Neuron Empirical Gradient: Connecting Neurons' Linear Controllability and Representational Capacity

Anonymous ACL submission

Abstract

Although neurons in the feed-forward layers of pre-trained language models (PLMs) can store factual knowledge, most prior analyses remain qualitative, leaving the quantitative relationship among knowledge representation, neuron activations, and model output poorly understood. In this study, by performing neuron-wise interventions using factual probing datasets, we first reveal the linear relationship between neuron activations and output token probabilities. We refer to the gradient of this linear relationship as **"neuron empirical gradients."** and propose NeurGrad, an efficient method for their calculation to facilitate quantitative neuron analysis. We next investigate whether neuron empirical gradients in PLMs encode general task knowledge by probing skill neurons. To this end, we introduce MCEval8k, a multi-choice knowledge evaluation benchmark spanning six genres and 22 tasks. Our experiments confirm that neuron empirical gradients effectively capture knowledge, while skill neurons exhibit efficiency, generality, inclusivity, and interdependency. These findings link knowledge to PLM outputs via neuron empirical gradients, shedding light on how PLMs store knowledge. The code and dataset are released[1].

1 Introduction

Although Transformer (Vaswani et al., 2017)-based language models (LMs) benefit from large-scale pre-training, the pre-trained LMs (PLMs) suffer from hallucination, where models generate incorrect knowledge. This issue makes it important to understand the mechanism by which PLMs store knowledge within their parameters (Dai et al., 2022; Niu et al., 2024; Wang et al., 2024a, 2022).

In Transformer-based LMs, feed-forward (FF) layers serve as key-value memory (Geva et al., 2021), with neurons possessing the ability to retrieve knowledge. Previous work reveals that specific facts correlate with a limited number of neurons (knowledge neurons) (Dai et al., 2022; Yu and Ananiadou, 2024; Wang et al., 2024b), and even specific neurons own the abilities to perform various language skills (Wang et al., 2022; Tan et al., 2024). Although these studies reveal neurons' role in handling knowledge and skills, the numerical relationship between neuron activations and model outputs remains poorly understood.

In this study, we first quantitatively analyze how neuron activations affect model outputs through factual knowledge probing (§ 2). To observe model generation under varying neuron activations, we conduct a neuron-wise intervention on PLMs using MyriadLAMA (Zhao et al., 2024), a factual knowledge probing dataset. For given changes in neuron activations, we observe the resulting changes in the probabilities of target tokens for correct knowledge (hereafter, "output probabilities"). Notably, we find that for some neurons, within a certain range of activations, shifts in their activations (hereafter, "activation shifts") have a linear relationship with the output probabilities. We also find that neurons differ in the direction they shift output probabilities as their activations increase — a property we call *polarity*, which allows us to classify neurons as either positive or negative. Our evaluation of six PLMs, including Llama2-70B, confirms that neurons generally exhibit both linearity and polarity. We term the gradient of this linear relationship between a specific neuron with a token in response to a prompt as the *(neuron) empirical gradient*.

While empirical gradients quantify a neuron's importance and direction in shaping PLM outputs, their calculation is costly due to variability across prompts, neurons, and target tokens. To facilitate quantitative neuron analysis, we thus propose NeurGrad, an efficient method for estimating empirical gradients, and validate its performance on the MyriadLAMA dataset (§ 3). Our results on the above six, diverse PLMs show that NeurGrad outperforms

[1] https://anonymous.4open.science/r/NeurGrad

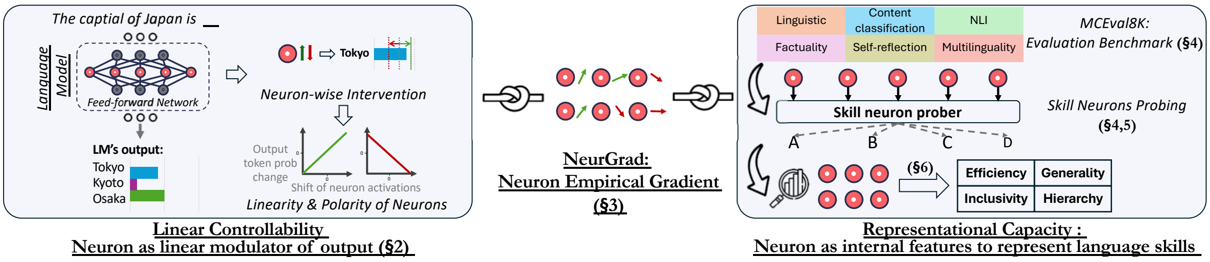

Figure 1: Overview of our contributions: i) observation on the linear controllability of PLM's outputs by shifting neuron activations, ii) an efficient method, NeurGrad, for computing the gradient of this linear relationship, and iii) skill neuron probing on MCEval8K to confirm empirical gradients capture diverse language skills.

baseline methods in both efficiency and precision.

We then leverage NeurGrad to investigate how empirical gradients represent language knowledge through skill neuron probing (Wang et al., 2022). Different from the factual knowledge probing in § 2, skill neuron probing aims to identify neurons associated with general language skills such as sentiment classification. We create a new multi-choice benchmark (MCEval8K) containing datasets conveying diverse language skills and train classifiers using empirical gradients calculated by NeurGrad as input. Neurons whose gradients provide valuable information for constructing the optimal classifier are identified as skill neurons for specific tasks.

Our contributions (Figrue 1) are as follows:

- We quantitatively confirm that neuron activations in PLMs have linear impacts on output token probabilities, introducing the concept of **neuron empirical gradients**. (§ 2)
- We present **NeurGrad**, an efficient method for estimating neuron empirical gradients. (§ 3)
- We confirm that empirical gradients serve as indicators of language skill representation via **skill neuron probing** (§ 4,§ 5); skill neurons demonstrate efficiency, generality, inclusivity, and interdependency (§ 6,§ C).
- We built **MCEval8K**, a multi-choice benchmark spanning various skill genres on language understanding. (§ 4.2)

2 Neuron as Linear Output Modulator

In this section, we aim to gain a deeper insight into how neurons in PLMs' FF layers influence model generations in a quantitative manner. Using factual knowledge probing as the target task, we perform neuron-wise intervention by adjusting neuron activations for the same prompt and observing the resulting change in output tokens' probabilities.

2.1 Settings

Models. To make the analysis result general, we experiment with two types of LMs, masked and causal LMs, with varied sizes and learning strategies. For masked LMs, we use three BERT (Vaswani et al., 2017; Devlin et al., 2019) models: $\text{BERT}_{\text{base}}$, $\text{BERT}_{\text{large}}$, and BERT_{wwm}. We construct masked prompts and let the model predict the masked token. For causal LMs, we examine three instruction-tuned LLMs of Llama2 family (Touvron et al., 2023), with sizes of 7B, 13B, and 70B. Following Zhao et al. (2024), we instruct them to generate single-token answers. See § B for model details.

Dataset. We utilize a multi-prompt knowledge probing dataset, MyriadLAMA[2] (Zhao et al., 2024), for neuron intervention. MyriadLAMA offers diverse prompts per fact, reducing the influence of specific linguistic expressions on probing results. We focus on single-token probing, where the target answer is represented by a single token. For each PLM, we randomly sample 1000 prompts from MyriadLAMA, where the model correctly predicts the target token. Due to differences in tokenizers, the probing prompts may vary across PLMs.

Neuron-wise intervention. We conduct neuron-wise intervention to analyze how activation shift affects model outputs. Specifically, we alter the neuron activations within a range of [-10, 10] with a step size of 0.2 to observe the resulting changes in target token output probabilities. Since observing the effect of a single neuron on one token for one prompt requires 100 inference runs and is costly, we only perform the neuron-wise intervention on specific neurons selected by either random sampling and choosing the top-k neurons with the highest absolute computational gradients.[3]

[2] https://huggingface.co/datasets/iszhaoxin/MyriadLAMA

[3] Computational gradient refers to the gradient computed

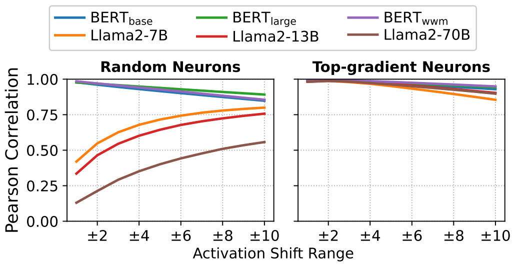

Figure 2: Average absolute Pearson correlation between activation shifts and output probabilities on 1000 neurons $\times$ 10 prompts with a step size of 0.2.

2.2 Results and Analysis

From experimental results, we reveal the numerical relationship between neuron activation shifts and output probabilities in PLMs.

Correlation vs. shift range. We first calculate the Pearson correlation between the shift ranges and the output probability of the correct tokens, considering only the absolute values to examine their linear relationship we call **neuron linearity**. The correlations are averaged over 10 prompts, each with 1000 neurons, for each activation shift size.[4]

Figure 2 depicts averaged correlations for the two neuron selection methods. The top-gradient neurons demonstrate high correlations across the PLMs and shift range, which is higher than the randomly sampled neurons. This suggests that the neuron linearity holds for top-gradient neurons (possibly, knowledge neurons).[5] Meanwhile, for top-gradient neurons, when setting the activation shift range to ± 2, the correlations in all models are close to 0.99, which we consider the threshold for indicating the linear relationship. Our subsequent analysis all uses the top-gradient neurons within a shift range of ± 2 by default.

Neuron linearity. We then present a quantitative analysis of the prevalence of neuron linearity and the generality of these neurons across different prompts and Transformer layers. Specifically, we report the ratio of neurons exhibiting **'linearity,'** defined as having correlations equal to or greater than 0.95 within a shift range of ± 2.[6] To enhance the coverage of analysis results, we use 1000 prompts paired with 100 top-gradient neurons, conducting 100K neuron intervention experiments per PLM.[7] The ratios of 'linear' neurons in BERT$_{\text{base/large/wwm}}$ models are 0.963/0.955/0.980, respectively, and the ratios for Llama-7B/13B/70B are 0.914/0.917/0.977, indicating that a large portion of neurons exhibit linearity. Our analysis in § A.2 reveals that linear neurons are common across layers and prompts.

Neuron polarity. In the following discussions, we consider the direction of change in output probabilities into our numerical analysis. We denote neurons are *positive/negative* if increasing/decreasing their activations enhances the target output probabilities.

3 Neuron Empirical Gradient

We quantify how important a neuron is in influencing the target token's probability by the gradient of the linear relationship we term **neuron empirical gradient**. To calculate the neuron empirical gradient, we fit a zero-intercept linear regression between activation shifts and output probability changes acquired through neuron intervention, and the regression coefficient is identified as the neuron empirical gradient, which requires extensive inferences for a specific neuron, prompt, and token.

To address this issue, we propose **NeurGrad**, inspired by the observation that computational gradients approximate empirical gradient magnitudes but fail to accurately capture neuron polarity, which is negatively correlated with activation signs.

$$\bar{G_E} = G_C \times -\operatorname{sign}(A), \qquad (1)$$

where $\bar{G_E}$, A, G_C, and $\operatorname{sign}(A)$ represents the estimated empirical gradient, activation, computational gradient, and sign of A[8] (1 for $A > 0$ and -1 for $A < 0$), respectively.

To validate NeurGrad's effectiveness, we obtain ground-truth empirical gradients via neuron-wise intervention experiments. Then, we measure the Pearson correlation between ground-truth empirical gradients and NeurGrad-estimated gradients.

from the computational graph through backpropagation.

[4]The mean/max/min activations over 1000 prompts on BERT$_{\text{base}}$ are -0.17/4.83/-0.04; On Llama2-7B: -21.6/7.13/0.

[5]The Llama2's lower correlations for smaller activation shift ranges are due to their small gradient magnitudes. We examine the gradient magnitudes of neurons in the PLMs. As a result, Llama2's gradients are five orders of magnitude smaller than BERT's, typically ranging from 10^{-5} to 10^{-8}. Given that gradients are in 16-bit floats with about 5.96×10^{-8} precision, random noise may overshadow true gradients in correlation calculation on Llama2 with randomly sampled neurons for smaller activation shift ranges.

[6]As there is no strict definition of linearity, we use the 0.95 as it indicates a strong linear relationship.

[7]We only chose 200 prompts and 100 neurons for Llama2-70B due to the large model size.

[8]For neurons with zero activation, we assign an empirical gradient of zero as such cases are rare. On average, there are at most around 1000 zero-activation neurons in the PLMs.

	G_C	IG.	NeurGrad
$\text{BERT}_{\text{large}}$	-.9307	.7360	.9998
$\text{BERT}_{\text{base}}$	-.8909	.7167	.9958
BERT_{wwm}	-.8914	.8584	.9989
Llama2-7B	.0115	.6728	.9769
Llama2-13B	-.0113	.6964	.9641
Llama2-70B	-.0391	n/a	.7811

Table 1: Pearson correlations between the estimated and ground-truth empirical gradients using sampled neurons; memory cost precludes results of IG. for Llama2-70B.

Specifically, we collect empirical gradients of 1000 prompts, with 100 random neurons per prompt. The activation shift range is set to [-2, 2] according to § 2.2. We estimate the empirical gradients using three different methods: computational gradient (G_c), integrated gradients (IG.) used for identifying knowledge neurons (Dai et al., 2022) that intervene neuron in small step sizes multiple times to simulate the gradient, and NeurGrad.

Table 1 reports the Pearson correlations between estimated gradients and empirical gradients. The results indicate NeurGrad's superiority in accurately measuring empirical gradients. NeurGrad is also much more efficient than IG. Regarding efficiency, calculating IG requires multiple iterations, each involving changes to neuron activations. In contrast, NeurGrad completes the calculation with just one inference pass, resulting in a computational cost nearly identical to that of computational gradients.

