

# NeuronMoE: Neuron-Guided Mixture-of-Experts for Efficient Multilingual LLM Extension

Rongzhi Li<sup>1,2</sup> Hitomi Yanaka<sup>1,2,3</sup>

<sup>1</sup>The University of Tokyo <sup>2</sup>Riken <sup>3</sup>Tohoku University  
{iimori-eiji, hyanaka}@is.s.u-tokyo.ac.jp

## Abstract

Extending large language models to low-resource languages is essential for global accessibility, but training separate models per language is prohibitively expensive. Mixture-of-Experts (MoE) architectures address this by adding sparse language-specific parameters, but determining how many experts each layer needs remains an open question. Current approaches allocate experts based on layer-level similarity, yet language processing exhibits fine-grained specialization at individual neurons. We propose **NeuronMoE**, a method that analyzes language-specific neurons across all transformer components to guide expert allocation per layer based on empirically measured cross-lingual neuron diversity. Applied to Llama-3.2-3B for low-resource languages (Greek, Turkish, and Hungarian), this approach achieves approximately 40% average parameter reduction while matching the performance of the LayerMoE baseline. We find that low-resource language experts independently develop neuron specialization patterns mirroring the high-resource language, which are concentrated in early and late layers. This reveals potential universal architectural principles in how multilingual models organize linguistic knowledge. Our approach generalizes across architectures, as validated on Qwen, and shows that allocation strategy matters more than total expert count.<sup>1</sup>

## 1 Introduction

Extending Large Language Models (LLMs) to support multiple languages is essential for global information access. However, building high-performance models for low-resource languages remains challenging due to data scarcity and computational costs (Joshi et al., 2020). Mixture-of-Experts (MoE) architectures offer a promising solution through sparse expert activation, enabling

<sup>1</sup>Our code is available at <https://github.com/yknlab/NeuronMoE>.

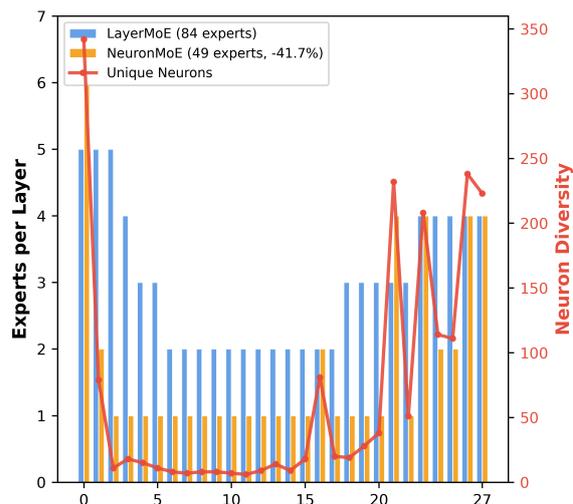


Figure 1: Expert allocation comparison on Llama-3.2-3B for Greek extension. Cross-lingual neuron diversity (red line) exhibits heterogeneous distribution across layers. NeuronMoE allocates experts following this distribution, achieving 41.7% parameter reduction (49 vs 84 experts) compared to LayerMoE’s uniform allocation strategy.

efficient model scaling. Recent work has applied MoE to multilingual and low-resource language extension (Li et al., 2023; Zhu et al., 2024; Zhou et al., 2025; Li et al., 2025), with layer-wise allocation strategies (Zhang et al., 2025) achieving parameter reduction by allocating experts based on cross-lingual similarity. However, similarity-based allocation provides only an indirect signal of language-specific processing requirements, as high similarity does not necessarily mean low capacity needs. They also rely on attention-layer similarity while ignoring MLP layers that comprise two-thirds of model parameters.

On the other hand, studies on language-specific neurons (Durrani et al., 2020; Tang et al., 2024; Kojima et al., 2024) reveal that individual neurons encode language knowledge, with heterogeneous distribution across layers, which is concentrated in

early and late layers and sparse in middle layers. However, their practical application mostly remains limited to observational studies.

This work bridges these two lines of research by using neuron-level language specialization analysis to guide MoE expert allocation. Figure 2 illustrates our approach. We analyze language-specific neurons across all transformer components (both attention and MLP layers) and then allocate experts based on empirically measured specialization needs.

We investigate two questions: (1) **Can neuron-level analysis enable more efficient expert allocation than layer-level similarity?**(2) **How do language-specific neurons develop within MoE experts during multilingual extension?**

To address (1), our experiments demonstrate that neuron-guided allocation on Llama-3.2-3B (Llama Team, 2024) achieves 42% average parameter reduction (47-50 vs. 84 experts) while maintaining comparable performance across different low-resource languages (Greek, Turkish, and Hungarian). By directly measuring language-specific capacity requirements rather than relying on similarity-based allocation, our method concentrates experts in layers with empirically identified high specialization needs. This efficiency generalizes across typologically diverse language families including Indo-European, Turkic, and Uralic. Validation on Qwen-1.5-1.8B (Bai et al., 2023) achieves 50% reduction, confirming that allocation strategy determines parameter efficiency.

Regarding (2), our analysis reveals that target language experts independently develop neuron specialization patterns mirroring those of the source language, whose neurons concentrate in early and late layers despite typological differences, indicating universal architectural principles governing how multilingual models organize linguistic knowledge.<sup>2</sup>

## 2 Related Work

### 2.1 Multilingual Extension via Mixture-of-Experts

Extending pre-trained LLMs to support additional languages presents a fundamental challenge: catastrophic forgetting of original language capabilities during acquisition of new language knowledge. Mixture-of-Experts (MoE) architectures ad-

dress this through sparse expert activation, enabling language-specific adaptation while preserving original capabilities.

Early MoE applications to multilingual scenarios employed uniform expert allocation, assigning identical expert counts across all layers (Lepikhin et al., 2021; Fedus et al., 2022; Du et al., 2022). Komatsuzaki et al. (2023) propose sparse upcycling, converting pre-trained dense models into MoE architectures by initializing experts from dense checkpoints. While these uniform allocation strategies demonstrate strong scaling properties, they do not exploit layer-specific optimization opportunities based on varying linguistic processing requirements across layers.

