyze the consistency of OI-PC between a related EA pair, we prepare four alternative datasets with various binding pattern as shown in Table 6. We use the distance on OI-PC to measure consistency and show partial result of mutual OI-PC based distance in Figure 8. We can observe that potentially related EA pairs (e.g., "E0_0" and "A0_0") tend to have lower OI-PC based distance than arbitrary pairs in all binding patterns, indicating that to some extent, OI-PC based distance could be seen as an important feature to represent binding. To further illustrate its significance, we attempt to classify related EA pairs only relying on their OI-PC based distances. Specifically, we search an optimal threshold value from a development set, and which is applied to classify potentially related EA pairs in a testing set. The results are shown in Figure 9. We

[4]The OI-PC of attribute is extracted from the query of attribute such as "(context) The coffee is in".

| **Input (original)** |
|---|
| "The *apple* is in *Box E*, the *bell* is in *Box F*, ..." |
| **Input (Non-related)** |
| "I see *apple*, somewhere else there is *Box E*, the *bell* and *Box F* are scattered around, ..." |

Table 5: An example of the dataset with non-related expression.

| **Input (7A-7E)** |
|---|
| "$A_0$ is in $E_0$, $A_1$ is in $E_1$, $A_2$ is in $E_2$, $A_3$ is in $E_3$, $A_4$ is in $E_4$, $A_5$ is in $E_5$, $A_6$ is in $E_6$." |
| **Input (7A-3E)** |
| "$A_0$ is in $E_0$, $A_1$ is in $E_0$, $A_2$ is in $E_0$, $A_3$ is in $E_1$, $A_4$ is in $E_1$, $A_5$ is in $E_2$, $A_6$ is in $E_2$." |
| **Input (7A-2E)** |
| "$A_0$ is in $E_0$, $A_1$ is in $E_0$, $A_2$ is in $E_0$, $A_3$ is in $E_1$, $A_4$ is in $E_1$, $A_5$ is in $E_1$, $A_6$ is in $E_1$." |
| **Input (7A-5E)** |
| "$A_0$ is in $E_0$, $A_1$ is in $E_0$, $A_2$ is in $E_0$, $A_3$ is in $E_1$, $A_4$ is in $E_2$, $A_5$ is in $E_3$, $A_6$ is in $E_4$." |

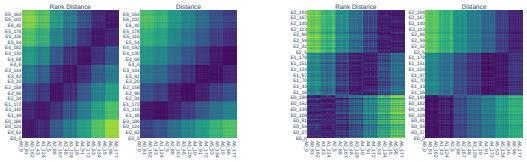
Table 6: Simplified expression of the datasets with various binding pattern, where "7A-3E" represents the pattern containing 7 attributes and 3 entities.

can observe that the performance of the OI-PC (i.e., "PC1") based distance is significantly better than other PCs across all binding patterns, indicating that comparing to other PCs, the OI-PC could be used to compute binding information.

### 4.7 OI Subspace and Relatedness

Binding of a EA pair means that the EA pair is bound by a binding relation such as property and location (Treisman, 1996). In turn, there is no binding when the EA pair is unrelated in its context. This raises a research question that if the correlation between OI-PC of attributes (e.g., OI-PC of "apple") and of their corresponding entities (e.g., OI-PC of "Box E") represents the relatedness namely the existence of a relation.

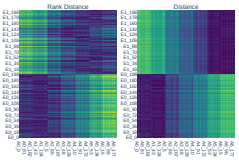
In order to uncover the relationship between OI-PC and the relatedness, we create an alternative dataset by converting relational expression of the entity tracking dataset (Kim and Schuster, 2023; Prakash et al., 2024) into non-related one. Specifically, we prepare a set of non-related expression templates and randomly select one to replace the original expression of relation as shown in Table 5. We can observe that the template could make a target EA pair (e.g., "Box E" and "apple") semanti-



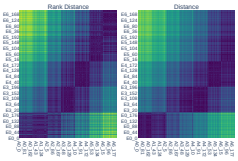
(a) 7A-7E



(b) 7A-3E



(c) 7A-2E



(d) 7A-5E

Figure 8: : OI-PC based distance heat map, where "Rank Distance" denotes the distance based on the rank of OI-PC value, "A$i$_" (or "E$i$_") is a sample of Attribute (or Entity) with OI= $i$, each cell represents the distance between a corresponding EA pair, and the darker the color, the smaller the distance is.

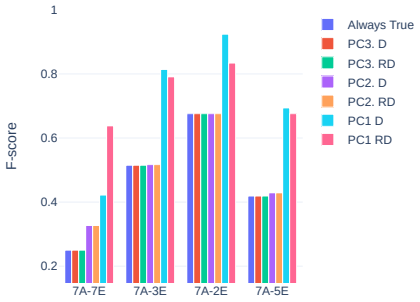



Figure 9: OI-PC distance based classification results, where "Always True" denotes the baseline that predicts all candidates as True, "PC$i$ D" denotes the Hamming distance on $i$-th PC of the OI subspace, and "PC$i$ RD" denotes the distance calculated by the rank of PC$i$'s value among samples.

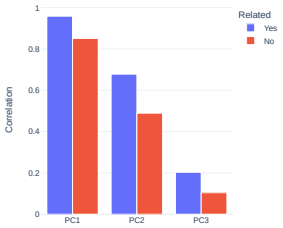



Figure 10: OI-PC based correlation between attributes and their corresponding entities, where "PC$i$" denotes the $i$-th PC of the OI subspace, "Yes" and "No" represent the related (i.e., original) and non-related dataset respectively.

cally unrelated but retain their OI (e.g., the OI of "apple" and "bell" are still 0 and 1 respectively). We select Spearman's rank correlation $\rho$ as the correlation metric and compare the $\rho$ of the non-related dataset with the related one in the Figure 10.

We can observe that $\rho$ of non-related dataset is slightly lower than the related (i.e., original) one, indicating that the OI-PC might contain limited relational information so that removing it can marginally decrease the $\rho$. However, there is still strong correlation between the non-related (or non-bound) entity attribute pair, indicating that the OI-PC primarily encodes the OI information but not binding information specifically the information of binding relation.

## 5 Related Work

**Linear Representation** Recent research found that sequence models trained only on next token prediction linearly represent various semantic concepts including Othello board positions (Li et al., 2022; Nanda et al., 2023), the truth value of assertions (Marks and Tegmark, 2023), sentiment (Tigges et al., 2023), and numeric values such as elevation, population, birth year, and death year (Gurnee and Tegmark, 2023; Heinzerling and Inui, 2024). Continuing this line of research, in this work, we discover that multiple LMs such as Llama2 can also linearly encode OI along a OI increasing direction in the activations.

See Appendix (§A.1) for additional related work.

## 6 Conclusion and Future Work

In this work, we study the in-context binding, a fundamental skill underlying many complex reasoning and natural language understanding tasks. We provide a novel view of the Binding ID mechanism introduced by Feng and Steinhardt (2023) that there exists a low-rank subspace in the hidden state (or activation) of LMs that primarily encodes the ordering information and which is used as the prototype of BIs to causally determine binding. Our future work includes: 1. the analysis of OI subspace in a more realistic setting; 2. the study of interaction between in-context binding and factual knowledge learned from pretraining; 3. OI subspace based mechanistic analysis.