Finally, Table 2 reports the ratios of positive/negative neurons. It shows that the number of positive neurons is nearly equivalent to negative neurons, indicating that PLMs show no preference for either positive or negative neurons.

In subsequent sections, we explore whether empirical gradients have the capacity to represent diverse language knowledge, termed "language skills." If validated, this would connect knowledge representation to model output through neurons, allowing neuron-level model behavior adjustment.

4 Skill Neuron Probing using NeurGard

We have demonstrated that neurons could linearly influence output probability on factual probing tasks, showcasing their potential to manipulate model outputs. Building on this, we propose to investigate whether empirical gradients can effectively encode diverse language skills through the skill neuron probing (Wang et al., 2022). Skill neuron probing aims to locate neurons that encode the skill to solve language tasks. While previous studies explore the effectiveness of using neuron activations to identify skill neurons (Wang et al., 2022; Song et al., 2024), the representational capacity of empirical gradients is still underexplored.

	$\text{BERT}_{\text{base/large/wwm}}$	Llama2-7/13/70B
Pos. ratio	.5019/.5008/.4996	.4604/.4664/.4484
Neg. ratio	.4981/.4992/.5004	.4592/.4660/.4480

Table 2: The pos/neg neuron ratios over 1000 prompts.

4.1 Task Definition

We formulate the skill neuron probing task as follows. A dataset conveying specific language skills $\mathcal{D}$ consists of language sequence pairs, including knowledge inquiries $\mathcal{Q} = \{q_1, ..., q_{|\mathcal{T}|}\}$ and answer sequences $\mathcal{A} = \{a_1, ..., a_{|\mathcal{T}|}\}$, where arbitrary a_i belongs to the answer candidate set $\hat{\mathcal{A}}_{\text{cands}}$. For example, in the sentiment classification task, Q is the documents set, and A is the ground-truth sentiment labels. We then build classifiers that take behaviors of arbitrary neuron subset $\mathcal{N}_s \subseteq \mathcal{N}$ as features to indicate the correct answer sequences a_i for the knowledge inquiry p_i. $\mathcal{N}$ refers to all the neurons.[9]

Our skill neuron prober aims to find $\mathcal{N}_s^*$ that can achieve optimal accuracy over the target dataset $\mathcal{D}$.

$$\mathcal{N}_s^* = \arg\max_{\mathcal{N}_s \subseteq \mathcal{N}} \text{Acc}(f(\mathcal{N}_s), D) \tag{2}$$

$$\text{Acc}(f(\mathcal{N}_s), D) = \frac{1}{|D|} \sum_{i=1}^{|D|} \mathbb{1}[f(\mathcal{N}_s, p_i) = a_i]. \tag{3}$$

Here, $f(\mathcal{N}_s, p_i)$ is the output of the classifier F using the neuron subset $\mathcal{N}_s$ for the prompt p_i. $\mathbb{1}[X = Y]$ is an indicator function that equals 1 if X matches Y, and 0 otherwise.

4.2 Evaluation Benchmark: MCEval8K

As skill neuron probing requires a fixed target token, it faces high computational costs due to the infinite possibility of answer sequences. We thus create a multi-choice language skill evaluation benchmark, MCEval8K, that forces PLMs to generate a single-token option label (A, B, etc.) named for category labels (A: positive, B: negative, etc.). MCEval8K encompasses 22 tasks across 6 distinct genres, conveying diverse language skills to evaluate the neurons' representation capacity. Since tasks

[9] We focus on intermediate outputs (neurons) of FF layers.

vary in different sizes, with some, such as cLang-8 (Rothe et al., 2021; Mizumoto et al., 2011), containing millions of data points, we standardize the evaluation by limiting each task to 8K queries.[10] It minimizes unnecessary computational costs while ensuring consistency across tasks. We also ensure the number of ground-truth options per task is balanced to eliminate bias introduced by imbalanced classification. The skill genres and contained tasks are shown below (detailed in § C).

Linguistic: Part-of-Speech tagging on Universal Dependencies (POS) (Nivre et al., 2017), phrase-chunking on CoNLL-2000 (CHUNK) (Tjong Kim Sang and Buchholz, 2000), named entity recognition on CoNLL-2003 (NER) (Tjong Kim Sang and De Meulder, 2003) and grammatical error detection on cLang-8 dataset (GED) (Rothe et al., 2021; Mizumoto et al., 2011).

Content classification: Sentiment (IMDB) (Maas et al., 2011), topic classification (Agnews) (Zhang et al., 2015), and Amazon reviews with numerical labels (Amazon) (Hou et al., 2024).

Natural language inference (NLI): textual entailment (MNLI) (Williams et al., 2018), paraphrase identification (PAWS) (Zhang et al., 2019), and grounded commonsense inference (SWAG) (Zellers et al., 2018).

Factuality: Fact-checking (FEVER) (Thorne et al., 2018), factual knowledge probing (Myriad-LAMA) (Zhao et al., 2024), commonsense knowledge (CSQA) (Talmor et al., 2019) and temporary facts probing (TempLAMA) (Dhingra et al., 2022).

Self-reflection: Examine PLMs' internal status, including hallucination (HaluEval) (Li et al., 2023), toxicity (Toxic) (cjadams et al., 2017) and stereotype (Stereoset) (Nadeem et al., 2021) detections.

Multilinguality: We select tasks containing queries in different languages, including language identification (LTI) (Brown, 2014; Lovenia et al., 2024), multilingual POS-tagging on Universal Dependencies (M-POS) (Nivre et al., 2017), Amazon review classification (M-Amazon) (Keung et al., 2020), factual knowledge probing (mLAMA) (Kassner et al., 2021) and textual entailment (XNLI) (Conneau et al., 2018).

[10]Only the Stereoset task has fewer than 8K queries due to the limited size of the original dataset.

5 Neuron Gradient as Knowledge Feature

We train skill neuron probers based on NeurGrad's estimated gradients to investigate whether and how empirical gradients encode language knowledge.

5.1 Gradient-based Skill Neuron Prober

For each task dataset $\mathcal{D}$, we split it into: training set $\mathcal{D}_{\text{train}}$ to train the classifiers, validation set $\mathcal{D}_{\text{valid}}$ to decide hyperparameters, and test set $\mathcal{D}_{\text{test}}$ for evaluation, with the ratio of 6:1:1. We train three probers with different designs for comparison.

Polarity-based majority vote (Polar-prober) adopts a simple majority-vote classifier, taking each neuron in $\mathcal{N}_s$ as one voter. A polarity-based classifier leverages the polarity of neurons (positive or negative) as features for classification. Given $\mathcal{D}_{\text{train}} = \{(q_i, a_i)\}$ and any neuron $n_k \in \mathcal{N}$, we identify the polarity as feature $\mathbf{x}^{n_k}_{q_i,a_i}$ for each (q_i, a_i) pair. For each n_k, we calculate the ratio of being positive and negative across all $|\mathcal{D}_{\text{train}}|$ examples and the dominant polarity is identified as their global polarity $\bar{\mathbf{x}}^{n_k}$. Neurons with more consistent polarity are ranked higher.

To make prediction of q_i, we measure all polarities of $\mathbf{x}^{n_k}_{q_i,a_j}$, where $a_j \in \hat{\mathcal{A}}_{\text{cands}}, n_j \in \mathcal{N}^*_s$. The prediction of each p_i is made as follows:

$$f(\mathcal{N}^*_s, p_i) = \underset{a_j \in \hat{\mathcal{A}}_{\text{cands}}}{\arg\max} \sum_{n_k \in \mathcal{N}^*_s} \mathbb{1}[\mathbf{x}^{n_k}_{q_i,a_j} = \bar{\mathbf{x}}^{n_k}] \tag{4}$$

We identify the optimal size of $\mathcal{N}^*_s$ with $\mathcal{D}_{\text{valid}}$.

Magnitude-based majority vote (Magn-prober) utilizes gradient magnitudes as features for a majority-vote classifier. During training, for a specific p_i and n_k, we compare the gradients between $a \in \hat{\mathcal{A}}_{\text{cands}}$. Neurons that consistently exhibit the largest or smallest gradients for the ground truth a_i compared to other candidates are used as skill indicators. We record each neuron's preference for being either the largest or smallest. Neurons exhibiting more consistent behavior are assigned higher importance and identified as skill neurons. During inference, similar to Eq. 4, the prediction is made by selecting a_j that satisfies the majority of $n_k \in \mathcal{N}^*_s$. This prober is designed to compare against the polarity-based prober, aiming to investigate the differences between using polarity and gradient magnitude as feature sources.

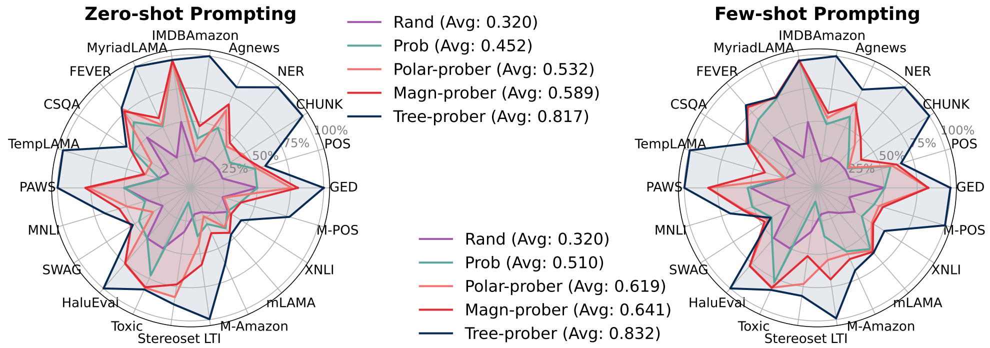

Figure 3: MCEval8K accuracies on Llama2-7B across tasks in zero-shot and few-shot settings, reported for Rand (random guess), TProb (token probability), and three proposed probers. The legend shows the average accuracy per method. See Table 8 and Table 9 for detailed accuracies values.

Random-forest classifier (Tree-prober) is finally introduced to understand the impact of considering the interdependency across skill neurons. We use the index (non-negative integers) of $a \in \hat{\mathcal{A}}_{\text{cands}}$ with the largest gradients as features of the neurons for training. The hyperparameter includes the number of trees (#n_trees) and layers (#n_layers) used in each tree. See more details in § D.2.

5.2 Experiment Setup

Dataset & Prompt settings Since our probing method restricts the output sequence length to 1, we carefully craft instructions and options for all datasets in MCEval8K through human effort. We evaluate both zero-shot and few-shot settings, ensuring in few-shot experiments that all candidate tokens appear once in the demonstrations to prevent majority label bias (Zhao et al., 2021). See § F for the designed instructions for all tasks.

Prober settings During validation, we select the optimal neuron size for majority-vote probers from 2^n (0<=n<=13). For the random-forest prober, we report accuracy using scikit-learn's default settings, where the optimal subset of features is selected automatically: 100 trees with no depth limitation. See § D.1 for detailed prober settings.

Model We perform skill neuron probing on Llama2-7B using three probers, all datasets in MCEval8K, and the full training set (6000) per task. For Llama2-70B, due to high cost, we probe one dataset per genre—NER, Agnews, PAWS, CSQA, HaluEval, and mLAMA—using 1,024 training examples and only train major-vote probers.

5.3 Result and Analysis

Skill neuron-based classifier accuracy is compared to two baselines: random guessing (**Rand**), and answer token probability-based classification (**LM-Prob**) which selects the candidate token with the highest probability as the prediction, serving as a benchmark for the LLMs' prompting performance.

Tasks	Llama2-7B		Llama2-70B	
	LM-Prob	Magn-Prober	LM-Prob	Magn-Prober
NER	.3610	.4980	.7900	.8170
Agnews	.5880	.7020	.7630	.8240
PAWS	.5240	.8150	.7790	.8460
CSQA	.6100	.6390	.7540	.7630
HaluEval	.5200	.7830	.7530	.8250
mLAMA	.6080	.6370	.7430	.7600

Table 3: Accuracies of 6 tasks on Llama2-7B and -70B.

Empirical gradients encode language skills. Figure 3 shows accuracies for all tasks in MCEval8K using the Llama2-7B, with both zero- and few-shot settings. The results demonstrate that LM-Prob outperforms Rand, indicating that Llama2-7B is capable of understanding instructions and recalling skills from its parameters. We also confirm the effectiveness of our skill neuron probers in addressing language tasks. The Tree-prober outperforms LM-Prob by nearly 30%, and even the two simple major-vote classifiers outperform LM-Prob in both zero- and few-shot settings. The per-task classification accuracies in Figure 3 show that skill neurons effectively represent diverse language skills, achieving consistently high results across tasks. See Table 8,9 for accuracy values.

Larger PLMs excel in skill recall. Table 3 compares accuracies of LM-Prob and Magn-prober across six tasks in the few-shot setting between Llama-7B and -70B. Llama-70B outperforms Llama-7B in both LM-Prob and skill neuron probing. However, the difference between LM-Prob

Neuron sizes	Tasks
$2^0 \sim 2^3$	Toxic, LTI, M-POS, FEVER, TempLAMA
$2^4 \sim 2^8$	GED, POS, CHUNK, NER, Amazon, IMDB, PAWS, MNLI, SWAG, HaluEval, XNLI, M-Amazon
$2^9 \sim 2^{13}$	Agnews, MyriadLAMA, CSQA, mLAMA

Table 4: Optimal number of skill neurons in probers.

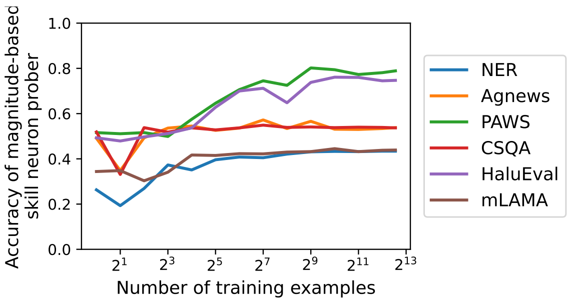

Figure 4: Accuracies with varying training sizes.

