Supervised Routing for MoE: Aligning Experts with their Knowledge

Anonymous ACL submission

Abstract

Mixture-of-Experts models have low computational cost despite having a large number of parameters. However, the problem of unbalanced expert selection during routing leads to inefficient use of parameters. Thus, an auxiliary loss is used to make the expert selection uniform, but it has been found that this interferes with the performance of the language model. In this paper, we propose a supervised learning approach to Mixture-of-Experts routing using token frequencies as the supervised signal. This method aims to align the expert selection with the knowledge they have acquired. As a case study, we focus on domain adaptation for law. The proposed method without the auxiliary loss achieved performance comparable to a baseline with the auxiliary loss.

1 Introduction

Mixture-of-Experts (MoE) is an ensemble technique that combines the outputs of multiple modules known as experts (Jacobs et al., 1991; Jordan and Jacobs, 1993). The ensemble uses the output of a module called a router to weight the experts' contributions. MoE has been applied to deep learning (Eigen et al., 2013) and language modeling (Shazeer et al., 2017). Moreover, the sparse selection of experts offers the advantage of reduced computational costs relative to the number of parameters (Shazeer et al., 2017).

However, a major issue with expert selection in MoE using routers is the biased distribution of selections. Recent studies in language models have attempted to mitigate this bias by introducing a load-balancing loss (LB loss), which is added to the language model loss to promote more uniform routing (Shazeer et al., 2017; Lepikhin et al., 2021; Fedus et al., 2022). Nevertheless, the LB loss can interfere with the performance of language models (Wang et al., 2024).

In light of these issues, we propose **SvMoE** (**S**upervised **M**ixture-**o**f-**E**xperts), a method that involves training the router through supervised learning. Our method matches the selected experts with the knowledge they have acquired. We assume that the token frequencies in the training data correspond to the experts that specialize in these tokens. First, we divide the training data by clustering using TF-IDF. Then the router is trained using normalized token frequencies as a supervised signal. Specifically, our approach involves: (1) clustering the training data with TF-IDF (2) training each expert on the divided data, (3) merging them into an MoE model, (4) training the router with token frequencies, and (5) training the entire MoE model, thereby achieving appropriate training without using the LB loss.

As a case study, we focus on domain adaptation in Japanese law. We perform continual training using Japanese legal documents on a pretrained model and evaluate it on a bar exam benchmark (Choi et al., 2024). The router trained by our proposed method was able to make selections corresponding to token frequencies, and our method, without the LB loss, achieved performance almost equivalent to a baseline method with the LB loss.

2 Related Work

2.1 Mixture-of-Experts

MoE has been extensively used in Transformer-based language models (Lepikhin et al., 2021; Fedus et al., 2022; Du et al., 2021; Team et al., 2022; Shen et al., 2024; Jiang et al., 2024; DeepSeek-AI et al., 2024). These models predominantly employ a structure where the feedforward network (FFN) layers of Transformer blocks are arranged in parallel. They are trained for a language modeling objective.

Although MoE offers efficiency despite its number of parameters, it is known that during train-

ing, only specific experts are predominantly activated (Shazeer et al., 2017). This activation bias undermines the specialization of each expert and hampers the efficient use of parameters. This prompts the adoption of the LB loss to promote more uniform routing (Shazeer et al., 2017; Lepikhin et al., 2021; Fedus et al., 2022). The LB loss is incorporated into the training process alongside the standard language model loss.

However, Wang et al. (2024) have shown that the LB loss can adversely affect the performance of language models. This study highlights the need for hyperparameter tuning due to trade-offs between language model performance and the LB loss, and achieves improvements in routing without using the LB loss. Specifically, they propose adding a dynamically learned bias term to the router's output to ensure load balancing without interfering with the gradients of the language model loss. This approach is also used in subsequent research (DeepSeek-AI et al., 2024). In line with this perspective, our study aims to enhance MoE model performance without relying on the LB loss.

2.2 MoEs for Continual Training

Some studies, such as Komatsuzaki et al.'s (2023), focus on building MoE models from pre-trained dense models, while others, as discussed in Section 2.1, build them from scratch.

