M-Wanda: Improving One-Shot Pruning for Multilingual LLMs

Rochelle Choenni[1] and **Ivan Titov**[1,2]
University of Amsterdam[1]
University of Edinburgh[2]
r.m.v.k.choenni@uva.nl, ititov@inf.ed.ac.uk

Abstract

Multilingual LLM performance is often critically dependent on model size. With an eye on efficiency, this has led to a surge in interest in one-shot pruning methods that retain the benefits of large-scale pretraining while shrinking the model size. However, as pruning tends to come with performance loss, it is important to understand the trade-offs between multilinguality and sparsification. In this work, we study multilingual performance under different sparsity constraints and show that moderate ratios already substantially harm performance. To help bridge this gap, we propose M-Wanda, a pruning method that models cross-lingual variation by incorporating language-aware activation statistics into its pruning criterion and dynamically adjusts layerwise sparsity based on cross-lingual importance. We show that M-Wanda consistently improves performance at minimal additional costs. We are the first to explicitly optimize pruning to retain multilingual performance, and hope to inspire future advances in multilingual pruning.[1]

1 Introduction

Large language models (LLMs) have demonstrated strong multilingual capabilities, with their ability to process and generate text in numerous languages improving substantially as the model size increases (He et al., 2024). This emergent multilingualism can largely be attributed to the vast amount of multilingual data used for pretraining and the increased model capacity that allows for better generalization over linguistic patterns across multiple languages. However, the steep increase in model scale comes with substantial computational and environmental costs, making efficient deployment in resource-constrained environments challenging (Ogueji et al., 2022). To address these challenges, model compression techniques, such as pruning, quantization, and distillation, have been widely explored to reduce model size while retaining performance (Zhu et al., 2024). However, despite the effectiveness of such methods, their evaluation focuses mainly on maintaining English performance (Yang et al., 2024), with limited consideration of their impact on multilingual performance (Zeng et al., 2024; Kurz et al., 2024; Ogueji et al., 2022). Given that multilingual performance is crucial for equitable LLMs, ensuring that compression does not disproportionately harm performance in non-English languages is essential.

In this paper, we study the effect of sparsity on the multilingual performance of six open-source LLMs of varying sizes. For model compression, we focus on a SOTA one-shot unstructured pruning method– Wanda (Sun et al., 2023)– and evaluate language modeling abilities and zero-shot task performance at varying sparsity levels across 15 languages and six downstream tasks. Our results show that Wanda, despite its strong performance in English, causes substantial degradation in multilingual performance, particularly at sparsity levels higher than 50% and in underrepresented languages. These findings highlight an important limitation. As Wanda was developed to optimize for global importance, we hypothesize that it fails to account for cross-lingual variation in neuron importance, despite being exposed to multilingual calibration data, which leads to the removal of weights that are important for specific languages.

To help bridge this gap, we propose a novel multilingual pruning method, M-Wanda, which is a multilingual extension of Wanda. M-Wanda improves on Wanda by incorporating language-aware input activation statistics to better inform pruning decisions at minimal additional costs. Moreover, M-Wanda dynamically adjusts sparsity ratios across layers based on cross-lingual correlation scores, ensuring that layers that are important for cross-lingual sharing are pruned less aggres-

[1] https://github.com/RochelleChoenni/M-Wanda.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 26951–26964
November 4-9, 2025 ©2025 Association for Computational Linguistics

sively. Together, these techniques allow us to better balance the contribution of shared and specialized neurons to weight importance. To the best of our knowledge, our work is the first to optimize pruning for multilingual retention, and to explicitly model cross-lingual activation variance and inter-language correlation to guide pruning decisions and layer-wise sparsity allocation.

We show that M-Wanda consistently reduces perplexity across all languages and that this translates into performance improvements on all downstream tasks. Importantly, we show that M-Wanda generalizes well beyond the set of languages included in the calibration data. In addition, we show that the techniques introduced in M-Wanda can also be integrated with RIA (Zhang et al., 2024), a more recent pruning method, thus showcasing their general usefulness in extending pruning to a multilingual setting. Finally, our findings highlight the need to evaluate pruning methods beyond English-centric compression benchmarks (Yang et al., 2024) and emphasize the importance of optimizing the pruning strategy to preserve multilingual performance. In doing so, we hope to contribute to the development of more efficient LLMs that remain effective in many languages.

2 Background and related work

2.1 Compression through pruning

Pruning reduces model size by removing unnecessary weights or neurons (LeCun et al., 1989) and can be grouped into iterative and one-shot methods. Iterative methods (Frankle and Carbin, 2018; Blalock et al., 2020) repeatedly prune a small percentage of weights, followed by retraining to recover performance, until a target threshold is met. While this does not require a predefined sparsity ratio, the additional training cycles can be expensive. One-shot methods, instead, remove a predefined fraction of weights in a single pass after the model is trained to convergence, and do not require retraining (Frantar and Alistarh, 2023; Sun et al., 2023). It has gained popularity because of its simplicity and ability to maintain competitive performance.

2.2 One-shot pruning methods

SparseGPT (Frantar and Alistarh, 2023) introduced one-shot pruning by sequentially processing model layers and solving a local quadratic optimization problem to minimize reconstruction error under sparsity constraints. Yet, this requires a weight update after pruning and backward passes for gradient computation. Sun et al. (2023) show that SparseGPT can be simplified to a gradient-free variant that achieves competitive performance without the need for parameter updates, i.e. Wanda.

Wanda method To determine the importance of model weights, Sun et al. (2023) propose to incorporate the absolute weight value and the norm of the input activations of the neurons into the pruning criterion. Formally, let the input to a layer be denoted by $X \in \mathbb{R}^{(N \times T) \times C_{\text{in}}}$, where N is the batch size and T the sequence length. The weight matrix $W \in \mathbb{R}^{C_{\text{out}} \times C_{\text{in}}}$ connects the input features to the output units. For each weight element $W_{i,j}$, the importance score is defined as:

$$\mathbf{S}_{i,j} = |W_{i,j}| \cdot \|X_j\|_2 \tag{1}$$

where $\|X_j\|_2$ denotes the ℓ_2-norm of the j-th input feature column across all $N \times T$ tokens. This score reflects the contribution of each weight based on both its magnitude and the aggregated strength of the corresponding input feature. Note that, collecting input activation statistics requires a set of input samples which we refer to as *calibration data*. Finally, a strong commonly used baseline is magnitude pruning (Han et al., 2015), in which only the weight magnitude: $\mathbf{S}_{i,j} = |W_{i,j}|$ is considered.