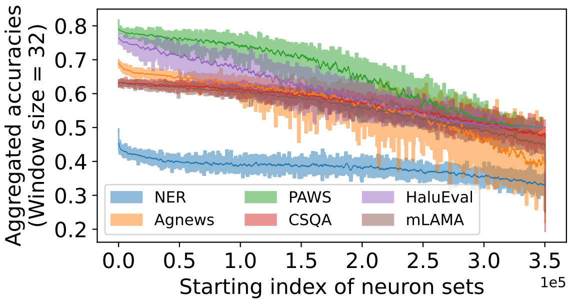

Figure 5: Accuracies of Magn-prober probers with different neuron sets, plotting the mean accuracy within each window, along with the accuracy ranges (min to max), as the envelope. Neuron sets are selected from all neurons in Llama2-7B in groups of 64, ranked by importance to be used as skill indicators.

and Magn-prober is smaller in Llama2-70B than in Llama2-7B, indicating the large model's strong ability to recall knowledge from its parameters.

6 Properties of Skill Neurons

6.1 Representation & Acquisition Efficiency

Representational efficiency: By finding the optimal neuron size on the validation set, we observe that skill-neuron prober can achieve high accuracy with a few neurons. We summarize optimal neuron sizes for all tasks with Magn-prober in Table 4. Most tasks achieved optimal accuracy within 256 neurons, demonstrating the efficiency of empirical gradients in representing language skills. Notably, factuality tasks, such as MyriadLAMA, CSQA, and mLAMA, engage a larger number of neurons, suggesting that handling facts requires more diverse neurons, reflecting the complexity of factual understanding tasks.

Acquisition efficiency: We report the accuracy of skill-neuron probers with different training examples in Figure 4. While adding training examples can consistently increase the probers' accuracy, the earnings slow down after 128, indicating the efficiency of acquiring skill neurons with limited data.

6.2 Generality Across Diverse Contexts

We investigate how skill neurons change when we provide different contexts, including instructions, demonstrations, and options for the same task. Given context X, we first acquire the skill neurons $\mathcal{N}_s^X$ and the accuracy $\mathrm{ACC}^X_{\mathcal{N}_s^X}$. Then, we use the classifier built with $\mathcal{N}_s^X$ to evaluate the task by context Y as $\mathrm{ACC}^Y_{\mathcal{N}_s^X}$. We denote the generality of $\mathcal{N}_s^X$ on context Y as $\frac{\max(\mathrm{ACC}^Y_{\mathcal{N}_s^X}-\alpha,0)}{\max(\mathrm{ACC}^Y_{\mathcal{N}_s^Y}-\alpha,0)}$, where α is the accuracy by Rand.

Using PAWS as an example, we create 12 distinct contexts by varying the instructions, the selection of demonstrations, and the output token styles. By measuring the generality for different combinations, we observe that the generality for prompting settings with different instructions and demonstrations is very high (close to 1), while the generality largely decreases if target tokens are changed. The results indicate that skill neurons maintain strong generality across different inputs, including variations in instructions and demonstrations. However, this generality diminishes when the output tokens are changed. See § G for details of experimental settings and results, including 12 designed contexts and generality results.

6.3 Are Neurons Exclusive in Skill Representations?

We investigate whether skill neurons exclusively represent specific skills or can be substituted by different neuron sets. We thus build Magn-probers using various neuron sets. Specifically, we select 64 consecutive neurons from the ranked list, ordered by their importance as skill indicators (§ 5.1).[11]

Figure 5 depicts the accuracies across six tasks, The result suggests that skill neurons are broadly

[11] We use 64-neurons units, which maintain high accuracies across tasks (§ A.2). With 352,256 neurons in Llama2-7B's FF layers, this yields 5,504 accuracy values per task.

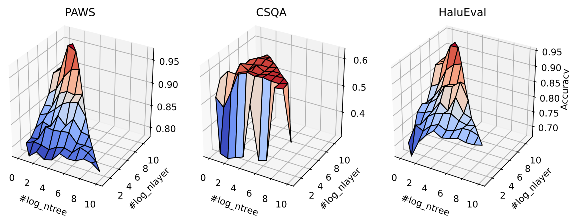

Figure 6: Accuracies of Tree-probers with varying depths and trees. **X-axis**: logarithm of trees' number; **Y-axis**: logarithm of tree depths; **Z-axis**: Accuracy.

distributed, with numerous neurons acting as skill indicators. Even when relying on less important neurons, the model's representational ability only gradually declines. Moreover, using only the least important neurons (end of each line) still yields better performance than random guesses, underscoring the inclusivity of skill neurons (See § E.3 for inclusivity evaluation on all datasets).

6.4 Do skill neurons depend on each other?

The majority-vote probers assume independence between neurons, while the Tree-prober considers their interdependencies by building hierarchical classifiers, which advantage over the major-vote prober in Figure 3 suggests that language skills can be better represented when considering the inter-neurons dependency. To see how important interdependency is in representing language skills, we train Tree-probers with varying hyperparameters, including the number of trees and depths per tree.[12] We report the resulting accuracies on PAWS, CSQA, and HaluEval in Figure 6. Their different shapes indicate that the interdependency levels required for different language skills are different. Some tasks (PAWS) prefer deep layers, while some (CSQA) prefer more trees, and some (HaluEval) require a balance between depths and trees. See § E.4 for more details.

7 Related Work

Mechanistic interpretability and knowledge attribution methods. Existing studies built the understanding of connections between knowledge and diverse modules in Transformers, such as attention heads (Clark et al., 2019; Olsson et al., 2022; Oymak et al., 2023), neurons in FF layers (Geva et al., 2021, 2022; Dai et al., 2022; Wang et al., 2024b), and the circuits within the models (Meng et al., 2022; Lieberum et al., 2023; Yao et al., 2024). They drive the development of knowledge attribution methods that assign importance scores to groups of features, indicating their relevance to the model output for a given input, including gradient-based method (Dai et al., 2022; Sundararajan et al., 2017), casual intervention methods that modify the internal status of models and observe the causal effect (Meng et al., 2022; Goldowsky-Dill et al., 2023) and automatic-tool-based methods relying on self-explanation with LLMs (Conmy et al., 2023; Singh et al., 2023). While these methods offer valuable insights into the interpretability of LLMs, such as neuron-ngram (Voita et al., 2024) or neuron-fact connections (Dai et al., 2022), they provide only qualitative measures of neuron importance, leaving the quantitative relationship between neurons and model output unexplored.

Skill neuron probing. Neurons in FF layers show the ability to convey specific skills so that using the neuron activations solely can tackle the language tasks, which these neurons are referred to as skill neurons (Wang et al., 2022; Song et al., 2024). Existing studies found that neurons can express semantic knowledge, solving tasks like sentiment classification (Wang et al., 2022; Song et al., 2024). Neurons are also discovered to represent more complex skills, including style transfer (Lai et al., 2024) and translation (Tan et al., 2024). Previous research viewed neuron activations as knowledge indicators, highlighting their representational ability but ignoring their limited influence on model output. In contrast, our empirical gradient findings provide a stronger basis for knowledge control.

8 Conclusions

Our study uncovers a linear relationship between individual neurons and model outputs through neuron intervention experiments. We quantify this linearity by "neuron empirical gradients" and propose NeurGrad, an efficient and effective method for estimating these gradients. We demonstrate empirical gradients' utility in representing language skills through skill neuron probing experiments. Our analyses reveal key properties of skill neurons—efficiency, generality, inclusivity, and interdependency. To our knowledge, this is the first study to establish a quantitative link between a model's internal representation and its output through gradients, laying a foundation for PLM output control via neuron-level adjustment.

[12] The numbers of trees and depths are set to 2^N and 2^M, respectively, where $0 \leq N \leq 10$, $1 \leq M \leq 11$, and $N + M < 12$.

9 Limitations

Our research establishes a framework for measuring neurons' influence on model output and demonstrates the effectiveness of empirical gradients in representing language skills, linking language skill representation to model output through neuron-level empirical gradients. However, the potential for achieving skill-level model output adjustment by tuning neuron values remains unexplored. Directly adjusting neuron values could offer a more efficient alternative to traditional weight-level tuning methods. This approach may enable dynamic behavior modification without altering the underlying parameters of LLMs, potentially reducing computational costs and enabling more flexible model adaptation.

References

Ralf D Brown. 2014. Non-linear mapping for improved identification of 1300+ languages. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 627–632.

cjadams, Jeffrey Sorensen, Julia Elliott, Lucas Dixon, Mark McDonald, nithum, and Will Cukierski. 2017. Toxic comment classification challenge. Kaggle.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

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

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502, Dublin, Ireland. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. 2022. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10:257–273.

Mor Geva, Avi Caciularu, Kevin Wang, and Yoav Goldberg. 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*, pages 30–45, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Nicholas Goldowsky-Dill, Chris MacLeod, Lucas Sato, and Aryaman Arora. 2023. Localizing model behavior with path patching. *Preprint*, arXiv:2304.05969.

D. Heath, S. Kasif, and S. Salzberg. 1993. k-dt: A multi-tree learning method. In *Proceedings of the Second Intl. Workshop on Multistrategy Learning*, pages 138–149.

Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. 2024. Bridging language and items for retrieval and recommendation. *Preprint*, arXiv:2403.03952.

Nora Kassner, Philipp Dufter, and Hinrich Schütze. 2021. Multilingual LAMA: investigating knowledge in multilingual pretrained language models. *CoRR*, abs/2102.00894. To appear in EACL2021.

Phillip Keung, Yichao Lu, György Szarvas, and Noah A. Smith. 2020. The multilingual amazon reviews corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.

Wen Lai, Viktor Hangya, and Alexander Fraser. 2024. Style-specific neurons for steering LLMs in text style transfer. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 13427–13443, Miami, Florida, USA. Association for Computational Linguistics.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6449–6464, Singapore. Association for Computational Linguistics.

Tom Lieberum, Matthew Rahtz, János Kramár, Neel Nanda, Geoffrey Irving, Rohin Shah, and Vladimir Mikulik. 2023. Does circuit analysis interpretability scale? evidence from multiple choice capabilities in chinchilla. *Preprint*, arXiv:2307.09458.

Holy Lovenia, Rahmad Mahendra, Salsabil Maulana Akbar, Lester James V. Miranda, Jennifer Santoso, Elyanah Aco, Akhdan Fadhilah, Jonibek Mansurov, Joseph Marvin Imperial, Onno P. Kampman, Joel Ruben Antony Moniz, Muhammad Ravi Shulthan Habibi, Frederikus Hudi, Railey Montalan, Ryan Ignatius, Joanito Agili Lopo, William Nixon, Börje F. Karlsson, James Jaya, Ryandito Diandaru, Yuze Gao, Patrick Amadeus, Bin Wang, Jan Christian Blaise Cruz, Chenxi Whitehouse, Ivan Halim Parmonangan, Maria Khelli, Wenyu Zhang, Lucky Susanto, Reynard Adha Ryanda, Sonny Lazuardi Hermawan, Dan John Velasco, Muhammad Dehan Al Kautsar, Willy Fitra Hendria, Yasmin Moslem, Noah Flynn, Muhammad Farid Adilazuarda, Haochen Li, Johanes Lee, R. Damanhuri, Shuo Sun, Muhammad Reza Qorib, Amirbek Djanibekov, Wei Qi Leong, Quyet V. Do, Niklas Muennighoff, Tanrada Pansuwan, Ilham Firdausi Putra, Yan Xu, Ngee Chia Tai, Ayu Purwarianti, Sebastian Ruder, William Tjhi, Peerat Limkonchotiwat, Alham Fikri Aji, Sedrick Keh, Genta Indra Winata, Ruochen Zhang, Fajri Koto, Zheng-Xin Yong, and Samuel Cahyawijaya. 2024. Seacrowd: A multilingual multimodal data hub and benchmark suite for southeast asian languages. *arXiv preprint arXiv: 2406.10118*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

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

Tomoya Mizumoto, Mamoru Komachi, Masaaki Nagata, and Yuji Matsumoto. 2011. Mining revision log of language learning SNS for automated Japanese error correction of second language learners. In *Proc. of 5th International Joint Conference on Natural Language Processing*, pages 147–155.

Moin Nadeem, Anna Bethke, and Siva Reddy. 2021. StereoSet: Measuring stereotypical bias in pretrained language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5356–5371, Online. Association for Computational Linguistics.

Jingcheng Niu, Andrew Liu, Zining Zhu, and Gerald Penn. 2024. What does the knowledge neuron thesis have to do with knowledge? In *The Twelfth International Conference on Learning Representations*.

Joakim Nivre, Daniel Zeman, Filip Ginter, and Francis Tyers. 2017. Universal Dependencies. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Tutorial Abstracts*, Valencia, Spain. Association for Computational Linguistics.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. 2022. In-context learning and induction heads. *Preprint*, arXiv:2209.11895.

Samet Oymak, Ankit Singh Rawat, Mahdi Soltanolkotabi, and Christos Thrampoulidis. 2023. On the role of attention in prompt-tuning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 26724–26768. PMLR.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Sascha Rothe, Jonathan Mallinson, Eric Malmi, Sebastian Krause, and Aliaksei Severyn. 2021. A Simple Recipe for Multilingual Grammatical Error Correction. In *Proc. of ACL-IJCNLP*.

Chandan Singh, Aliyah R. Hsu, Richard Antonello, Shailee Jain, Alexander G. Huth, Bin Yu, and Jianfeng Gao. 2023. Explaining black box text modules in natural language with language models. *Preprint*, arXiv:2305.09863.