Zhou et al. (2025) introduce MoE-LPR (Language Priors Routing), a two-stage training framework for multilingual extension. Stage 1 freezes original model parameters while training newly added experts on target language data, preserving base model knowledge. Stage 2 trains routing mechanisms with minimal replay data (less than 1% of Stage 1 data), incorporating language priors to recover original capabilities. While this framework effectively mitigates catastrophic forgetting, it maintains uniform expert allocation across all layers.

Building on MoE-LPR, Zhang et al. (2025) introduce layer-wise allocation based on cross-lingual similarity (hereafter, we call this framework Layer-MoE). They allocate experts inversely proportional to attention-layer similarity: layers with high cross-lingual similarity receive fewer experts, while layers with low similarity receive more experts. This strategy achieves 60% parameter reduction over uniform allocation in single-language extension and 33.3% in continual multilingual scenarios.

However, their approach faces limitations that motivate our work. Their similarity computation considers only attention layers, omitting MLP layers that comprise two-thirds of model parameters and encode substantial linguistic knowledge (Durani et al., 2020). Layer-level similarity provides coarse allocation signals that may not capture fine-grained language-specific processing requirements within layers. Moreover, their similarity-based allocation strategy lacks grounding in empirically measured language specialization patterns identified based on neuron-level analysis.

We address these limitations through neuron-level language specialization analysis across all transformer components, providing more direct

<sup>2</sup>Code is available at <https://github.com/ynklab/NeuronMoE>

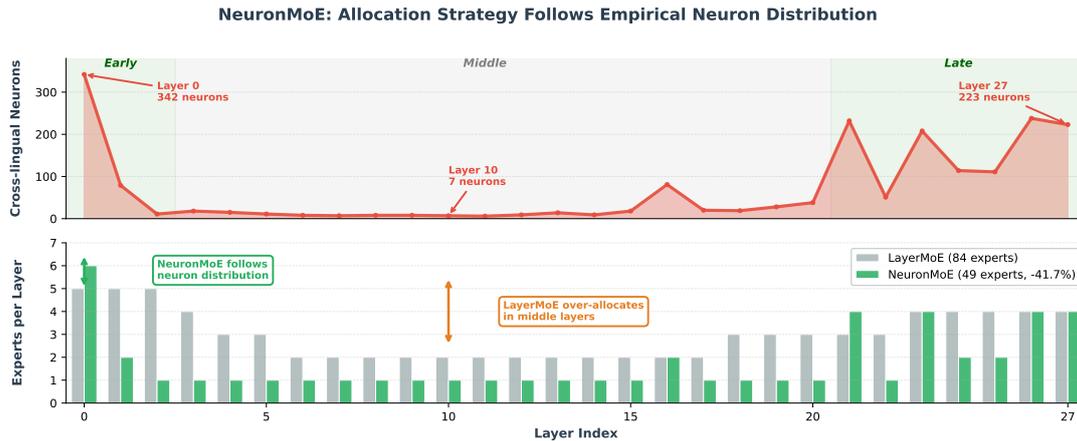


Figure 2: Overview of NeuronMoE’s core insight and allocation strategy. **Top:** Cross-lingual neuron diversity exhibits dramatic heterogeneity across layers. For example, Layer 0 contains 342 unique language-specific neurons while Layer 10 contains only 7, demonstrating that different layers have vastly different language-specific processing requirements. **Bottom:** NeuronMoE allocates experts following this empirical distribution (49 experts total), concentrating capacity where specialization occurs. In contrast, LayerMoE’s similarity-based approach over-allocates in middle layers (84 experts total). This neuron-guided strategy achieves 41.7% parameter reduction while maintaining comparable performance.

and comprehensive signals for expert allocation decisions.

## 2.2 Language-Specific Neurons in Multilingual Models

A parallel line of research investigates how multilingual models encode language-specific knowledge at the neuron level, revealing fine-grained specialization patterns within model architectures.

Durrani et al. (2020) pioneer neuron-level analysis of pre-trained models, demonstrating that core linguistic properties can be predicted from small neuron subsets, suggesting concentrated language encoding. Recent work extends this to decoder-only LLMs with varying methodological approaches for identifying language-specific neurons.

**Identification Methods:** Tang et al. (2024) propose Language Activation Probability Entropy (LAPE) for MLP layers, measuring activation probability entropy across language-specific inputs. Neurons exhibiting low entropy for specific languages are identified as language-specific. Kojima et al. (2024) define language-specific neurons as those exhibiting statistically significant activation patterns for individual languages across both attention and MLP layers, measuring significance through Average Precision scores. Their analysis reveals minimal overlap between languages, with each language developing distinct specialized neu-

ron populations.

**Layer-wise Heterogeneity:** Despite methodological differences, these works converge on a critical observation: language specialization exhibits heterogeneous distribution across layers. Tang et al. (2024); Kojima et al. (2024) demonstrate that language-specific neurons concentrate in early layers and late layers, while middle layers contain minimal language-specific neurons, suggesting abstract, language-agnostic processing. This heterogeneous pattern has implications for architectural design: layers with high language specialization may benefit from multiple experts to accommodate diverse linguistic patterns, while language-agnostic layers may require fewer parameters.

**Current Applications and Limitations:** Recent work explores multilingual activation patterns (Liu et al., 2024) and linguistic neuron overlap for cross-lingual transfer (Chen et al., 2025; Xu et al., 2025), though Mondal et al. (2025) question their utility for transfer. Practical applications have been limited to observational studies, model interpretation, and output manipulation. Leveraging neuron-level signals to guide architectural design decisions, such as expert allocation in MoE, remains unexplored.

**Bridging Neuron Analysis and MoE Allocation:** The observed heterogeneous distribution of language-specific neurons provides a direct signal for layer-specific expert allocation. Unlike layer-

wise similarity, which indirectly infers capacity needs by measuring representation alignment, neuron counting directly quantifies how many specialized units each layer requires for multilingual processing. This key insight motivates our approach: if a layer contains 342 unique language-specific neurons across two languages while another contains only 7, the former empirically requires greater capacity for language-specific computation. By measuring this distribution before MoE training, we can allocate experts proportionally to empirically observed specialization rather than relying on similarity-based allocation.

### 3 Method

We bridge neuron-level analysis and MoE architecture design by using empirically measured language specialization patterns to guide expert allocation. Our framework is compatible with any neuron specificity metric that quantifies language-specific activation patterns. In this work, we adopt Kojima et al. (2024)’s Average Precision method for its comprehensive coverage of both attention and MLP layers; alternative metrics such as LAPE (Tang et al., 2024) could also be applied. Rather than relying on indirect measures such as attention-layer similarity, we directly measure cross-lingual neuron diversity to determine layer-specific expert requirements.