Layerwise sparsity allocation Sparsity allocation methods were developed to mitigate model degradation by enforcing different sparsity ratios across layers, and have been shown to improve performance (Li et al., 2024; Huang et al., 2025). Rather than pruning uniformly, such methods estimate how much to prune based on the layers' sensitivity or redundancy. Concretely, given a global sparsity ratio R, the goal is to derive a set of target layerwise sparsity ratios $[r_0, r_1, .., r_L]$ such that: $\frac{1}{L+1}\sum_{n=0}^{L} r_n = R$.

One such method is Outlier Weighted Layerwise sparsity (OWL) (Yin et al., 2024), which uses per-layer outlier counts (i.e. activations that exceed M times the mean) as global importance scores $C = [c_0, c_1, .. ,c_L]$ for allocating sparsity. To prevent extreme imbalances between layers, they introduce a hyperparameter γ that restricts each ratio to fall within a small interval around the global sparsity rate, specifically $r_n \in [R-\gamma, R+\gamma]$, while maintaining a mean sparsity ratio of R across all layers. To achieve this, the raw importance scores C are rescaled to the range $[0, 2\lambda]$ and shifted so

that the resulting values are centered around R. Following the intuition that layers that are more important should be pruned less, the sparsity ratios are then defined as: $r_n = 1 - c_n$.

2.3 Multilingual pruning

Ogueji et al. (2022) first studied the effect of model pruning on multilingual performance. However, their scope was limited to iterative methods, smaller models, and a single task. More recently, Zeng et al. (2024); Kurz et al. (2024) studied multilingual performance of LLMs using SparseGPT and Wanda. They both study how varying the composition of calibration data from different languages affect performance, and show that using a mixture of languages yields better results. However, both studies are limited to modifying the calibration data, without altering the pruning method itself, and restricting analysis to compression at 50%. In this work, we show that 50% sparsity already substantially harms multilingual performance. Moreover, this sparsity ratio is enforced uniformly across model layers, despite substantial evidence that model layers play different roles in language-specific and cross-lingual processing (Tang et al., 2024; Kojima et al., 2024). We, instead, study multilingual performance under different sparsity constraints and introduce M-Wanda, a novel pruning method that builds on Wanda by using multilingual calibration data, incorporating language-aware scoring, and combining it with an OWL-inspired dynamic sparsity allocation method.

3 M-Wanda method

While Zeng et al. (2024); Kurz et al. (2024) show that using Wanda with multilingual calibration data improves performance, Wanda was developed to preserve weights that are globally important, and by averaging input activations across languages, we might suppress language-specific signals that are essential for multilingual retention. Thus, we enhance Wanda in three key ways: (1) We assign ***layerwise sparsity*** based on the degree of cross-lingual activation similarity, applying less aggressive pruning to layers that are more important for cross-lingual sharing. (2) We incorporate ***cross-lingual activation variance*** into the pruning criterion to encourage retention of specialized neurons that might, for instance, support underrepresented or typologically distinct languages. (3) We introduce an ***activation probability*** term to discourage retention of high-variance neurons that rarely activate, helping to filter out noisy or spurious features. Together, these additions bias pruning toward preserving both shared and consistently active specialized neurons, thereby improving multilingual retention at minimal additional costs.

3.1 Correlation Weighted Layerwise (CWL) sparsity

We introduce *Correlation Weighted Layerwise (CWL) sparsity* to guide sparsity allocation decisions across model layers. In contrast to OWL (Yin et al., 2024), which scores layer importance based on outlier counts, CWL uses Pearson correlation coefficients to approximate activation similarity both across and between languages to determine importance. We hypothesize that layers that exhibit high inter-language activation similarity are more involved in cross-lingual sharing and better facilitate multilingual generalization. As such, we apply less aggressive pruning to them. However, when intra-language correlation scores are low, this suggests instable or noisy representations. To correct for this, we adjust the inter-language correlation score using intra-language scores.

Concretely, we first compute Pearson correlation scores between the mean input activation (aggregated across tokens) for each language and sublayer $\mu_\ell^{(k)}$ of the attention or MLP block.[2] To compute the average inter-language correlation score for sublayer k and a set of languages $\mathcal{L}$, we then take the mean of all pairwise correlations:

$$\text{Inter}^{(k)} = \frac{2}{|\mathcal{L}|(|\mathcal{L}|-1)} \sum_{i<j} \text{corr}(\mu_{\ell_i}^{(k)}, \mu_{\ell_j}^{(k)}) \quad (2)$$

Moreover, we adjust inter-language scores using intra-language scores, by assigning more importance when both are high, yielding:

$$c^{(k)} = \text{Inter}^{(k)} \cdot \sum_{\ell \in \mathcal{L}} \text{Intra}_\ell^{(k)} \quad (3)$$

This score reflects how shared representations are between languages and how stable they are within languages. To obtain a single importance score for each layer n: $[c_0, c_1, .. c_L]$, we take the average over all sublayers. We then apply the same procedure as OWL, described in Section 2.2, to ensure that the mean sparsity is equal to the global ratio R, and assign layers with more importance, lower

[2] Note that input activations are shared between query, key and value, and between the MLP gate and up projection layers.

ratios: $r_n = 1 - c_n$. We find that setting λ to 0.04 generally works well across LLMs.

3.2 Cross-lingual activation variance

Recall from Eq 1 that Wanda incorporates both weight importance and activation strength. We now enhance the activation scores by storing the mean of activation values per language μ_ℓ and computing the variance in neuron activation across languages:

$$\text{Var}_{inter} = \frac{1}{|\mathcal{L}|} \sum_{\ell \in \mathcal{L}} (\mu_\ell - \bar{\mu})^2 \quad (4)$$

By adding this inter-language variance score, we give more importance to neurons whose input activations show highly variable responses across languages, meaning that they might be very important to *some* specific languages.