## Acknowledgement

This work was supported by JST CREST Grant Number JPMJCR20D2 and JSPS KAKENHI Grant Number 21K17814. We are grateful to the anonymous reviewers for their constructive comments.

## Limitations

The limitations of our research include the following points: 1. We only analyze OI subspace on the attribute prediction task, but not on the entity inference task (i.e., given an attribute to infer its entity); 2. Although PCA based analysis is empirically proven to be effective, we lack the theoretical analysis on why PCA could capture OI subspace; 3. We lack the analysis on how predicate (or relation) affect the OI subspace, and how the results of OI subspace based intervention differ with the type of predicate; 4. Although we use a publicly available entity tracking dataset, it is still a synthesized dataset. Therefore, for uncovering how LMs bind and track entity in reality, it is necessary to analyze the binding via a real world dataset; 5. We only analyze the activations in the query part instead of in the context part, thus this work can not explain how LMs encode OI in the context and how it is used for OI encoding in query part and how it contributes the binding computation; 6. We infer the change of BI information via the result of model output but the interaction between OI and BI information is not directly observed. This research thus lacks the mechanistic interpretation on the interaction between them; 7. We only analyze binding from the perspective of representation and localize OI subspace. However, we have not answered what is the mechanism that generates the subspace and what is the circuit that utilizes the subspace for binding.

## Ethical Statement

The existing dataset (Kim and Schuster, 2023; Prakash et al., 2024) and LMs (i.e., Llama2-7B, Float-7B, Llama3-8B, Qwen1.5-7B and Pythia-6.9B) are applied according to their intended research purpose. The synthetic datasets we adopted in this work are automatically created by strictly following the rule (or pattern) of the existing dataset, where the entities and attributes are sampled from a pool of wide variety of one-token names and concepts. Therefore, there is no ethical concern on human annotation bias and semantic biases.

## References

AI@Meta. 2024. Llama 3 model card.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. arXiv preprint arXiv:2309.16609.

Regina Barzilay and Mirella Lapata. 2008. Modeling local coherence: An entity-based approach. Computational Linguistics, 34(1):1–34.

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.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2021. Knowledge neurons in pretrained transformers. arXiv preprint arXiv:2104.08696.

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, 1:1.

Joshua Engels, Isaac Liao, Eric J. Michaud, Wes Gurnee, and Max Tegmark. 2024. Not all language model features are linear.

Jiahai Feng and Jacob Steinhardt. 2023. How do language models bind entities in context? arXiv preprint arXiv:2310.17191.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. 2021. Causal abstractions of neural networks. Advances in Neural Information Processing Systems, 34:9574–9586.

Atticus Geiger, Kyle Richardson, and Christopher Potts. 2020. Neural natural language inference models partially embed theories of lexical entailment and negation. arXiv preprint arXiv:2004.14623.

Atticus Geiger, Zhengxuan Wu, Hanson Lu, Josh Rozner, Elisa Kreiss, Thomas Icard, Noah Goodman, and Christopher Potts. 2022. Inducing causal structure for interpretable neural networks. In International Conference on Machine Learning, pages 7324–7338. PMLR.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. arXiv preprint arXiv:2304.14767.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2020. Transformer feed-forward layers are key-value memories. arXiv preprint arXiv:2012.14913.

Wes Gurnee and Max Tegmark. 2023. Language models represent space and time. arXiv preprint arXiv:2310.02207.

Michael Hanna, Ollie Liu, and Alexandre Variengien. 2024. How does gpt-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model. Advances in Neural Information Processing Systems, 36.

Irene Heim. 1983. File change semantics and the familiarity theory of definiteness. Semantics Critical Concepts in Linguistics, pages 108–135.

Benjamin Heinzerling and Kentaro Inui. 2024. Monotonic representation of numeric properties in language models. arXiv preprint arXiv:2403.10381.

Evan Hernandez, Arnab Sen Sharma, Tal Haklay, Kevin Meng, Martin Wattenberg, Jacob Andreas, Yonatan Belinkov, and David Bau. 2023. Linearity of relation decoding in transformer language models. arXiv preprint arXiv:2308.09124.

Hans Kamp, Josef Van Genabith, and Uwe Reyle. 2010. Discourse representation theory. In Handbook of Philosophical Logic: Volume 15, pages 125–394. Springer.

Lauri Karttunen. 1976. Discourse referents. In Notes from the linguistic underground, pages 363–385. Brill.

Najoung Kim and Sebastian Schuster. 2023. Entity tracking in language models. arXiv preprint arXiv:2305.02363.

Najoung Kim, Sebastian Schuster, and Shubham Toshniwal. 2024. Code pretraining improves entity tracking abilities of language models. arXiv preprint arXiv:2405.21068.

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.

Vedang Lad, Wes Gurnee, and Max Tegmark. 2024. The remarkable robustness of llms: Stages of inference? arXiv preprint arXiv:2406.19384.

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.

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.

Yuta Matsumoto, Benjamin Heinzerling, Masashi Yoshikawa, and Kentaro Inui. 2023. Tracing and manipulating intermediate values in neural math problem solvers. arXiv preprint arXiv:2301.06758.

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.

Neel Nanda, Andrew Lee, and Martin Wattenberg. 2023. Emergent linear representations in world models of self-supervised sequence models. arXiv preprint arXiv:2309.00941.

Mante S Nieuwland and Jos JA Van Berkum. 2006. When peanuts fall in love: N400 evidence for the power of discourse. Journal of cognitive neuroscience, 18(7):1098–1111.

Nikhil Prakash, Tamar Rott Shaham, Tal Haklay, Yonatan Belinkov, and David Bau. 2024. Fine-tuning enhances existing mechanisms: A case study on entity tracking. arXiv preprint arXiv:2402.14811.

Alessandro Stolfo, Yonatan Belinkov, and Mrinmaya Sachan. 2023. A mechanistic interpretation of arithmetic reasoning in language models using causal mediation analysis. arXiv preprint arXiv:2305.15054.

Curt Tigges, Oskar John Hollinsworth, Atticus Geiger, and Neel Nanda. 2023. Linear representations of sentiment in large language models. arXiv preprint arXiv:2310.15154.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Anne Treisman. 1996. The binding problem. Current opinion in neurobiology, 6(2):171–178.

Alex Turner, Lisa Thiergart, David Udell, Gavin Leech, Ulisse Mini, and Monte MacDiarmid. 2023. Activation addition: Steering language models without optimization. arXiv preprint arXiv:2308.10248.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. 2020. Investigating gender bias in language models using causal mediation analysis. Advances

in neural information processing systems, 33:12388–12401.

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.

Svante Wold, Michael Sjöström, and Lennart Eriksson. 2001. Pls-regression: a basic tool of chemometrics chemometr. Intell. Lab, 58(2):109–130.

Zhengxuan Wu, Atticus Geiger, Thomas Icard, Christopher Potts, and Noah Goodman. 2024. Interpretability at scale: Identifying causal mechanisms in alpaca. Advances in Neural Information Processing Systems, 36.

