Identifying Query-Relevant Neurons in Large Language Models for Long-Form Texts

Lihu Chen, Adam Dejl, Francesca Toni
Imperial College London
`firstname.surname@imperial.ac.uk`

Abstract

Large Language Models (LLMs) possess vast amounts of knowledge within their parameters, prompting research into methods for locating and editing this knowledge. Previous work has largely focused on locating entity-related (often *single-token*) facts in smaller models. However, several key questions remain unanswered: (1) *How can we effectively locate query-relevant neurons in contemporary autoregressive LLMs, such as Llama and Mistral?* (2) *How can we address the challenge of long-form text generation?* (3) *Are there localized knowledge regions in LLMs?* In this study, we introduce Query-Relevant Neuron Cluster Attribution (QRNCA), a novel architecture-agnostic framework capable of identifying query-relevant neurons in LLMs. QRNCA allows for the examination of long-form answers beyond triplet facts by employing the proxy task of multi-choice question answering. To evaluate the effectiveness of our detected neurons, we build two multi-choice QA datasets spanning diverse domains and languages. Empirical evaluations demonstrate that our method outperforms baseline methods significantly. Further, analysis of neuron distributions reveals the presence of visible localized regions, particularly within different domains. Finally, we show potential applications of our detected neurons in knowledge editing and neuron-based prediction.

1 Introduction

Large Language Models (LLMs) contain substantial amounts of knowledge within their neurons (or parameters). Recent research has focused on identifying and localizing these knowledge neurons to gain insights into the information processing mechanisms of LLMs. *Activation-based methods* (Voita, Ferrando, and Nalmpantis 2023) examine activation patterns to elucidate the role of neurons in the reasoning process. However, these methods often struggle to directly attribute specific outputs to corresponding inputs, thereby limiting their effectiveness in accurately identifying relevant knowledge. *Gradient-based methods* (Dai et al. 2022) measure the sensitivity of model outputs to internal components in response to specific inputs, which enables the effective identification of neurons relevant to particular queries. However, these methods typically employ fill-in-the-blank tasks, such as ``Paris is the capital of ___'', to ascertain correlations of *triplet facts*. *Causality-based methods* (Meng et al. 2022a) take a different approach by employing causal mediation

Methods	Long-Form Texts	Neuron-Level Location	Decoder Models	⩾ 7B LLMs
Knowledge Neuron (2022)	✗	✓	✗	✗
ROME (2022a)	✗	✗	✓	✗
Knowledge Subnetwork (2023)	✗	✓	✓	✗
QRNCA (Ours)	✓	✓	✓	✓

Table 1: Comparison of general-domain knowledge locating methods. Here, we do not consider task-specific approaches like Language Neuron (Chen et al. 2024b) and Privacy Neuron (Wu et al. 2023).

analysis to pinpoint *layers* within LLMs that store factual associations. Another branch of pioneering research attempts to locate functional regions in small-size language models such as BERT (Kenton and Toutanova 2019) and GPT-small (Radford et al. 2019), including linguistic regions (Zhang et al. 2024b), factual subnetworks (Ren and Zhu 2022; Bayazit et al. 2023), and modular structures (Zhang et al. 2023; Conmy et al. 2023).

While these studies successfully identify knowledge associations stored within LLMs, three significant questions remain underexplored: (1) How can we effectively locate query-relevant neurons in contemporary autoregressive LLMs, such as Llama (Touvron et al. 2023) and Mistral (Jiang et al. 2023), given their large model size and different architectures? (2) How can we address the challenge of long-form text generation, as previous methods have been limited to triplet facts? (3) Are there localized knowledge regions in LLMs analogous to the localized functional regions observed in human brains (Brett, Johnsrude, and Owen 2002)?

To address the first two questions, we introduce a novel framework named ***Query-Relevant Neuron Cluster Attribution (QRNCA)*** designed to identify query-relevant neurons in LLMs. The principal advantages of our framework are its architecture-agnostic nature and its capability of handling long-form text generation effectively, as shown in Table 1. QRNCA aims to extract Query-Relevant (QR) neurons for each input query. The process begins by transforming an open-ended generation task into a multiple-choice question-answering format. By employing prompt engineering, we constrain LLMs to generate only the option letter rather than the complete answer. This approach allows for the examination of long-form generation beyond single tokens. Subsequently, we adapt the Knowledge Attribution method (Dai
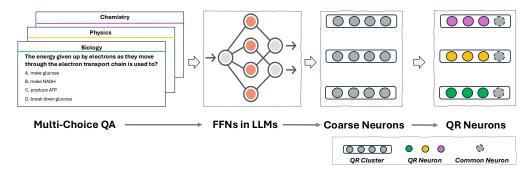
Figure 1: Our overall framework, which aims to detect Query-Relevant (QR) neurons with regard to specific queries.

et al. 2022) to compute *Neuron Attribution*, which elucidates the relationship between neurons and the input query. We then gather clusters for a series of queries and calculate the *Inverse Cluster Attribution*. This step mitigates the influence of neurons that recur across clusters (or queries). The final step involves multiplying the neuron attribution and inverse cluster attribution values to pinpoint correlated neurons. Additionally, we identify certain *Common Neurons* that are associated with common words, punctuation marks, and option letters. Excluding these common neurons enhances the detection of QR neurons. Empirical evaluations demonstrate that our proposed method outperforms baseline approaches.

To investigate the existence of localized knowledge regions, we construct two multi-choice QA datasets encompassing various *domains* and *languages*. Then, we visualize the geographical locations of the detected neurons in Llama. Our findings indicate that distinct localized regions emerge in the middle layers, particularly for domain-specific neurons. This suggests that LLMs tend to complete the formation of domain-specific concepts within these middle layers. Conversely, language-specific neurons are more sparsely distributed, indicating that LLMs likely draw on linguistic knowledge at all processing levels. Additionally, we observed that common neurons are concentrated in the top layer, predominantly expressing frequently used tokens.

In summary, our main contribution is four-fold: (1) **A scalable method:** we propose QRNCA to detect query-relevant neurons in LLMs; the QRNCA method is architecture-agnostic and can deal with long-form generations. (2) **Two new datasets:** we curate two multi-choice QA datasets that contain different types of knowledge, namely Domain Knowledge and Language knowledge. (3) **In-depth studies:** we visualize distributions of detected neurons and we are the first to show that there are visible localized regions in Llama. (4) **Potential applications:** we show that QRNCA might be useful for knowledge editing and neuron-based prediction.