Ran Song, Shizhu He, Shuting Jiang, Yantuan Xian, Shengxiang Gao, Kang Liu, and Zhengtao Yu. 2024. Does large language model contain task-specific neurons? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7101–7113, Miami, Florida, USA. Association for Computational Linguistics.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. *Preprint*, arXiv:1703.01365.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Shaomu Tan, Di Wu, and Christof Monz. 2024. Neuron specialization: Leveraging intrinsic task modularity for multilingual machine translation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6506–6527, Miami, Florida, USA. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *NAACL-HLT*.

Erik F. Tjong Kim Sang and Sabine Buchholz. 2000. Introduction to the CoNLL-2000 shared task chunking. In *Fourth Conference on Computational Natural Language Learning and the Second Learning Language in Logic Workshop*.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Elena Voita, Javier Ferrando, and Christoforos Nalmpantis. 2024. Neurons in large language models: Dead, n-gram, positional. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1288–1301, Bangkok, Thailand. Association for Computational Linguistics.

Mengru Wang, Yunzhi Yao, Ziwen Xu, Shuofei Qiao, Shumin Deng, Peng Wang, Xiang Chen, Jia-Chen Gu, Yong Jiang, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. 2024a. Knowledge mechanisms in large language models: A survey and perspective. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7097–7135, Miami, Florida, USA. Association for Computational Linguistics.

Xiaozhi Wang, Kaiyue Wen, Zhengyan Zhang, Lei Hou, Zhiyuan Liu, and Juanzi Li. 2022. Finding skill neurons in pre-trained transformer-based language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11132–11152, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yifei Wang, Yuheng Chen, Wanting Wen, Yu Sheng, Linjing Li, and Daniel Dajun Zeng. 2024b. Unveiling factual recall behaviors of large language models through knowledge neurons. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7388–7402, Miami, Florida, USA. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Yunzhi Yao, Ningyu Zhang, Zekun Xi, Mengru Wang, Ziwen Xu, Shumin Deng, and Huajun Chen. 2024. Knowledge circuits in pretrained transformers. *CoRR*, abs/2405.17969.

Zeping Yu and Sophia Ananiadou. 2024. Neuron-level knowledge attribution in large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3267–3280, Miami, Florida, USA. Association for Computational Linguistics.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: Paraphrase adversaries from word scrambling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1298–1308, Minneapolis, Minnesota. Association for Computational Linguistics.

Xin Zhao, Naoki Yoshinaga, and Daisuke Oba. 2024. What matters in memorizing and recalling facts? multifaceted benchmarks for knowledge probing in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 13186–13214, Miami, Florida, USA. Association for Computational Linguistics.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR.

A Neuron Linearity Analysis

A.1 Impact of Gradient Magnitudes in Linearity Analysis

To understand this divergence between randomly sampled and top-gradient neurons, we measure the percentage of neurons exceeding specific gradient magnitudes across all neurons in PLMs. As shown in Figure 7, Llama2's gradients are five orders of magnitude smaller than BERT's, typically ranging from 10^{-5} to 10^{-8}. Given that gradients are in 16-bit floats with about 5.96×10^{-8} precision and small gradient magnitudes on Llama2, random noise may overshadow true gradients in correlation calculation on Llama2 with randomly sampled neurons. This explains why correlations increase even with larger shift ranges for randomly sampled neurons (the left-hand side of Figure 2): the increased number of data points likely reduces the impact of noise on the results. We then focus on correlations using neurons with top gradient magnitudes to mitigate the random noise effect.

To reduce the impact of noise on the correlation, we select neurons with high absolute gradient values. We use the gradient computed from the computational graph through network backpropagation (hereafter, "computational gradient"). Specifically, we measure the correlations from the 1,000 neurons with the highest absolute computational gradients (Figure 2, right). The right-hand side of Figure 2 indicates that activation shifts tend to show stronger correlations with output tokens at smaller shift ranges, consistent across six models. Specifically, when setting the range to ± 2, the correlations in all models are close to 0.99, which we consider the threshold for indicating the linear relationship. Our subsequent analysis all uses the top-gradient neurons within a shift range of ± 2 by default.

A.2 Generality of Neuron Linearity

In this section, we provide additional evidence to verify that linearity is a general property for neurons in LLMs. Specifically, we want to verify whether the linear neurons exist widely across different Transformer feed-forward layers and within

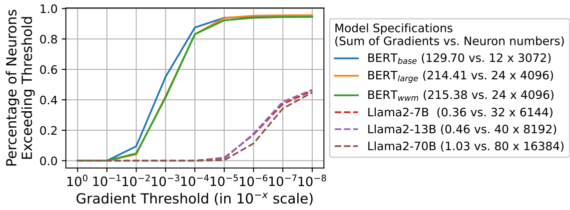

Figure 7: Ratio of neurons exceeding threshold as a function of gradient magnitudes. The X-axis shows gradient magnitudes, while the Y-axis represents the percentage of neurons with gradients exceeding those magnitudes.

different prompts. We use the metrics of layer generality (**LG**) and prompt generality (**PG**) to measure the prevalence of their existence. Intuitively, we can consider a simplified problem as follows: suppose we have many colored balls (green, blue, ...) and 10 bins, and if we want to verify whether the blue ball has "generality," it means (1) **high coverage**: the blue ball exists in most of the bins; (2) **even distribution**: the number of blue balls in each bin hardly differs from others. For our neuron generality, the "balls" are the "linear neurons," and the "bins" refer to either "feed-forward layers" (for **LG**) or "different prompt" (for **PG**). To address these two aspects simultaneously, we define **LG** and **PG** as follows:

$$\mathbf{LG} \triangleq \text{coverage}_{\text{layer}} \times \text{distribution}_{\text{layer}}, \quad (5)$$

$$\mathbf{PG} \triangleq \text{coverage}_{\text{prompt}} \times \text{distribution}_{\text{prompt}}, \quad (6)$$

where coverage and distribution are defined as:

$$\text{coverage}_{\text{x}} = \frac{\Sigma_i \mathbb{1}(\text{linear neuron exists in } x_i)}{\#\text{ of } x}, \quad (7)$$

$$\text{distribution}_x = 1 - \frac{\text{Var}(\#\text{neurons in } x)}{\max\text{Var}(\#\text{neurons in } x)}, \quad (8)$$

where x refers to either layer or prompt, $\max\text{Var}(\cdot)$ denotes the max possible variance. High coverage and distribution are desirable; a perfect generality then achieves coverage of one and distribution of one.

A.3 Dynamic Knowledge Store Hypothesis

The empirical gradient reveals a perspective that differs from the existing explanations in knowledge representation (Dai et al., 2022; Geva et al.,

	Linear neuron ratio	Prompt-wise gen.	Layer-wise gen.
BERT$_{\text{base}}$	.9625	.9999	.9844
BERT$_{\text{large}}$	.9546	.9999	.9492
BERT$_{\text{wwm}}$	.9799	.9999	.9494
Llama2-7B	.9137	.9999	.9518
Llama2-13B	.9187	.9999	.9833
Llama2-70B	.9769	.9999	.9780

Table 5: Neuron linearity statistics. We choose 1000 prompts and their corresponding 100 neurons randomly. For Llama2-70B, since the model is giant, we only chose 200 prompts and 100 neurons due to the high computational cost. The shift range is set to ± 2.

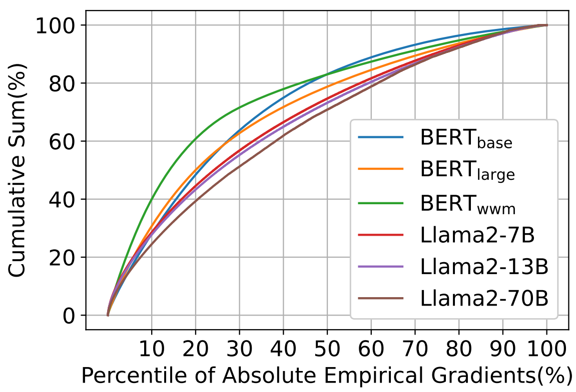

Figure 8: Cumulative distribution of empirical gradient magnitudes, sorted by descending empirical gradient volume. (**X-axis**: the percentiles of absolute empirical gradients; **Y-axis**: the cumulative contribution of these gradients to the total magnitude).

2022; Yu and Ananiadou, 2024; Voita et al., 2024; Geva et al., 2021). These explanations, such as the knowledge neuron theory, posit that knowledge is decisively represented by a few neurons (Dai et al., 2022; Geva et al., 2022; Yu and Ananiadou, 2024). Some studies have also used activations as indicators of knowledge representation (Voita et al., 2024; Geva et al., 2021), suggesting that if a neuron has a neuron activation of zero, it is not involved in representing the knowledge. We refer to this perspective as the static knowledge store hypothesis.

The empirical gradient offers a dynamic knowledge store hypothesis: *the expression of knowledge in a model is not determinative but a balanced status that can be reimplemented by modifying neuron activations.* For instance, by simultaneously increasing the activations of both positive and negative neurons, the model can use different activations to achieve the same output probability. This hypothesis provides a different perspective from the statistical hypothesis. Firstly, our experiments show that setting the activations of different neurons from positive to zero yields different effects. This suppresses the representation of knowledge in positive neurons while it activates the knowledge in negative neurons. We report the ratio of positive and negative neurons in Table 2. The percentage of positive and negative neurons is similar across the PLMs. All neurons in the BERT family exhibit non-zero empirical gradients, while only a few neurons in Llama2 models show non-zero empirical gradients.

Secondly, we found that a substantial number of neurons can alter the PLMs' output, indicating that while specific neurons can control the expression of certain knowledge, this relationship is not exclusive—other neurons also have this capacity. Figure 8 shows the cumulative distribution of empirical gradient magnitudes for all neurons in PLMs, calculated from 1000 prompts and sorted in descending order. We can observe that although different PLM families have varying distributions of empirical gradient values, as shown in Figure 7, their cumulative distributions are similar. Moreover, the figure shows that the rising curves do not converge until all neurons are accounted for. This steady increase suggests that a wide range of neurons can influence the PLMs' output. This suggests that no "decisive" knowledge neurons can absolutely control knowledge representation, while others have zero effect. Instead, knowledge representation in PLMs seems to emerge from the collective contributions of numerous neurons. The overall state of PLMs' ability to map factual inquiry to correct answers is balanced by the activations of many neurons rather than being dominated by a select few.

B Model cards

Here are the links from Hugging Face to load each model:

BERT$_{\text{base}}$: https://huggingface.co/bert-base-uncased

BERT$_{\text{large}}$: https://huggingface.co/bert-large-uncased

BERT$_{\text{wwm}}$: https://huggingface.co/bert-large-uncased-whole-word-masking

Llama2-7B: https://huggingface.co/meta-llama/Llama-2-7B-hf

Llama2-13B: https://huggingface.co/meta-llama/Llama-2-13B-hf

Model	#n_layers	#neurons_per_layer
$\text{BERT}_{\text{base}}$	12	3,072
$\text{BERT}_{\text{large}}$	24	4,096
BERT_{wwm}	24	4,096
LLama2-7B	32	11,008
Llama2-13B	40	13,824
Llama2-70B	80	28,672

Table 6: Number of Layers and Intermediate Neurons per Layer for BERT and Llama2 Models

Llama2-70B: https://huggingface.co/meta-llama/Llama-2-70B-hf

The statistics of these six PLMs, including the number of layers (#n_layers) and neurons per layer (#neurons_per_layer) are listed in Table 6.

C Construction of MCEval8K

The motivation behind creating MCEval8K is to establish a comprehensive benchmark that spans diverse knowledge genres and language skills. Since our goal is to facilitate skill neuron probing experiments where a single token must represent answers, we adopt a multi-choice task format. Additionally, we aim for the benchmark to be adaptable while avoiding redundancy for effective evaluation. In summary, we adhere to several guiding principles to design MCEval8K.

1. All datasets must be in multi-choice format.
2. Avoid including datasets that covey similar language skills.
3. To eliminate potential bias from imbalanced classifications, we ensure that the number of correct options is evenly distributed across all answer choices. This balance helps maintain fairness and accuracy in the analysis results.
4. We use a unified number (8000) of data to avoid high computational costs.

Multi-choice format: We created MCEval8K to include six different genres with 22 tasks, which are linguistic, content classification, natural language inference (NLI), factuality, self-reflection, and multilingualism. All the genres and tasks are listed in Table 7. For datasets that are not multi-choice tasks, we create options for each inquiry following rules. These datasets include POS, CHUNK, NER, MyriadLAMA, TempLAMA, Stereoset, M-POS, and mLAMA. The rules we adhere to create options are listed below:

POS We use weighted sampling across all POS tags to select three additional tags alongside the ground-truth tag.

CHUNK The process is analogous to POS.

NER The process is analogous to POS.

MyriadLAMA For factual inquiries formed from $< \text{sub}_i, \text{rel}_j >$, we collect all objects that appear as the target of the rel_j within the dataset and perform sampling to select three additional objects alongside the ground-truth tag.

TempLAMA We randomly sample three additional candidate years from the range 2009 to 2020, alongside the ground-truth tag.

M-POS The process is similar to POS, applied separately for each language.

mLAMA The process is similar to MyriadLAMA, applied separately for each language.

Balanced Options: Most datasets, except for Stereoset, contain more than 8000 data points. To ensure balance across all options, we perform balanced sampling so that each option has an equal number of examples. From these datasets, we split 8000 examples into training, validation, and test sets, allocating 6,000, 1000, and 1000 examples, respectively. For instance, in the case of mLAMA, where each inquiry has four options, we ensure that the correct answer is represented equally across all four positions. This results in 1,500 occurrences (6,000/4) per position in the training set and 250 occurrences per position in both the validation and test sets.