#### 3.1 Neuron Specialization Measurement

**Language-Specific Neuron Definition:** Following Kojima et al. (2024), we define language-specific neurons as neurons that exhibit statistically significant activation patterns for individual languages. For a given neuron  $n$  in layer  $l$  and language  $\ell$ , we compute activation values  $\{a_n^{(i)}\}$  across a corpus of samples  $\{x^{(i)}\}$ , where  $x^{(i)}$  is labeled with language  $y^{(i)} \in \{\ell_1, \dots, \ell_K\}$ .

The language specificity of a neuron  $n$  for language  $\ell$  is measured using Average Precision (AP). AP represents the degree to which samples of language  $\ell$  are concentrated at the top when sorted by activation values in descending order:

$$\text{AP}(n, \ell) = \frac{1}{|\{i : y^{(i)} = \ell\}|} \sum_{k=1}^N \text{P}(k) \cdot \mathbf{1}[y^{(\pi(k))} = \ell]$$

where  $\pi$  is the descending sort of activation values, and  $\text{P}(k)$  is the proportion of language  $\ell$  in the top

$k$  items. A neuron is considered more specific to language  $\ell$  as AP approaches 1.0.

**Cross-Lingual Neuron Diversity for Expert Allocation:** Following Kojima et al. (2024), we identify the top 1000 language-specific neurons per language across all layers based on their AP scores, using  $\text{AP} > 0.5$  as the threshold. We then determine expert allocation based on the diversity of language-specific neurons within each layer across the source language and a single target language. The layer specialization score  $\mathcal{S}_l$  is defined as the total number of unique language-specific neurons in layer  $l$ , aggregated across the language pair. For Greek extension, this is formulated as:

$$\mathcal{S}_l = \left| \bigcup_{\text{lang} \in \{\text{en}, \text{el}\}} N_{l, \text{lang}} \right|$$

where  $N_{l, \text{lang}}$  is the set of language-specific neurons for layer  $l$  and language  $\text{lang}$ . This score reflects the total pool of specialized neurons a layer needs to handle both languages. The same formulation applies to Turkish (en, tr) and Hungarian (en, hu) extensions.

#### 3.2 Expert Allocation Strategy

The number of experts for each layer,  $E_l$ , is determined by linearly scaling the unique neuron count  $\mathcal{S}_l$  between a predefined minimum and maximum number of experts.

First, the unique neuron counts are normalized to a  $[0, 1]$  range:

$$\text{norm}(\mathcal{S}_l) = \frac{\mathcal{S}_l - \min(\mathcal{S})}{\max(\mathcal{S}) - \min(\mathcal{S})}$$

Then, the number of experts is calculated as:

$$E_l = E_{\min} + \text{round}(\text{norm}(\mathcal{S}_l) \times (E_{\max} - E_{\min}))$$

where  $E_{\min}$  and  $E_{\max}$  are the minimum and maximum allowed experts per layer. This data-driven approach allocates more experts to layers with a greater diversity of language-specific neurons.

#### 3.3 Two-Stage Training Process

Following the MoE-LPR framework (Zhou et al., 2025), we adopt a two-stage training approach where our neuron-guided allocation replaces the similarity-based allocation for determining expert counts per layer.

**Stage 1 (Expert Initialization):** We freeze the original pre-trained model parameters and add new MoE experts to each layer according to our neuron-guided allocation strategy. The model is then trained on target language data (Greek, Turkish, or Hungarian) to initialize these experts. At this stage, expert counts are determined by the layer specialization scores  $\mathcal{S}_l$  computed from cross-lingual neuron diversity while LayerMoE (Zhang et al., 2025) uses attention-layer similarities for the expert allocation. This stage focuses on adapting experts to the new language while preserving original parameters.

**Stage 2 (Router Training):** Following LayerMoE (Zhang et al., 2025), we train the routing mechanism using a small amount of replay data from the source languages mixed with target language data, incorporating language priors to recover original language capabilities and refine expert selection. The key difference between our method and LayerMoE lies solely in Stage 1’s allocation strategy; Stage 2 routing training is identical for both methods.

## 4 Experimental Setup

To enable direct comparison with LayerMoE (Zhang et al., 2025), we adopt their experimental setup: Llama-3.2-3B (28 layers) (Llama Team, 2024) as our primary model and Qwen-1.5-1.8B (24 layers) (Bai et al., 2023) for cross-architecture validation. Following their language selection, we extend these models to Greek (EL), Turkish (TR), and Hungarian (HU) using 2B tokens per language from CulturaX. These languages represent diverse families: Indo-European, Turkic, and Uralic. We evaluate using the lm-evaluation-harness framework (Gao et al., 2024), which provides multilingual versions of ARC Challenge, MMLU, and HellaSwag translated by Okapi (Lai et al., 2023). We also use Belebele (Bandarkar et al., 2024) for reading comprehension. Evaluation settings are 5-shot for MMLU, 25-shot for ARC Challenge, and 0-shot for HellaSwag and Belebele. All results are reported as single-run evaluations.

We conduct experiments across three target languages and two model architectures:

**Greek extension:** Full evaluation on Llama-3.2-3B covering both training stages, plus cross-architecture validation on Qwen-1.5-1.8B (Stage 1 only). Baselines include Dense (no MoE), LayerMoE-S1/LayerMoE (84 experts for Llama,

72 for Qwen, using attention-layer similarity allocation (Zhang et al., 2025)), and NeuronMoE variants (49 experts for Llama, 36 for Qwen). We include an ablation NeuronMoE-EN (37 experts) that allocates based solely on English neuron distributions, testing whether cross-lingual neuron analysis is essential.

**Turkish and Hungarian extensions:** Validation experiments on Llama-3.2-3B at Stage 1 only, comparing Dense, LayerMoE (84 experts), and NeuronMoE (50 and 47 experts respectively) to assess cross-lingual generalization across typologically diverse language families (Indo-European, Turkic, Uralic).

All models are trained using the AdamW optimizer with learning rate  $1e-4$ , cosine scheduler, batch size 32, and 15K training steps. Full hyperparameters are detailed in Appendix D. The neuron analysis preprocessing requires approximately 6 minutes per language on a single NVIDIA GH200 GPU.