Ziyi Yang, Yinfei Yang, Daniel Cer, and Eric Darve. 2021. A simple and effective method to eliminate the self language bias in multilingual representations. arXiv preprint arXiv:2109.04727.

# A Appendix

## A.1 Other Related Work

**Knowledge Localization** Many works aim to localize and edit factual relations (e.g., "capital of") that LMs learn from pretraining and are stored into model weights (Geva et al., 2020; Dai et al., 2021; Meng et al., 2022; Geva et al., 2023; Hernandez et al., 2023). Different from this line of research, this work studies in-context representations of relations and analyzes how they are represented in model activations.

**Mechanistic Interpretability** Notable progress has been made in uncovering circuits performing various tasks within LMs (Elhage et al., 2021; Wang et al., 2022; Wu et al., 2024). Recently, Prakash et al. (2024) identify the circuit for entity tracking task. Feng and Steinhardt (2023) introduce a Binding ID Mechanism for explaining the binding problem, state that LMs use the abstract concept BI to internally mark entity-attribute pairs. However, they have not captured the OI information from entity activations that directly determines the binding behaviour.

## A.2 Partial least squares regression and PCA

Besides PCA, a commonly used unsupervised Dimension Reduction (DR) method, we also attempt Partial Least Squares regression (PLS) (Wold et al., 2001), a supervised DR method. PLS extracts a set of ordered latent variables that maximizes the co-variability between the features (e.g., activations) and the scores to be predicted (e.g., OI). We perform PCA and PLS on a development set and compare their regression cures in Figure 11. We can observe that both the first PCA component and the first PLS direction contain almost all information about OI of target entity, because their regression score is close to one. The consistency indicates that PCA is an effective method to capture OI subspace.

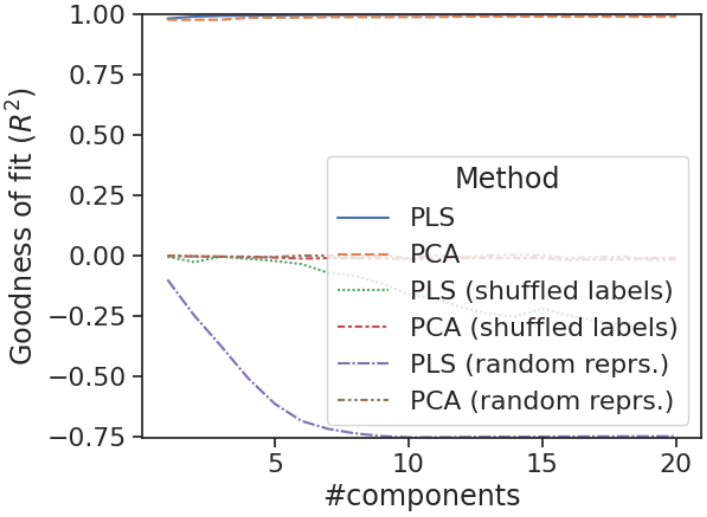



Figure 11: Regression curves for PLS and PCA.

## A.3 ICA for OI Subspace

Independent Component Analysis (ICA) is a statistical method to reveal hidden subcomponent from multivariate signal. Similar to PCA, ICA is utilized as a dimension reduction method. We apply ICA to analyze the activations of entities and visualize the subspace in Figure 12. We can observe that the distribution is generally similar to the one of PCA. In addition, we also conduct causal intervention (i.e., activation steering) based on the subspace and present the results in Figure 13. We can observe that the results are also similar to those from PCA based intervention. This indicates that besides PCA, ICA can also be used as a computational method to capture OI subspace.

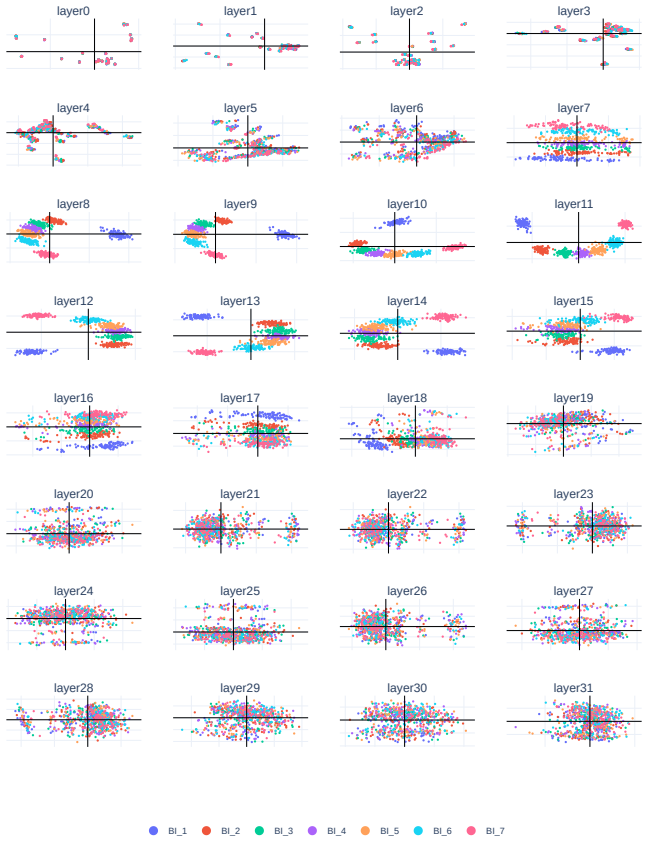



Figure 12: Subspace visualization from ICA on Llama2-7B, where "BI" primarily denotes OI.

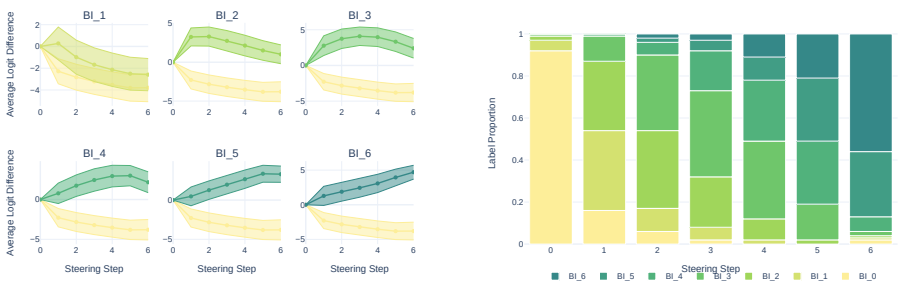
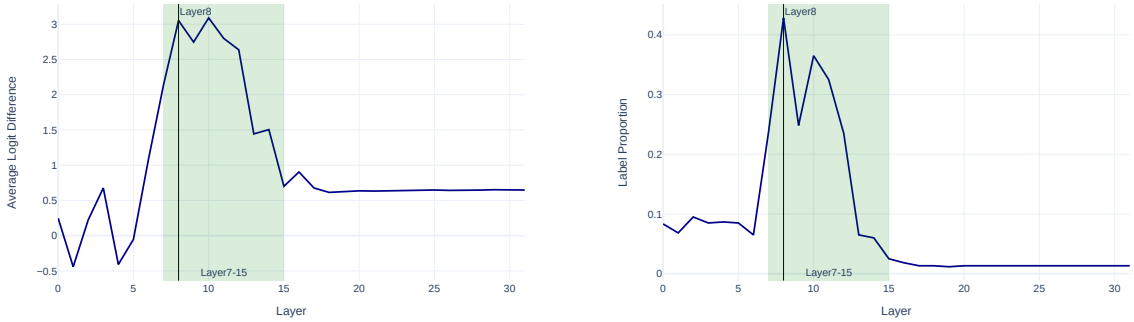



Figure 13: Logit Difference and Logit Flip for ICA based activation steering on the entity tracking dataset (i.e., r: is_in) on Llama2-7B.