Creation of multilingual tasks: For multilingual datasets, we focus on five languages: English (en), German (de), Spanish (es), French (fr), and Chinese (zh). These languages vary significantly in linguistic distance, with English being closer to German, French closer to Spanish, and Chinese being distant from all of them. This selection allows for a deeper analysis considering linguistic distances between languages. We ensure that 5 languages have the same number of datAn examples in each dataset (1,600 per language). Furthermore, for datasets like mLAMA, XNLI, and M-AMAZON, we ensure that each piece of knowledge is expressed in all five languages. This consistency enables direct comparisons of language understanding abilities across different languages.

D Details of Skill Neuron Probing

D.1 Per-task Probing Result

In this section, we report the details of our skill neuron probing evaluation, including the full optimal accuracies on all tasks with zero-shot prompt setting (Table 8), few-shot prompt setting (Table 9). For two major vote probers, optimal accuracies are acquired by performing a hyper-parameter (optimal neuron size) search on the validation set and evaluating the test set. We report the optimal neuron sizes for all tasks along with the accuracies in the table. For the random-forest probing (Tree-prober), we directly use the gradients of all neurons to train the random forest tree. As the random forest training algorithm only takes important features to construct the decision trees, we also report the number of neurons used to construct random forests. The details of random-forest-based prober are introduced in § D.2.

D.2 Random Forest-based Prober

What is the random forest algorithm? Random forests is an ensemble learning algorithm (Heath et al., 1993) that works by creating a multitude of decision trees during training. For our multi-choice classification tasks in MCEval8K, the output of the random forest is the option selected by most trees. A decision tree is a supervised learning model that makes predictions by recursively splitting data based on feature values. During training, the tree builds nodes by selecting features that best separate the data according to a chosen metric, such as Gini impurity. Splitting continues until the data in each leaf node is sufficiently pure or a maximum depth is reached. During inference, a new input is passed through the tree by following the feature-based decisions from the root to a leaf, where the final prediction is determined by the majority label or average value of samples in that leaf.

Feature design: The objective of our study is to explore the effectiveness of using empirical gradients as features for knowledge representation and conduct further analysis. Therefore, the inputs for training and inference in the random forest model are constructed solely based on gradients estimated by NeurGrad. Specifically, each neuron is assigned an integer value for a given prompt. In our classification tasks, a neuron's feature is set to i if the gradient associated with the i-th token for that neuron is the largest among the gradients computed for all other candidate tokens (options). We ignore information on the smallest gradients to reduce the size of feature spaces.

Implementation details: For the implementation, we directly use *RandomForestClassifier* in scikit-learn (Pedregosa et al., 2011) for training and inference. We use the default parameters of RandomForestClassifier besides the number of trees (#n_trees) and layers (#n_layers) used in each tree. The number of trees refers to the number of decision trees used to ensemble the random forest. The number of layers refers to the layer depth for each tree. Noted that RandomForestClassifier constructs binary trees; thus, the number of features used in each tree is equal or less than $2^{\#\text{n\_layers}} - 1$.

Visualization: We present an example of a single-tree random forest model learned from the PAWS dataset in the few-shot setting, illustrated in Figure 9. The number of trees and layers is set to 1 and 8 for learning this decision tree. The PAWS dataset is a binary classification task with candidate tokens "yes" and "no." To construct features for each neuron, we compare the empirical gradients computed by NeurGrad for the prompt-"yes" and prompt-"no" pairs. If the gradient estimated for the prompt-"yes" pair exceeds that of the prompt-"no" pair, we assign a feature value of 1; otherwise, we assign 0.

E Additional Analysis on Probing Results

E.1 Interpreting in-context learning with empirical gradients.

To understand why simple majority-vote classifiers achieve high accuracy, we analyze the gradients associated with each answer choice. Using PAWS (binary classification) as an example, we inspect the gradient pairs for target tokens (yes/no) across all training prompts. We find that 97.21% of neurons display opposite signs for yes/no tokens. Moreover, the Pearson correlation between yes/no gradients is -0.9996. This pronounced inverse correlation suggests that empirical gradients are sharply polarized, making it easier for a majority-vote approach to distinguish between the target tokens. Furthermore, we examine how zero-shot and few-shot prompting differ from the perspective of empirical gradients. Our analysis reveals that the total gradient magnitudes in few-shot scenarios over 22 tasks are 5.36 times greater than in zero-shot. This indicates that demonstrations in context can effectively activate skill neurons, leading to better task understanding.

E.2 More Data about Efficiency

We report the accuracies of major-vote probers with different neuron sizes for all tasks to provide additional evidence for the discussion about the representation and acquisition efficiency of skill neurons in § 6.1. The results are demonstrated in Figure 12 and Figure 13 for zero-shot and few-shot prompting settings.

E.3 Probing With Varying Neuron Sets

We report the aggregated accuracies across all 22 tasks in MCEval8K in Figure 14 to provide additional evidence for discussion in § 6.3. It demonstrates that many neurons can construct the classifiers in solving the language tasks, showing their ability to represent language skills and knowledge.

E.4 Tree-prober: Flatteness vs. Hierarchy

To investigate the balance between hierarchy and independence of skill neurons, we train Tree-prober with fixed neuron features (2^{10}) but with different depths and trees. For each task, we train 10 Tree-probers, varying the number of trees (#n_tree $\in$ ($2^0 \sim 2^9$)) and the tree depth (#n_layer $\in$ ($2^{10} \sim 2^1$)), which fewer trees with deeper layers indicate a more hierarchical structure. All tasks show a camel curve given stronger hierarchies. We report the optimal #n_layer for different tasks as follows: CSQA(4), MNLI(16), SWAG(16), Stereoset(16), Agnews(32), MyriadLAMA(32), mLAMA(32), XNLI(32), POS(64), FEVER(64), Toxic(64), LTI(64), GED(128), IMDB(128), M-Amazon(128), CHUNK(256), NER(256), Amazon(256), PAWS(256), HaluEval(256), M-POS(256), TempLAMA(1024). This demonstrates that Different language skills require different hierarchy levels. For instance, factual tasks benefit from flatter structures, while linguistic tasks prefer deeper hierarchies.

For all tasks in MCEval8K, we plot the accuracies of trained models with varying hyperparameters, including the number of trees and layers per tree. The number of trees is set to 2^N, where N ranges from 0 to 10, and the number of layers is set to 2^M, where M ranges from 1 to 11. Training is conducted only for configurations where $N + M < 12$. The results are visualized as 3D surfaces, where the x-axis represents the logarithm of the number of trees (#log_ntree), the y-axis shows the logarithm of the number of layers (#log_nlayer), and the z-axis indicates the accuracy evaluated on the test set. We display the results for all tasks under the zero-shot setting in Figure 16 and those under the few-shot setting in Figure 17.

F Prompting Setups

In this subsection, we list all the instructions we use for each task in MCEval8K. It includes design instructions, options, and a selection of few-shot examples. As mentioned in § 5.2, we adopt two instruction settings, zero-shot and few-shot. For few-shot prompting, we set the number of examples to the same number as the number of options and ensure each option only appears once to prevent majority label bias (Zhao et al., 2021). All the few-shot examples are sampled from the training set. Finally, we list all the instructions and options we used for skill neuron probing examples by showing one zero-shot prompt.

GED

```
### Instruction: Which of the sentence
below is linguistically acceptable?
### Sentences:
a.I set the alarm for 10:00 PM but I could
n't wake up then .
b.I set the alarm for 10:00PM but I could
n't wake up then .
### Answer:
```

POS

```
### Instruction: Determine the
part-of-speech (POS) tag for the
highlighted target word in the given
text. Choose the correct tag from the
provided options.
### Input text:One of the largest
population centers in pre-Columbian
America and home to more than 100,000
people at its height in about 500 CE,
Teotihuacan was located about thirty
miles northeast of modern Mexico City.
### Target word:'pre-Columbian'
### Options:
a.DET
b.ADJ
c.PRON
d.PUNCT
### Answer:
```

CHUNK

```
### Instruction: Identify the chunk type
for the specified target phrase in the
sentence and select the correct label from
```

```
the provided options.
### Input text:B.A.T said it purchased
2.5 million shares at 785 .
### Target phrase:'said'
### Options:
a.PP
b.VP
c.NP
d.ADVP
### Answer:
```

NER

```
### Identify the named entity type for
the specified target phrase in the given
text. Choose the correct type from the
provided options
### Input text:With one out in the fifth
Ken Griffey Jr and Edgar Martinez stroked
back-to-back singles off Orioles starter
Rocky Coppinger ( 7-5 ) and Jay Buhner
walked .
### Target phrase:'Orioles'
### Options:
a.LOC
b.ORG
c.MISC
d.PER
### Answer:
```

Agnews

```
### Instruction: Determine the genre of
the news article. Please choose from
the following options: a.World b.Sports
c.Business d.science. Select the letter
corresponding to the most appropriate
genre.
### Text:Context Specific Mirroring
"Now, its not that I dont want to have
this content here. Far from it. Ill
always post everything to somewhere on
this site. I just want to treat each
individual posting as a single entity
and place it in as fertile a set of beds
as possible. I want context specific
mirroring. I want to be able to
newlinechoose
multiple endpoints for a post, and
publish to all of them with a single
button
click."

### Genres:
a.World
b.Sports
c.Business
d.Science
### Answer:
```

Amazon

```
### Instruction: Analyze the sentiment
of the given Amazon review and assign a
score from 1 (very negative) to 5 (very
positive) based on the review. Output
only the score.
### Input Review:I never write reviews,
but this one really works, doesn't float
up, is clean and fun. Kids can finally
take a bath!
### Output Score:
```

IMDB

```
### Instruction: Based the review, is the
movie good or bad?
### Review:Stewart is a Wyoming cattleman
who dreams to make enough money to
buy a small ranch in Utah ranch
<...abbreviation...>. In spontaneous
manner, Stewart is lost between the
ostentatious saloon owner and the
wife-candidate...
### Answer:
```

MyriadLAMA

```
### Instruction: Predict the [MASK] in
the sentence from the options. Do not
provide any additional information or
explanation.
### Question:What is the native language
of Bernard Tapie? [MASK].
### Options:
a.Dutch
b.Telugu
c.Russian
d.French
### Answer:
```

CSQA

```
### Instruction: Please select the most
accurate and relevant answer based on the
context.
### Context: What does a lead for a
journalist lead to?
### Options:
a.very heavy
b.lead pencil
c.store
d.card game
```

e.news article
Answer:

TempLAMA

Instruction: Select the correct year from the provided options that match the temporal fact in the sentence. Output the index of the correct year.
Question:Pete Hoekstra holds the position of United States representative.
Options:
a.2013
b.2014
c.2018
d.2011
Answer:

PAWS

Instruction: Is the second sentence a paraphrase of the first? Answer exactly 'yes' or 'no'.
Sentence 1: It is directed by Kamala Lopez and produced by Cameron Crain , Richard Shelgren and Kamala Lopez .
Sentence 2: It was produced by Cameron Crain , Richard Shelgren and Kamala Lopez and directed by Kamala Lopez .
Answer:

MNLI

Instruction: Given a premise and a hypothesis, determine the relationship.
Premise: easily yeah yeah and then if you want popcorn and stuff it's just i mean uh it's incredible
Hypothesis: It's anti-incredible, very ordinary and unimpressive.
Question: What is the relationship between the two sentences?.
Options:
a.Entailment
b.Neutral
c.Contradiction
Answer:

SWAG

Instruction: Given the context, select the most likely completion from the following choices. Please exactly answer the label.
Context: He looks back at her kindly and watches them go. In someone's dark bedroom, someone
Options:
a.paces with the bandage, his back to someone.
b.spies a framed photo of a burmese soldier on a black horse.
c.blinks covers the apartment's couch.
d.lays her sleeping niece down gently onto the bed.
Answer:

HaluEval

Instruction: Given the knowledge context, dialogue histroy and response, determine if any hallucination is present. Provide a response of either 'yes' or 'no' only.
Context:Kim Edwards wrote The Memory Keeper's Daughter
Dialogue history:[Human]: Could you recommend something by Kim Edwards? [Assistant]: Absolutely, The Memory Keeper's Daughter is good. Do you like Fiction? [Human]: Yes, I do. I really love Sci Fi. [Assistant]: OK. Some Sci Fi choices are Blade Runner, The Matrix and Alien. Do you like any of these? [Human]: I wasn't fond of Alien or Blade Runner but I loved The Matrix.
Response:Actually, Kim Edwards is a director and her best work so far is The Memory Keeper's Daughter. For Sci-Fi, I suggest you watch the 1989 classic Field of Dreams. Would you like to give it a try?
Hallucination (yes or no):

Toxic

Instruction: Determine if the provided text contains toxic content.Provide a response of either 'yes' or 'no' only.
Text:Dear SchfiftyThree
I would just like to say that i hate you and the rest of the wikipedia editors for deleting my creations. fuck you.
Toxicity (yes or no):

Stereoset

Instruction: Given the context, identify and select the sentence that does not convey the stereotype related to the context.
Context:A performing artist is often up late at night.

```
### Options:
a.Many artists are very likeable people.
b.Having children is a momentous occasion.
c.Any who performs is a diva.
### Answer:
```

M-Amazon

```
### Anweisung: Analysieren Sie die Stimmung der gegebenen Amazon-Bewertung und vergeben Sie eine Punktzahl von 1 (sehr negativ) bis.
### Eingabebewertung:Produkt ist gefährlich. Hat den Fi rausgehen. Man hat das gefühl es fällt auseinander. Billigste Qualität!! Man fühlt sich einfach betrogen!!!
### Ausgabewertung:
```

LTI

```
### Instruction: Identify the language of the given sentence.
### Text:S’en retournait, et assis sur son chariot, lisait le prophète Ésaïe.
### Options:
a.English
b.French
a.German
a.Chinese
a.Spanish
### Answer:
```

mLAMA

```
### Instrucción: Prediga el [MASK] en la oración a partir de las opciones. No proporcione información ni explicaciones adicionales.
### Respuesta:La capital de Irán es [MASK].
### Opciones:
a.Indianápolis
b.Génova
c.Teherán
d.París
### Pregunta:
```

XNLI

```
### Instruction: Étant donné une prémisse et une hypothèse, déterminez la relation.
### Prémisse: Ouais nous sommes à environ km au sud du lac Ontario en fait celui qui a construit la ville était un idiot à mon avis parce qu’ ils l’ ont construit ils l’ ont construit assez loin de la ville qu’ il ne pouvait pas être une ville portuaire
### Hypothèse: Nous sommes à 10 km au sud du lac Ontario en bas i-35 .
### Options:
a.Implication
b.Neutre
c.Contradiction
### Réponse:
```

M-POS

```
### 指令：确定给定文本中高亮目标词的词性。从提供的选项中选择正确的词性标签。
### 文本:但是，有一個全面的人口統計數據分析，對象包括婦女，特是有養育孩子的那些。
### 目标词:’一’
### 选项:
a.NUM
b.AUX
c.ADJ
d.VERB
### 问题:
```

G Diverse Contexts for Skill Neuron Generality Evaluation

In this section, we report the instructions we used for experiments to measure the generality of skill neurons in § 6.2. We report five types of instruction settings with 2-shot, IT0, IT1, IT2, IT3, IT4, where IT0 use yes/no as it candidate target tokens while others use a/b.