## 5 Results

### 5.1 Greek Extension on Llama-3.2-3B

Table 1 presents results for Greek extension on Llama-3.2-3B across both training stages. NeuronMoE-S1 achieves 41.7% parameter reduction with 49 experts compared to LayerMoE-S1’s 84, while mitigating catastrophic forgetting in English with consistent +0.5-2% gains across benchmarks and maintaining comparable Greek performance. After Stage 2 routing training, NeuronMoE maintains near-Dense English performance while providing substantial Greek capability (+3.09% over Dense baseline on ARC), demonstrating successful multilingual extension without sacrificing original language proficiency.

NeuronMoE achieves 41.7% parameter reduction at the cost of 1-2.8% target language performance on specific benchmarks. The tradeoffs are task-dependent: ARC Challenge shows 2.0-2.5% degradation, while language understanding tasks such as Belebele, HellaSwag, and MMLU show smaller gaps of 0.4-2.8%. This pattern reflects our allocation strategy that concentrates experts in early/late layers handling language-specific processing while minimizing allocation in middle layers performing abstract reasoning.

Model	#Exp	ARC		Belebele		HellaSwag		MMLU	
		EN	EL	EN	EL	EN	EL	EN	EL
<i>Llama-3.2-3B (28 layers):</i>									
Dense	-	51.11	31.93	74.11	65.22	76.33	43.38	56.45	41.17
<i>Stage 1 (Expert Initialization):</i>									
LayerMoE-S1	84	47.27	35.19	73.89	66.33	75.84	52.58	55.40	43.41
NeuronMoE-EN	37 (-56%)	49.15	33.13	73.56	63.67	76.45	49.03	56.89	43.26
NeuronMoE-S1	49 (-42%)	49.15	34.16	75.33	63.56	76.34	50.69	56.61	43.95
<i>Stage 2 (Routing Training):</i>									
LayerMoE	84	49.32	37.50	74.78	67.33	76.50	52.41	55.79	44.06
NeuronMoE	49 (-42%)	50.17	35.02	75.11	64.56	76.53	50.53	56.48	43.66
<i>Qwen-1.5-1.8B (24 layers, Stage 1 only):</i>									
LayerMoE-S1	72	37.29	25.26	54.89	31.44	59.88	35.21	45.46	31.26
NeuronMoE-S1	36 (-50%)	37.88	23.80	58.67	32.00	61.20	31.26	45.46	29.23

Table 1: Performance on Greek extension across Llama-3.2-3B and Qwen-1.5-1.8B. #Exp indicates the number of experts. NeuronMoE achieves 41.7-50.0% parameter reduction with comparable performance, demonstrating cross-architecture generalization.

## 5.2 Ablation Study: Importance of Target Language Analysis

The NeuronMoE-EN ablation allocates 37 experts using only English neuron distribution. Despite achieving strong English performance at 49.15% on ARC, Greek performance drops to 33.13% compared to 34.16% for full NeuronMoE. This confirms that effective MoE allocation requires analyzing neuron specialization patterns for the target language, not just source language patterns.

## 5.3 Cross-Architecture Validation

Qwen-1.5-1.8B results in Table 1 demonstrate that neuron-guided allocation generalizes across architectures. Despite architectural differences between the 24-layer Qwen and 28-layer Llama with different hidden dimensions, NeuronMoE achieves 50.0% parameter reduction. At Stage 1, Qwen NeuronMoE-S1 shows strong English improvements over LayerMoE-S1 on Belebele (+3.78%) and HellaSwag (+1.32%), paralleling Llama results. Greek ARC performance decreases 1.46% compared to LayerMoE-S1, suggesting that while neuron-level specialization patterns reflect universal principles, smaller models may exhibit different sensitivities for reasoning tasks.

## 5.4 Cross-Lingual Generalization

To evaluate cross-lingual generalization, we extend Llama-3.2-3B to Turkish and Hungarian using the same neuron-guided allocation strategy (50 and 47 experts respectively). Table 3 shows that NeuronMoE successfully extends to both languages while maintaining approximately 2% ARC degradation

Layers	LayerMoE	NeuronMoE
0-2 (Early)	5,5,5	6,2,1
3-10 (Early-Mid)	4,3,3,2,2,2,2,2	1,1,1,1,1,1,1,1
11-18 (Mid-Late)	2,2,2,2,2,2,2,3	1,1,1,1,1,1,1,2
19-27 (Late)	3,3,3,3,4,4,4,4,4	1,1,4,1,4,2,2,4,4
<b>Total</b>	<b>84</b>	<b>49</b>

Table 2: Layer-wise expert allocation comparison (Llama-3.2-3B, Greek extension). NeuronMoE achieves 41.7% parameter reduction by concentrating experts in high-diversity layers while minimizing allocation in middle layers.

compared to LayerMoE (84 experts), consistent with the Greek results. Performance remains substantially above Dense baseline (+0.94% for Turkish, +2.82% for Hungarian on ARC). The consistent performance-efficiency tradeoff pattern across three typologically diverse language families (Indo-European, Turkic, Uralic) validates cross-lingual generalization.

## 5.5 Analysis of Allocation Strategy

Table 5 in Appendix A shows cross-lingual neuron diversity across layers. Early layers exhibit high neuron counts, with L0 containing 342 unique neurons for EN+EL. Late layers show moderate to high counts ranging from 111 to 238 neurons at L21 and L23-27. Middle layers contain minimal specialization, with L7 and L10 having only 7 neurons each.

Table 2 compares layer-wise expert allocation between LayerMoE and NeuronMoE. LayerMoE employs gradual reduction (5→2→3→4), while NeuronMoE shows substantial reduction in middle

Language	Model	#Exp	ARC		Belebele		HellaSwag		MMLU	
			EN	Target	EN	Target	EN	Target	EN	Target
Turkish	Dense	-	51.11	33.56	74.11	59.11	76.33	42.80	56.45	42.35
	LayerMoE	84	49.91	36.55	75.33	61.78	76.28	49.60	56.05	43.51
	NeuronMoE	50 (-40%)	49.83	34.50	74.44	61.33	76.22	47.83	56.21	43.37
Hungarian	Dense	-	51.11	34.85	74.11	59.67	76.33	44.63	56.45	43.68
	LayerMoE	84	49.91	39.73	75.33	64.22	76.28	51.66	56.05	44.86
	NeuronMoE	47 (-44%)	50.43	37.67	74.78	60.11	76.47	49.90	56.10	44.30

Table 3: Stage 1 results for Turkish and Hungarian extension on Llama-3.2-3B. NeuronMoE achieves substantial parameter efficiency while successfully extending to additional low-resource languages. All values in %. Results for LayerMoE baseline are from Zhang et al. (2025).

layers (average 1.1 experts for layers 3-20) with selective concentration in early (layer 0: 6 experts) and late layers (layers 21, 23-27: 2-4 experts).