Yet, the input activations within a language also fluctuate between input samples. If the intra-language variance is high, it introduces noise into our pruning metric, making the inter-language variance less reliable. Therefore, we assess how much neuron activation varies between languages relative to how much it varies within individual languages:

$$VAR = \frac{\text{Var}_{inter}}{\frac{1}{|\mathcal{L}|} \sum_{\ell=1}^{\mathcal{L}} \text{Var}^{\ell}_{intra}} \quad (5)$$

This means that we assign higher scores to neurons that exhibit high inter-language variance but low intra-language variance.

$$A_{X_j} = \|X_j\|_2 + \lambda \cdot VAR \quad (6)$$

To balance the trade-off between language-specificity and generalization, we add a scaling term λ for which the optimal value is found through a grid search. Also, note that before adding variance scores, we apply min-max normalization.

3.3 Activation probability

Finally, we correct the overall weight importance scores based on the average activation probability across languages. This is motivated by the idea that high-variance neurons that are upweighted by Eq. 6, but rarely activate, are noisy and should be filtered out. To compute this activation probability, we simply count how many times the input activations are higher than some threshold value ϵ. Given that recent LLMs rely on activation functions that also allow for negative activation that can contain meaningful information, we consider absolute activation values.[3] As such, we end up with the following pruning metric:

$$\mathbf{S}_{i,j} = (|W_{i,j}| \cdot A_{X_j}) \cdot P(\mathbb{I}(abs(X_j) > \epsilon)) \quad (7)$$

where $\mathbb{I}(\cdot)$ is the indicator function.

4 Experiments

Calibration and test languages For calibration and evaluation we use 15 languages: English, German, Spanish, French, Italian, Portuguese, Hindi, Russian, Korean, Japanese, Vietnamese, Chinese, Indonesian, Turkish, Arabic. These languages belong to 8 different families, 11 sub-families and cover 7 writing scripts.

Calibration data Following prior work, we use 128 random samples of 2048 tokens from the multilingual C4 (MC4) dataset for calibration (Raffel et al., 2020). While recent studies show that the data source affects pruning quality (Williams and Aletras, 2024; Ji et al., 2024; Bandari et al., 2024), we use MC4 to limit the scope of this work and ensure comparability with existing literature. To adhere to the 128 samples maximum, our calibration data includes 16 samples from English and 8 from all other test languages.

Models We study six open-source LLMs at different model sizes: Llama3 (1B, 3B and 8B) (Grattafiori et al., 2024), Aya-23 (8B) (Dang et al., 2024), OLMo-7B (Groeneveld et al., 2024) and Bloomz-7b1 (Muennighoff et al., 2023).

Zero-shot performance We perform zero-shot evaluation on six tasks that test the LLMs ability on reasoning (Xstorycloze, Xcopa) (Ponti et al., 2020), coreference resolution (Xwinograd) (Muennighoff et al., 2023), reading comprehension (Lambada) (Paperno et al., 2016), natural language understanding (XNLI) (Conneau et al., 2018), and paraphrasing (PAWS-X) (Yang et al., 2019). For consistent evaluation we employ the eleuther-AI evaluation harness.[4]

Model perplexity We test general language modeling abilities by measuring perplexity on datasets different from the one used for calibration. Specifically, we evaluate perplexity on the entire Flores-101 (`dev+devtest`) dataset which contains parallel

[3] Using absolute values was found to work better than positive ones, showing that negative signals carry information.

[4] https://github.com/EleutherAI/lm-evaluation-harness

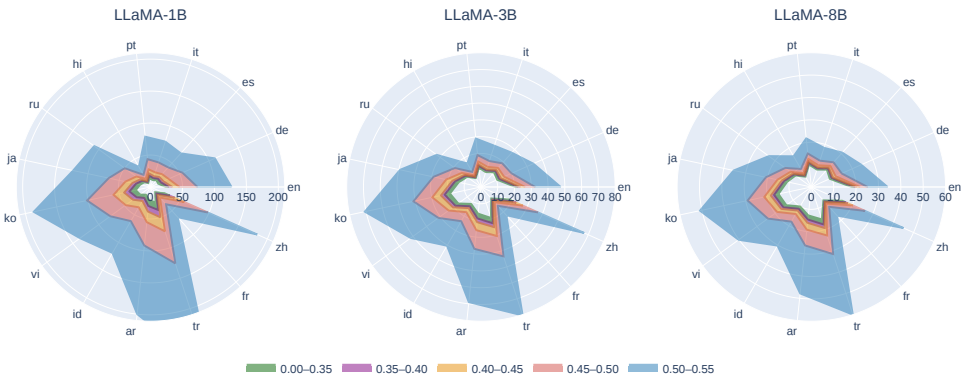

Figure 1: The effect of Wanda pruning under different sparsity ratios on the perplexity of each calibration language. Colored areas denote the increase in perplexity when increasing the sparsity ratio. Note that the perplexity scores are on different scales across models.

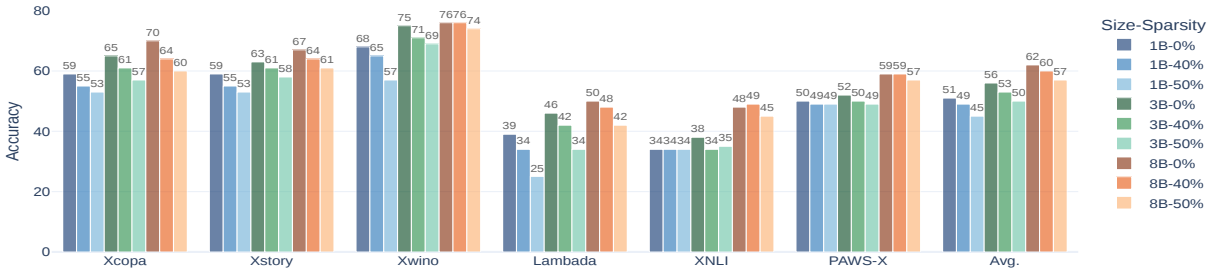

Figure 2: Performance in accuracy (%) given different sparsity ratios used on different sizes of Llama3. Zero-shot results are averaged across test languages per downstream task.

data from Wikipedia. To test whether our results are robust across different domains, we also evaluate on the XL-Sum dataset that contains high-quality articles from BBC (Hasan et al., 2021).

5 Results

In Section 5.1, we first show how pruning under different sparsity constraints affects multilingual LLMs of different sizes. Motivated by these findings, we show in Section 5.2 how our M-Wanda method can help mitigate some of the multilingual performance loss induced by pruning.