2 Related Work

2.1 Locating Knowledge in LLMs

Large Language Models contain a vast range of knowledge within their parameters, spanning factual (Petroni et al. 2019; Zhou et al. 2020; Jiang et al. 2020; Roberts, Raffel, and Shazeer 2020; Pezeshkpour 2023), linguistic (Liu

et al. 2019; Jawahar, Sagot, and Seddah 2019; Chen, Varoquaux, and Suchanek 2023), and domain-specific information (Sung et al. 2021; Frieder et al. 2024). Recent mechanistic studies suggest that knowledge is primarily stored in the FFNs (Feed-forward Networks) layers of Transformers (Geva et al. 2021, 2022), which prompts ongoing research efforts aimed at developing methods to identify and locate this knowledge within these layers. *Activation-based methods* (Voita, Ferrando, and Nalmpantis 2023; Gurnee et al. 2024) investigate the activation patterns of neurons to interpret how the network processes information at different stages. However, a key limitation of these methods is their inability to directly attribute the model's output to specific inputs, which limits their precision in identifying relevant knowledge. *Gradient-based methods* (Ancona et al. 2019; Dai et al. 2022), on the other hand, offer fine-grained attribution by quantifying the sensitivity of model outputs to internal components in response to a given input. This approach effectively identifies neurons relevant to specific queries. Nonetheless, current gradient-based techniques have primarily focused on single-token triplet facts. Another approach, *Causality-based methods*, employs causal mediation analysis to discern the particular layers associated with a given factual input (Meng et al. 2022a). This line of research has evolved into a locate-and-edit paradigm, aimed at refining knowledge within LLMs (Meng et al. 2022b; Ju and Zhang 2023; Zhang et al. 2024a). In addition to general knowledge locating approaches, recent studies have focused on identifying neurons responsible for specific tasks, such as linguistic (Chen et al. 2024b; Tang et al. 2024; Kojima et al. 2024), privacy-related (Wu et al. 2023; Chen et al. 2024a) and bias-related neurons (Yang, Kang, and Jung 2023).

In this work, we propose a novel gradient-based attribution method aimed at locating input-output knowledge within LLMs. Unlike previous methodologies, our approach mainly focuses on long-form texts beyond entity facts.

2.2 Analyzing Knowledge Distribution in LLMs

Given the human-like reasoning capabilities observed in LLMs across various tasks (Zhao et al. 2023), and since our brain contains functional locations associated with distinct cognitive processes (Brett, Johnsrude, and Owen 2002; Bjaalie 2002; Gholipour et al. 2007), we ask whether there are similar regions in LLMs. Previous investigations have

explored the behaviors of individual neurons indicating that a neuron can encode multiple concepts (Bolukbasi et al. 2021) while a concept can also be distributed across multiple neurons (Dalvi et al. 2019; Durrani et al. 2020; Chen et al. 2024b). Subsequent endeavors have sought to identify functional regions in LLMs, encompassing linguistic regions (Zhang et al. 2024b), factual subnetworks (Ren and Zhu 2022; Bayazit et al. 2023), and modular structures (Zhang et al. 2023; Conmy et al. 2023). These studies have investigated localized behaviors in smaller-scale language models, such as BERT and GPT-small. Building upon these foundations, our research embarks on the examination of knowledge locations in larger-size LLMs, specifically those with 7B parameters, spanning multiple knowledge domains.

3 Background

Feed-forward Networks in LLMs Feed-forward networks (FFNs) are widely used by transformer-based language models. Geva et al. (2021) reveal that FFNs emulate key-value memories and their outputs are responsible for refining the final output distribution over the vocabulary. Although traditional two-layer FFNs in BERT (Kenton and Toutanova 2019) and GPT-2 (Radford et al. 2019) have been studied well, the behaviors of FFNs in modern LLMs such as Llama (Touvron et al. 2023) and Mistral (Jiang et al. 2023), are not well-explored. These LLMs adopt Gated Linear Units (GLUs) (Dauphin et al. 2017) in their FFNs, which can be formulated as follows:

$$\text{FFN}(\mathbf{X}) = (\mathbf{X}\mathbf{W}^U \odot \text{SiLU}(\mathbf{X}\mathbf{W}^G)) \mathbf{W}^D \qquad (1)$$

Here, $\mathbf{X} \in \mathbb{R}^{n \times d}$ is the input sequence, n is the number of tokens and d is the dimension of input vectors; $\mathbf{W}^U \in \mathbb{R}^{d \times m}$, $\mathbf{W}^G \in \mathbb{R}^{d \times m}$, $\mathbf{W}^D \in \mathbb{R}^{m \times d}$ are parameter matrices, m is the hidden dimension of the FFN and $\odot$ is the Hadamard product; finally SiLU (Elfwing, Uchibe, and Doya 2018) is the activation function.

Knowledge Neurons Dai et al. (2022) propose a gradient-based *Knowledge Attribution* to identify the knowledge neurons in BERT by using the fill-in-the-blank cloze task. Their method evaluates the contribution of each neuron in FFNs to the knowledge predictions. Given a prompt q (''Paris is the capital of ＿＿＿''), the probability of the correct answer predicted by an LLM can be formulated as:

$$P_q(\hat{w}_i^l) = p(y^*|q, w_i^l = \hat{w}_i^l) \qquad (2)$$

where y^* is the correct answer (France); w_i^l denotes the i-th intermediate neuron in the l-th layer in FFNs; $\hat{w}_i^l$ is a constant we assign to w_i^l.

In order to measure the attribution score (or contribution) of a neuron, they gradually change the w_i^l from 0 to its original value computed during the forward pass and integrate the gradients (Sundararajan, Taly, and Yan 2017):

$$\text{Attr}(w_i^l) = \bar{w}_i^l \int_{\alpha=0}^{1} \frac{\partial P_q(\alpha \bar{w}_i^l)}{\partial w_i^l} d\alpha \qquad (3)$$

where $\frac{\partial P_q(\alpha \bar{w}_i^l)}{\partial w_i^l}$ is the gradient with regard to w_i^l. Attr$(\cdot)$ accumulates the output probability change as α gradually varies from 0 to 1. The attribution measures the contribution of the neuron w_i^l to the correct answer. In practice, the score is estimated by using Riemann Approximation:

$$\hat{\text{Attr}}(w_i^l) = \frac{\bar{w}_i^l}{m} \sum_{k=1}^{m} \frac{\partial P_q(\frac{k}{m} \bar{w}_i^l)}{\partial w_i^l} \qquad (4)$$

where m is the number of the estimation steps. Finally, they identify a coarse set of knowledge neurons whose attribution scores are greater than a threshold t.

4 Locating Query-Relevant (QR) Neurons in Autoregressive LLMs

While Knowledge Attribution (Dai et al. 2022) effectively identifies neurons linked to factual queries, its applicability is limited to encoder-only architectures, and it mandates the output to be a single-token word. To address these constraints, we propose a new framework named Query-Relevant Neuron Cluster Attribution (QRNCA). The framework is architecture-agnostic and capable of handling long-form generation.