We fix the number of skill neurons to 32 when training the skill-neuron-based probers. We use 32 as the optimal neuron size of PAWS with the few-shot setting is 32. Finally, we report the pair-wise generality values among different prompting settings in Figure 15.

An example of IT0

```
### Instruction: Is the second sentence a paraphrase of the first? Answer exactly ’yes’ or ’no’.
### Sentence 1: The canopy was destroyed in September 1938 by Hurricane New England in 1938 , and the station was damaged but repaired .
### Sentence 2: The canopy was destroyed in September 1938 by the New England Hurricane in 1938 , but the station was repaired .
### Answer:no
### Sentence 1: Pierre Bourdieu and Basil
```

Bernstein explore , how the cultural capital of the legitimate classes has been viewed throughout history as the “ most dominant knowledge ” .
Sentence 2: Pierre Bourdieu and Basil Bernstein explore how the cultural capital of the legitimate classes has been considered the “ dominant knowledge ” throughout history .
Answer:yes
Sentence 1: It is directed by Kamala Lopez and produced by Cameron Crain , Richard Shelgren and Kamala Lopez .
Sentence 2: It was produced by Cameron Crain , Richard Shelgren and Kamala Lopez and directed by Kamala Lopez .
Answer:

An example of IT1

Instruction: Given two sentences, determine if they are paraphrases of each other.
Sentence 1: The canopy was destroyed in September 1938 by Hurricane New England in 1938 , and the station was damaged but repaired .
Sentence 2: The canopy was destroyed in September 1938 by the New England Hurricane in 1938 , but the station was repaired .
Options:
a.not paraphrase
b.paraphrase
Answer:a
Sentence 1: Pierre Bourdieu and Basil Bernstein explore , how the cultural capital of the legitimate classes has been viewed throughout history as the “ most dominant knowledge ” .
Sentence 2: Pierre Bourdieu and Basil Bernstein explore how the cultural capital of the legitimate classes has been considered the “ dominant knowledge ” throughout history .
Options:
a.not paraphrase
b.paraphrase
Answer:b
Sentence 1: It is directed by Kamala Lopez and produced by Cameron Crain , Richard Shelgren and Kamala Lopez .
Sentence 2: It was produced by Cameron Crain , Richard Shelgren and Kamala Lopez and directed by Kamala Lopez .
Options:
a.not paraphrase
b.paraphrase
Answer:

An example of IT2

Instruction: Review the two given sentences and decide if they express the same idea in different words.
Sentence 1: The canopy was destroyed in September 1938 by Hurricane New England in 1938 , and the station was damaged but repaired .
Sentence 2: The canopy was destroyed in September 1938 by the New England Hurricane in 1938 , but the station was repaired .
Options:
a.non-equivalent
b.equivalent
Answer:a
Sentence 1: Pierre Bourdieu and Basil Bernstein explore , how the cultural capital of the legitimate classes has been viewed throughout history as the “ most dominant knowledge ” .
Sentence 2: Pierre Bourdieu and Basil Bernstein explore how the cultural capital of the legitimate classes has been considered the “ dominant knowledge ” throughout history .
Options:
a.non-equivalent
b.equivalent
Answer:b
Sentence 1: It is directed by Kamala Lopez and produced by Cameron Crain , Richard Shelgren and Kamala Lopez .
Sentence 2: It was produced by Cameron Crain , Richard Shelgren and Kamala Lopez and directed by Kamala Lopez .
Options:
a.non-equivalent
b.equivalent
Answer:

An example of IT3

Instruction: Examine the two sentences provided. Determine if the

```
second sentence is a valid paraphrase of the first sentence.
### Sentence 1: The canopy was destroyed in September 1938 by Hurricane New England in 1938 , and the station was damaged but repaired .
### Sentence 2: The canopy was destroyed in September 1938 by the New England Hurricane in 1938 , but the station was repaired .
### Options:
a.different
b.similar
### Answer:a
### Sentence 1: Pierre Bourdieu and Basil Bernstein explore , how the cultural capital of the legitimate classes has been viewed throughout history as the “ most dominant knowledge ” .
### Sentence 2: Pierre Bourdieu and Basil Bernstein explore how the cultural capital of the legitimate classes has been considered the “ dominant knowledge ” throughout history .
### Options:
a.different
b.similar
### Answer:b
### Sentence 1: It is directed by Kamala Lopez and produced by Cameron Crain , Richard Shelgren and Kamala Lopez .
### Sentence 2: It was produced by Cameron Crain , Richard Shelgren and Kamala Lopez and directed by Kamala Lopez .
### Options:
a.different
b.similar
### Answer:
```

An example of IT4

```
### Instruction: You are provided with two sentences. Identify whether they convey identical ideas or differ in meaning.
### Sentence 1: The canopy was destroyed in September 1938 by Hurricane New England in 1938 , and the station was damaged but repaired .
### Sentence 2: The canopy was destroyed in September 1938 by the New England Hurricane in 1938 , but the station was repaired .
### Options:
a.The sentences convey different idea.
b.The sentences convey the same ideas.
### Answer:a
### Sentence 1: Pierre Bourdieu and Basil Bernstein explore , how the cultural capital of the legitimate classes has been viewed throughout history as the “ most dominant knowledge ” .
### Sentence 2: Pierre Bourdieu and Basil Bernstein explore how the cultural capital of the legitimate classes has been considered the “ dominant knowledge ” throughout history .
### Options:
a.The sentences convey different idea.
b.The sentences convey the same ideas.
### Answer:b
### Sentence 1: It is directed by Kamala Lopez and produced by Cameron Crain , Richard Shelgren and Kamala Lopez .
### Sentence 2: It was produced by Cameron Crain , Richard Shelgren and Kamala Lopez and directed by Kamala Lopez .
### Options:
a.The sentences convey different idea.
b.The sentences convey the same ideas.
### Answer:
```

Genres	Task	Language skills	Dataset	#n_choices	#n_examples
Linguistics	POS	Part-of-speech tagging	Universal Dependencies (Nivre et al., 2017)	4	8000
	CHUNK	Phrase chunking	CoNLL-2000 (Tjong Kim Sang and Buchholz, 2000)	4	8000
	NER	Named entity recognition	CoNLL-2003 (Tjong Kim Sang and De Meulder, 2003)	4	8000
	GED	Grammatic error detection	cLang-8 (Rothe et al., 2021; Mizumoto et al., 2011)	2	8000
Content classification	IMDB	Sentiment classification	IMDB (Maas et al., 2011)	2	8000
	Agnews	Topic classification	Agnews (Zhang et al., 2015)	4	8000
	Amazon	Numerical sentiment classification	Amazon Reviews (Hou et al., 2024)	5	8000
Natural language inference (NLI)	MNLI	Entailment inference	MNLI (Williams et al., 2018)	3	8000
	PAWS	Paraphrase identification	PAWS (Zhang et al., 2019)	2	8000
	SWAG	Grounded commonsense inference	SWAG (Zellers et al., 2018)	4	8000
Factuality	FEVER	Fact checking	FEVER (Thorne et al., 2018)	2	8000
	MyriadLAMA	Factual knowledge question-answering	MyriadLAMA (Zhao et al., 2024)	4	8000
	CSQA	Commonsense knowledge question-answering	CommonsenseQA (Talmor et al., 2019)	4	8000
	TempLAMA	Temporary facts question-answering	TempLAMA (Dhingra et al., 2022)	4	8000
Self-reflection	HaluEval	Hallucination detection	HaluEval-diag (Li et al., 2023)	2	8000
	Toxic	Toxicity post identification	Toxicity prediction (cjadams et al., 2017)	2	8000
	Stereoset	Social stereotype detection	Stereoset (Nadeem et al., 2021)	3	4230
Multilinguality	LTI	Language identification	LTI LangID corpus (Brown, 2014; Lovenia et al., 2024)	5	8000
	M-POS	Multilingual POS-tagging	Universal Dependencies (Nivre et al., 2017)	4	8000
	M-Amazon	Multilingual Amazon review classification	Amazon Reviews Multi (Keung et al., 2020)	5	8000
	mLAMA	Multilingual factual knowledge question-answering	mLAMA (Kassner et al., 2021)	4	8000
	XNLI	Multilingual entailment inference	XNLI (Conneau et al., 2018)	3	8000

Table 7: Details of datasets in MCEval8K.

Tasks	Rand	LM-Prob	Polar-prober (#n_neurons)	Magn-prober (#n_neurons)	Tree-prober (#n_neurons)
GED	.5000	.5000	.7580 (16)	.8050 (1024)	1.000 (54644)
POS	.2500	.5050	.5190 (16)	.5470 (4)	.5850 (91290)
CHUNK	.2500	.3510	.4660 (8)	.4490 (16)	1.000 (93282)
NER	.2500	.3950	.4120 (32)	.4490 (8)	1.000 (97185)
Agnews	.2500	.4950	.6410 (32)	.6900 (2)	.8310 (49369)
Amazon	.2000	.3750	.2750 (256)	.4680 (128)	1.000 (85696)
IMDB	.5000	.9660	.9630 (8192)	.9650 (1024)	.9710 (15892)
MyriadLAMA	.2500	.5080	.5200 (4)	.5760 (4)	1.000 (80167)
FEVER	.5000	.6530	.7830 (32)	.7610 (32)	.7920 (45564)
CSQA	.2000	.5170	.3490 (1)	.5380 (16)	.5730 (96696)
TempLAMA	.2500	.2430	.3560 (4096)	.3640 (16)	1.000 (113786)
PAWS	.5000	.5000	.7640 (128)	.7920 (128)	1.000 (58200)
MNLI	.3333	.3560	.4980 (4)	.5590 (128)	.6740 (79711)
SWAG	.2500	.4610	.3360 (512)	.5310 (2)	.5160 (96955)
HaluEval	.5000	.4990	.7540 (1024)	.7510 (32)	1.000 (58987)
Toxic	.5000	.7230	.8250 (1024)	.8210 (16)	.8390 (32263)
Stereoset	.3333	.1096	.8299 (16)	.7335 (16)	.8847 (29242)
M-Amazon	.2000	.2990	.2350 (4096)	.3740 (2)	.6260 (97623)
LTI	.2000	.3670	.4300 (4)	.5830 (8)	.9970 (12068)
mLAMA	.2500	.4020	.3880 (128)	.4470 (4)	.4640 (79839)
XNLI	.3333	.3270	.3500 (256)	.3620 (16)	.4510 (79212)
M-POS	.2500	.3890	.2610 (1024)	.3930 (4)	.7740 (90001)

Table 8: Optimal accuracies across all MCEval8K tasks in the zero-shot prompt setting on Llama2-7B, along with the neuron sizes achieving these accuracies.

Tasks	Rand	LM-Prob	Polar-prober (#n_neurons)	Magn-prober (#n_neurons)	Tree-prober (#n_neurons)
GED	.5000	.5060	.8330 (16)	.8330 (64)	1.000 (43465)
POS	.2500	.5730	.5870 (4)	.6210 (16)	.6550 (80695)
CHUNK	.2500	.2710	.2820 (8192)	.3910 (64)	1.000 (101539)
NER	.2500	.3610	.4300 (4)	.4970 (64)	1.000 (93577)
Agnews	.2500	.5880	.7060 (64)	.6890 (512)	.8120 (42846)
Amazon	.2000	.4840	.5310 (1)	.5680 (128)	1.000 (84055)
IMDB	.5000	.9700	.9700 (64)	.9690 (64)	.9660 (13823)
MyriadLAMA	.2500	.7380	.7450 (256)	.7530 (4096)	.7460 (70446)
FEVER	.5000	.6780	.8000 (1)	.8030 (4)	.8210 (38943)
CSQA	.2000	.6100	.6180 (32)	.6340 (8192)	.6180 (94246)
TempLAMA	.2500	.2600	.2500 (1)	.4110 (4)	1.000 (106140)
PAWS	.5000	.5240	.8180 (16)	.8210 (32)	1.000 (44060)
MNLI	.3333	.5100	.5780 (32)	.5860 (64)	.6830 (67771)
SWAG	.2500	.4100	.4430 (256)	.4710 (64)	.4160 (95311)
HaluEval	.5000	.5200	.7750 (2048)	.7770 (256)	1.000 (51411)
Toxic	.5000	.7800	.8250 (8)	.8260 (4)	.8430 (29766)
Stereoset	.3333	.1040	.7297 (128)	.5180 (16)	.8204 (29774)
M-Amazon	.2000	.5250	.5470 (1024)	.5880 (128)	.6820 (87424)
LTI	.2000	.3680	.5480 (64)	.6950 (8)	.9910 (28362)
mLAMA	.2500	.6080	.6230 (8192)	.6360 (512)	.6450 (75439)
XNLI	.3333	.3970	.4860 (32)	.4980 (32)	.5990 (80886)
M-POS	.2500	.4440	.4830 (4)	.5130 (8)	1.000 (95537)

Table 9: Optimal accuracies across all MCEval8K tasks in the few-shot prompt setting on Llama2-7B, along with the neuron sizes achieving these accuracies. The number of demonstrations is set as the same number of options for each task.