5.1 Wanda's impact on multilinguality

We prune our models using Wanda with sparsity ratios between 35 and 60% at 5 percent intervals. In Figure 1, we see that across all languages and different sizes of Llama, the perplexity has already substantially increased when going from 45 to 50% sparsity (red area), especially in underrepresented languages (typically not from the Indo-European family). This sheds doubt on the common practice of adopting the default sparsity ratio of 50% in the multilingual setting (Zeng et al., 2024; Kurz et al., 2024). Importantly, this same degradation is not found in English when only using English calibration data (see Appendix D), the setting used in the original paper (Sun et al., 2023).

Similarly, a clear degradation is visible in all downstream tasks when sparsity increases from 40 to 50%; see Figure 2. In fact, when studying how larger models pruned to 50% of their original capacity compared to their smaller dense counterparts (i.e. Llama 3B at 50% versus Llama 1B and Llama 8B at 50% versus Llama 3B), we see that they are not able to outperform them despite still having a larger capacity.

5.2 Improvements with M-Wanda

In Section 5.1, we show that Wanda with 50% sparsity leads to a substantial drop in multilingual performance. This degradation highlights an area of potential improvement for M-Wanda, and we hypothesize that more optimally balancing the importance between specialized and shared neurons would allow us to better retain multilingual performance. In Table 1, we show how M-Wanda is able to reduce the average perplexity across languages for all models on the Flores dataset (see Appendix A for the optimal hyperparameters selected for each model and Appendix C for results

Method	Llama3-1B	Llama3-3B	Llama3-8B	Aya-23-8B	Bloomz-7b1	OLMo-7B
Magnitude	17605	1579	403	36.12	29.64	33.55
RIA*	71.75	27.88	20.45	25.28	**24.05**	30.45
Wanda	67.51	26.42	19.63	24.34	24.71	23.23
M-Wanda	**59.56** (12%↓)	**24.52** (7%↓)	**18.57** (5%↓)	**23.87** (2%↓)	24.32 (2%↓)	**21.54** (7%↓)

Table 1: Average perplexity on Flores across all calibration languages at a sparsity ratio of 50%. For M-Wanda, we also report the relative percentage decrease compared to Wanda.*Refer to Section 7 for an introduction to RIA.

	Xcopa	Xstory	Xwino	Lambada	XNLI	PAWS-X	Avg.
Wanda	60.36	60.86	73.81	42.08	45.07	57.43	56.60
M-Wanda	**61.16**	**61.49**	**74.54**	**44.68**	**46.51**	**58.23**	**57.77**

Table 2: Average performance (%) on downstream tasks when using Wanda versus M-Wanda on Llama-8B.

on XL-Sum).[5] Moreover, we find that this holds across different model sizes. Importantly, while lower perplexity does not always guarantee better performance on downstream tasks, in Table 2 we show that the improvements achieved by M-Wanda are substantial enough to improve performance in all six downstream tasks.

When taking a closer look at the effect of M-Wanda on individual test languages in Figure 3, we see that M-Wanda consistently reduces the perplexity on all 15 languages for Llama-8B. Moreover, we see that languages most typologically different from English, i.e. Arabic, Turkish, Vietnamese, Chinese, Korean and Japanese, obtain larger absolute gains from M-Wanda than the Indo-European languages. Similarly, when looking at downstream task improvements, we find that M-Wanda tends to consistently improve performance on all individual test languages, with a few exceptions (mostly English), see Appendix F for full results.

5.2.1 Generalization to unseen languages

Previously, we used the same set of languages for calibration and testing. We now test whether the performance improvements also generalize beyond our calibration languages. As such, we use 15 different languages for evaluation: Czech, Polish, Ukranian, Bulgarian, Tamil, Marathi, Urdu, Bengali, Kannada, Gujarati, Javanese, Thai, Swahili, Zulu, and Persian. In Figure 4, we show that our method consistently reduces perplexity across all languages, despite them not being included in the calibration set. Specifically, M-Wanda results in

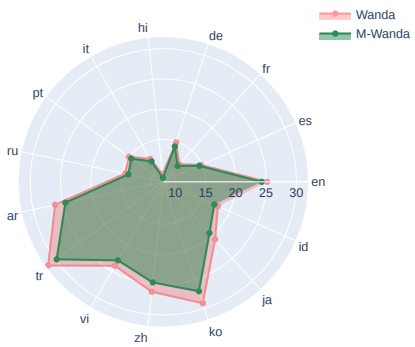

Figure 3: Perplexity scores per language from Llama-8B pruned using Wanda versus M-Wanda.

a 6% decrease in average perplexity compared to Wanda (18.98 versus 20.09), which is higher than on the calibration languages itself. This is likely because our unseen languages include many more underrepresented languages, and our method seems to book larger performance gains on those. Importantly, however, this suggests that M-Wanda more generally helps to preserve language variance and is not only adjusting to the language-specific patterns of the calibration languages.

5.2.2 Effectiveness at different sparsity levels

While we already saw that M-Wanda improves performance across different model sizes, we now also test its effectiveness across different sparsity ratios. In Figure 5, we plot the average perplexity scores obtained with Wanda and M-Wanda at different sparsity levels. We see that M-Wanda remains effective at higher ratios and that the average improvement of M-Wanda over Wanda increases substantially when applying more aggressive pruning. At the extreme sparsity ratio of 70% we find that

[5]Perplexity from magnitude pruning is notably higher on Llama. We find that performance is reasonable in English, yet explodes on other languages, yielding high average scores.