To clarify the main concepts in our framework, we provide the following key notions: **QR Neuron** is an individual neuron highly correlated with a specific query, capable of influencing the corresponding knowledge expression. Our primary objective is to identify a set of QR neurons for a given input. **QR Cluster** represents a coarse grouping of neurons associated with a specific query. This cluster may include noisy neurons and require further refinement. **Common Neuron** is consistently activated by a wide range of inputs, representing general knowledge or concepts.

The overall framework is shown in Figure 1. Our framework first resorts to the proxy task of *Multi-Choice QA* to deal with long-form texts. Starting with a given query, the framework employs *Neuron Attribution* to derive a QR Cluster. Each neuron within this cluster is assigned an attribution score that indicates its relevance to the query. Next, we apply *Inverse Cluster Attribution* to attenuate the influence of neurons that frequently occur across multiple clusters. Finally, we identify *Common Neurons*, as those lacking discriminative power in determining query relevance and representing common knowledge or concepts. Refining the extraction of QR neurons by excluding these common neurons enhances the precision in identifying critical neural correlates.

In the following paragraphs, we introduce the details of these key components in our framework: Multi-Choice QA Transformation, Neuron Attribution, Inverse Cluster Attribution, and Common Neurons.

4.1 Multi-Choice QA Transformation

Multi-choice QA is widely used in a variety of real-world educational assessments and standardized tests. Meanwhile, many known benchmarks such as MMLU (Hendrycks et al. 2020) and Big-bench (Srivastava et al. 2023) use multi-choice QA to evaluate the breadth and depth of a model's knowledge. Therefore, we adopt the proxy task of multi-choice QA to study the knowledge associations in LLMs.

Given the biology question *"The energy given up by electrons as they move through the electron transport chain is used to?"*, the correct answer can be the long-form text *"produce ATP"*. To deal with long-form answers, we advocate for the transformation of questions and their corresponding answers into a multiple-choice framework, as illustrated in Figure 1. This approach involves the generation of incorrect options by randomly sampling answers within the same domain. Following this, the LLM is prompted to produce only the option letter. Subsequently, we investigate the neurons correlated with the input query. To mitigate the impact of randomness, we devise multiple prompt templates and systematically shuffle the order of options to prevent the model from learning spurious correlations based on option letters. These prompt templates are detailed in Table A3 in the Supplementary Material (SM).

4.2 Neuron Attribution

To extend our methodology to Gated Linear Units (GLUs), which comprise two linear transformations followed by a gating mechanism, we adapt the Knowledge Attribution approach (Eq 5). In GLUs, the linear transformations involve computing a linear combination of input features, denoted by $f = \mathbf{X}\mathbf{W}^U$. Additionally, the gating mechanism, represented by $g = \mathrm{SiLU}(\mathbf{X}\mathbf{W}^G)$, determines the extent to which each input component should be forwarded, thereby enabling the model to emphasize important features while suppressing irrelevant ones. To compute the relevant attribution, we can use either $\frac{\partial P_q}{\partial f}$ or $\frac{\partial P_q}{\partial g}$ and we choose to use the former since our empirical study shows it can obtain better QR neurons (see details in Figure A5 in the SM). Given a query q, instantiation using our templates yields a query set $\mathcal{Q} = \{q_1, q_2, \ldots, q_{|\mathcal{Q}|}\}$, and the attribution score of the neuron n_i^l can be denoted as:

$$\mathrm{na}(n_i^l) = \frac{\sum_{j=1}^{|\mathcal{Q}|} \frac{\bar{f}_i^l}{m} \sum_{k=1}^{m} \frac{\partial P_{q_j}(\frac{k}{m}\bar{f}_i^l)}{\partial f_i^l}}{Z} \tag{5}$$

Here, the numerator means that we sum up the scores of different instantiated templates together as the initial attribution score. The denominator Z is the normalized factor by summing the initial attribution scores of all neurons. Since the number of prompts for each query may be varied and the initial attribution scores may be scaled differently, we use normalization to make the attribution scores comparable across queries.

4.3 Inverse Cluster Attribution

With the attribution score, we can obtain a list of coarse clusters for each query $\mathcal{C} = \{c_1, c_2, \ldots, c_{|\mathcal{C}|}\}$, where c is a cluster that consists of neurons whose attribution score is higher than some threshold t. The frequent appearance of some neurons across queries of different fields reveals that they are not critical neurons to the input query. To decrease their impact, we calculate the inverse cluster attribution:

$$\mathrm{ica}(n_i^l) = \log \frac{|\mathcal{C}|}{|\{c : c \in \mathcal{C} \text{ and } n_i^l \in c\}| + 1} \tag{6}$$

4.4 Common Neurons

We observe that some neurons with a relatively high attribution score are still shared across clusters. Through case studies (as shown in Table 4), we demonstrate that they express commonly used concepts such as option letters (``A'' and ``B'') or stop words (``and'' and ``the''). Therefore, we count the frequency of each neuron across clusters. If the frequency is higher than the $u\%$ of total clusters, we assign the given neuron into the common neuron set.

4.5 Obtaining QR Neurons

Given a query, the final score of a neuron can be computed as:

$$\mathrm{naica}(n_i^l) = \mathrm{na}(n_i^l) \times \mathrm{ica}(n_i^l) \tag{7}$$

We select top-v neurons with the highest score from the detected cluster and further remove common neurons to refine the QR neuron set.

5 Analyzing Detected QR Neurons

5.1 Experimental Settings

Dataset Construction We construct two datasets to locate knowledge neurons that cover two different categories: *subject domains and languages*.

Domain Dataset is derived from MMLU (Hendrycks et al. 2020), a multi-choice QA benchmark designed to evaluate models across a wide array of subjects with varying difficulty levels. The subjects encompass traditional disciplines such as mathematics and history, as well as specialized fields like law and ethics. In our study, we select six high school exam subjects from the test set: *Biology, Physics, Chemistry, Mathematics, Computer Science*, and *Geography*.

Language Dataset is adapted from Multilingual LAMA (Kassner, Dufter, and Schütze 2021), which is a dataset to investigate knowledge in language models in a multilingual setting covering 53 languages. We select six languages: *Arabic, English, French, Japanese, Russian* and *Chinese*. Each language subset includes queries that cover five different relations: `birth_place`, `employer`, `instrument`, `headquarters_location`, and `host_country`.

To mitigate sensitivity to prompts and option orders, each query is instantiated with multiple distinct templates (as shown in Table A3 in the SM), and the option orders are shuffled each time. The statistics of our datasets are shown in Table 2 and examples can be found in Table A4 in the SM.