Performance varies systematically across benchmark types. ARC Challenge (commonsense reasoning) exhibits the largest gaps (2.0-2.5% degradation vs. LayerMoE), while language understanding tasks show smaller differences: Belebele (0.5-2.8%), HellaSwag (1.8-3.4%), and MMLU (0.1-0.4%). This pattern reflects our allocation strategy that minimizes experts in middle layers performing abstract reasoning while concentrating capacity in early/late layers handling language-specific processing.

## 6 Analysis

To understand the mechanisms underlying neuron-guided allocation’s efficiency, we analyze language-specific neuron distributions in trained MoE models. We examine post-training expert specialization patterns, compare them with source language distributions, and validate the empirical basis for our allocation strategy.

### 6.1 Language-Specific Neuron Distribution in MoE Experts

We analyze language-specific neuron development in Greek experts after MoE extension using Average Precision (AP) scores.

**Analysis Method:** Measuring language-specific neurons in sparse MoE architectures requires addressing the challenge that each expert processes only token subsets determined by routing decisions. Thus we extend the methodology of Kojima et al. (2024) by tracking routing information  $R_l = \{\text{expert\_ids}, \text{routing\_weights}\}$  at each layer and aggregating token-level activations to sample-

level representations per expert:

$$h_e^{(s)} = \frac{1}{|T_e^{(s)}|} \sum_{t \in T_e^{(s)}} a_e^{(t)}$$

where  $T_e^{(s)}$  denotes the set of tokens from sample  $s$  routed to expert  $e$ , and  $a_e^{(t)}$  represents expert  $e$ ’s activation vector for token  $t$ .

Following the AP algorithm, we measure how strongly each neuron correlates with specific languages. For neuron  $n$  in expert  $e$ , we collect its activation values  $h_{e,n}^{(1)}, h_{e,n}^{(2)}, \dots, h_{e,n}^{(N)}$  across all  $N$  samples and compute:

$$\text{AP}_{\text{exp}}(n, e, \ell) = \text{AP}(h_{e,n}^{(1:N)}, y_\ell^{(1:N)})$$

where  $y_\ell^{(i)} \in \{0, 1\}$  indicates whether sample  $i$  belongs to language  $\ell$ . High AP scores approaching 1.0 identify neurons that consistently activate strongly for specific languages. We define high-AP neurons as those with  $\text{AP} \geq 0.9$ , representing strong language specialization. These scores quantify language-specific capacity that develops during MoE training, enabling us to validate whether our allocation strategy correctly identifies layers requiring expert capacity.

**Results:** Table 4 quantifies high-AP neuron counts per expert across selected layers. In layer 0 (6 experts allocated), expert\_3 contains 61 Greek-specific neurons (0.31% of layer neurons), expert\_1 contains 27 neurons (0.14%), while remaining experts show negligible specialization. In layer 27 (4 experts allocated), expert\_2 exhibits concentrated specialization with 405 Greek-specific neurons (2.08%), whereas expert\_0 and expert\_1 contain zero Greek-specific neurons. Middle layers (3-20) with single-expert allocation exhibit minimal language-specific neuron development, with

Layer-Expert	High-AP	Ratio (%)
L0 base_layer	0	0.00
L0 expert_0	0	0.00
L0 expert_1	27	0.14
L0 expert_2	0	0.00
L0 expert_3	61	0.31
L0 expert_4	19	0.10
<hr/>		
L26 base_layer	0	0.00
L26 expert_0	278	1.43
L26 expert_1	154	0.79
L26 expert_2	0	0.00
<hr/>		
L27 base_layer	179	0.92
L27 expert_0	0	0.00
L27 expert_1	0	0.00
L27 expert_2	405	2.08

Table 4: High-AP neuron counts for Greek across experts in selected layers (total neurons per layer/expert: 19,456). Multi-expert layers show distinct specialization patterns, validating allocation decisions.

most layers containing fewer than 10 specialized neurons total.

Figure 3 visualizes this heterogeneous distribution: high-AP neuron ratios concentrate in early layers (0-2) and late layers (21, 23-27), while middle layers show minimal language-specific processing across all experts.

## 6.2 Implications for Allocation Generalization

The observed convergence provides an explanation for why neuron-guided allocation generalizes across languages and architectures. Our allocation strategy derives from analyzing source language (English) neuron distributions to determine per-layer expert counts. Post-training analysis reveals the mechanism behind this convergence: Greek-specific neurons in middle layers remain minimal (Figure 3 shows base layer and experts contain 0.0-0.4% ratios for layers 16-23), while late layers develop concentrated Greek specialization (Expert\_0 has 1.4% in layer 26, Expert\_2 reaches 2.1% in layer 27).

This pattern demonstrates that convergence occurs not through middle layers adapting to match English distribution, but through target languages developing their own late-layer concentration that complements the existing early-layer focus. Greek maintains its language-specific characteristics in middle layers while adding substantial capacity in late layers (Table 6), resulting in a distribution pattern similar to English despite different absolute values and layer-specific peaks.