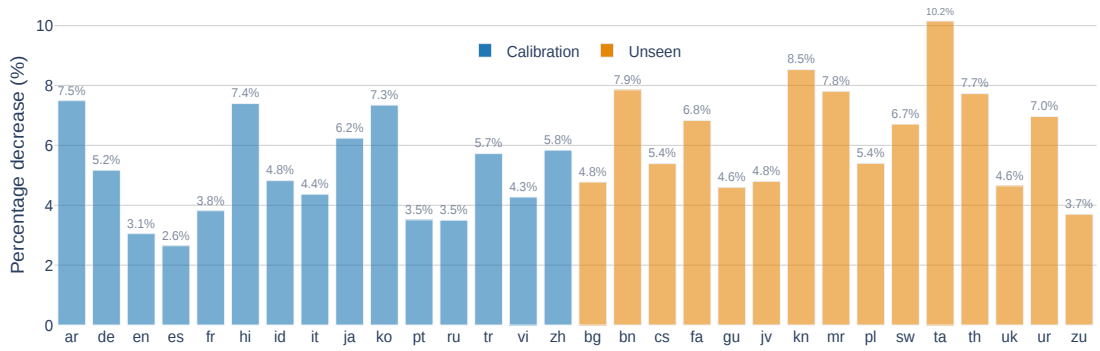

Figure 4: Relative percentage decrease in perplexity when using M-Wanda compared to Wanda for all 15 calibration and 15 unseen test languages. Results are reported for Llama-8B.

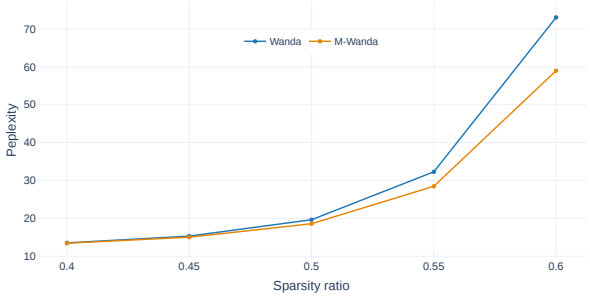

Figure 5: Average perplexity scores across languages as an effect of higher sparsity ratios when applying Wanda and M-Wanda to Llama-8B.

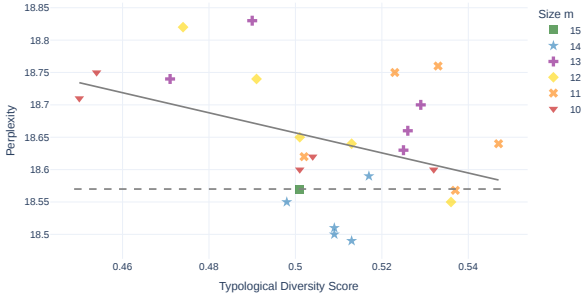

Figure 6: Average M-Wanda perplexity on Llama-8B as a function of the typological coverage of the calibration languages. Subsets are colored based on their size m.

M-Wanda reduces average perplexity by as much as 52% (see Appendix B for full results).

5.2.3 Robustness analysis

Sensitivity to calibration samples While we limited the scope of this paper to randomly selecting calibration samples from the MC4 dataset, we now also test the sensitivity of our method to the calibration set. Specifically, we use 3 random seeds to select calibration data and recompute average perplexity using Wanda versus M-Wanda. We find that across all three runs, M-Wanda outperforms Wanda. On average Wanda obtains a perplexity of 19.37±0.27 and M-Wanda 18.63±0.22.

Sensitivity to calibration languages Finally, we now study how selecting different subsets of languages from the full calibration set affects performance. These subsets vary both in size, which impacts the number of samples per language and, consequently, the robustness of language-specific signals, and in their typological composition, which might influence how well calibration generalizes across languages. To study this, we draw multiple random subsets of languages for calibration. Specifically, we each time sample 5 unique subsets of size m uniformly at random from the set of all languages $\mathcal{L}$: $\mathcal{S}_m^{(i)} \sim \text{Unif}\left(\{S \subset \mathcal{L} : |S| = m\}\right)$. In Figure 6, we plot the average perplexity on the full calibration set $\mathcal{L}$ as a function of the typological diversity of the calibration languages in the different language subsets of size m. These diversity scores are defined as the mean of pairwise cosine similarity between their URIEL language representations (Malaviya et al., 2017).[6] In general, we find that higher typological diversity leads to better performance. However, we also observe that a few language subsets can outperform the full calibration set, suggesting that optimal calibration may depend more on carefully selecting which languages are chosen than on increasing its size.

6 Ablation study

To understand where the performance improvements of M-Wanda come from, we now perform an ablation study, isolating the impact of individual enhancements that were added to the original Wanda method. When we combine the original Wanda metric with OWL instead of CWL allocation, we

[6]We use `syntax_knn` features from the `Lang2Vec` library.

Model	Wanda	+OWL	+CWL	+CWL+Var	+CWL+Act	M-Wanda
1B	67.51	67.43	61.12	59.85	60.23	**59.56**
3B	26.42	26.02	24.56	24.58	24.59	**24.52**
8B	19.63	19.13	18.61	18.58	18.59	**18.57**

Table 3: Full M-Wanda ablation on the Llama3 models. We analyze the contributions of different components of M-Wanda (Wanda$_{+\text{CWL}\ +\text{Var}\ +\text{Act}}$) by evaluating variants that incorporate OWL, CWL, cross-lingual activation variance (Var), activation probability (Act), and combinations thereof. We report average perplexity on the calibration languages.

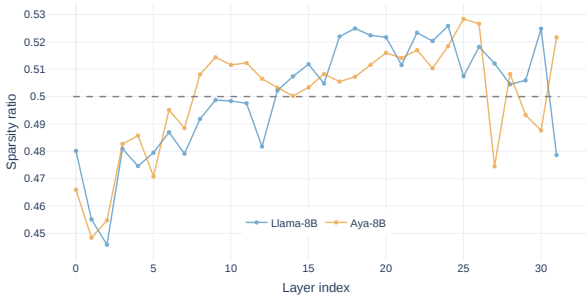

Figure 7: Layerwise sparsity allocation using CWL.

	RIA	M-RIA
Llama3.2-1B	71.75	**66.22** (8%↓)
Llama3.2-3B	27.88	**25.53** (8%↓)
Llama3.1-8B	20.45	**19.03** (7%↓)
Aya-23-8B	25.28	**24.82** (2%↓)
Bloomz-7b1	24.05	**23.65** (2%↓)
OLMo-7B	30.45	**26.21** (14%↓)

Table 4: Average perplexity scores on the calibration languages for Flores using RIA (α=0.5) at 50% sparsity.

find that OWL[7] reduces average perplexity to a lesser extent, see Table 3. Importantly, we also find that enhancing Wanda$_{+\text{OWL}}$ with cross-lingual variation and activation probability further improves the performance to 19.09 on Llama-8B. This shows that our proposed enhancements can more generally be paired with different allocation methods and do not work exclusively in combination with CWL.