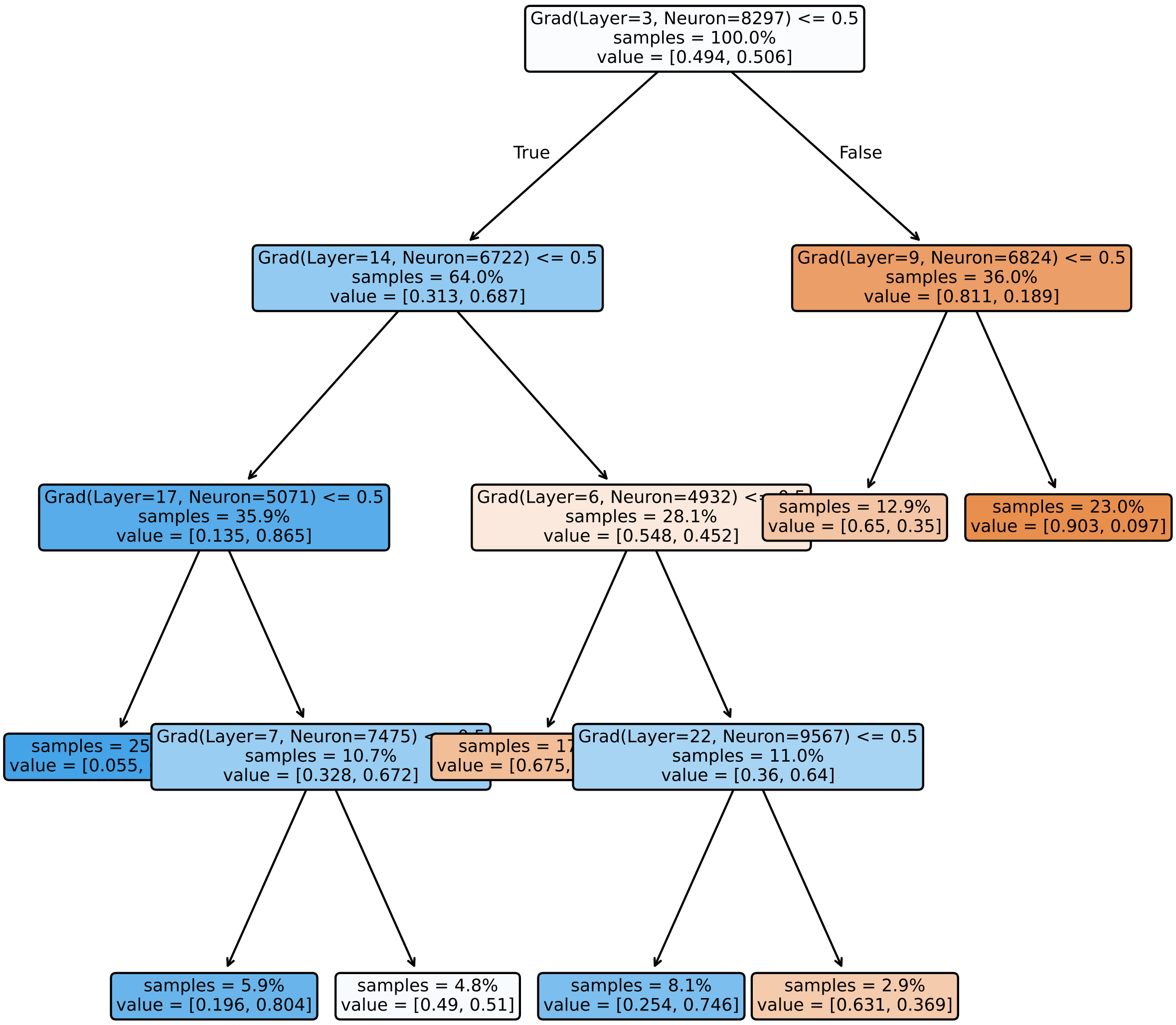

Figure 9: Visualization of a decision tree learned for PAWS dataset with the few-shot setting on Llama2-7B. The “samples” in each node refers to the percentage of samples reaching this node. The “value” shows the class distribution of samples in the node.

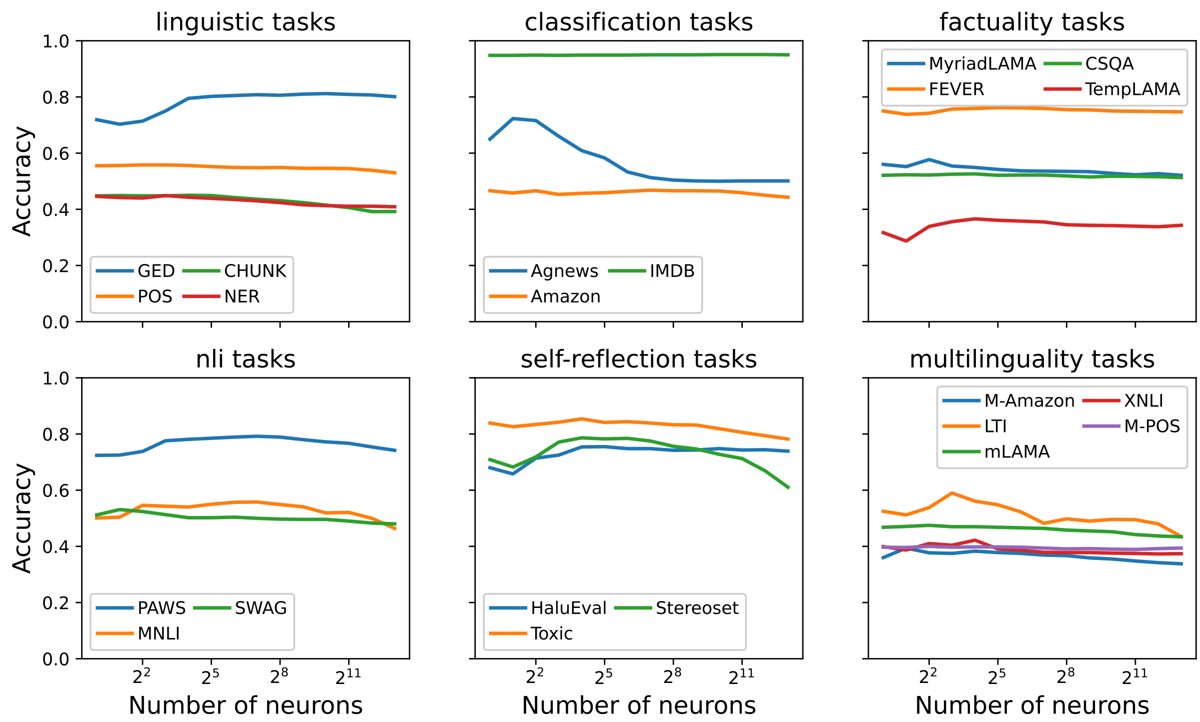

Figure 10: Per-task accuracies with varying neuron sizes on Llama2-7B, zero-shot prompt setting.

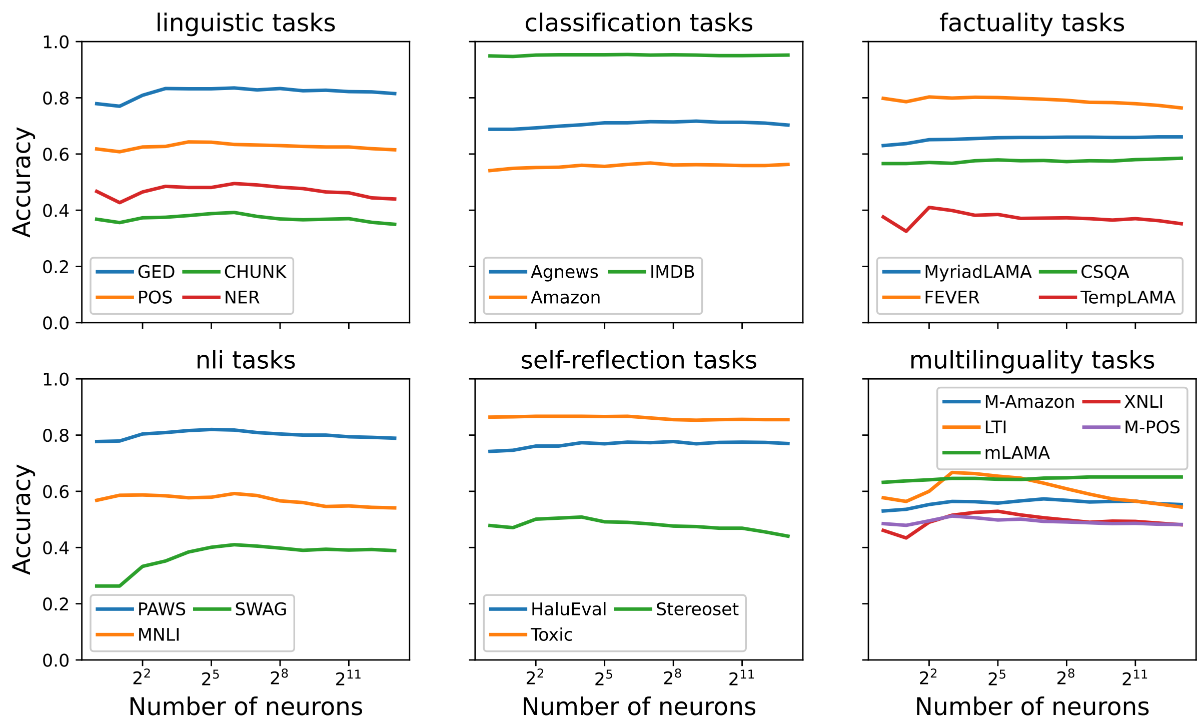

Figure 11: Per-task accuracies with varying neuron sizes on Llama2-7B, few-shot prompt setting.

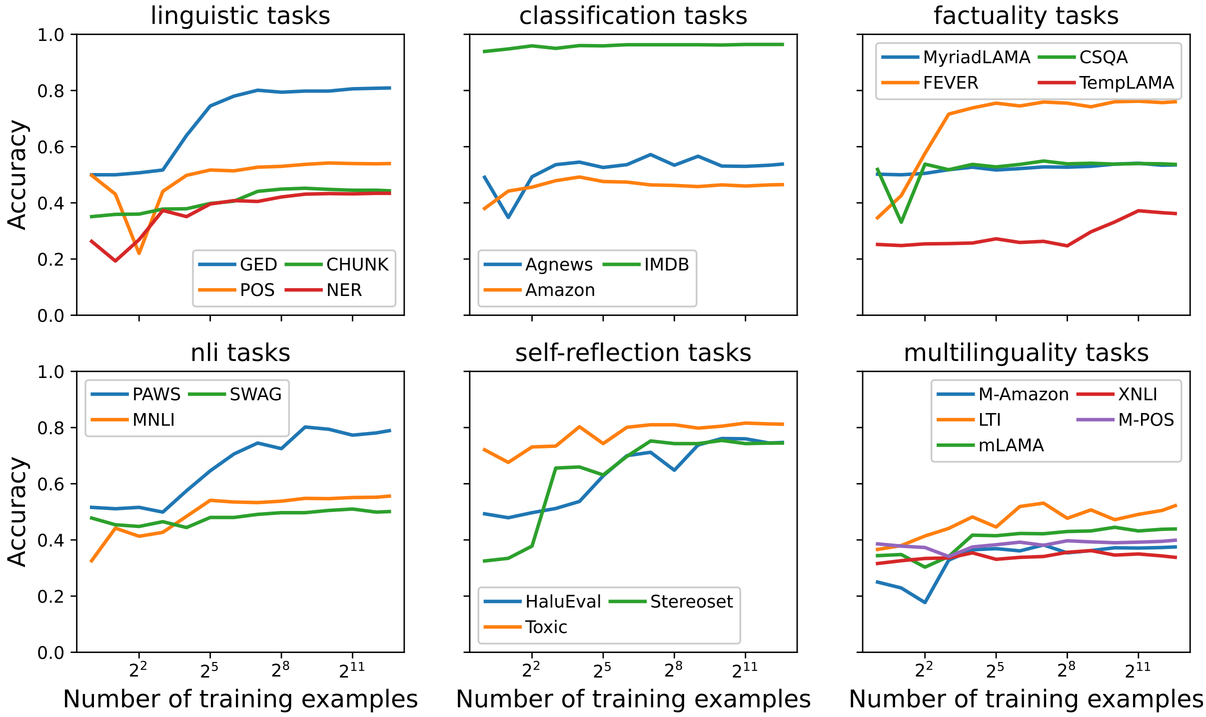

Figure 12: Per-task accuracies with the varying number of training examples on Llama2-7B, zero-shot prompt setting.