This mechanism indicates our method ex-

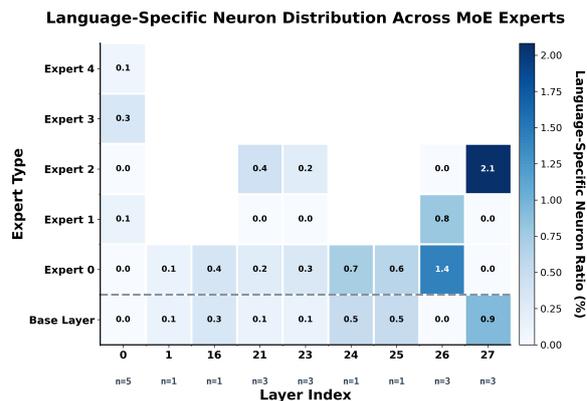


Figure 3: Heatmap of high-AP neuron ratios for Greek across all experts. Color intensity indicates Greek specialization strength (red: high, blue: low/none). Multi-expert layers (0, 21, 23-27) show distinct specialization patterns across experts, while single-expert middle layers remain largely unspecialized, validating our neuron-guided allocation strategy.

ploits the principles: transformer models organize language-specific processing at layer boundaries (early for input encoding, late for output generation) while middle layers handle language-agnostic abstract reasoning. The strategy concentrates experts where linguistic specialization empirically manifests, explaining the 40% parameter reduction while maintaining performance across typologically diverse languages.

## 7 Discussion

Our 40-50% parameter reduction stems from identifying that middle layers require minimal language-specific capacity, enabling single-expert allocation where LayerMoE allocates 2-3 experts. The performance tradeoffs are task-dependent: ARC Challenge (commonsense reasoning) shows 2.0-2.5% degradation, while Belebele and MMLU (language understanding) show smaller gaps (0.5-2.8% and 0.1-0.4% respectively), and HellaSwag (reading comprehension) shows intermediate degradation (1.8-3.4%). This pattern suggests middle layer reduction disproportionately affects commonsense reasoning over language understanding tasks. The convergence of specialization patterns across typologically diverse languages provides evidence for universal functional organization, implying practitioners can leverage allocation strategies from high-resource languages when extending to new languages.

## 8 Conclusion

We propose neuron-guided MoE allocation for efficient multilingual LLM extension. Unlike layer-wise similarity-based approaches, we directly measure language-specific neurons across all transformer components to guide expert allocation based on empirical specialization patterns. Experiments on Llama-3.2-3B (Greek, Turkish, Hungarian) and Qwen-1.5-1.8B achieve 40-50% parameter reduction with comparable performance. The key finding is that early and late layers concentrate language-specific neurons requiring multiple experts, while middle layers show minimal specialization enabling single-expert allocation. Target languages independently develop similar patterns despite typological differences, indicating potential universal architectural principles that explain generalization across languages and architectures.

## Limitations

Our experiments cover three typologically diverse languages from Indo-European, Turkic, and Uralic families. While our current study covers typologically diverse languages, extending validation to broader language families such as Indo-Aryan (e.g., Bengali, Nepali), Sino-Tibetan, and Niger-Congo would be a valuable direction for future research. Regarding the hyperparameters  $E_{min}$  and  $E_{max}$  used in expert allocation scaling, our ablation study (NeuronMoE-EN) suggests that performance is relatively robust to allocation variations; however, further sensitivity analysis could provide additional insights into optimal hyperparameter selection. Neuron analysis requires one-time preprocessing to measure language-specific activations, though this cost amortizes across subsequent training runs for the same architecture and language pair. Finally, since our evaluation is limited to multiple-choice QA tasks, generalization to other tasks is not guaranteed.

## Acknowledgements

We thank the three anonymous reviewers for their helpful comments and feedback. We also thank Taisei Yamamoto, Xiaotian Wang and Daiki Mat-suoka for their insightful discussions. This work was supported by JST CREST Grant Number JP-MJCR2565, Japan and JST BOOST Program Grant Number JPMJBY24H5, Japan.

## References

- Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, and 29 others. 2023. [Qwen technical report](#). *CoRR*, abs/2309.16609.
- Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. 2024. [The belebele benchmark: a parallel reading comprehension dataset in 122 language variants](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 749–775, Bangkok, Thailand. Association for Computational Linguistics.
- Zhihao Chen, Yufei Zhang, Chengming Liu, Bei Chen, and Wai Lam. 2025. [Linguistic neuron overlap patterns to facilitate cross-lingual transfer on low-resource languages](#). *arXiv preprint arXiv:2508.17078*.
- Nan Du, Yanping Huang, Andrew M. Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, Barret Zoph, Liam Fedus, Maarten P. Bosma, Zongwei Zhou, Tao Wang, Yu Emma Wang, Kellie Webster, Marie Pellat, Kevin Robinson, and 8 others. 2022. [Glam: Efficient scaling of language models with mixture-of-experts](#). In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 5547–5569. PMLR.
- Nadir Durrani, Hassan Sajjad, Fahim Dalvi, and Yonatan Belinkov. 2020. [Analyzing individual neurons in pre-trained language models](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4865–4880, Online. Association for Computational Linguistics.
- William Fedus, Barret Zoph, and Noam Shazeer. 2022. [Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity](#). *J. Mach. Learn. Res.*, 23:120:1–120:39.
- Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac’h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, and 5 others. 2024. [The language model evaluation harness](#).
- Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. [The state and fate of linguistic diversity and inclusion in the nlp world](#). In *Proceedings of the 58th Annual Meeting of*

