Journey to the Center of the Knowledge Neurons: Discoveries of Language-Independent Knowledge Neurons and Degenerate Knowledge Neurons

Yuheng Chen[1,2 *], **Pengfei Cao**[1,2 *], **Yubo Chen**[1,2 †], **Kang Liu**[1,2], **Jun Zhao**[1,2]

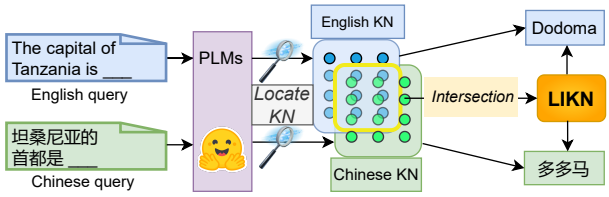
[1] The Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
chenyuheng22@ia.ac.cn, {pengfei.cao, yubo.chen, kliu, jzhao}@nlpr.ia.ac.cn

Abstract

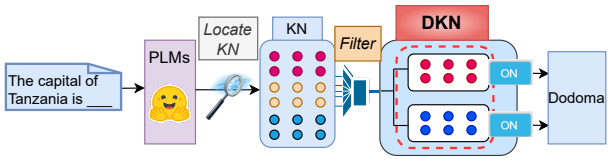
Pre-trained language models (PLMs) contain vast amounts of factual knowledge, but how the knowledge is stored in the parameters remains unclear. This paper delves into the complex task of understanding how factual knowledge is stored in multilingual PLMs, and introduces the Architecture-adapted Multilingual Integrated Gradients method, which successfully localizes knowledge neurons more precisely compared to current methods, and is more universal across various architectures and languages. Moreover, we conduct an in-depth exploration of knowledge neurons, leading to the following two important discoveries: (1) The discovery of Language-Independent Knowledge Neurons, which store factual knowledge in a form that transcends language. We design cross-lingual knowledge editing experiments, demonstrating that the PLMs can accomplish this task based on language-independent neurons; (2) The discovery of Degenerate Knowledge Neurons, a novel type of neuron showing that different knowledge neurons can store the same fact. Its property of functional overlap endows the PLMs with a robust mastery of factual knowledge. We design fact-checking experiments, proving that the degenerate knowledge neurons can help the PLMs to detect wrong facts. Experiments corroborate these findings, shedding light on the mechanisms of factual knowledge storage in multilingual PLMs, and contribute valuable insights to the field. The code is available at https://github.com/heng840/AMIG.

Introduction

Pre-trained language models (PLMs) (Devlin et al. 2018; Radford et al. 2019; Shliazhko et al. 2022; OpenAI 2023; Touvron et al. 2023; Wang et al. 2023) have revolutionized the field of natural language processing, due to their exceptional performance across a broad spectrum of tasks. These models, trained on extensive corpora such as Wikipedia, are widely believed to encapsulate vast amounts of factual knowledge (Petroni et al. 2019b; Jiang et al. 2020), but how the knowledge is stored in the parameters remains unclear (Kandpal et al. 2023). Investigating knowledge storage mechanisms will facilitate deeper comprehension and mastery of knowledge in PLMs (Zhen et al. 2022; Zhao et al. 2023). In this paper, we conduct an in-depth study on the Knowledge Localization task (Hase et al. 2023; Andreas 2022), which seeks to determine the storage location of specific factual knowledge in the model parameters, where such parameters are named *Knowledge Neurons* (Dai et al. 2022).



(a) Language-Independent Knowledge Neurons: Acquisition process and functionality.



(b) Degenerate Knowledge Neurons: Acquisition process and functionality. "ON" indicates the PLMs must activate at least one corresponding degenerate knowledge neuron.

Figure 1: Explanation of Language-Independent Knowledge Neurons (LIKN) and Degenerate Knowledge Neurons (DKN). KN denotes knowledge neurons.

Recently, several established approaches strive to elucidate the knowledge storage mechanism in PLMs. One strategy is the gradient-based method (Ancona et al. 2019), which assesses the contribution of each neuron by calculating its attribution score using integrated gradients. Another is the causal-inspired method (Cao et al. 2023), which employs a tracing algorithm to follow causal influences across model layers. Despite successful efforts in the knowledge localization task, these methods still face two major challenges: (1) *Lack of Universal Method for Different PLM Architectures*: Factual knowledge is observed to emerge in all kinds of PLM architectures, including auto-encoding models (e.g., BERT) (Devlin et al. 2018) and auto-regressive models (e.g., GPT) (Shliazhko et al. 2022). However, while some methods are suited for auto-encoding models and perform poorly

*These authors contributed equally to this work.
†Corresponding author.

with auto-regressive models (Meng et al. 2022a), others are designed specifically for auto-regressive models and are not well-adapted to auto-encoding models(Li et al. 2022), leaving a gap in a universal approach that performs well across both PLM architectures. (2) *Lack of Exploration in Multiple Languages*: Substantial knowledge is independent of language, and current LLMs support multilingualism. However, existing methods, with their focus solely on English datasets, may fail to provide comprehensive insights into the knowledge storage mechanism across different languages, limiting the ability to draw multilingual conclusions.

In order to localize knowledge neurons more precisely, we follow the gradient-based method and propose a novel knowledge localization method, termed Architecture-adapted Multilingual Integrated Gradients (AMIG). Firstly, for the lack of universal method in different PLM architectures, we design an architecture adaptation technique, making the baseline vectors in the integrated gradients algorithm (Lundstrom, Huang, and Razaviyayn 2022) universally compatible across different PLM architectures. Secondly, for the lack of exploration in multiple languages, we introduce a multilingual threshold adjustment technique, adjusting the thresholds in the integrated gradient calculations for different languages. Experimental results on multilingual datasets demonstrate that our method can localize the knowledge neurons more precisely compared to previous state-of-the-art models. In addition, we also conduct an in-depth exploration of knowledge neurons, leading to the following two important discoveries.