### A.4 Layer-wise Intervention

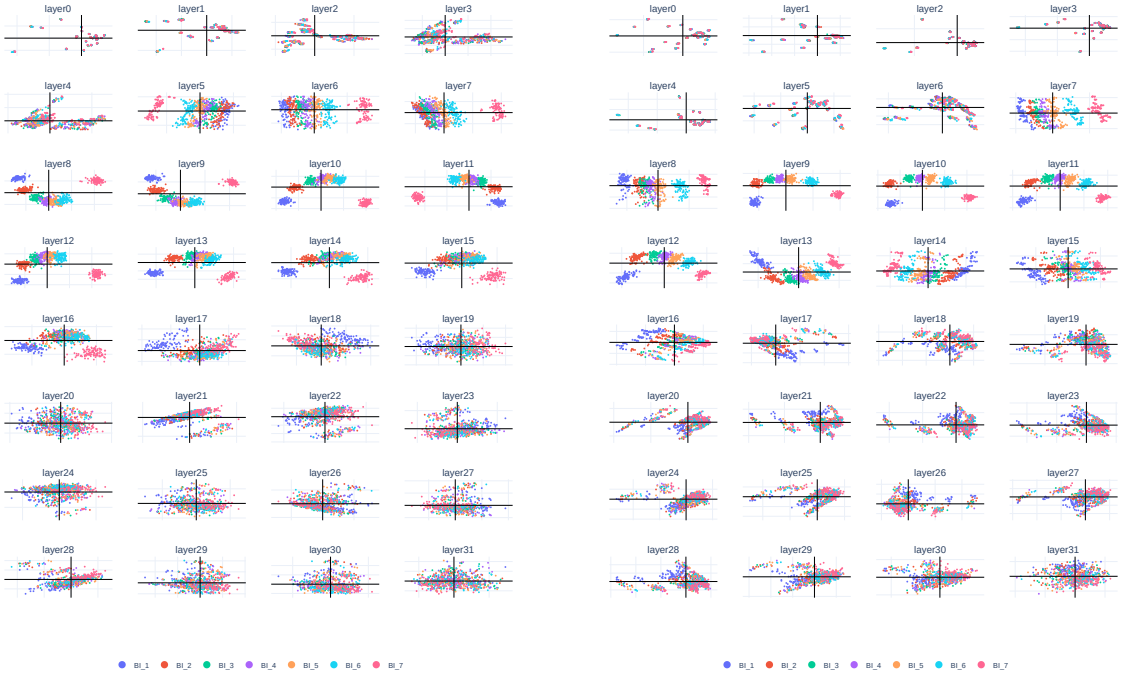
To localize the layer that contributes to binding behaviour, we perform layer-wise OI-PC based intervention mentioned in Section (§4.3) on our development set. In Figure 14, we can observe that OI subspace from middle layers (i.e., from layer7 to layer15, especially layer8) significantly affect the computation of binding, and interestingly, these layers also overlap with the ones that clearly encode OI information, as shown in Figure 2. Based on the analysis, we select the layer to perform activation patching.




(a) (b)

Figure 14: Average Logit Difference (LD) and logit flip for layer-wise OI-PC based intervention on Llama2-7B, where x axis denotes the layer, the colored zone indicates the layers that are sensitive to the intervention, and the vertical line represents the most effective layer (i.e., Layer 8), Y axis in Figure 14a and Figure 14b denotes the average LD and the proportion of inferred attributes (excluding the original one) respectively.

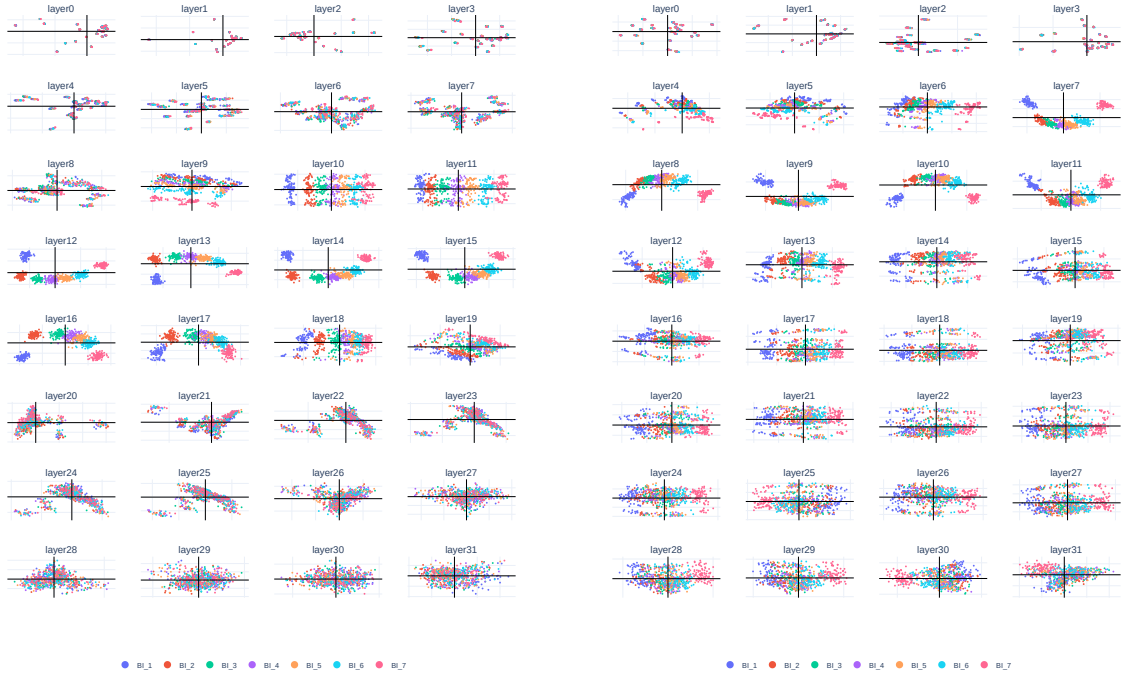
## A.5 Layer-wise Embedding Visualization on Llama3-8B and Float-7B




(a) Llama3-8B (b) Float-7B

Figure 15: Layer-wise OI subspace visualization on Llama3-8B and Float-7B.