- the Association for Computational Linguistics*, pages 6282–6293.
- Takeshi Kojima, Itsuki Okimura, Yusuke Iwasawa, Hitomi Yanaka, and Yutaka Matsuo. 2024. [On the multilingual ability of decoder-based pre-trained language models: Finding and controlling language-specific neurons](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6919–6971, Mexico City, Mexico. Association for Computational Linguistics.
- Aran Komatsuzaki, Joan Puigcerver, James Lee-Thorp, Carlos Riquelme Ruiz, Basil Mustafa, Joshua Ainslie, Yi Tay, Mostafa Dehghani, and Neil Houlsby. 2023. [Sparse upcycling: Training mixture-of-experts from dense checkpoints](#). In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.
- Viet Lai, Chien Nguyen, Nghia Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan Rossi, and Thien Nguyen. 2023. [Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 318–327, Singapore. Association for Computational Linguistics.
- Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. 2021. [Gshard: Scaling giant models with conditional computation and automatic sharding](#). In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.
- Chong Li, Yingzhuo Deng, Jiajun Zhang, and Chengqing Zong. 2025. [Group then scale: Dynamic mixture-of-experts multilingual language model](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 1730–1754, Vienna, Austria. Association for Computational Linguistics.
- Shangjie Li, Xiangpeng Wei, Shaolin Zhu, Jun Xie, Baosong Yang, and Deyi Xiong. 2023. [MMNMT: Modularizing multilingual neural machine translation with flexibly assembled MoE and dense blocks](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4978–4990, Singapore. Association for Computational Linguistics.
- Weize Liu, Yinlong Xu, Hongxia Xu, Jintai Chen, Xuming Hu, and Jian Wu. 2024. [Unraveling Babel: Exploring multilingual activation patterns of LLMs and their applications](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11855–11881, Miami, Florida, USA. Association for Computational Linguistics.
- Llama Team. 2024. [The llama 3 herd of models](#). *CoRR*, abs/2407.21783.
- Soumen Kumar Mondal, Sayambhu Sen, Abhishek Singhania, and Preethi Jyothi. 2025. [Language-specific neurons do not facilitate cross-lingual transfer](#). In *The Sixth Workshop on Insights from Negative Results in NLP*, pages 46–62, Albuquerque, New Mexico. Association for Computational Linguistics.
- Thuat Nguyen, Chien Van Nguyen, Viet Dac Lai, Hieu Man, Nghia Trung Ngo, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2024. [CulturaX: A cleaned, enormous, and multilingual dataset for large language models in 167 languages](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 4226–4237, Torino, Italia. ELRA and ICCL.
- Daria Soboleva, Faisal Al-Khateeb, Robert Myers, Jacob R Steeves, Joel Hestness, and Nolan Dey. 2023. [SlimPajama: A 627B token cleaned and deduplicated version of RedPajama](#). <https://www.cerebras.net/blog/slimpajama-a-627b-token-cleaned-and-deduplicated-version-of-redpajama>.
- Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. 2024. [Language-specific neurons: The key to multilingual capabilities in large language models](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5701–5715, Bangkok, Thailand. Association for Computational Linguistics.
- Tianwen Wei, Liang Zhao, Lichang Zhang, Bo Zhu, Lijie Wang, Haihua Yang, Biye Li, Cheng Cheng, Weiwei Lü, Rui Hu, Chenxia Li, Liu Yang, Xilin Luo, Xuejie Wu, Lunan Liu, Wenjun Cheng, Peng Cheng, Jianhao Zhang, Xiaoyu Zhang, and 11 others. 2023. [Skywork: A more open bilingual foundation model](#). *CoRR*, abs/2310.19341.
- Yuemei Xu, Kexin Xu, Jian Zhou, Ling Hu, and Lin Gui. 2025. [Linguistic neuron overlap patterns to facilitate cross-lingual transfer on low-resource languages](#). *ArXiv*, abs/2508.17078.
- Xue Zhang, Yunlong Liang, Fandong Meng, Songming Zhang, Yufeng Chen, Jinan Xu, and Jie Zhou. 2025. [Less, but better: Efficient multilingual expansion for LLMs via layer-wise mixture-of-experts](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 17948–17963, Vienna, Austria. Association for Computational Linguistics.
- Hao Zhou, Zhijun Wang, Shujian Huang, Xin Huang, Xue Han, Junlan Feng, Chao Deng, Weihua Luo, and Jiajun Chen. 2025. [Moe-lpr: Multilingual extension of large language models through mixture-of-experts with language priors routing](#). In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pages 26092–26100. AAAI Press.

Tong Zhu, Xiaoye Qu, Daize Dong, Jiacheng Ruan, Jingqi Tong, Conghui He, and Yu Cheng. 2024. [LLaMA-MoE: Building mixture-of-experts from LLaMA with continual pre-training](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15913–15923, Miami, Florida, USA. Association for Computational Linguistics.

## A Cross-Lingual Neuron Distribution

This section provides the complete cross-lingual neuron diversity measurements (EN+EL) that inform NeuronMoE’s allocation strategy (Table 5).

Layers	Unique Neurons (EN+EL)
0-2	342, 79, 11
3-10	18, 15, 11, 8, 7, 8, 8, 7
11-18	6, 9, 14, 9, 18, 81, 20, 19
19-27	28, 38, 232, 51, 208, 114, 111, 238, 223
<b>Total</b>	<b>1,933</b>

Table 5: Cross-lingual neuron diversity across layers (Llama-3.2-3B, Greek extension).

## B Cross-Lingual Neuron Distribution Comparison

This section presents the complete layer-by-layer comparison of language-specific neuron counts across English (source) and three target languages (Greek, Turkish, Hungarian) representing three distinct language families (Table 6). Despite typological differences, all languages exhibit the same pattern: minimal specialization in middle layers with concentration at layer boundaries, supporting the universal architectural principles discussed in Section 6.

## C Expert Allocation Patterns for Additional Languages

This section provides the complete expert allocation patterns for Turkish and Hungarian extensions (Table 8), complementing the Greek allocation shown in Table 2. These allocations demonstrate that the neuron-guided strategy generalizes across typologically diverse languages from different families.

Table 8 shows the layer-wise expert allocation patterns for Turkish and Hungarian, following the same neuron-guided allocation strategy applied to Greek. Both languages exhibit similar heterogeneous patterns with concentration in early and late

L	EN	EL	TR	HU	L	EN	EL	TR	HU
0	322	21	8	8	14	7	2	1	1
1	73	6	5	3	15	10	8	4	6
2	8	3	3	1	16	11	70	10	8
3	11	7	17	6	17	4	16	15	8
4	13	2	0	1	18	10	9	9	15
5	9	2	2	2	19	9	19	22	23
6	7	1	0	1	20	16	22	27	18
7	6	1	1	0	21	17	215	68	92
8	5	3	1	0	22	24	27	47	40
9	7	1	1	0	23	70	139	137	117
10	7	0	1	1	24	63	51	173	190
11	5	1	1	1	25	60	52	185	303
12	8	1	0	0	26	75	163	180	124
13	9	5	2	0	27	108	118	70	24

Table 6: Complete layer-wise comparison of language-specific neuron counts across English (base Llama-3.2-3B), Greek, Turkish, and Hungarian

layers, validating the universality of our approach across diverse language families.

The allocation patterns reveal consistent principles across languages: early layers (0-2) receive 5-6 experts for language-specific input processing, middle layers (3-20) predominantly use single experts for language-agnostic reasoning, and late layers (19-27) show varying concentration (2-6 experts) based on language-specific generation requirements. Despite minor differences in total expert count (47-50, all three languages achieve substantial parameter reduction compared to LayerMoE’s relatively uniform 84 experts while maintaining strong performance.