Language-Independent Knowledge Neurons: We discover a new type of neuron in multilingual PLMs that is capable of storing factual knowledge across languages. We name them *Language-Independent Knowledge Neurons*, since their existence transcends the boundaries of specific languages.

As illustrated in Figure 1a, these neurons are obtained by intersecting knowledge neurons derived from different languages, encapsulating knowledge representations that are consistent across multiple languages. Language-independent knowledge neurons can help cross-lingual knowledge editing tasks: a single edit to certain knowledge can simultaneously affect the corresponding knowledge in all languages. For example, if we edit the language-independent neuron corresponding to the fact $\langle$*Tanzania*, *Capital*, *Dar es Salaam*$\rangle$ to $\langle$*Tanzania*, *Capital*, *Dodoma*$\rangle$, this fact will be changed correspondingly in all languages. We design experiments to verify the role of language-independent knowledge neurons. Compared with existing cross-lingual knowledge editing models, the editing performance of our method is superior. This experiment demonstrates the potential of our method in cross-lingual knowledge editing applications.

Degenerate Knowledge Neurons: We discover an interesting phenomenon, corresponding to a completely new type of neurons. Given a fact and its corresponding knowledge neurons, some subsets of knowledge neurons exhibit unique properties. Even if some elements in this subset are suppressed, the model can still express the fact correctly; however, if all elements in the subset are suppressed, the model can no longer express the fact correctly. This phenomenon demonstrates that some knowledge neurons store the same factual knowledge, and the model needs to activate at least one of the neurons to express the facts correctly. It is very similar to the "degenerate" phenomenon in biological systems (Tononi, Sporns, and Edelman 1999; Mason 2015), so we name this type of neuron *Degenerate Knowledge Neurons*. Unlike redundancy, degenerate knowledge neurons cannot simply be deleted because they only partially overlap. A degenerate knowledge neuron may store multiple pieces of factual knowledge, the deletion of it has no effect on specific knowledge but may affect other knowledge.

Figure 1b illustrates the acquisition process of degenerate knowledge neurons. In detail, we first localize the knowledge neurons, then aggregate and filter them to obtain degenerate knowledge neurons. For the query "*The capital of Tanzania is* __", the PLM must activate at least one corresponding degenerate knowledge neuron to predict the correct fact *Dodoma*. Intuitively, the property of functional overlap in degenerate knowledge neurons endows the PLMs with a robust understanding of factual knowledge, ensuring that its mastery of facts remains stable and less prone to errors. Inspired by this, we design an experiment to use degenerate knowledge neurons for fact-checking. Our experiment demonstrates that the degenerate knowledge neurons can help the PLMs to detect wrong facts, thus illustrating that their presence enhances the PLMs' stable mastery of factual knowledge.

Overall, the main contributions are summarized as follows:

(1) We propose a novel knowledge localization method named architecture-adapted multilingual integrated gradients, which can effectively address the two challenges of traditional methods: the lack of a universal method for different PLM architectures and the lack of exploration in multiple languages, thus achieving more precise localization of knowledge neurons.

(2) We discover language-independent knowledge neurons, which store factual knowledge in a form that transcends language barriers. Experimental results demonstrate that they are beneficial for the cross-lingual knowledge editing task.

(3) We discover degenerate knowledge neurons, a new type of neuron that possesses properties of functional overlap, making the model's mastery of factual knowledge more robust. Experiments prove that they can help detect incorrect facts.

Methodology

Figure 2 schematically visualizes our proposed framework. It consists of three main modules, including knowledge neuron localization (module 1), language-independent knowledge neuron detection (module 2), and degenerate knowledge neuron detection (module 3). We illustrate each module in detail.

Knowledge Neuron Localization

Module 1 of Figure 2 showcases the knowledge localization module, which aims to pinpoint the exact locations of the knowledge neurons within a PLM. Using the fill-in-the-blank cloze task (Petroni et al. 2019a), we evaluate the understanding of a PLM of specific facts. For example, given a fact $\langle$*Tanzania*, *Capital*, *Dodoma*$\rangle$ with corresponding query "*The capital of Tanzania is* __", Petroni et al.(2019a) describe that a model knows a fact if it can predict the correct answer. In this study, we extend this analysis by introducing

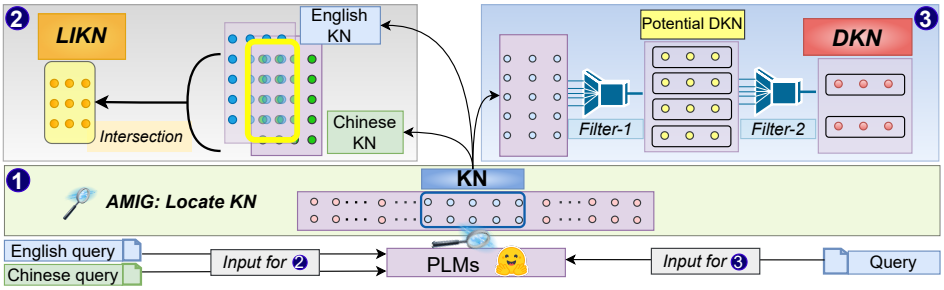


Figure 2: Overall Algorithm Flow, describing (1) our architecture-adapted multilingual integrated gradients (AMIG) method for locating knowledge neurons (KN), (2) the process of detecting language-independent knowledge neurons (LIKN), and (3) the process of detecting degenerate knowledge neurons (DKN).

the Architecture Adapted Multilingual Integrated Gradients method to localize the neurons responsible for processing factual information specifically.