Metric We modify the values of neurons to observe their impact on knowledge expression. For each query, we record the percentage change in the probability of the correct answer, thereby assessing the extent to which the QR neurons influence the predictions of LLMs. We compare our approach to other baseline methods and include a control group to determine whether the same detected neurons affect the predictions of randomly selected queries from unrelated fields (*Unrelated*). The Probability Change Ratio (PCR) for a dataset is calculated by $\frac{|\text{Related}|}{|\text{Unrelated}|}$, where $|\text{Related}|$ and $|\text{Unrelated}|$ means the average probability change of the related and unrelated samples, respectively. A higher value of

Domain	Bio	Phys	Chem	Math	CS	Geo	Total
Num	100	100	100	100	52	100	552
Language	Ar	En	Fr	Ja	Ru	Zh	Total
Num	100	100	100	100	100	100	600

Table 2: Statistics of our constructed datasets.

	Domain		Language	
Method	⇑ Boost	⇑ Suppress	⇑ Boost	⇑ Suppress
Random Neuron	1.0	0.55	2.0	1.0
Activation	1.0	1.0	1.1	1.1
Knowledge Neuron*	1.0	1.0	6.7	1.8
QRNCA *wo/* ICA	2.5	1.1	6.5	2.2
QRNCA *w/* Common Neuron	2.8	1.8	10.4	8.5
QRNCA	**4.4**	**5.6**	**41.2**	**36.0**

Table 3: Comparisons of different knowledge locating methods for Llama-2-7B. The metric here is the Probability Change Ratio (PCR) described in Section 5.1. Details are shown in Table A2 in the appendix.

PCR shows detected neurons can have a higher influence on the query, which is an indicator of better neurons (Chen et al. 2024b).

Baselines We compare our QRNCA to other neuron-level baselines[1]: **Random Neuron** are randomly selected from FFNs and we make sure they have the same number of neurons as QRNCA; **Activation** selects neurons with high activated values. **Kowledge Neuron*** is adapted from knowledge attribution (Dai et al. 2022) by using the multi-choice QA task; **QRNCA *wo/* ICA** only uses neuron attribution (Eq 5) to obtain relevant neurons, which dose not involve the computation of Inverse Cluster Attribution; **QRNCA *w/* Common Neuron** is a variant without removing common neurons.

Implementations We mainly study the knowledge neurons in Llama-2-7B (Touvron et al. 2023) and we use the instruction-tuned version so that the model is more responsive to our prompts. Llama-2-7B consists of 32 layers with the FFN hidden dimension of 11008. Besides, we also conduct experiments for Mistral-7B (Jiang et al. 2023) to validate whether our method can obtain consistent findings over different models. Note that our framework can be easily extended to larger-size LLMs.

As for the used hyper-parameters, the number of estimation steps was set to $m = 16$ and the attribution threshold t to 0.2 times the maximum attribution score. The template number was $|\mathcal{Q}| = 3$, the frequency u for obtaining common neurons was 30%, and the top-v for select coarse neurons was 20. We ran all experiments on three NVIDIA-V100. It took 120 seconds on average to locate neurons for a query with three prompt templates. For each domain and language,

[1]We do not compare to ROME (Meng et al. 2022a) since it locates layers instead of neurons. Also, we do not compare to task-specific methods.

the average number of detected QR neurons is between 12 and 17 (as shown in Table A1 in the SM).

5.2 Statistics of Detected QR Neurons

We are curious about the distribution of different knowledge storage in neurons: *Do different categories of knowledge share neurons?* To this end, we study the overlap rate. First, we aggregate detected neurons of all queries in a domain or language. Next, the rate is obtained by counting the number of shared neurons between different domains or languages. Figure 2a illustrates the overlap rates among different domains and languages. We observe that interdisciplinary or interconnected languages share a higher overlap rate such as (*geography, biology*) and (*Chinese, Japanese*), which is in line with our intuition. A surprising finding is that domains have higher overlap rates than languages, which indicates that LLMs tend to allow the storage of multiple domain-specific concepts in a single neuron (polysemantic). Although language-specific neurons are not monosemantic (Chen et al. 2024b), they prefer to encode one specific language concepts, which is also consistent with recent findings (Tang et al. 2024).

Regarding layer distribution, the QR neurons are predominantly located in the middle layers (15-18) and the top layers (around 30), as depicted in Figure 2b. This finding indicates knowledge concepts are mainly stored in the middle and top layers, and we may only modify these neurons for efficient knowledge updating (Ding et al. 2023).

5.3 QR Neurons Can Impact the Knowledge Expression

To validate the impact of our identified QR neurons, we replicate the experiments by Dai et al. (2022), updating the values of QR neurons using two methods: given a query and the value of $\bar{f}_i^l$, we either (1) boost the neurons by doubling the value $f_i^l = 2 \times \bar{f}_i^l$; or (2) suppress the neuron by making $f_i^l = 0$. After one operation, we record the PCR on a specific dataset to show the quality of these neurons.

Table 3 presents the overall performance of various methods. Our QRNCA method consistently outperforms other baselines, evidenced by its higher PCR. This indicates that our identified QR neurons significantly affect the probability of correct answers while exerting a relatively low impact on unrelated queries. For instance, our method achieves a boosting ratio of 41.2 on the language dataset, the highest among the baselines. Additionally, both our proposed ICA and the removal of common neurons can bring benefits to locating neurons, which is validated by the worse performance of the two QRNCA variants.

Furthermore, Figure 3 illustrates the percentage change in probability for each domain and language after boosting neuron values. Again, we can clearly observe the effectiveness of our detected QR neurons. Additionally, we performed supplementary experiments on Mistral-7B. The results, presented in Figure A4 in the SM, consistently support our conclusions.

(a) Overlap Rate

(b) Layer Distribution

Figure 2: Overlap rates and layer distributions of found QR neurons.
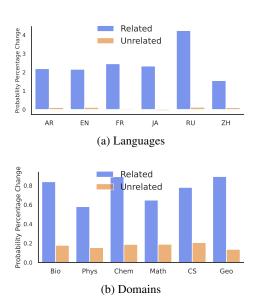

(a) Languages

(b) Domains

Figure 3: The correct probability percentage change by boosting QR neurons. The LLM here is Llama-2-7B (Touvron et al. 2023). The suppression results are shown in Figure A3 in the SM.

5.4 Are There Localized Regions in LLMs?

Given our ability to identify QR neurons for each query, it is intriguing to explore whether LLMs exhibit localized regions for each domain or language, analogous to the functional localizations in the human brain (Brett, Johnsrude, and Owen 2002). To investigate this, we visualize domain- or language-specific neurons on a 2D geographical heatmap. The width of the heatmap corresponds to the dimension of FFNs in Llama-2-7B (11008), and the length represents the layer depth (32). We accumulate the value of $\text{naica}(n_i^l)$ to populate the heatmap. Figure 4 displays the geographical locations of QR neurons in Llama-2-7B across various academic domains and languages. The distribution of QR neurons appears sparse but with distinct regions, particularly for different domains. Notably, certain regions are visible in the middle layers (10-15), suggesting specific neuron patterns. In contrast, language neurons are more sparsely distributed

with smaller regions, and languages like Arabic and Russian exhibit less localized properties.