## D Training Configuration Details

All MoE extension experiments used identical hyperparameters across languages (Greek, Turkish, Hungarian) and architectures (Llama-3.2-3B, Qwen-1.5-1.8B) to ensure fair comparison.

**Training Data:** Stage 1 uses 2B tokens per target language from CulturaX (Nguyen et al., 2024). Stage 2 router training samples 50K examples each from SlimPajama (Soboleva et al., 2023) for English, SkyPile-150B (Wei et al., 2023) for Chinese, and CulturaX for Spanish.

**Baselines:** Dense (no MoE), LayerMoE-S1/LayerMoE (84 experts for Llama, 72 for Qwen), and NeuronMoE variants (49 experts for Llama, 36 for Qwen). NeuronMoE-EN (37 experts) allocates based solely on English neuron distributions.

**Computational Cost:** Neuron analysis requires approximately six GPU minutes per language on a single NVIDIA GH200. Training uses DeepSpeed

ZeRO Stage 2 on NVIDIA GH200 96GB GPUs, requiring approximately 120 GPU hours total.

Complete hyperparameters are shown in Table 7.

Parameter	Value
<i>Optimization:</i>	
Optimizer	AdamW
Learning rate	1e-4
Adam $\beta_1, \beta_2$	0.9, 0.999
Adam $\epsilon$	1e-8
Weight decay	0.0
LR scheduler	Cosine
Warmup steps	0
<i>Training:</i>	
Batch size per device	32
Gradient accumulation steps	1
Effective batch size	32
Total training steps	15,000
Max gradient norm	1.0
Precision	BF16
Seed	42
<i>MoE-specific:</i>	
Top-K routing	2
Auxiliary loss coefficient	0.01
Min experts per layer ( $E_{\min}$ )	1
Max experts per layer ( $E_{\max}$ )	6

Table 7: Complete training hyperparameters for all experiments. All models trained with identical settings for fair comparison.

Table 9 shows the detailed per-layer expert allocation for both LayerMoE and NeuronMoE across Llama-3.2-3B and Qwen-1.5-1.8B architectures.

NeuronMoE’s allocation pattern reflects the empirical finding that language-specific neurons concentrate in early and late layers, while middle layers remain largely language-agnostic. This neuron-guided strategy achieves 41.3% (Llama) and 62.5% (Qwen) parameter reduction compared to LayerMoE’s attention-based allocation.

## E Detailed Expert Specialization Statistics

Table 10 presents the complete statistics of high-AP neuron counts for Greek across all experts in layers that received multiple experts under our neuron-guided allocation strategy. These measurements were taken after Stage 1 training.

## F Detailed Results for Additional Languages

We also provide preliminary Stage 1 (Expert Initialization) results for Bengali (Indo-Aryan) and Nepali (Indo-Aryan) in Table 11. These results

Layers	Greek	Turkish	Hungarian
0-2 (Early)	6,2,1	6,2,1	5,2,1
3-10 (Early-Mid)	1×8	1×8	1×8
11-18 (Mid-Late)	1×7,2	1×8	1×8
19-27 (Late)	1,1,4,1,4, 2,2,4,4	1,1,1,2,2, 4,4,4,4,3	1,1,1,2,1, 3,4,6,3,2
<b>Total</b>	<b>49</b>	<b>50</b>	<b>47</b>

Table 8: Layer-wise expert allocation for Turkish and Hungarian on Llama-3.2-3B (28 layers). Both languages show similar allocation patterns to Greek, with concentration in input-proximal and output-proximal layers.

Method	Expert Allocation (Layer 0-23)
LayerMoE	[5, 4, 3, 5, 4, 3, 2, 3, 3, 2, 2, 2, 2, 2, 2, 2, 3, 3, 4, 4, 4, 3, 3]
NeuronMoE	[4, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 6]

Table 9: Per-layer expert allocation for LayerMoE and NeuronMoE on Qwen-1.5-1.8B (24 layers). LayerMoE uses attention-layer similarity; NeuronMoE uses cross-lingual neuron diversity analysis.

further demonstrate the extensibility of our method to languages beyond the primary experimental set.

<b>L</b>	<b>Expert</b>	<b>Total</b>	<b>High-AP</b>	<b>Ratio (%)</b>
0	base	19,456	0	0.00
0	exp_0	19,456	0	0.00
0	exp_1	19,456	27	0.14
0	exp_2	19,456	0	0.00
0	exp_3	19,456	61	0.31
0	exp_4	19,456	19	0.10
1	base	19,456	27	0.14
1	exp_0	19,456	27	0.14
16	base	19,456	64	0.33
16	exp_0	19,456	71	0.36
21	base	19,456	26	0.13
21	exp_0	19,456	46	0.24
21	exp_1	19,456	0	0.00
21	exp_2	19,456	82	0.42
23	base	19,456	26	0.13
23	exp_0	19,456	57	0.29
23	exp_1	19,456	0	0.00
23	exp_2	19,456	46	0.24
24	base	19,456	100	0.51
24	exp_0	19,456	139	0.71
25	base	19,456	98	0.50
25	exp_0	19,456	118	0.61
26	base	19,456	0	0.00
26	exp_0	19,456	278	1.43
26	exp_1	19,456	154	0.79
26	exp_2	19,456	0	0.00
27	base	19,456	179	0.92
27	exp_0	19,456	0	0.00
27	exp_1	19,456	0	0.00
27	exp_2	19,456	405	2.08

Table 10: Complete expert specialization statistics for Greek. High-AP neurons indicate strong Greek specialization developed during Stage 1 training. Total neurons per layer/expert: 19,456.

<b>Model</b>	<b>Language</b>	<b>ARC</b>	<b>Belebele</b>	<b>HellaSwag</b>	<b>MMLU</b>
Dense	English	51.11	74.11	76.33	56.45
	Nepali	24.12	42.44	32.54	33.77
	Bengali	26.26	44.78	33.26	33.48
NeuronMoE-S1 (NE)	English	47.44	71.44	75.29	54.71
	Nepali	24.89	44.67	33.07	34.63
NeuronMoE-S1 (BN)	English	48.72	75.33	76.01	55.28
	Bengali	26.35	46.89	33.57	33.47

Table 11: Stage 1 results for Bengali and Nepali extensions on Llama-3.2-3B.