Mathematically, given a query q, the probability of the correct answer predicted by a PLM can be defined as:

$$\mathrm{F}(\hat{w}_j^{(l)}) = p(y^*|q, w_j^{(l)} = \hat{w}_j^{(l)}), \tag{1}$$

where y^* is the correct answer, $w_j^{(l)}$ is the j-th neuron of l-th layer, and $\hat{w}_j^{(l)}$ is a value that $w_j^{(l)}$ is assigned to. To compute the attribution score for each neuron, we use integrated gradients(Sundararajan, Taly, and Yan 2017). Consider a neuron $w_j^{(l)}$, we can calculate its attribution score:

$$\mathrm{Attr}(w_j^{(l)}) = (\overline{w}_j^{(l)} - {w'}_j^{(l)}) \int_0^1 \frac{\partial \mathrm{F}({w'}_j^{(l)} + \alpha(\overline{w}_j^{(l)} - {w'}_j^{(l)}))}{\partial w_j^{(l)}} d\alpha, \tag{2}$$

where $\overline{w}_j^{(l)}$ is the value of $w_j^{(l)}$, ${w'}_j^{(l)}$ is the baseline vector of $w_j^{(l)}$, and $\frac{\partial \mathrm{F}({w'}_j^{(l)} + \alpha(w_j^{(l)} - {w'}_j^{(l)}))}{\partial w_j^{(l)}}$ calculates the gradient. As α changes from 0 to 1, $({w'}_j^{(l)} + \alpha(w_j^{(l)} - {w'}_j^{(l)}))$ changes from ${w'}_j^{(l)}$ to $w_j^{(l)}$, so the $\mathrm{Attr}(w_j^{(l)})$ can accumulates the probability changes caused by the change of $w_j^{(l)}$ through integrating the gradients. The ideal baseline vector ${w'}_j^{(l)}$, typically approximated by a zero vector (Liu et al. 2022), lacks consideration for diverse PLM architectures, resulting in sub-optimal performance. We address this by introducing an architecture adaptation technique to compute baseline vectors suitable for different PLM architectures.

First, in order to minimize the information content in the baseline vectors, we follow the method of Enguehard(2023), dividing the input query q into m words, and then feeding each word separately into the PLM to calculate the activation score for the neurons corresponding to each word q_i. Then, we meticulously design the baseline sentence corresponding to q_i be q'_i, and q'_i contains m words, with a length consistent with q, denoted as $q'_i = (q'_{i1} \dots q'_{ik} \dots q'_{im})$, where:

$$q'_{ik} = \begin{cases} \langle \text{mask} \rangle & \text{if } k = i \text{ (for auto-encoding models)} \\ \langle \text{eos} \rangle & \text{if } k = i \text{ (for auto-regressive models)} \\ q_k & \text{otherwise} \end{cases}, \tag{3}$$

where $\langle \text{mask} \rangle$ is used for masking auto-encoding models, $\langle \text{eos} \rangle$ stands for "end of sequence" in auto-regressive models, and q_k is the k-th word of the query. In this design, the i-th neuron in the l-th layer, represented by $w_j^{(l)}$, corresponds to q_i, and its associated baseline vector ${w'}_j^{(l)}$ corresponds to q'_i. We can then calculate the attribution score $Attr_i(w_j^{(l)})$ for each neuron when q_i is used as input, according to Equation (2). To calculate the integral, we use the Riemann approximation:

$$Attr_i(w_j^l) \approx \frac{\overline{w}_j^{(l)}}{N} \sum_{k=1}^{N} \frac{\partial F({w'}_j^{(l)} + \frac{k}{N} \times (\overline{w}_j^{(l)} - {w'}_j^{(l)})}{\partial w_j^{(l)}}, \tag{4}$$

where N is the number of approximation steps. The attribution for each word q_i is then summed and normalized, leading to the final attribution score for the query:

$$Attr(w_j^l) = \frac{\sum_{i=1}^{m} Attr_i(w_j^l)}{\sum_{j=1}^{n} \sum_{i=1}^{m} Attr_i(w_j^l)}, \tag{5}$$

where n is the number of neurons in the l-th layer. Finally, we can find the neurons with attribution scores greater than the threshold τ, and consider them as knowledge neurons, denote as $\mathcal{N}$.

Language-Independent Knowledge Neuron Detection

Explanation Many PLMs support multilingualism, and a significant portion of factual knowledge within these models is language-independent (Xu et al. 2023; Wang, Lipton, and Tsvetkov 2020). This necessity has become increasingly important in exploring the storage mechanism of factual knowledge in multilingual PLMs. We define neurons that store factual knowledge common to multiple languages as *Language-Independent Knowledge Neurons*, denoted as $\mathcal{L}$. To identify these type of knowledge neurons, we devise a detection algorithm that is illustrated as follows.

Algorithm As shown in the module 2 of Figure 2, given factual triples in K languages with identical semantics, let the corresponding queries be denoted by q^k for $k = 1, 2, \dots, K$. For each query, we use knowledge neuron localization module to obtain the corresponding knowledge neurons, where the attribution score of neuron $w_i^{(l)}$ is recorded as

$Attr_k(w_i^{(l)})$. The sensitivity of the multilingual PLMs to different languages varies, resulting in significant differences in attribution scores for queries in different languages. Therefore, it is difficult to obtain knowledge neurons for all languages by setting a unified threshold. To solve this problem, we design a multilingual threshold adjustment technique. We set different scaling factors τ_k for different languages, and record the maximum attribution score of the neurons corresponding to query q_k, and then determine the dynamic threshold:

$$T_k = \max_{i,l} Attr_k(w_i^{(l)}) \times \tau_k, \tag{6}$$

Then, we identify knowledge neurons $\mathcal{N}_k$ for the k-th language using threshold filtering as follows:

$$\mathcal{N}_k = \left\{ w_i^{(l)} \mid Attr_k(w_i^{(l)}) > T_k, \forall i, l \right\}, \tag{7}$$

Finally, we compute the intersection of the knowledge neurons across all languages:

$$\mathcal{L} = \bigcap_{k=1}^{K} \mathcal{N}_k, \tag{8}$$

where $\mathcal{L}$ represents the language-independent knowledge neurons, encoding factual knowledge consistent across all considered languages. Through the aforementioned algorithm, we can ultimately obtain them.