Based on prior studies, LLMs process and represent information in a hierarchical manner (Geva et al. 2022; Wendler et al. 2024; Tang et al. 2024). The early layers are primarily responsible for extracting low-level features , while the middle layers begin to integrate this information, forming more complex semantic representations. The late layers are typically dedicated to generating the final output. Therefore, we suppose that domain-specific knowledge representation is built in the middle layer and the top layers are then mainly responsible for next-token prediction, which may explain the visible regions for different subject domains. Regarding language-specific neurons, their role in accessing linguistic knowledge across different layers likely accounts for their more sparse and distributed locations. This distribution reflects the necessity of engaging with language-specific neurons at multiple stages of information processing.

5.5 The Function of Common Neurons

To gain insights into the function of common neurons, we project the matrix $\mathbf{W}^D$ in Equation 1 to the vocabulary space and select the top-k tokens with the highest probability. Table 4 lists the predicted tokens, which include common words, punctuation marks, and option letters. These findings reinforce the notion that common neurons are not critical for specific queries. We also visualize their locations within Llama-2-7B and we observe that they tend to appear at the top layer (as shown in Figure A2 in the SM).

We also analyzed the token predicted by QR neurons, but we found that middle-layer neurons do not have a clear semantic meaning and human-readable concepts mostly appear in the top layer (Wendler et al. 2024). In Section A in the SM we conduct semantic meaning analyses of neurons.

6 Potential Applications

We provide two usage examples to showcase the potential applications of our detected QR neurons: *Knowledge Editing* and *Neuron-Based Prediction*.
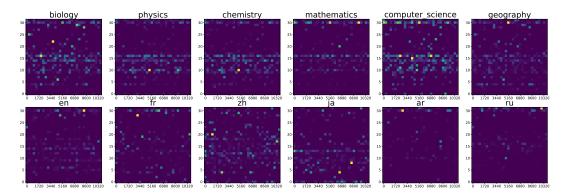
Figure 4: Geographical heatmap of detected QR neurons for different domains and languages. The value is calculated by our $naica(n_i^l)$. Brighter colors indicate higher naica values. The LLM here is Llama-2-7B (11008×32) (Touvron et al. 2023)

Neuron	Top-k tokens
n_{2725}^{31}	_in, _and, _to, _for, _today, _at, _as
n_{10676}^{31}	_July, _June, _March, _April, _November
n_{10075}^{30}	., _, (, :,), [, −
n_{5202}^{31}	_respectively, _while, _and
n_{5778}^{31}	_C, C, _c, c, '_ced'
n_{7670}^{31}	_B, B, _Bill, _Bh, '_Bureau'

Table 4: Tokens predicted by the common neurons.

Method	Domain		Language	
	Boost Δ (%)	Suppress Δ (%)	Boost Δ (%)	Suppress Δ (%)
Random Neuron	0.0	0.3	0.2	0.3
Activation	0.0	0.1	0.0	0.3
Knowledge Neuron*	1.4	3.8	14.3	16.0
QRNCA	**12.6**	**18.2**	**16.6**	**24.8**

Table 5: Successful rates of knowledge editing. Δ measures how well we can flip the predictions (*correct → incorrect* or vice versa).

6.1 Knowledge Editing

Apart from using the metric of PCR in Section 5.3, we are also interested in whether the detected QR neurons can be used for knowledge editing. For this goal, we adjust the values of QR neurons by either boosting or suppressing them to determine if we can change the prediction of a query from incorrect to correct or vice versa. Table 5 presents the success rates of knowledge editing on our constructed language datasets. Our observations indicate that QRNCA achieves higher success rates than other baselines.

6.2 Neuron-Based Prediction

The intuition behind neuron-based prediction is that for a domain-specific question, if the corresponding localized regions are properly activated, the LLM is more likely to generate truthful answers. Otherwise, the LLM may produce hallucinated answers. To this end, we test whether the correct answers to domain-specific questions can be predicted solely based on the activity of the associated neurons. Since we harvest QR neurons for queries in different subject domains, we can group all neurons for a domain to obtain a set of *domain-specific neurons*. We experiment on a specifically constructed MMLU (Hendrycks et al. 2020) validation

Method	Biology	Chemistry	Geography
Random guess	0.25	0.25	0.25
Prompt-based model pred.	0.96	0.71	0.89
Neuron-based pred.	0.96	0.67	0.89

Table 6: Accuracy of neuron-based prediction on selected domains in comparison with the standard prompt-based model prediction.

set with a different set of questions than those used to determine the QR neurons (see Section B in the SM for details on our experimental strategy). The results are summarised in Table 6. We observe that the accuracy of the neuron-based predictions is very close to the accuracy of the prompt-based method of using the entire model (the used templates are shown in Table A3 in the SM). This suggests that the activity of identified neurons can reflect the model's reasoning process to some extent. Investigating how this finding could be leveraged in applications like fact-checking and hallucination detection presents a promising line of future work.

7 Conclusion

In this study, we introduce a novel framework, QRNCA, for identifying query-relevant neurons in contemporary autoregressive language models. QRNCA leverages a multi-choice QA proxy task to address the complexity of long-form answers, extending beyond triplet facts. Meanwhile, it adopts strategies of inverse cluster attribution and common neuron removal to refine QR neurons. To validate our approach, we curate two datasets encompassing diverse domains and languages. Our experimental results show that our method outperforms existing baselines in identifying associated neurons. Additionally, this study pioneers the exploration of localized knowledge regions in LLMs and demonstrates Llama contains knowledge-specific regions in the middle layers while language-specific neurons tend to be distributed across different layers. Further, we prototype two potential usages of identified neurons in applications such as knowledge editing and neuron-based prediction. We hope that our findings are beneficial for further research in understanding the knowledge mechanisms underlying LLMs.

References

Ancona, M.; Ceolini, E.; Öztireli, C.; and Gross, M. 2019. Gradient-based attribution methods. *Explainable AI: Interpreting, explaining and visualizing deep learning*, 169–191.

Bayazit, D.; Foroutan, N.; Chen, Z.; Weiss, G.; and Bosselut, A. 2023. Discovering knowledge-critical subnetworks in pretrained language models. *arXiv preprint arXiv:2310.03084*.

Bjaalie, J. G. 2002. Localization in the brain: new solutions emerging. *Nature reviews neuroscience*, 3(4): 322–325.