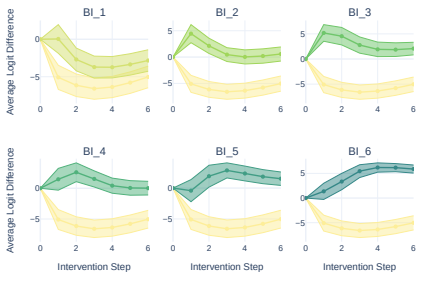
## A.6 OI Subspace on other LM Families



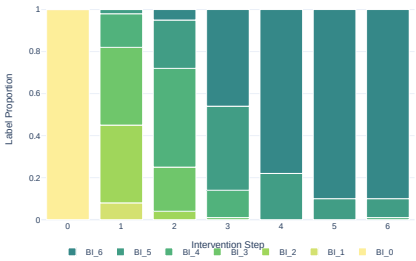

(a) Qwen1.5-7B

(b) Pythia-6.9B

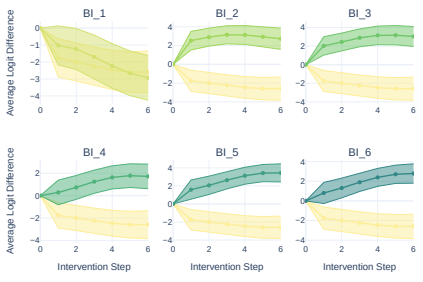
Figure 16: Layer-wise OI subspace visualization on Qwen1.5-7B and Pythia-6.9B.



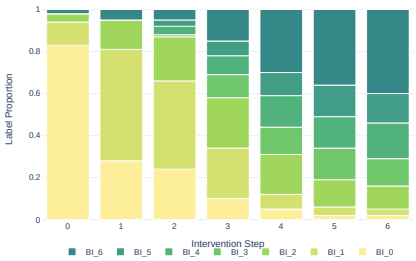

(a) LD for Qwen1.5-7B.




(b) LF for Qwen1.5-7B.




(c) LD for Pythia-6.9B.




(d) LF for Pythia-6.9B.

Figure 17: Logit Difference (LD) and Logit Flip (LF) for activation patching on the entity tracking dataset (i.e., r: is_in).

## A.7 Case Study on Llama2-7B

| Context | Query | Answer for # Step | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| The bugis sold by person Esta, the spawn is sold by person Fritz, the wine is sold by person Inga, the paste is sold by person Ward, the poison is sold by person Albert, the crow is sold by person Davis, the nest is sold by person Val . | Person Esta is selling the | spawn | wine | paste | poison | crow | nest |
| The virus is sold by person Anna, the fur is sold by person Earl, the pill is sold by person Flor, the bean is sold by person Roy, the spawn is sold by person Kam, the farm is sold by person Young, the sheep is sold by person Billy. | Person Anna is selling the | fur | pill | spawn | spawn | farm | sheep |
| The root is sold by person Carl, the mouse is sold by person Marco, the fruit is sold by person Luke, the bug is sold by person Paul, the grass is sold by person Inga, the pie is sold by person Pok, the cookie is sold by person George. | Person Carl is selling the | mouse | fruit | bug | grass | cookie | cookie |

Table 7: Attributes inferred by Llama2-7B as a result of directed activation patching along OI-PC on the dataset of "r: sell", where color denotes the BI.

| Context | Query | Answer for # Step | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| The carbon is applied by person Wei, the liquid is applied by person Season, the bath is applied by person Robert, the fog is applied by person Daniel, the heavy is applied by person Roma, the motor is applied by person Ara, the pool is applied by person Jorge | Person Wei applies the | liquid | bath | fog | motor | motor | pool |
| The rain is applied by person Kurt, the gauge is applied by person Jon, the dust is applied by person Newton, the jet is applied by person Dan, the floor is applied by person Alfred, the low is applied by person Mike, the basket is applied by person April | Person Kurt applies the | gauge | dust | jet | floor | basket | basket |
| The lamp is applied by person Angel, the bucket is applied by person Carl, the canvas is applied by person Bert, the cargo is applied by person Otto, the plain is applied by person Johnny, the floor is applied by person John, the heavy is applied by person Era. | Person Angel applies the | bucket | canvas | cargo | plain | floor | heavy |

Table 8: Attributes inferred by Llama2-7B as a result of directed activation patching along OI-PC on the dataset of "r: apply", where color denotes the BI.

| Context | Query | Answer for # Step | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| The lip is moved by person Mack, the tract is moved by person Sommer, the pen is moved by person Son, the tip is moved by person August, the bat is moved by person Monte, the socket is moved by person Marco, the hook is moved by person Paul. | Person Mack moved the | tract | pen | tip | bat | hook | hook |
| The mask is moved by person Jules, the timer is moved by person Ward, the bullet is moved by person Ana, the eye is moved by person Val, the button is moved by person Andy, the lock is moved by person Arnold, the colon is moved by person Betty. | Person Jules moved the | timer | bullet | button | lock | lock | colon |
| The mask is moved by person Cole, the neck is moved by person Donald, the pad is moved by person Beth, the cone is moved by person Jorge, the tail is moved by person Lou, the thread is moved by person Alfred, the toe is moved by person Edward. | Person Cole moved the | neck | pad | cone | tail | toe | toe |

Table 9: Attributes inferred by Llama2-7B as a result of directed activation patching along OI-PC on the dataset of "r: move", where color denotes the BI.

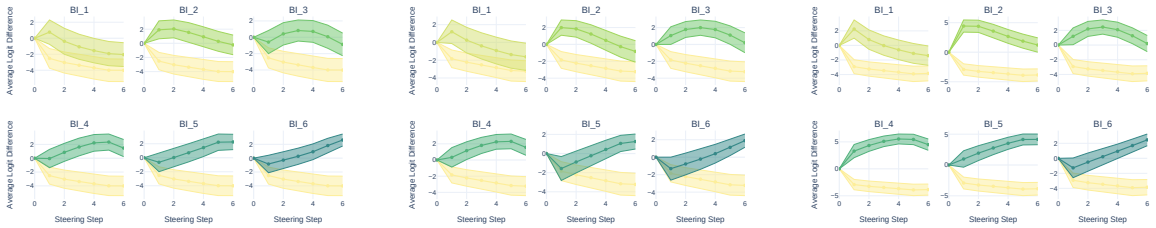
| Context | Query | Answer for # Step | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| The creature is brought by person Tam, the guitar is brought by person Frank, the dress is brought by person Stuart, the block is brought by person Victor, the brain is brought by person David, the coffee is brought by person Mack, the radio is brought by person Roger. | Person Tam brings the | guitar | dress | block | brain | coffee | radio |
| The boat is brought by person Luke, the pipe is brought by person Clara, the pot is brought by person Han, the bill is brought by person Chi, the milk is brought by person Scott, the card is brought by person Henry, the brick is brought by person Morris | Person Luke brings the | pipe | pot | bill | card | brick | brick |
| The fan is brought by person Van, the note is brought by person Clara, the block is brought by person Alex, the newspaper is brought by person Peg, the crown is brought by person Jan, the car is brought by person Pok, the magnet is brought by person Golden. | Person Van brings the | note | block | crown | car | magnet | magnet |

Table 10: Attributes inferred by Llama2-7B as a result of directed activation patching along OI-PC on the dataset of "r: bring", where color denotes the BI.