Degenerate Knowledge Neuron Detection

Explanation By conducting in-depth analysis, we identify an intriguing phenomenon: distinct sets of neurons are responsible for storing identical factual knowledge. For example, for a specific fact denoted as $\langle h, r, t \rangle$, suppose we localize 10 knowledge neurons labeled as $N = \{1, 2, \ldots, 10\}$. If we suppress the neurons of sets $A = \{1, 2\}$ or $B = \{3, 4, 5\}$, both subsets of N, we observe no significant decrease in prediction probability. Conversely, suppression of the neurons of these two sets simultaneous (i.e., $A \cup B$) leads to a substantial loss of prediction probability. This suggests that both sets A and B house the same factual knowledge, at least one must be active for the model to accurately comprehend the fact. Furthermore, these two sets of neurons are not mutually redundant. That is to say, besides the fact $\langle h, r, t \rangle$, A may also store the fact $\langle h_1, r_1, t_1 \rangle$, while B may store $\langle h_2, r_2, t_2 \rangle$, thus playing additional roles in PLMs. Given the resemblance of this behavior to the *degenerate* phenomenon in biological neural networks (Tononi, Sporns, and Edelman 1999; Mason 2015), we coin the term *Degenerate Knowledge Neurons* for these neurons. This concept is introduced in detail next.

Algorithm Formally, let $\mathcal{N} = \{n_1, \ldots, n_k\}$ be the set of all localized knowledge neurons[1], we define degenerate knowledge neurons as $\mathcal{D} = \{d_1^{\mathcal{D}}, \ldots, d_m^{\mathcal{D}}\}$, where each $d_i^{\mathcal{D}} = \{n_{i1}, \ldots, n_{iv}\}$ contains v knowledge neurons, and satisfies the following conditions:

$$Prob(\mathcal{N}) - Prob(\mathcal{N} \setminus P_{\text{s}}(n_i)) \leq T_{\text{low}}, \forall P_{\text{s}}(n_i), \tag{9}$$

$$Prob(\mathcal{N}) - Prob(\mathcal{N} \setminus \bigcup_{j=1}^{v} n_{ij}) > T_{\text{high}}, \tag{10}$$

[1]This can be further generalized, where each n_k is itself a set, making $\mathcal{N}$ a set of sets.

Algorithm 1: Identification of Degenerate Knowledge Neurons ($\mathcal{D}$)

Input: Query q, thresholds T_{low} and T_{high}.
Output: Degenerate knowledge neurons $\mathcal{D}$.

Localize knowledge neurons $\mathcal{N} = \{n_1, n_2, \ldots, n_k\}$.
Let $P_{\text{d}} \leftarrow \emptyset$ (potential degenerate knowledge neurons).
for each n_i in $\mathcal{N}$ **do**
  if $Prob(\mathcal{N})$ - $Prob(\mathcal{N} \setminus \{n_i\}) \leq T_{\text{low}}$ **then**
    $P_{\text{d}} \leftarrow P_{\text{d}} \cup \{n_i\}$
  end if
end for
Let $\mathcal{D} \leftarrow \emptyset$ (degenerate knowledge neurons).
for each n_{i1}, n_{i2} in P_{d}, $n_{i1} \neq n_{i2}$ **do**
  if $Prob(\mathcal{N})$ - $Prob(\mathcal{N} \setminus \{n_{i1}, n_{i2}\}) > T_{\text{high}}$ **then**
    $\mathcal{D} \leftarrow \mathcal{D} \cup \{n_{i1}, n_{i2}\}$, where $\{n_{i1}, n_{i2}\}$ is a $d_i^{\mathcal{D}}$ within $\mathcal{D}$.
  end if
end for
return $\mathcal{D}$

where $P_{\text{s}}(n_i)$ is a proper subset of the union $\bigcup_{j=1}^{v} n_{ij}$, i.e., $P_{\text{s}}(n_i) \subsetneq \bigcup_{j=1}^{v} n_{ij}$. $Prob(X)$ is the prediction probability of the model when the set of neurons X is activated, and T_{low} and T_{high} are predefined thresholds of acceptable prediction probability difference. Equation (9) indicates that suppressing any proper subset of $d_i^{\mathcal{D}}$, i.e., $P_{\text{s}}(n_i)$, will not result in a significant decrease in prediction probability; whereas Equation (10) shows that suppressing all the neurons in $d_i^{\mathcal{D}}$ will lead to a significant decrease in prediction probability. This demonstrates that these neurons store the same knowledge.

In the general case, considering n knowledge neurons and we need to evaluate all possible subsets, the complexity of finding $\mathcal{D}$ is $O(2^n)$. To make the problem tractable, we simplified the problem by assuming that each $d_i^{\mathcal{D}}$ only contains two knowledge neurons. This assumption reduces the problem complexity to $O(n^2)$. To further reduce the computation, we design a two-step filtering process. Depicted in Algorithm 1 and the module 3 of Figure 2, we first suppress each neuron and record neurons that do not cause a significant decrease in prediction probability, which are regarded as potential degenerate knowledge neurons P_{d}. For the elements in P_{d}, perform secondary filtering: suppress the pair of neurons in it, and if this operation leads to a significant decrease in the prediction probability of the model, record the pair of neurons as a degenerate knowledge neuron $d_i^{\mathcal{D}}$. Finally, we can return the degenerate knowledge neurons as $\mathcal{D}$.

Experiments

Experimental Settings

Model Selection and Dataset For our experiments, we opt for two distinct multilingual PLMs: m-BERT (Devlin et al. 2018) and m-GPT (Shliazhko et al. 2022). The m-BERT, an auto-encoding model, is pre-trained on a diverse collection of multilingual data, while the m-GPT, an auto-regressive model, is designed to process a wide-ranging corpus of 61 languages. Regarding the datasets, we employ mLAMA (Kass-