Bolukbasi, T.; Pearce, A.; Yuan, A.; Coenen, A.; Reif, E.; Viégas, F.; and Wattenberg, M. 2021. An interpretability illusion for bert. *arXiv preprint arXiv:2104.07143*.

Brett, M.; Johnsrude, I. S.; and Owen, A. M. 2002. The problem of functional localization in the human brain. *Nature reviews neuroscience*, 3(3): 243–249.

Chen, L.; Varoquaux, G.; and Suchanek, F. 2023. The Locality and Symmetry of Positional Encodings. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 14313–14331.

Chen, R.; Hu, T.; Feng, Y.; and Liu, Z. 2024a. Learnable Privacy Neurons Localization in Language Models. *arXiv preprint arXiv:2405.10989*.

Chen, Y.; Cao, P.; Chen, Y.; Liu, K.; and Zhao, J. 2024b. Journey to the center of the knowledge neurons: Discoveries of language-independent knowledge neurons and degenerate knowledge neurons. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 17817–17825.

Conmy, A.; Mavor-Parker, A.; Lynch, A.; Heimersheim, S.; and Garriga-Alonso, A. 2023. Towards automated circuit discovery for mechanistic interpretability. *Advances in Neural Information Processing Systems*, 36: 16318–16352.

Dai, D.; Dong, L.; Hao, Y.; Sui, Z.; Chang, B.; and Wei, F. 2022. Knowledge Neurons in Pretrained Transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 8493–8502.

Dalvi, F.; Durrani, N.; Sajjad, H.; Belinkov, Y.; Bau, A.; and Glass, J. 2019. What is one grain of sand in the desert? analyzing individual neurons in deep nlp models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 6309–6317.

Dauphin, Y. N.; Fan, A.; Auli, M.; and Grangier, D. 2017. Language modeling with gated convolutional networks. In *International conference on machine learning*, 933–941. PMLR.

Ding, N.; Qin, Y.; Yang, G.; Wei, F.; Yang, Z.; Su, Y.; Hu, S.; Chen, Y.; Chan, C.-M.; Chen, W.; et al. 2023. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3): 220–235.

Durrani, N.; Sajjad, H.; Dalvi, F.; and Belinkov, Y. 2020. Analyzing Individual Neurons in Pre-trained Language Models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 4865–4880.

Elfwing, S.; Uchibe, E.; and Doya, K. 2018. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks*, 107: 3–11.

Frieder, S.; Pinchetti, L.; Griffiths, R.-R.; Salvatori, T.; Lukasiewicz, T.; Petersen, P.; and Berner, J. 2024. Mathematical capabilities of chatgpt. *Advances in Neural Information Processing Systems*, 36.

Geva, M.; Caciularu, A.; Wang, K.; and Goldberg, Y. 2022. Transformer Feed-Forward Layers Build Predictions by Promoting Concepts in the Vocabulary Space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 30–45.

Geva, M.; Schuster, R.; Berant, J.; and Levy, O. 2021. Transformer Feed-Forward Layers Are Key-Value Memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 5484–5495.

Gholipour, A.; Kehtarnavaz, N.; Briggs, R.; Devous, M.; and Gopinath, K. 2007. Brain functional localization: a survey of image registration techniques. *IEEE transactions on medical imaging*, 26(4): 427–451.

Gurnee, W.; Horsley, T.; Guo, Z. C.; Kheirkhah, T. R.; Sun, Q.; Hathaway, W.; Nanda, N.; and Bertsimas, D. 2024. Universal neurons in gpt2 language models. *arXiv preprint arXiv:2401.12181*.

Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; and Steinhardt, J. 2020. Measuring Massive Multitask Language Understanding. In *International Conference on Learning Representations*.

Jawahar, G.; Sagot, B.; and Seddah, D. 2019. What Does BERT Learn about the Structure of Language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3651–3657.

Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D. S.; Casas, D. d. l.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825*.

Jiang, Z.; Anastasopoulos, A.; Araki, J.; Ding, H.; and Neubig, G. 2020. X-FACTR: Multilingual factual knowledge retrieval from pretrained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 5943–5959.

Ju, Y.; and Zhang, Z. 2023. Klob: a benchmark for assessing knowledge locating methods in language models. *arXiv preprint arXiv:2309.16535*.

Kassner, N.; Dufter, P.; and Schütze, H. 2021. Multilingual LAMA: Investigating Knowledge in Multilingual Pretrained Language Models. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 3250–3258.

Kenton, J. D. M.-W. C.; and Toutanova, L. K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL-HLT*, 4171–4186.

Kojima, T.; Okimura, I.; Iwasawa, Y.; Yanaka, H.; and Matsuo, Y. 2024. On the Multilingual Ability of Decoder-based Pre-trained Language Models: Finding and Controlling Language-Specific Neurons. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 6912–6964.

Liu, N. F.; Gardner, M.; Belinkov, Y.; Peters, M. E.; and Smith, N. A. 2019. Linguistic Knowledge and Transferability of Contextual Representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 1073–1094.

McInnes, L.; Healy, J.; Saul, N.; and Großberger, L. 2018. UMAP: Uniform Manifold Approximation and Projection. *J. Open Source Softw.*, 3(29): 861.

Meng, K.; Bau, D.; Andonian, A.; and Belinkov, Y. 2022a. Locating and editing factual associations in GPT. *Advances in Neural Information Processing Systems*, 35: 17359–17372.

Meng, K.; Sharma, A. S.; Andonian, A. J.; Belinkov, Y.; and Bau, D. 2022b. Mass-Editing Memory in a Transformer. In *The Eleventh International Conference on Learning Representations*.

Petroni, F.; Rocktäschel, T.; Riedel, S.; Lewis, P.; Bakhtin, A.; Wu, Y.; and Miller, A. 2019. Language Models as Knowledge Bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2463–2473.

Pezeshkpour, P. 2023. Measuring and modifying factual knowledge in large language models. *arXiv preprint arXiv:2306.06264*.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.

Ren, S.; and Zhu, K. 2022. Specializing Pre-trained Language Models for Better Relational Reasoning via Network Pruning. In *Findings of the Association for Computational Linguistics: NAACL 2022*, 2195–2207.

Roberts, A.; Raffel, C.; and Shazeer, N. 2020. How Much Knowledge Can You Pack Into the Parameters of a Language Model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 5418–5426.

Srivastava, A.; Rastogi, A.; Rao, A.; Shoeb, A. A. M.; Abid, A.; Fisch, A.; Brown, A. R.; Santoro, A.; Gupta, A.; Garriga-Alonso, A.; et al. 2023. Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic attribution for deep networks. In *International conference on machine learning*, 3319–3328. PMLR.