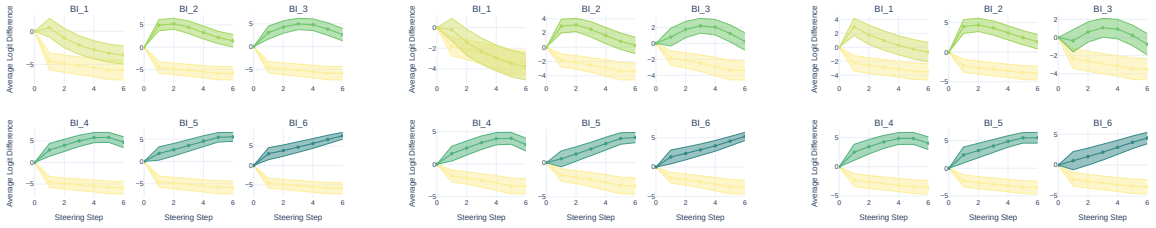
| Context | Query | Answer for # Step | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 |
| The load is pushed by person Mike, the atom is pushed by person Mira, the tin is pushed by person Juli, the stud is pushed by person Sam, the sedan is pushed by person Pia, the bath is pushed by person Leo, the growth is pushed by person Pat. | Person Mike pushes the | atom | tin | stud | bath | growth | growth |
| The mud is pushed by person Thomas, the heavy is pushed by person Ralph, the tile is pushed by person Pierre, the import is pushed by person Perry, the arm is pushed by person Robert, the lung is pushed by person Kurt, the cabin is pushed by person Ernest. | Person Thomas pushes the | heavy | tile | import | arm | cabin | cabin |
| The bed is pushed by person Fran, the lever is pushed by person Lan, the cord is pushed by person Paris, the vent is pushed by person Gene, the thumb is pushed by person Marie, the mouth is pushed by person Asia, the ear is pushed by person Lang. | Person Fran pushes the | lever | cord | vent | thumb | thumb | ear |

Table 11: Attributes inferred by Llama2-7B as a result of directed activation patching along OI-PC on the dataset of "r: push", where color denotes the BI.

## A.8 Activation Steering on Llama2-7B

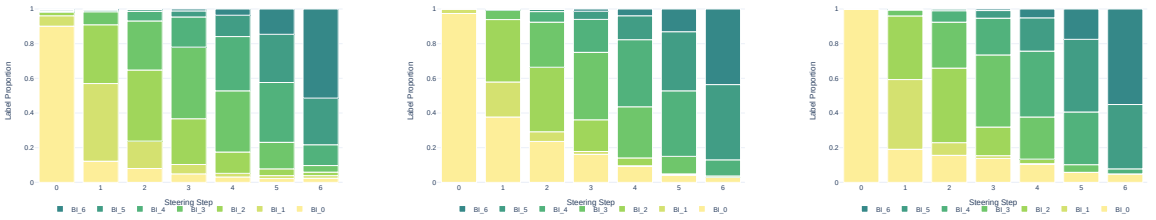



(a) r: is_in (b) r: sell (c) r: apply

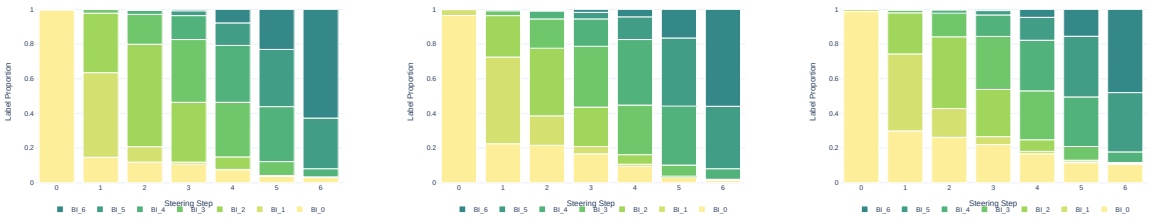



(d) r: move (e) r:bring (f) r: push

Figure 18: Logit Difference (LD) for OI subspace based activation steering across datasets on Llama2-7B, where x axis represents the intervention of $\mathbf{s}_{0\to bi}$ on the activation of $e_0$. Here, $l = 8$ and $\alpha = 1.25$.




(a) r: is_in (b) r: sell (c) r: apply

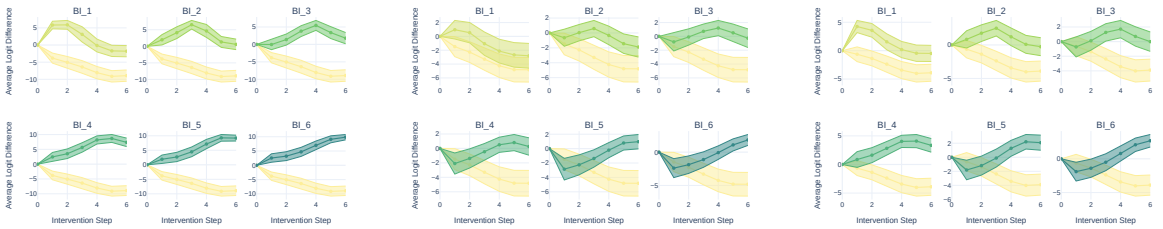



(d) r: move (e) r: bring (f) r: push

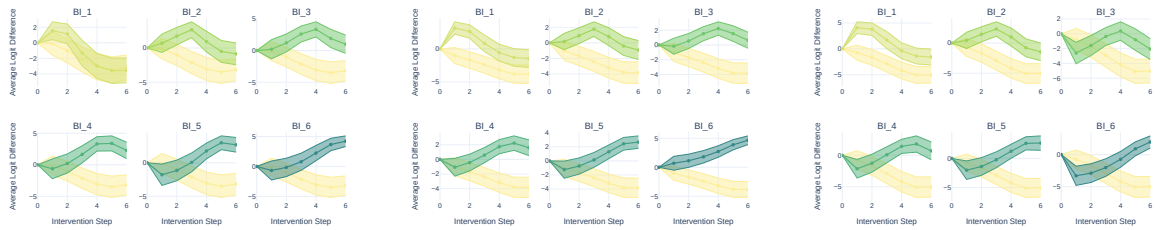
Figure 19: Logit flip for OI subspace based activation steering across datasets on Llama2-7B, where x axis represents the intervention of $\mathbf{s}_{0\to bi}$ on the activation of $e_0$.

### A.9 Results on Fine-Tuned LM

Kramár et al. (2024); Kim et al. (2024) claim that the code fine-tuned LM, such as Float-7B (Prakash et al., 2024) outperforms the pretrained LM on the entity tracking task (Kim and Schuster, 2023; Prakash et al., 2024). Since the code fine-tuned LM performs well on the entity tracking task that requires the OI subspace based computation, we hypothesize that OI subspace also exists in the code fine-tuned LM and the intervention along OI-PC will causally affect the model output. To prove the hypothesis, we conduct the intervention on Float-7B and show results in Figure 20 and Figure 21. We found that the OI subspace based intervention on Float-7B achieves the similar results as on Llama2-7B, indicating that the OI subspace not only exists in the pretrained LM but also in the fine-tuned one. In addition, adding the same step value (i.e., $v$) on Float-7B will achieve higher LD value than Llama2-7B, indicating that the code fine-tuned LM is more sensitive to the OI subspace based intervention. For instance, the maximum LD of $a_4$ in the former is around 10, and it is 2 times larger than the one in the latter, which is around 5. This might partially explains why the code fine-tuned LM performs better than the original one, because code fine-tuning might enhance the function of OI subspace so that it is more sensitive on the intervention and more effective on the in-context entity tracking task.