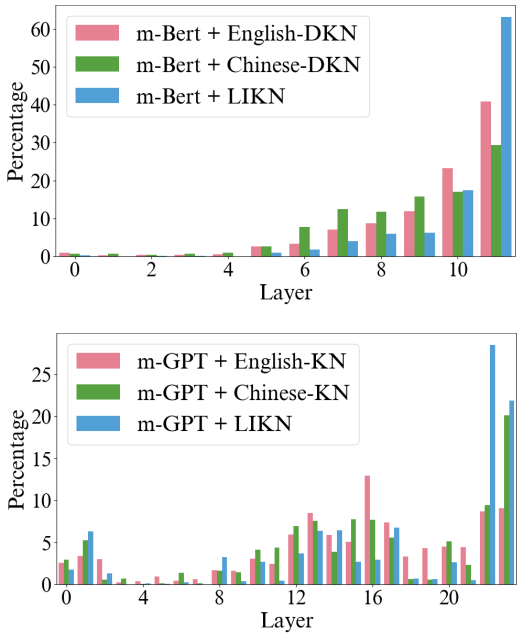


Figure 3: The distributions of knowledge neurons in m-BERT and m-GPT models under two languages (English-KN and Chinese-KN) and language-independent knowledge neurons (LIKN).

ner, Dufter, and Schütze 2021), a multilingual extension of the original LAMA (Petroni et al. 2019a, 2020) to localize the knowledge in multilingual PLMs.

Evaluation Metrics We apply the same neuron editing manipulation to both methods, where the detected knowledge neurons are suppressed or enhanced, followed by calculating the prediction probability of the PLM for both relevant and irrelevant facts. To compare the precision of knowledge localization across different methods in a comprehensive manner, we propose a new evaluation metric to assess the results of knowledge localization across the entire dataset:

$$SR_x = \frac{\Delta Prob_{rx}}{\Delta Prob_{ix}} \tag{11}$$

where SR_x is the editing success rate, and x represents our editing operation to suppress or enhance the neurons. Given a query, it itself is considered as a relevant fact, and a fact of a different type is randomly selected as its irrelevant fact. $\Delta Prob_{rx}$ and $\Delta Prob_{ix}$ represent the average changes in prediction probability under operation x for relevant and irrelevant facts, respectively. Overall, we hope that relevant facts change with the change of knowledge neurons, while irrelevant facts remain unchanged; thus, the higher the success rate, the better the localization results[2]. Since we perform suppress and enhancement operations on neurons separately, the success rates of these two cases are summed up as the final success rate: $SR = SR_{\text{enhance}} + SR_{\text{suppress}}$.

[2]Data with extremely high $\Delta Prob_{rx}$ or $\Delta Prob_{ix}$, reflecting unmastered facts, is excluded in order to localize storage of mastered facts.

Dataset	Method	m-BERT	m-GPT
English	B-KN	3.94	5.21
	AMIG (Ours)	**4.04** (↑2.3%)	**5.60** (↑7.6%)
Chinese	B-KN	5.58	5.44
	AMIG (Ours)	**10.29** (↑84.3%)	**7.86** (↑44.5%)

Table 1: Results of the localization of knowledge neurons. B-KN is the baseline method, the symbol "↑" indicates the increase in success rate compared to B-KN for our method, which can be expressed as: $\frac{\text{AMIG}-\text{B-KN}}{\text{B-KN}}$, and bold indicates the method with a higher SR.

Localization of Knowledge Neurons

We carry out experiments using the module 1 on both m-BERT and m-GPT models across English and Chinese datasets, and take the method proposed by Dai et al.(2022) as the baseline, which we denote as B-KN. The findings from our study are presented in Table 1 and Figure 3, from which we derive several key insights.

(1) Our method achieves better results in all settings. In Table 1, we use AMIG to represent our method, and the results in the table represents the average success rate SR. Under all settings, our method outperforms B-KN, especially for the Chinese dataset, where the success rates for m-BERT and m-GPT have increased by 84.34% and 44.49% respectively. This demonstrates that the knowledge neurons localized by our method are more precise.

(2) In m-BERT, knowledge neurons are primarily in the final layers, whereas in m-GPT they are in the early, middle, and final layers, as shown in Figure 3, where the *x* and *y* axes represent the PLM layers and the percentage of knowledge neurons, respectively. This might be due to the auto-encoding models (e.g., m-BERT), which share encoding space and encode high-level features in the final few layers, while the auto-regressive models (e.g., m-GPT) gradually refine the features at each layer to predict the next word.

(3) The distributions of knowledge neurons for Chinese and English are relatively similar, but differences persist. Similarities could be due to facts having the same meaning across languages, while differences might result from the inherent structural and syntactic differences between the languages or from variations in the quality of the pretraining corpora.

Language-Independence Neurons and Cross-Lingual Knowledge Editing

Localization of Language-Independence Neurons Through our experiment with module 2, we capture the results in Figure 3. The findings reveal that, whether in m-BERT or m-GPT, language-independent knowledge neurons are primarily concentrated in the final one or two layers. This might be because language-independent facts serve as high-level features, and the PLM is only able to successfully encode them in the final few layers.

Cross-Lingual Knowledge Editing Experimental Settings and Results We design cross-lingual editing experiments based on language-independent knowledge neurons. Similar

Dataset	Method	m-BERT	m-GPT
English	LIKN (Ours)	2.36 (↑ 10.3%)	2.54 (↑ 5.9%)
	Mono-KN	2.14	2.40
	Seq-KN	3.80	4.29
Chinese	LIKN (Ours)	7.18 (↑ 213%)	8.87 (↑ 277%)
	Mono-KN	2.29	2.35
	Seq-KN	4.09 (↓ 43.0%)	3.65 (↓ 58.8%)

Table 2: Results of cross-lingual knowledge editing. LIKN represents editing language-independent knowledge neurons, Mono-KN denotes editing knowledge neurons in one language's dataset corresponding to another, and Seq-KN denotes sequentially editing knowledge neurons in two languages. The symbol '↑' shows a success rate increase in LIKN over Mono-KN, represented as $\frac{\text{LIKN}-\text{Mono-KN}}{\text{Mono-KN}}$, and '↓' indicates a decrease in LIKN compared to Seq-KN, represented as $\frac{\text{LIKN}-\text{Seq-KN}}{\text{LIKN}}$.

to the setup of knowledge localization experiments, we suppress or enhance language-independent knowledge neurons and calculate the editing success rate SR. To demonstrate the role of language-independent knowledge neurons, we design two comparative experiments: (1) Editing the knowledge neurons of one language and observing the changes in the corresponding facts in another language. (2) Sequentially editing the knowledge neurons of two languages, observing the changes in the corresponding facts in both languages. Our analysis of Table 2 brings to light two insights.