Sung, M.; Lee, J.; Yi, S.; Jeon, M.; Kim, S.; and Kang, J. 2021. Can Language Models be Biomedical Knowledge Bases? In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 4723–4734.

Tang, T.; Luo, W.; Huang, H.; Zhang, D.; Wang, X.; Zhao, X.; Wei, F.; and Wen, J.-R. 2024. Language-specific neurons: The key to multilingual capabilities in large language models. *arXiv preprint arXiv:2402.16438*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Voita, E.; Ferrando, J.; and Nalmpantis, C. 2023. Neurons in large language models: Dead, n-gram, positional. *arXiv preprint arXiv:2309.04827*.

Wendler, C.; Veselovsky, V.; Monea, G.; and West, R. 2024. Do llamas work in english? on the latent language of multilingual transformers. *arXiv preprint arXiv:2402.10588*.

Wu, X.; Li, J.; Xu, M.; Dong, W.; Wu, S.; Bian, C.; and Xiong, D. 2023. DEPN: Detecting and Editing Privacy Neurons in Pretrained Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2875–2886.

Yang, N.; Kang, T.; and Jung, K. 2023. CRISPR: Eliminating Bias Neurons from an Instruction-following Language Model. *arXiv preprint arXiv:2311.09627*.

Zhang, N.; Yao, Y.; Tian, B.; Wang, P.; Deng, S.; Wang, M.; Xi, Z.; Mao, S.; Zhang, J.; Ni, Y.; et al. 2024a. A comprehensive study of knowledge editing for large language models. *arXiv preprint arXiv:2401.01286*.

Zhang, Z.; Zeng, Z.; Lin, Y.; Xiao, C.; Wang, X.; Han, X.; Liu, Z.; Xie, R.; Sun, M.; and Zhou, J. 2023. Emergent Modularity in Pre-trained Transformers. In *Findings of the Association for Computational Linguistics: ACL 2023*, 4066–4083.

Zhang, Z.; Zhao, J.; Zhang, Q.; Gui, T.; and Huang, X. 2024b. Unveiling Linguistic Regions in Large Language Models. *arXiv preprint arXiv:2402.14700*.

Zhao, W. X.; Zhou, K.; Li, J.; Tang, T.; Wang, X.; Hou, Y.; Min, Y.; Zhang, B.; Zhang, J.; Dong, Z.; et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Zhou, X.; Zhang, Y.; Cui, L.; and Huang, D. 2020. Evaluating commonsense in pre-trained language models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 9733–9740.

A Semantic Analysis of Neurons

According to the previous study, Logit Lens[2], the vocabulary probabilistic predictions are a linear function of the activations in Transformer's final layer but we can obtain reasonable distributions if we apply the same function to the activations of intermediate layers, i.e., an interpretable next-token distribution can be obtained by intermediate states. This finding suggests that intermediate states are capable of representing specific semantic meanings. In Section 5.4, we focused on examining the geographical distribution of domain-specific neurons but did not consider their semantic positions. Consequently, a natural question arises: *what are the properties of the memory cells associated with the QR neurons for the different domains and if they are clustered in the corresponding semantic space*.

To this end, we study the hidden activations of $\mathbf{W}^D$ (Eq 1), since transformer feed-forward layers can be viewed as key-value memory units. As a first step in our analysis, we visualize the $\mathbf{W}^D$ vectors associated with the QR neurons from the different domains using UMAP (McInnes et al. 2018) for dimensionality reduction (with cosine similarity used as the distance metric). For comparison, we additionally include the vectors from the unembedding matrix. The results are shown in Figure A1. As can be seen from the figure, the distribution of the vectors associated with QR neurons appears to be significantly different from that of vector unembeddings. Thus, it appears that the contents of the internal memory cells used by Llama 2 are not directly aligned with the final output space. This indicates that Llama 2 tends to form an abstract representation, *usually human-unreadable*, in intermediate layers (Wendler et al. 2024).

Since the 2D visualization produced by UMAP might not accurately reflect the true properties of the data manifold, we additionally examined the highest-likelihood tokens predicted by QR neurons. Domain-specific neurons are mainly centralized in middle layers, and we found the predicted tokens less human-interpretable, including tokens like textt, archivi, _Kontrola, _totalité or _Einzeln. Apart from the above tokens, there are certain neurons scattered in top layers still representing option letters, which need further refinement. In summary, since the detected neurons centralize in middle layers, it is hard to interpret their predicted tokens. We may need to explore a better semantic space to study their localized regions.

---

[2]https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens

Domain	Bio	Phys	Chem	Math	CS	Geo	Total
Num	13.1	13.3	12.8	11.1	14.3	12.7	12.9
Language	Ar	En	Fr	Ja	Ru	Zh	Total
Num	12.4	14.4	12.7	16.6	15.8	15.0	15.2

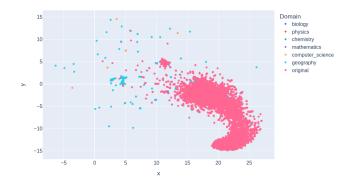Table A1: Average number of detected QR neurons per query.

Figure A1: UMAP visualisation of $\mathbf{W}^D$ vectors associated with the QR neurons and the token unembeddings
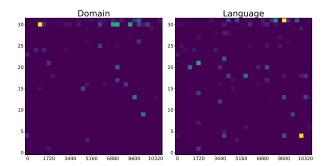

Figure A2: The distribution of common neurons.

B Neuron-Based Prediction Details

In the neuron-based prediction case study, we experiment on the MMLU (Hendrycks et al. 2020) validation set $\mathcal{D}_{val}$ to ensure there is no overlap between the dataset used to mine the QR neurons and the test set $\mathcal{D}_{test}$. As a further post-processing step, we randomly select three options from other domains to replace the incorrect options in each query. Additionally, we manually remove questions that become invalid due to this post-processing, including queries such as *"Which of the following is LEAST valid?"* and *"All of the following statements are true EXCEPT"*. These operations result in ~20 test samples per domain. To obtain *domain-specific neurons*, detected QR neurons for each query in a particular domain are grouped together, and we hope the activation of these neurons can be an indicator for predicting the correct answer. To perform the neuron-based prediction, we compute the gradient of the probability of each option token with respect to the QR neurons for the domain of the considered query, and select the option with the highest total gradient. For comparison, we include a normal prompt-based prediction, which employs designed prompts to query LLM without accessing the internal states (used prompts are shown in Table A3).
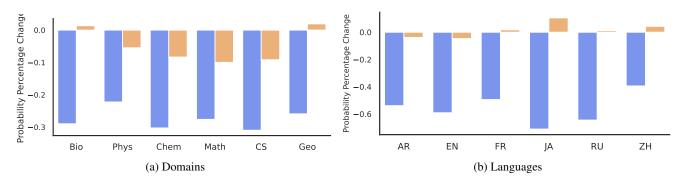
(a) Domains