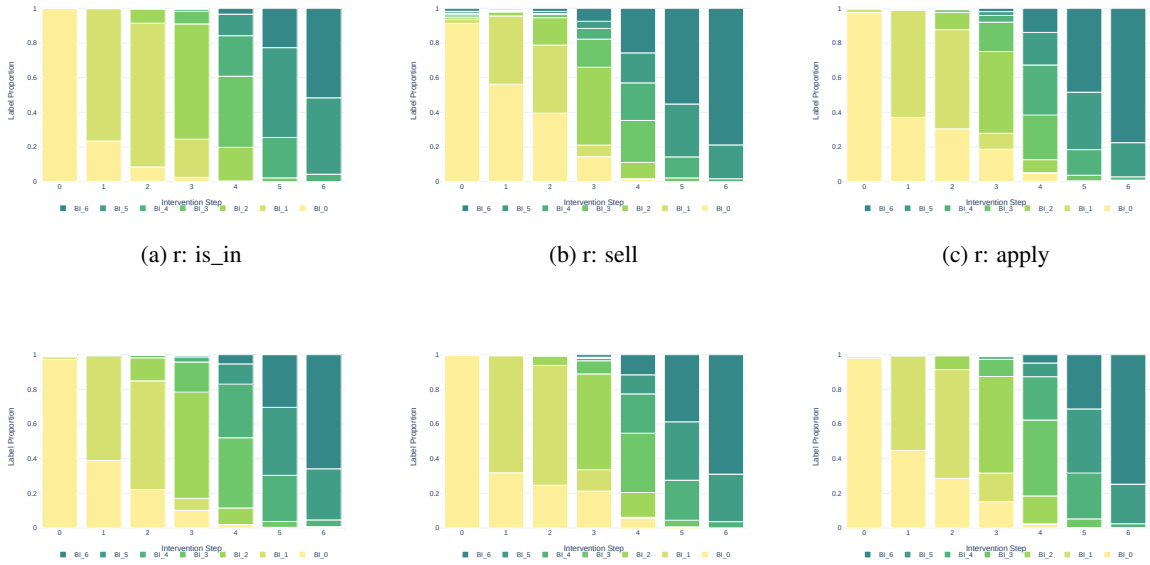


(a) r: is_in (b) r: sell (c) r: apply



(d) r: move (e) r:bring (f) r: push

Figure 20: Logit Difference (LD) for OI-PC based intervention across datasets on Float-7B, where x axis denotes the number of intervention steps on $e_0$, y axis does the LD, BI_i represents each target attribute and the light yellow bottom line indicates the LD of original attribute (i.e., $a_0$). Here, $l = 10$, $v = 2.55$, and $\alpha = 5.0$.

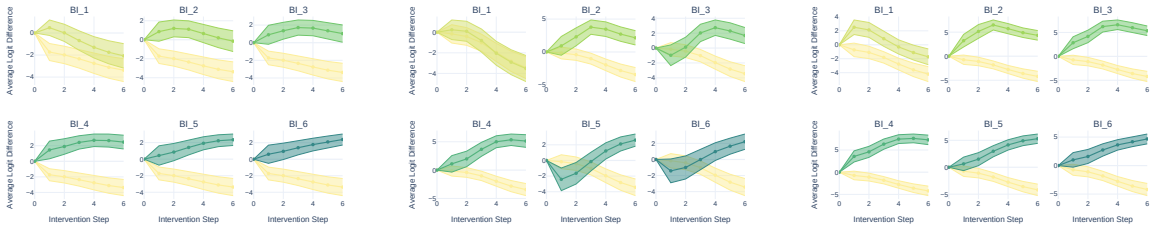



(a) r: is_in (b) r: sell (c) r: apply

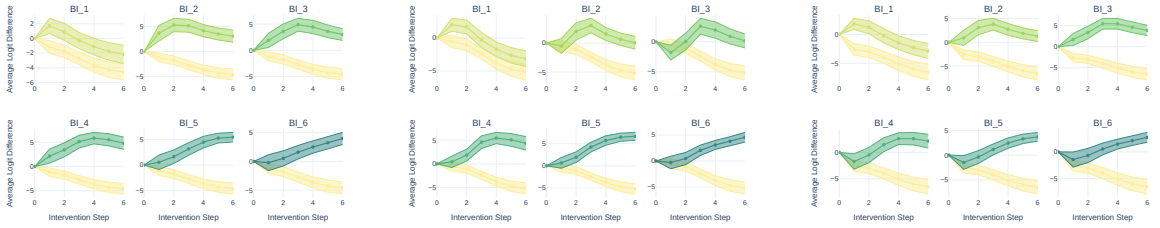
(d) r: move (e) r: bring (f) r: push

Figure 21: Logit flip for OI-PC based intervention across datasets on Float-7B, where x axis denotes the number of intervention steps on $e_0$, y axis does the proportion of each inferred attribute in model output.

## A.10 Activation Patching on Llama3-8B

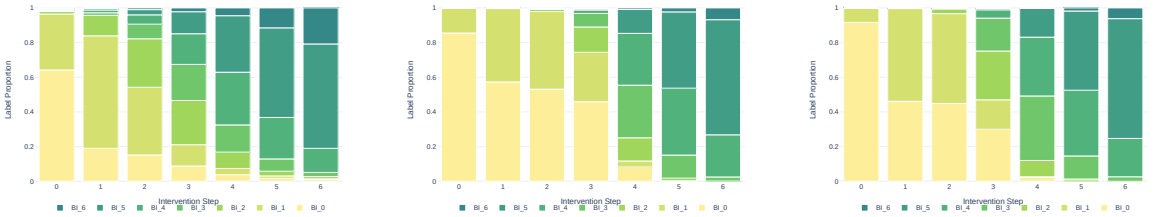



(a) r: is_in (b) r: sell (c) r: apply

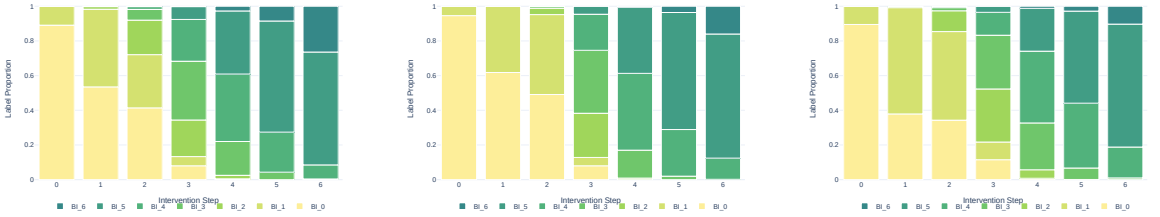



(d) r: move (e) r:bring (f) r: push

Figure 22: Logit Difference (LD) for OI-PC based intervention across datasets on Llama3-8B, where x axis denotes the number of intervention steps on $e_0$, y axis does the LD, BI_i represents each target attribute and the blue line indicates the LD of original attribute (i.e., $a_0$). Here, $l = 10$, $v = 0.65$, and $\alpha = 2.0$.