(1) Language-independent knowledge neurons facilitate cross-lingual editing. Compared to editing in Chinese or English, editing language-independent knowledge neurons has a higher success rate in all settings; in the Chinese dataset, the success rates for m-BERT and m-GPT increased by 213.05% and 277.36%. This indicates that while editing facts in one language and expecting changes in another is challenging, language-independent neurons provide a viable solution.

(2) Editing each language separately does not guarantee better results. Though one might intuitively edit each language to achieve cross-lingual changes, our experiments show that this method not only relies on more computational resources but also might underperform. Sequential editing led to 42.97% and 58.80% lower success rates for m-BERT and m-GPT respectively, compared to using language-independent neurons, possibly due to confusion from multiple edits. This emphasizes the importance of language-independent neurons.

Degenerate Knowledge Neurons and Fact-Checking Experiment

Identification of Degenerate Knowledge Neurons in Multilingual PLMs We set up an experiment using module 3 to investigate the degenerate knowledge neurons, and the results are displayed in Figure 4. From our observations, degenerate knowledge neurons in m-BERT and m-GPT exhibit distribution patterns similar to knowledge neurons. This not only demonstrates a strong correlation between the degeneracy of factual knowledge and the facts themselves, but also reflects

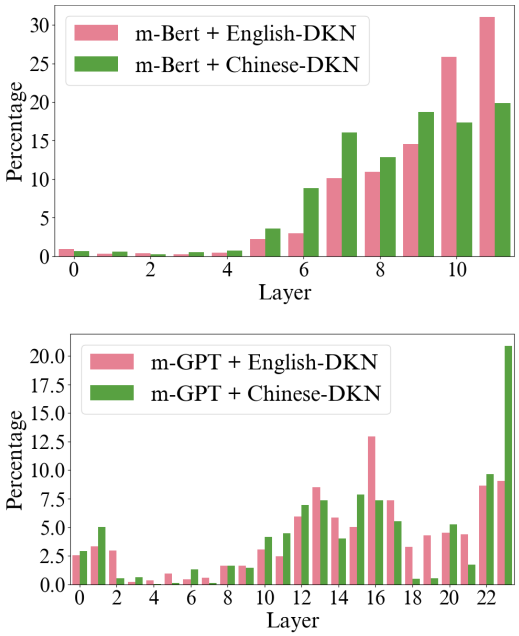


Figure 4: The distributions of degenerate knowledge neurons (DKN) in multilingual PLMs under two languages.

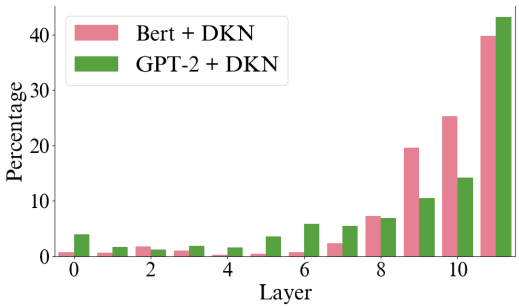


Figure 5: The distributions of degenerate knowledge neurons (DKN) in monolingual PLMs under two languages.

the PLMs' mastery of the facts.

Identification of degenerate knowledge neurons in Monolingual PLMs In our experiments with monolingual PLMs, we successfully identify the degenerate knowledge neurons and prove that they are inherently present within the PLMs. A possible question regarding degenerate knowledge neurons is: does the PLMs store the same fact in multiple languages, thus utilizing multiple neuron sets for the same information? To dispel this notion and demonstrate that the existence of degenerate knowledge neurons is unrelated to the support of multilingualism in the PLMs, we extend our exploration to monolingual PLMs, specifically in BERT and GPT-2. The distributions of these degenerate knowledge neurons is depicted in Figure 5, further reinforcing our conclusion.

Fact-Checking Experimental Settings and Results PLMs may conceal false facts (Edwards 2023; Pitt 2022), and current solutions often rely on external data for fact-checking (Vladika and Matthes 2023). Considering the nature of the

Dataset	Model	Method	P	R	F1
English	m-Bert	W/O	0.22	0.99	0.36
		W/	0.49	0.60	**0.54** (↑0.49)
	m-GPT	W/O	0.01	1.00	0.02
		W/	0.31	0.71	**0.43** (↑ 19.6)
Chinese	m-Bert	W/O	0.01	1.00	0.02
		W/	0.87	0.52	**0.65** (↑ 31.7)
	m-GPT	W/O	2e-4	1.00	4e-4
		W/	0.97	0.51	**0.67** (↑ 1672)
English	Bert	W/O	0.30	0.98	0.46
		W/	0.50	0.57	**0.54** (↑ 0.16)
	GPT-2	W/O	0.01	1.00	0.02
		W/	0.32	0.61	**0.42** (↑ 18.8)

Table 3: Fact-checking experiment results comparing methods with (W/) and without (W/O) degenerate knowledge neurons. The symbol "↑" shows F1-score improvement in W/ over W/O as $\frac{\text{W/}-\text{W/O}}{\text{W/O}}$, with bold indicating the higher score.

functional overlap of degenerate knowledge neurons, we design a fact-checking experiment to detect wrong facts based on degenerate knowledge neurons without relying on external data. Next, we introduce our experimental settings in detail.