Moreover, we find that combining Wanda$_{+\text{CWL}}$ with cross-lingual activation variance or activation probability separately yields mixed results. However, incorporating both scores further reduces the average perplexity across all models. As motivated in Section 3, we hypothesize that this is because the activation probability can act as a regularizer when incorporating the variance scores.

Lastly, in Figure 7 we plot the sparsity ratio allocated per layer for Llama-8B and Aya-8B using CWL. In general, we see that the lower layers and the last few top layers receive less aggressive pruning. The fact that this results in better multilingual performance can likely be connected to the fact that these layers have been shown to be more involved in cross-lingual processing (Zhao et al., 2024) (see Appendix E for allocation results using OWL).

[7]We tested the optimal hyperparameters reported in the original paper i.e., $M \in [3, 5]$ and $\gamma = 0.08$, but found that $\gamma = 0.04$, as used for CWL, works better and thus report scores using the latter for a more fair comparison.

7 Extendability to other pruning methods

Relative Importance and Activations (RIA) is a SOTA pruning method that has been shown to outperform Wanda (Zhang et al., 2024). It aims to improve upon Wanda by re-evaluating the importance of each weight element W_{ij} based on all connections that originate from the input neuron i or lead to the output neuron j:

$$\begin{aligned}\mathbf{RIA}_{i,j} &= \mathbf{RI}_{i,j} \cdot (\|X_j\|_2)^{\alpha} \\ &= \left(\frac{|W_{i,j}|}{\sum |W_{*j}|} + \frac{|W_{i,j}|}{\sum |W_{i*}|}\right) \cdot (\|X_j\|_2)^{\alpha}\end{aligned} \tag{8}$$

where $\sum |W_{*j}|$ and $\sum |W_{*i}|$ sum over the absolute values of the weights in input channel j and output channel i respectively. Yet, while we find that RIA outperforms Wanda on English at 50% sparsity (25.05 versus 25.16 on Llama-8B), the average perplexity across all 15 calibration languages tends to increase instead (20.45 versus 19.59 on Llama-8B). This further highlights the need for multilingual evaluation of pruning methods. Nonetheless, to test the compatibility of our proposed method with different pruning criterion, we now add cross-lingual variance and activation probability to RIA and apply CWL to obtain layerwise sparsity, yield-

ing M-RIA:

$$\mathbf{S}_{i,j} = (\mathbf{RI}_{i,j} \cdot A_{X_j}) \cdot P(\mathbb{I}(|X_j| > \epsilon))$$
$$\text{where} \quad A_{X_j} = (\|X_j\|_2)^{\alpha} + \lambda \cdot VAR \tag{9}$$

Note that we adopt α=0.5 which Zhang et al. (2024) found to be optimal for various LLMs. In Table 4 we show how M-RIA is also able to consistently improve over RIA, nicely demonstrating the general advantage of our proposed method for adaptation to a multilingual setting.

8 Conclusion

In this paper, we shed light on the limitations of SOTA pruning methods in a multilingual setting and introduce M-Wanda, a novel pruning method that explicitly models cross-lingual variation in weight importance. By incorporating language-aware activation statistics and adaptive sparsity allocation, M-Wanda substantially improves multilingual retention over existing methods, particularly for underrepresented languages and at high sparsity ratios. Our results show that multilingual pruning requires strategies that go beyond global importance scoring and highlight the benefits of considering the importance of specialized neurons. We hope that these insights help advance the state of multilingual pruning by underscoring the broader need for multilingual evaluation and design in LLM sparsification, and inspire new directions to improve multilingual pruning beyond the modification of calibration data.

9 Limitations

Our improvements to the original Wanda method come at minimal additional computational costs. Specifically, we only compute additional statistics from the activation inputs that would already need to be collected for the original method. Moreover, we still use 128 calibration samples in total across all of our calibration languages. However, while we show that input activation statistics can help inform pruning decisions, unlike Wanda, M-Wanda does require tuning of the hyperparameters. To alleviate the need for manual tuning, future work could investigate how hyperparameters could automatically be adjusted based on the scale of the weights and activations.

In addition, we limited the scope of this project to studying unstructured pruning, the setting for which Wanda was originally developed. However, Sun et al. (2023) show that Wanda can also be extended to structured N:M pruning, which requires at most N out of every M contiguous weights to be non-zero (e.g. 2:4 or 4:8) (Mishra et al., 2021). While this usually results in lower performance, it is more amenable to practical inference speed-ups. Thus, future work should investigate how the core ideas behind M-Wanda generalize to structured pruning settings.

Acknowledgement

This work is supported by the Dutch National Science Foundation (NWO Vici VI.C.212.053).

References

Abhinav Bandari, Lu Yin, Cheng-Yu Hsieh, Ajay Jaiswal, Tianlong Chen, Li Shen, Ranjay Krishna, and Shiwei Liu. 2024. Is c4 dataset optimal for pruning? an investigation of calibration data for llm pruning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 18089–18099.

Davis Blalock, Jose Javier Gonzalez Ortiz, Jonathan Frankle, and John Guttag. 2020. What is the state of neural network pruning? *Proceedings of machine learning and systems*, 2:129–146.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel 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*, pages 2475–2485.

John Dang, Shivalika Singh, Daniel D'souza, Arash Ahmadian, Alejandro Salamanca, Madeline Smith, Aidan Peppin, Sungjin Hong, Manoj Govindassamy, Terrence Zhao, et al. 2024. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *arXiv preprint arXiv:2412.04261*.

Jonathan Frankle and Michael Carbin. 2018. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*.

Elias Frantar and Dan Alistarh. 2023. Sparsegpt: Massive language models can be accurately pruned in one-shot. In *International Conference on Machine Learning*, pages 10323–10337. PMLR.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh

Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, Will Smith, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah A. Smith, and Hannaneh Hajishirzi. 2024. Olmo: Accelerating the science of language models. *Preprint*.

Song Han, Jeff Pool, John Tran, and William Dally. 2015. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28.

Tahmid Hasan, Abhik Bhattacharjee, Md Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M Sohel Rahman, and Rifat Shahriyar. 2021. Xl-sum: Large-scale multilingual abstractive summarization for 44 languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4693–4703.