(a) r: is_in (b) r: sell (c) r: apply




(d) r: move (e) r: bring (f) r: push

Figure 23: Logit flip for OI-PC based intervention across datasets on Llama3-8B, where x axis denotes the number of intervention steps on $e_0$, y axis does the proportion of each inferred attribute in model output.

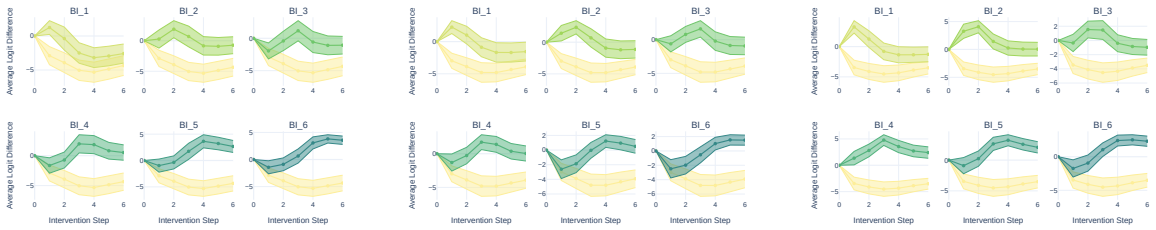
### A.11 New Dataset with Interjections

The dataset is created by inserting a sequence of interjections after the first attribute entity pair, as illustrated in Table 12. Since there is no BI information in the interjection (e.g., $j^{p3}$), adding it only changes the PI of its following entities and attributes. We set the number of interjections as that the PI of last interjection token is larger than the last PI of its original input (e.g., $pi > p5$ in Table 12).

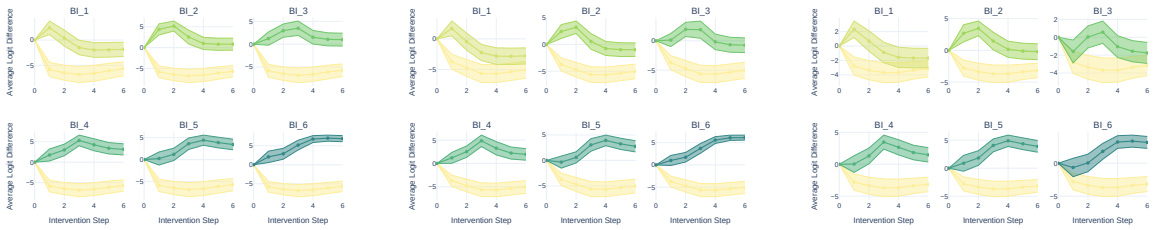
Based on this dataset, we conduct the same intervention on its OI subspace, as mentioned in Section (§4.3). The counter argument is that the subspace only captures Position Information (PI), and the intervening step only changes the PI information. Specifically, adding one unit of $v$ on $\overrightarrow{e_0^{p1}}$ might convert the PI of target entity from $p1$ to $p3$, and $p3$ is the PI of $e_1$, and its attribute is $a_1$, as shown in Table 12, and thus the LM swaps the answer from $a_0$ to $a_1$. If it is true, then the same intervention will not change the answer on the new dataset, because following the counter argument, after adding one unit of $v$ on $\overrightarrow{e_0^{p1}}$, the PI of target entity becomes $p3$, and $p3$ is the PI of $j^{p3}$ (i.e., an interjection token). The LM thus would not select $a_1$ as its answer. However, the results on Figure 24 and Figure 25 show that the subspace intervention on the new dataset achieves similar results as the original one, as shown in Figure 3 and Figure 4, proving the counter argument wrong and indicating the independence between OI subspace and PI.

| **Input (original)** |
|---|
| $a_0^{p0}\ r\ e_0^{p1}\ ,\ a_1^{p2}\ r\ e_1^{p3}\ ,\ a_2^{p4}\ r\ e_2^{p5}\ .\quad e_0^{p1}\ r^{-1}\ ?$ |
| **Input (with interjection)** |
| $a_0^{p0}\ r\ e_0^{p1}\ ,\ j^{p2}\ j\ j^{p3}\ ...j^{pi}\ a_1^{pi+1}\ r\ e_1^{pi+1}\ ...\quad e_0^{p1}\ r^{-1}\ ?$ |

Table 12: Simplified expression of original input and the one modified with a sequence of interjections, where $j^{p3}$ denotes an interjection $j$, such as "ah", with position of $p3$, which is also the position of $e_1^{p3}$ in the original input.

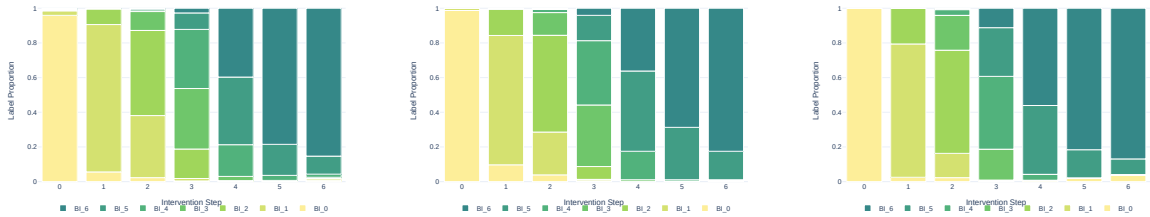


(a) r: is_in (b) r: sell (c) r: apply

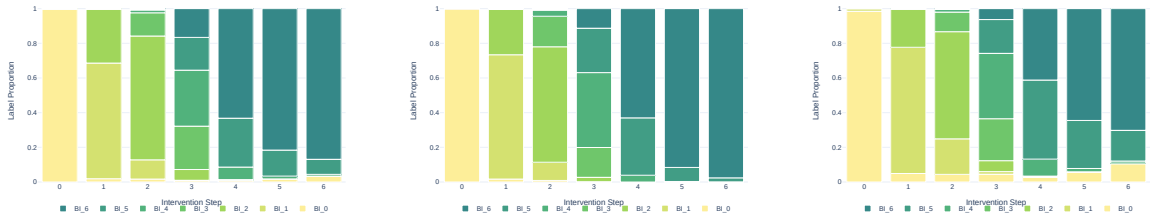


(d) r: move (e) r:bring (f) r: push

Figure 24: Logit Difference for activation patching on the dataset with interjections. Here, $l = 8$, $v = 2.5$, and $\alpha = 3.0$.




(a) r: is_in (b) r: sell (c) r: apply




(d) r: move (e) r: bring (f) r: push

Figure 25: Logit Flip for activation patching on the dataset with interjections.