First, the mLAMA dataset is modified to include a *wrong fact* attribute. For a fact triple associated with a certain relation name of fact, such as $\langle \textit{Tanzania}, \textit{Capital}, \textit{Dodoma} \rangle$, we randomly select an object (e.g., *Dar es Salaam*) from the same relation name as a wrong fact. Then, to validate the practical implications of our findings, we divide each type of query in the dataset into two parts proportionally. For each type, the first segment is used to obtain degenerate knowledge neurons, and we identify those exceeding a certain threshold of $t\%$ in quantity. Then, we take the queries from the second part, along with the corresponding correct or incorrect facts, as input and compute the average activation score of the degenerate knowledge neurons. If the average activation score surpasses a threshold λ, the fact is classified as correct. We use the original PLMs to directly evaluate the correctness of facts for comparative analysis. This configuration prevents the PLMs from employing the degenerate knowledge neurons of the query itself for fact-checking, rendering the experiments more convincing. We denote our method as "W/" in the Table 3. Finally, since the current fact-checking method must rely on external data, we use the PLMs to directly perform fact-checking as the baseline of our method, denoted as "W/O" in the Table 3. We use Precision, Recall and F1-score as evaluation metrics.

The results in Table 3 lead us to the following conclusions. (1) Degenerate knowledge neurons can help the PLMs detect wrong facts. Under various settings, our method is better than the baseline method, especially for Chinese datasets and auto-regressive models. For instance, in the context of m-GPT and Chinese datasets, the F1 score of our method has increased by 167150% compared to the baseline.This substantial improvement indicates that the presence of degenerate knowledge neurons enhances the PLMs' stable mastery of factual knowledge. (2) Using PLMs for fact-checking, they often judge a fact as correct, leading to extremely high Recall. This aligns with observations that generative language models may produce incorrect information if presented with a false premise (Edwards 2022; Lakshmanan 2022; Metz 2022). It is essential to recognize that a model's low predictive probability does not hinder the accurate identification of knowledge neurons. As shown in Equation 1, using the true value y^* allows for correct knowledge neuron localization even when the model's output is erroneous. (3) Auto-regressive models show higher Recall than auto-encoding models. This may be due to the auto-regressive design favoring coherence over accuracy, and the auto-encoding possibly being more conservative (Zhou et al. 2023). (4) The existence of degenerate knowledge neurons is unrelated to the support of multilingualism in the PLMs. In the monolingual PLMs, i.e., BERT and GPT-2, fact-checking can also be performed based on degenerate knowledge neurons. This result further proves the existence of degenerate knowledge neurons and its usefulness.

Related Work

Knowledge Localization Existing methods roughly fall into two categories: (1) Gradient-based method: Dai et al.(2022) first introduces the concept of knowledge neurons and localizes them by assessing the contribution of each neuron (Geva et al. 2021) through calculating their attribution scores using integrated gradients. (2) Causal-inspired method, introduced by Meng et al.(2022a), defines knowledge neurons as the neuron activations within PLMs that have the strongest causal effect on predicting certain factual knowledge, and this method has inspired the creation of knowledge editing algorithms such as ROME (Meng et al. 2022a), MEMIT (Meng et al. 2022b), and MEND (Mitchell et al. 2022). However, current methods lack a universal approach for different PLM architectures and exploration in multiple languages.

Axiomatic Attribution Methods Sundararajan, Taly, and Yan(2017) introduces the axiomatic attribution method, emphasizing Sensitivity and Implementation Invariance as the core axioms for attribution methods, leading to Integrated Gradients (IG). Subsequent research includes Discretized IG (Sanyal and Ren 2021), which uses interpolation strategies for gradient accuracy; Sequential IG (Enguehard 2023) designed for word importance evaluation; and Effective Shapley value along with Shapley IG, developed by Liu et al.(2022) to enhance efficiency and effectiveness. We improve the baseline vectors for IG to minimize their information content.

Conclusion

In this research, we explore factual knowledge localization in multilingual PLMs using our architecture-adapted multilingual integrated gradient method. We further design two modules, leading to two discoveries of language-independent knowledge neurons and degenerate knowledge neurons. The former affirms that a portion of the knowledge exists in a form that transcends language, while the latter presents a new type of neuron characterized by degeneracy.

Acknowledgments

This work is supported by the National Key Research and Development Program of China (No. 2020AAA0106400), the National Natural Science Foundation of China (No. 61976211, 62176257). This work is also supported by the Strategic Priority Research Program of Chinese Academy of Sciences (Grant No.XDA27020100), the Youth Innovation Promotion Association CAS, and Yunnan Provincial Major Science and Technology Special Plan Projects (No.202202AD080004).

References

Ancona, M.; et al. 2019. Gradient-based attribution methods. In *Explainable AI: Interpreting, explaining and visualizing deep learning*, 169–191.

Andreas, J. 2022. Language Models as Agent Models. arXiv:2212.01681.

Cao, B.; Lin, H.; Han, X.; and Sun, L. 2023. The Life Cycle of Knowledge in Big Language Models: A Survey. *Machine Intelligence Research*, 1–22.

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), ACL 2022, Dublin, Ireland, May 22-27, 2022*, 8493–8502.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, abs/1810.04805.

Edwards, B. 2022. OpenAI invites everyone to test ChatGPT, a new AI-powered chatbot—with amusing results. Retrieved 29 December 2022.

Edwards, B. 2023. Why ChatGPT and Bing Chat are so good at making things up. Retrieved 11 June 2023.

Enguehard, J. 2023. Sequential Integrated Gradients: a simple but effective method for explaining language models. arXiv:2305.15853.

Geva, M.; Schuster, R.; Berant, J.; and Levy, O. 2021. Transformer Feed-Forward Layers Are Key-Value Memories. arXiv:2012.14913.

Hase, P.; Bansal, M.; Kim, B.; and Ghandeharioun, A. 2023. Does Localization Inform Editing? Surprising Differences in Causality-Based Localization vs. Knowledge Editing in Language Models. arXiv:2301.04213.

Jiang, Z.; Xu, F. F.; Araki, J.; and Neubig, G. 2020. How Can We Know What Language Models Know? arXiv:1911.12543.

Kandpal, N.; Deng, H.; Roberts, A.; Wallace, E.; and Raffel, C. 2023. Large language models struggle to learn long-tail knowledge. In *International Conference on Machine Learning*, 15696–15707. PMLR.

Kassner, N.; Dufter, P.; and Schütze, H. 2021. Multilingual LAMA: Investigating Knowledge in Multilingual Pretrained Language Models. arXiv:2102.00894.