Yifei He, Alon Benhaim, Barun Patra, Praneetha Vaddamanu, Sanchit Ahuja, Parul Chopra, Vishrav Chaudhary, Han Zhao, and Xia Song. 2024. Scaling laws for multilingual language models. *arXiv preprint arXiv:2410.12883*.

Weizhong Huang, Yuxin Zhang, Xiawu Zheng, Fei Chao, and Rongrong Ji. 2025. Determining layer-wise sparsity for large language models through a theoretical perspective. *arXiv preprint arXiv:2502.14770*.

Yixin Ji, Yang Xiang, Juntao Li, Qingrong Xia, Ping Li, Xinyu Duan, Zhefeng Wang, and Min Zhang. 2024. Beware of calibration data for pruning large language models. *arXiv preprint arXiv:2410.17711*.

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. Association for Computational Linguistics.

Simon Kurz, Jian-Jia Chen, Lucie Flek, and Zhixue Zhao. 2024. Investigating language-specific calibration for pruning multilingual large language models. *arXiv preprint arXiv:2408.14398*.

Yann LeCun, John Denker, and Sara Solla. 1989. Optimal brain damage. *Advances in neural information processing systems*, 2.

Lujun Li, Peijie Dong, Zhenheng Tang, Xiang Liu, Qiang Wang, Wenhan Luo, Wei Xue, Qifeng Liu, Xiaowen Chu, and Yike Guo. 2024. Discovering sparsity allocation for layer-wise pruning of large language models. *Advances in Neural Information Processing Systems*, 37:141292–141317.

Chaitanya Malaviya, Graham Neubig, and Patrick Littell. 2017. Learning language representations for typology prediction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2529–2535.

Asit Mishra, Jorge Albericio Latorre, Jeff Pool, Darko Stosic, Dusan Stosic, Ganesh Venkatesh, Chong Yu, and Paulius Micikevicius. 2021. Accelerating sparse deep neural networks. *arXiv preprint arXiv:2104.08378*.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, et al. 2023. Crosslingual generalization through multitask finetuning. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Kelechi Ogueji, Orevaoghene Ahia, Gbemileke Onilude, Sebastian Gehrmann, Sara Hooker, and Julia Kreutzer. 2022. Intriguing properties of compression on multilingual models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9092–9110.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc-Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The lambada dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1525–1534.

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. 2020. Xcopa: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2362–2376.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Mingjie Sun, Zhuang Liu, Anna Bair, and J Zico Kolter. 2023. A simple and effective pruning approach for large language models. *arXiv preprint arXiv:2306.11695*.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Wayne 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.

Miles Williams and Nikolaos Aletras. 2024. On the impact of calibration data in post-training quantization and pruning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10100–10118.

Ge Yang, Changyi He, Jinyang Guo, Jianyu Wu, Yifu Ding, Aishan Liu, Haotong Qin, Pengliang Ji, and Xianglong Liu. 2024. Llmcbench: Benchmarking large language model compression for efficient deployment. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. Paws-x: A cross-lingual adversarial dataset for paraphrase identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, page 3687. Association for Computational Linguistics.

Lu Yin, You Wu, Zhenyu Zhang, Cheng-Yu Hsieh, Yaqing Wang, Yiling Jia, Gen Li, Ajay Kumar Jaiswal, Mykola Pechenizkiy, Yi Liang, et al. 2024. Outlier weighed layerwise sparsity (owl): A missing secret sauce for pruning llms to high sparsity. In *International Conference on Machine Learning*, pages 57101–57115. PMLR.

Hongchuan Zeng, Hongshen Xu, Lu Chen, and Kai Yu. 2024. Multilingual brain surgeon: Large language models can be compressed leaving no language behind. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 11794–11812.

Yingtao Zhang, Haoli Bai, Haokun Lin, Jialin Zhao, Lu Hou, and Carlo Vittorio Cannistraci. 2024. Plug-and-play: An efficient post-training pruning method for large language models. In *The Twelfth International Conference on Learning Representations*.

Yiran Zhao, Wenxuan Zhang, Guizhen Chen, Kenji Kawaguchi, and Lidong Bing. 2024. How do large language models handle multilingualism? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Xunyu Zhu, Jian Li, Yong Liu, Can Ma, and Weiping Wang. 2024. A survey on model compression for large language models. *Transactions of the Association for Computational Linguistics*, 12:1556–1577.

A Hyperparameter selection

	λ	ϵ	γ	CWL block
Llama3-1B	0.2	5e-5	0.04	attn
Llama3-3B	0.02	1e-7	0.04	attn
Llama3-8B	0.2	5e-5	0.04	attn
Aya-23-8B	0.2	1e-7	0.04	MLP
OLMo-7B	0.2	1e-7	0.04	MLP
Bloomz-7b1	6	0	0.01	attn

Table 5: Optimal hyperparameters found for M-Wanda after a small search $\lambda \in [0.02, 0.2]$, $\epsilon \in$ [5e-5, 1e-7], $\gamma = [0.01, 0.04]$ and CWL block $\in$ [attn, MLP] .

Llama3-1B	λ = 0.2	λ = 0.02
ϵ = 5e-5	**59.56**	60.30
ϵ = 1e-7	59.56	60.73

Llama3-3B	λ = 0.2	λ = 0.02
ϵ = 5e-5	24.67	24.59
ϵ = 1e-7	24.72	**24.52**

Llama3-8B	λ = 0.2	λ = 0.02
ϵ = 5e-5	**18.57**	18.62
ϵ = 1e-7	18.57	18.61

Table 6: Gridsearch results of varying λ and ϵ on Llama models under M-Wanda. We report average perplexity on calibration languages.

In Table 5, we report the optimal hyperparameters used when applying M-Wanda to each model. Note that these are the hyperparameters used for both the results on the Flores dataset, reported in Table 1, and the XL-Sum dataset, reported in Table 8. We find that, generally, $\lambda = 0.2$ and $\gamma = 0.04$ work well on most LLMs. See Table 6 for the gridsearch results.

However, interestingly, the optimal hyperparameter value for λ on Bloomz-7b1 is on a completely different scale, so we ran a different grid search for this model: $\lambda \in [6, 12]$. Moreover, we noticed that input activations from Bloomz-7b1 are sometimes equal to zero, resulting in suboptimal performance when applying the activation probabilities. Thus, while we found that this model benefits from cross-lingual variance and CWL, activation probability should be disabled ($\epsilon = 0$) for the improvements reported in the main paper.