(b) Languages

Figure A3: The correct probability percentage change after **suppressing**. The LLM here is Llama-2-7B (Touvron et al. 2023)

	Domain						Language					
	Boost			Suppress			Boost			Suppress		
Model	Related	Unrelated	⇑ Ratio	Related	Unrelated	⇑ Ratio	Related	Unrelated	⇑ Ratio	Related	Unrelated	⇑ Ratio
Random Neurons	-0.03	-0.03	1.0	+0.06	+0.11	0.55	+0.08	+0.04	2.0	-0.01	-0.01	1.0
Activation	+92.53	+91.73	1.0	-45.44	-45.14	1.0	+44.17	+40.28	1.1	-31.04	-28.88	1.1
Knowledge Neurons* (2022)	+932.05	+921.84	1.0	-85.70	-85.34	1.0	+1081.33	+161.98	6.7	-86.74	-48.18	1.8
QRNCA *wo/* ICA	+2403.60	+982.52	2.5	-82.82	-74.09	1.1	+1225.27	+190.03	6.5	-81.62	-36.93	2.2
QRNCA *w/* Common Neurons	+919.03	+328.49	2.8	-59.34	-33.59	1.8	+606.54	+54.84	10.4	-71.45	-8.40	8.5
QRNCA	+77.23	+17.55	**4.4**	-27.65	-4.95	**5.6**	+248.64	+6.91	**41.2**	-56.20	+1.56	**36.0**

Table A2: Details of average probability percentage changes of related and unrelated queries. The LLM here is Llama-2-7B (Touvron et al. 2023)
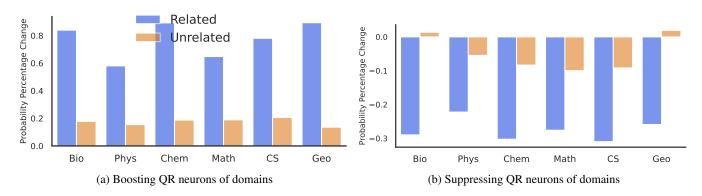

(a) Boosting QR neurons of domains

(b) Suppressing QR neurons of domains

Figure A4: The correct probability percentage change across different domains. The LLM here is **Mistral-7B** (Jiang et al. 2023)

(a) Boosting QR neurons of domains
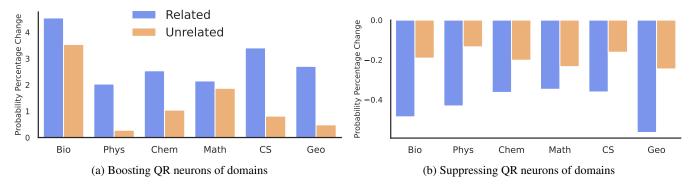
(b) Suppressing QR neurons of domains

Figure A5: An ablation study of using $\frac{\partial P_q}{\partial g}$ to compute naica scores. The LLM here is Llama-2-7B (Touvron et al. 2023).

Prompt ID	Template
Domain Prompt 1	You will be asked a multiple-choice question. Respond with the letter which corresponds to the correct answer, followed by a period. There is no need to provide an explanation, so your response should be very short.\nNow here is the question:\n{Question}\n A. {A}\n B. {B}\n C. {C}\n D. {D}\nResponse:
Domain Prompt 2	Prepare to answer a multiple-choice question. Provide the letter that corresponds to the correct answer, followed by a period. Keep your response brief; no explanations are necessary.\nHere is the question:\n{Question}\n A. {A}\n B. {B}\n C. {C}\n D. {D}\nResponse:
Domain Prompt 3	Below is a multiple-choice question. Respond with the letter that best answers the question. Keep your response brief, stating only the letter corresponding to your answer, followed by a period, with no explanation.\nThe question is:\n{Question}\n A. {A}\n B. {B}\n C. {C}\n D. {D}\nResponse:
Language Prompt 1	You will be asked a multiple-choice question. Respond with the letter which corresponds to the correct answer, followed by a period. There is no need to provide an explanation, so your response should be very short. \nNow here is the question:\n{Question} \nHere the [Y] is most likely to be? \n A. {A}\n B. {B}\n C. {C}\n D. {D}\nResponse:
Language Prompt 2	Prepare to answer a multiple-choice question. Provide the letter that corresponds to the correct answer, followed by a period. Keep your response brief; no explanations are necessary. \nNow here is the question:\n{Question} \nHere the [Y] is most likely to be? \n A. {A}\n B. {B}\n C. {C}\n D. {D}\nResponse:
Language Prompt 3	Below is a multiple-choice question. Respond with the letter that best answers the question. Keep your response brief, stating only the letter corresponding to your answer, followed by a period, with no explanation. \nNow here is the question:\n{Question} \nHere the [Y] is most likely to be? \n A. {A}\n B. {B}\n C. {C}\n D. {D}\nResponse:

Table A3: Prompt templates for constructing multi-choice QA datasets. We use ChatGPT to translate English templates to other languages.

Field	Question	Options
Biology	*The energy given up by electrons as they move through the electron transport chain is used to?*	A. make glucose B. make NADH C. produce ATP D. break down glucose
Physics	*An object is placed 100 cm from a plane mirror. How far is the image from the object?*	A. 50 cm B. 200 cm C. 100 cm D. 300 cm
Chemistry	*Three half-lives after an isotope is prepared:*	A. 12.5% of the isotope decayed B. 25% of the isotope decayed C. 25% of the isotope is left D. 12.5% of the isotope is left
Mathematics	*Suppose the graph of f is both increasing and concave up on a ¡= x ¡= b. Then, using the same number of subdivisions, and with L, R, M, and T denoting, respectively, left, right, midpoint, and trapezoid sums, it follows that:*	A. R <= T <= M <= L B. L <= M <= T <= R C. R <= M <= T <= L D. L <= T <= M <= R
Computer Science	*A programmer is writing a program that is intended to be able to process large amounts of data. Which of the following considerations is LEAST likely to affect the ability of the program to process larger data sets?*	A. How long the program takes to run B. How many programming statements the program contains C. How much storage space the program requires as it runs D. How much memory the program requires as it runs
Geography	*The tendency for migration to decrease with distance is called?*	A. push factors. B. migration selectivity. C. distance decay. D. pull factors.
English	*Sergey Lavrov was born in [Y]. Here the [Y] is most likely to be?*	A. Montevideo B. Bengaluru C. Parsons D. Moscow

Table A4: Examples in our constructed datasets. For the language dataset, we only show one English example as multi-lingual samples are obtained bu using traslator (Kassner, Dufter, and Schütze 2021)