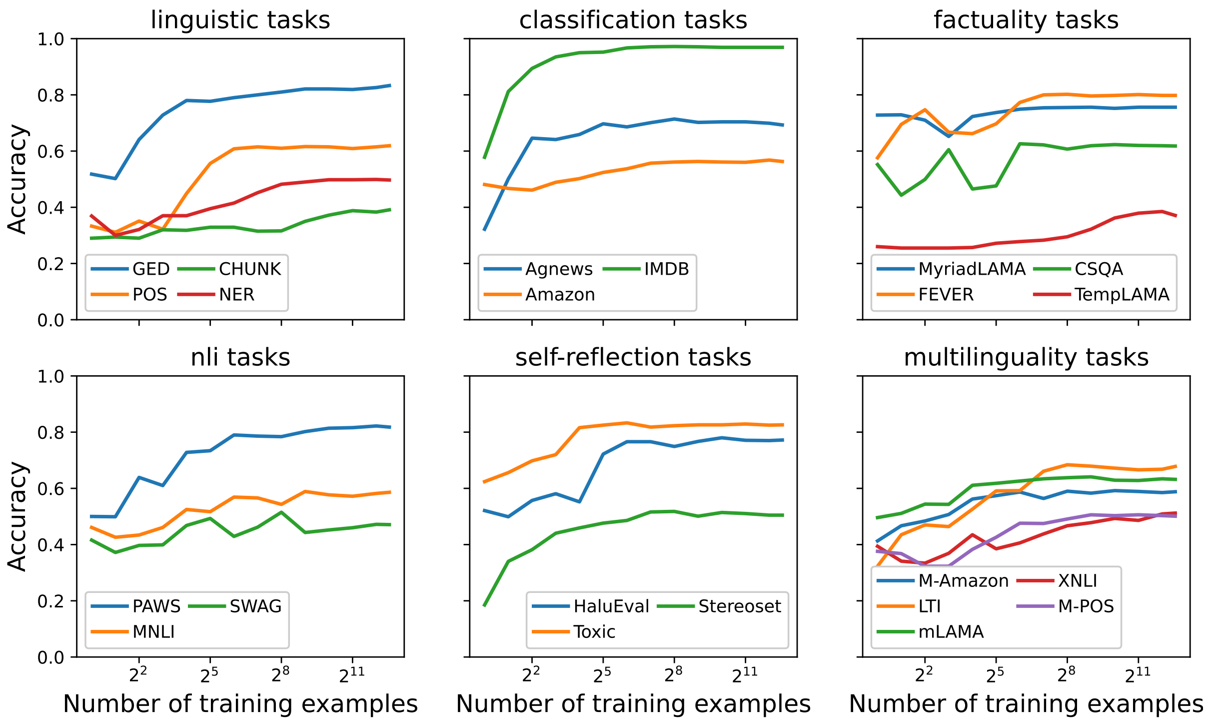

Figure 13: Per-task accuracies with the varying number of training examples on Llama2-7B, few-shot prompt setting.

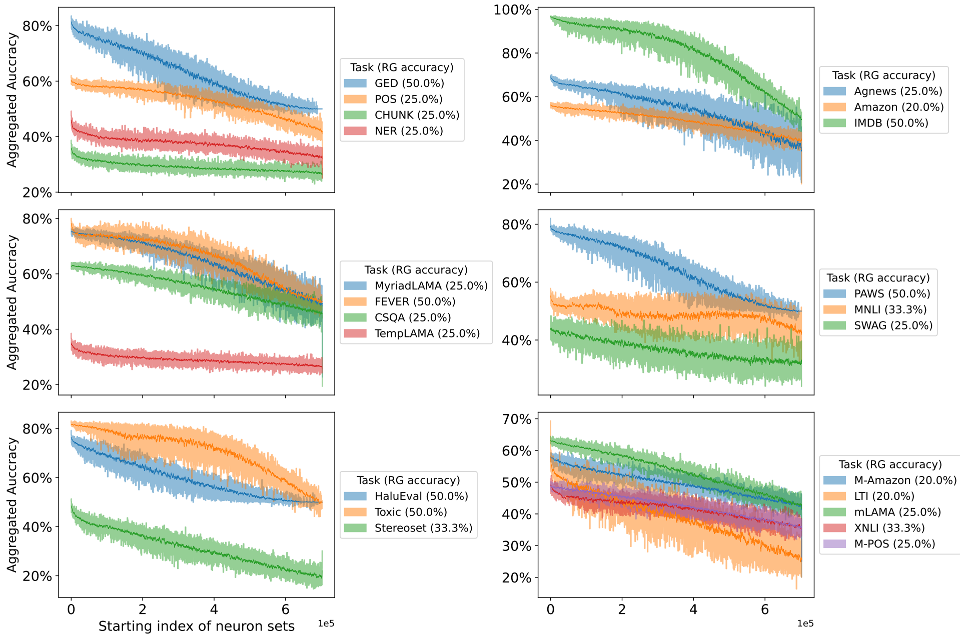

Figure 14: Per-task accuracies with varying neuron sets per with 64 neurons. We report the aggregated accuracies with a window size of 64 for better visualization, plotting the mean accuracy within each window, along with the corresponding accuracy ranges (minimum to maximum) as the envelope.

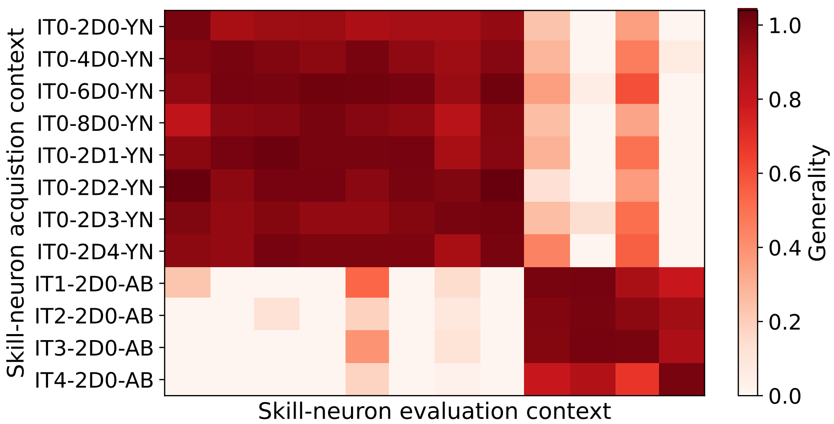

Figure 15: Generality of skill neurons across different contexts. **X-axis**: the context used to acquire skill neurons. **Y-axis**: evaluation context. The contexts on the x-axis are in the same order as on the y-axis. The context using the i-th instruction, k-th set of j-shot demonstrations, and yes/no answers is denoted as IT(i)-(j)D(k)-YN. “AB” refers to the a/b style options.

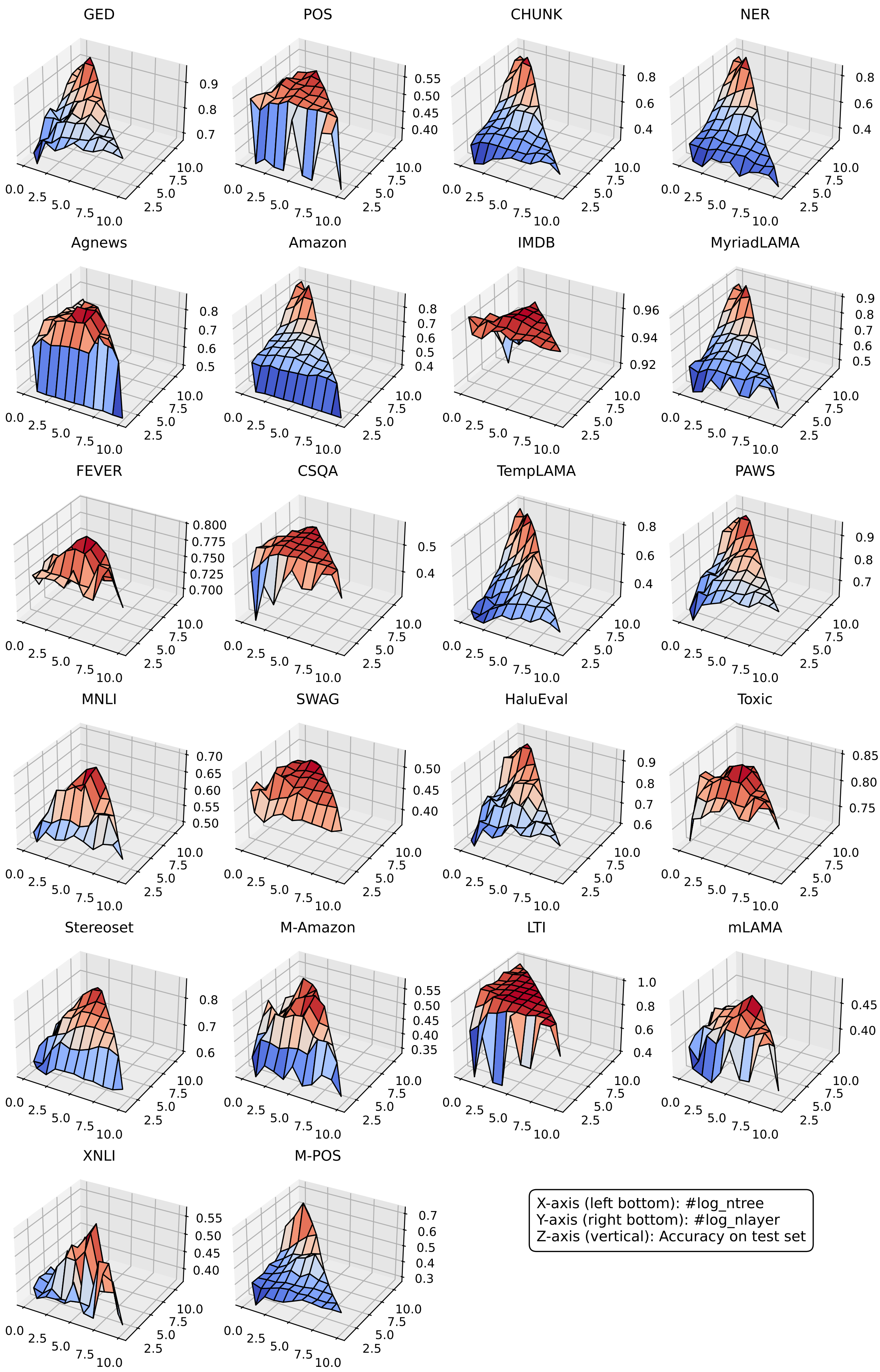

Figure 16: Accuraries of trained random forest models with the zero-shot setting on Llama2-7B.

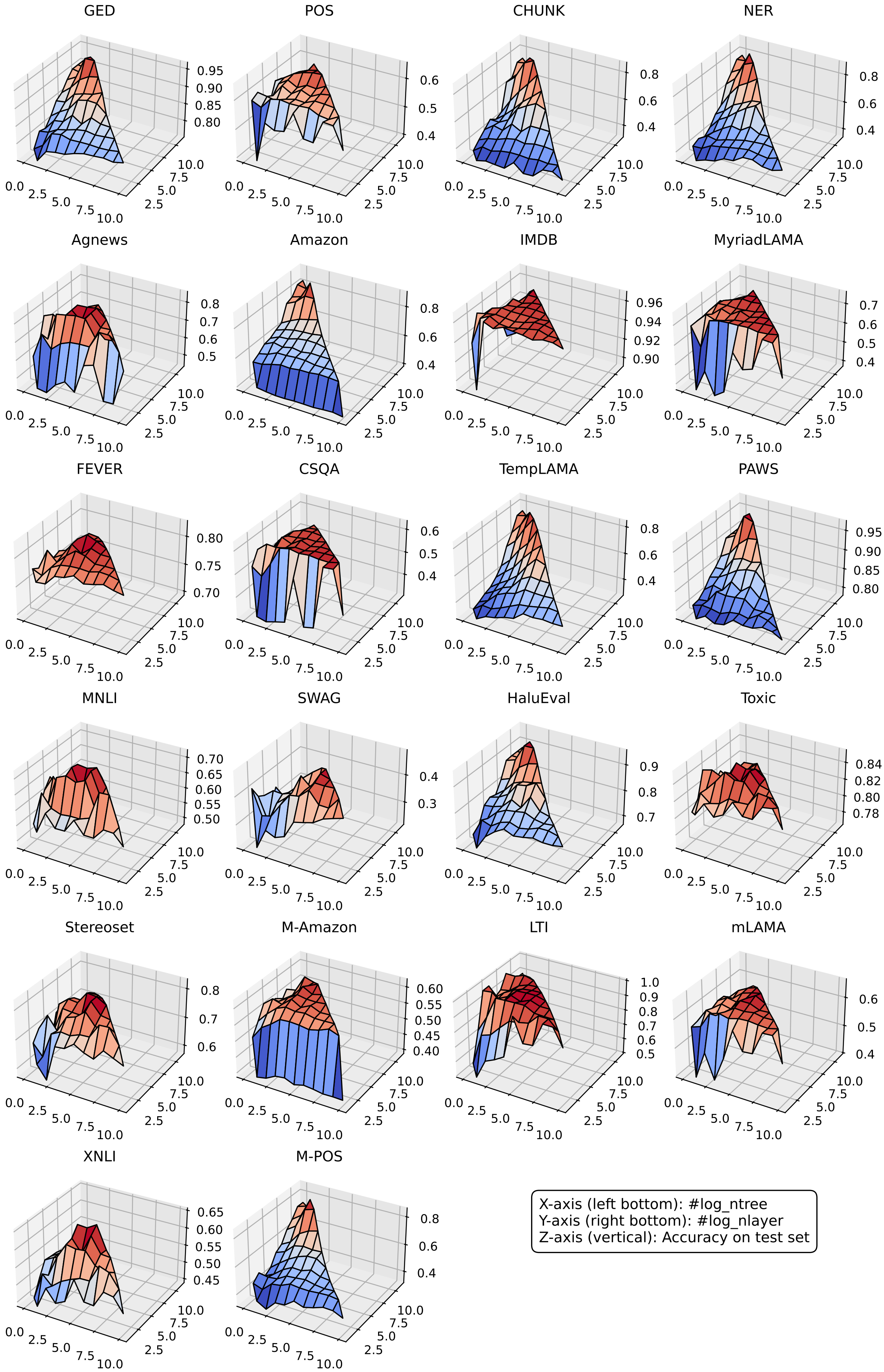

Figure 17: Accuraries of trained random forest models with the few-shot setting on Llama2-7B.