In addition, Wanda$_{\textbf{+OWL}}$ results reported in Table 3, required the tuning of the hyperparameter M. We found that for Llama 1B and 3B $M = 3$ is optimal, yet for Llama 8B $M = 5$ yields better results. Similarly, while $\gamma = 0.08$ was reported to generally work best with OWL, we found that $\gamma = 0.04$, as used for CWL, performed better. As such, those are the values used for the results reported in the table. All experiments were performed using a single NVIDIA A100 GPU.

B Effectiveness of M-Wanda at different sparsity ratios

Sparsity	30%	35%	40%	45%	50%	55%	60%	65%	70%
Wanda	12.12	12.62	13.52	15.31	19.63	32.27	73.02	174.70	1743.14
M-Wanda	12.12	**12.60**	**13.43**	**15.02**	**18.57**	**28.46**	**58.96**	**159.48**	**835.63**

Table 7: Average perplexity scores of Wanda and M-Wanda across different sparsity levels. For reference, the average performance of the full model is 11.38. Results are reported on Llama-8B.

C XL-Sum results

	Wanda	M-Wanda
Llama3.2-1B	40.30	**36.88** (8% ↓)
Llama3.2-3B	16.40	**15.61** (5% ↓)
Llama3.1-8B	12.18	**11.57** (5% ↓)
Aya-23-8B	15.46	**15.14** (2% ↓)
Bloomz-7b1	20.35	**17.40** (14% ↓)
OLMo-7B	15.28	**14.34** (6% ↓)

Table 8: Average perplexity scores on the XL-Sum validation set across 13/15 languages at a sparsity ratio of 50%. We use 500 articles for each language. Note: German and Italian are not covered by this dataset.

D English-centric pruning

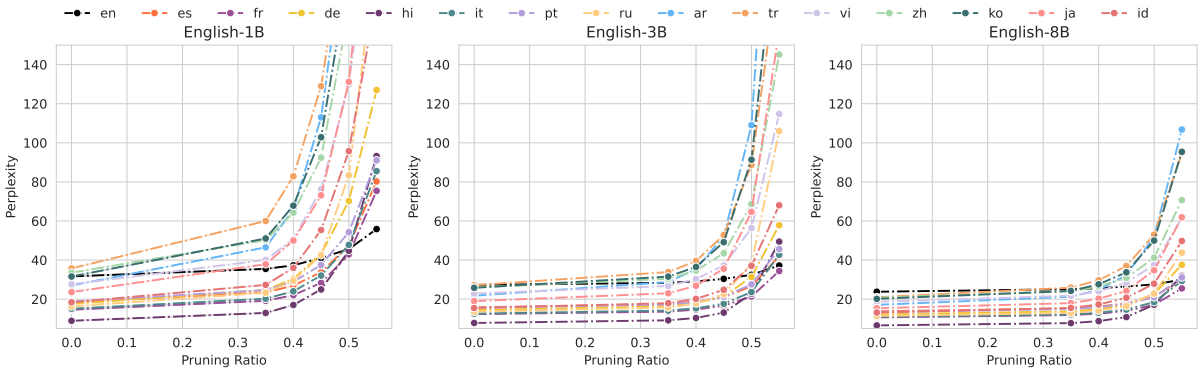

Figure 8: The effect of higher sparsity ratio's on the perplexity across languages. The calibration data is fully in English and perplexity is measured on the Flores dataset. Results are reported on the Llama3 models.

E Sparsity allocation with OWL

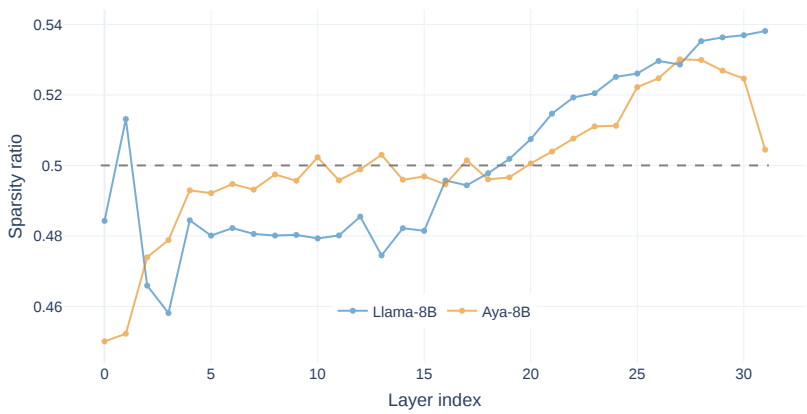

Figure 9: Layerwise sparsity allocation by OWL.

F Downstream task results per language

	Lang.	Wanda	M-Wanda
XCOPA	id	58.8	**60.4**
	it	62.8	**63.2**
	tr	57.8	**58.6**
	vi	59.6	**61.0**
	zh	**62.8**	62.6
XStory	ar	52.9	**53.8**
	en	**72.0**	71.8
	es	65.0	**65.2**
	hi	56.8	**57.8**
	id	58.7	**60.3**
	ru	61.7	**61.9**
	zh	58.9	**59.6**
XWinograd	en	**86.7**	86.0
	fr	69.9	69.9
	pt	73.8	**74.1**
	ru	66.7	**68.9**
	zh	72.0	**73.8**
Lambada	de	34.4	**36.3**
	en	**69.2**	71.9
	es	19.9	**22.8**
	fr	43.3	**45.8**
	it	43.6	**46.6**
XNLI	ar	32.8	**33.2**
	de	50.5	**51.9**
	en	**55.4**	55.2
	es	50.3	**52.3**
	fr	51.1	**52.3**
	hi	43.9	**46.5**
	ru	43.8	**47.3**
	tr	44.8	**45.6**
	vi	45.1	**45.3**
	zh	33.0	**35.5**
PAWS-X	de	63.3	**64.7**
	en	65.6	**65.7**
	es	58.4	**61.1**
	fr	58.5	**58.6**
	ja	**56.1**	53.2
	ko	49.3	**51.2**
	zh	50.8	**53.1**

Table 9: Per-language accuracy (%) for each downstream task using Wanda and M-Wanda on Llama-8B.