Lakshmanan, L. 2022. Why large language models like ChatGPT are bullshit artists. *becominghuman.ai*. Archived from the original on December 17, 2022.

Li, S.; Li, X.; Shang, L.; Dong, Z.; Sun, C.; Liu, B.; Ji, Z.; Jiang, X.; and Liu, Q. 2022. How Pre-trained Language Models Capture Factual Knowledge? A Causal-Inspired Analysis. arXiv:2203.16747.

Liu, S.; Fan, C.; Xiong, Y.; Wang, M.; Hu, Y.; Lv, T.; Chen, Z.; Wu, R.; and Gao, Y. 2022. The Effective coalitions of Shapley value For Integrated Gradients.

Lundstrom, D. D.; Huang, T.; and Razaviyayn, M. 2022. A rigorous study of integrated gradients method and extensions to internal neuron attributions. In *International Conference on Machine Learning*, 14485–14508. PMLR.

Mason, P. H. 2015. Degeneracy: Demystifying and destigmatizing a core concept in systems biology. *Complexity*, 20(3): 12–21.

Meng, K.; Bau, D.; Andonian, A.; and Belinkov, Y. 2022a. Locating and Editing Factual Associations in GPT. *Advances in Neural Information Processing Systems*, 36.

Meng, K.; Sen Sharma, A.; Andonian, A.; Belinkov, Y.; and Bau, D. 2022b. Mass Editing Memory in a Transformer. *arXiv preprint arXiv:2210.07229*.

Metz, C. 2022. The new chatbots could change the world. Can you trust them. *The New York Times*, 10.

Mitchell, E.; Lin, C.; Bosselut, A.; Finn, C.; and Manning, C. D. 2022. Fast Model Editing at Scale. In *International Conference on Learning Representations*.

OpenAI. 2023. GPT-4 Technical Report. arXiv:2303.08774.

Petroni, F.; Lewis, P.; Piktus, A.; Rocktäschel, T.; Wu, Y.; Miller, A. H.; and Riedel, S. 2020. How Context Affects Language Models' Factual Predictions. In *Automated Knowledge Base Construction*.

Petroni, F.; Rocktäschel, T.; Riedel, S.; Lewis, P.; Bakhtin, A.; Wu, Y.; and Miller, A. 2019a. 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. Hong Kong, China: Association for Computational Linguistics.

Petroni, F.; Rocktäschel, T.; Lewis, P.; Bakhtin, A.; Wu, Y.; Miller, A. H.; and Riedel, S. 2019b. Language Models as Knowledge Bases? arXiv:1909.01066.

Pitt, S. 2022. Google vs. ChatGPT: Here's what happened when I swapped services for a day. Retrieved 30 December 2022.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2019. Language Models are Unsupervised Multitask Learners.

Sanyal, S.; and Ren, X. 2021. Discretized Integrated Gradients for Explaining Language Models. arXiv:2108.13654.

Shliazhko, O.; Fenogenova, A.; Tikhonova, M.; Mikhailov, V.; Kozlova, A.; and Shavrina, T. 2022. mGPT: Few-Shot Learners Go Multilingual.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic Attribution for Deep Networks. arXiv:1703.01365.

Tononi, G.; Sporns, O.; and Edelman, G. M. 1999. Measures of degeneracy and redundancy in biological networks.

Proceedings of the National Academy of Sciences, 96(6): 3257–3262.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; Bikel, D.; Blecher, L.; Ferrer, C. C.; Chen, M.; Cucurull, G.; Esiobu, D.; Fernandes, J.; Fu, J.; Fu, W.; Fuller, B.; Gao, C.; Goswami, V.; Goyal, N.; Hartshorn, A.; Hosseini, S.; Hou, R.; Inan, H.; Kardas, M.; Kerkez, V.; Khabsa, M.; Kloumann, I.; Korenev, A.; Koura, P. S.; Lachaux, M.-A.; Lavril, T.; Lee, J.; Liskovich, D.; Lu, Y.; Mao, Y.; Martinet, X.; Mihaylov, T.; Mishra, P.; Molybog, I.; Nie, Y.; Poulton, A.; Reizenstein, J.; Rungta, R.; Saladi, K.; Schelten, A.; Silva, R.; Smith, E. M.; Subramanian, R.; Tan, X. E.; Tang, B.; Taylor, R.; Williams, A.; Kuan, J. X.; Xu, P.; Yan, Z.; Zarov, I.; Zhang, Y.; Fan, A.; Kambadur, M.; Narang, S.; Rodriguez, A.; Stojnic, R.; Edunov, S.; and Scialom, T. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv:2307.09288.

Vladika, J.; and Matthes, F. 2023. Scientific Fact-Checking: A Survey of Resources and Approaches. arXiv:2305.16859.

Wang, X.; Chen, G.; Qian, G.; Gao, P.; Wei, X.-Y.; Wang, Y.; Tian, Y.; and Gao, W. 2023. Large-scale Multi-modal Pre-trained Models: A Comprehensive Survey. *Machine Intelligence Research*, 20(4): 447–482.

Wang, Z.; Lipton, Z. C.; and Tsvetkov, Y. 2020. On Negative Interference in Multilingual Models: Findings and A Meta-Learning Treatment. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 4438–4450. Online: Association for Computational Linguistics.

Xu, Y.; Hou, Y.; Che, W.; and Zhang, M. 2023. Language Anisotropic Cross-Lingual Model Editing. arXiv:2205.12677.

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*.

Zhen, C.; Shang, Y.; Liu, X.; Li, Y.; Chen, Y.; and Zhang, D. 2022. A survey on knowledge-enhanced pre-trained language models. *arXiv preprint arXiv:2212.13428*.

Zhou, C.; Li, Q.; Li, C.; Yu, J.; Liu, Y.; Wang, G.; Zhang, K.; Ji, C.; Yan, Q.; He, L.; et al. 2023. A comprehensive survey on pretrained foundation models: A history from bert to chatgpt. *arXiv preprint arXiv:2302.09419*.