Branch-**T**rain-**M**erge (BTM) (Li et al., 2022) improves performance by merging individually trained expert models. Instead of parallelizing FFN layers, it parallelizes entire Transformer models, merging at the token logit level. Moreover, C-BTM (Gururangan et al., 2023) enables parallelized training across large datasets by using clustered training corpora along with the BTM approach. **B**ranch-**T**rain-Mi**X** (BTX) (Sukhbaatar et al., 2024) employs domain-partitioned data to fine-tune pre-trained Transformers. These models are then integrated by parallelizing their FFN layers to create an MoE model. To train the entire MoE model for routing, the LB loss is employed.

In this study, we use BTX as a baseline to explore routing learning without relying on the LB loss.

3 Supervised MoE

We propose **SvMoE**, a supervised Mixture-of-Experts method that trains the router using token frequencies from documents as supervised signals, which accounts for the negative impacts of the LB loss. We use TF-IDF to cluster the entire dataset into subdomains, where each expert is tailored to specialize in a particular subdomain. Instead of forced routing based on token frequency, the training of the router integrates the training data information and context information during inference. SvMoE consists of five stages: clustering the dataset, training the experts, merging the experts, supervised learning of the router, and fine-tuning the entire MoE model.

3.1 Creation of Training Data for the Router

In SvMoE, each expert is trained to specialize in a subdomain. Here, we create the supervised signals needed for routing that aligns with the data used to train each expert. First, we define the number of subdomains as N. We assume that the dataset for a particular domain $\mathcal{D}_0$ is pre-categorized into M categories where $M > N$. For each category, TF-IDF is computed by regarding a category as a document.

Using TF-IDF as features, we perform clustering to obtain N subdomain datasets d_n $(1 \leq n \leq N,\ d_i \cap d_j = \emptyset\ (\forall i \neq j))$. We set $\mathcal{D} = \bigcup_{n=1}^{N} d_n$ as the whole dataset for training.

Next, we extract the token frequencies that serve as the supervised signal for training the router. We count the tokens in each of the subdomain datasets, resulting in an $N \times |V|$ matrix, where V represents the vocabulary. The token frequency vector for each token $t \in V$ is expressed as

$$\mathrm{TF}_t = (\mathrm{tf}_{t,1}, \mathrm{tf}_{t,2}, \ldots, \mathrm{tf}_{t,N}) \in \mathbb{R}^N,$$

such that $\sum_{n=1}^{N} \mathrm{tf}_{t,n} = 1 \quad (\forall t \in V)$.

3.2 Model Construction

Training the Experts N experts $\{E_1, \ldots, E_N\}$ are trained using the dataset $\mathcal{D}$ prepared in Section 3.1. We use a pre-trained Transformer model and apply continual training to it separately on each subdomain for the purpose of language modeling, thereby obtaining a group of experts specialized in each subdomain.

Merging the Experts We merge the obtained experts to create the MoE model. Following the BTX approach, we set up the FFN layers of the expert group in parallel for each Transformer block, resulting in an MoE layer with N experts. For other layers, such as attention, we merge them by averaging, while we initialize the router of the MoE layers randomly.

Model	LB loss	PPL	CMR_1	S_{RL}
Dense	-	**1.156±0.462**	-	-
BTX	✓	1.187±0.407	4.063±2.237	1.805±0.198
	×	1.169±0.401	2.847±2.025	1.705±0.261
SvMoE	✓	1.245±0.443	2.538±2.097	**0.465±0.316**
	×	1.199±0.434	**2.533±2.092**	0.466±0.319

Table 1: Evaluation results on the test set. The best and second-best values are highlighted in **bold** and underlined, respectively. The proposed model is shown in the bottom row (SvMoE without the LB loss).

Supervised Learning of the Router We train the merged MoE model's router using the token frequency signals obtained in Section 3.1. At this stage, only the router's parameters are updated, and the rest remains frozen. Given that the model has L blocks, each block has its own router, resulting in L routers. We define the objective loss $\mathcal{L}_{\mathrm{SvMoE}}$ as follows when a training batch $b = t_1 t_2 \dots t_{|b|}$ ($t_i \in V$) is given as input:

$$\mathcal{L}_{\mathrm{SvMoE}} = \sum_{t \in b} \sum_{l=1}^{L} \ell_{\mathrm{CE}}(\mathrm{softmax}(RL_l), \mathrm{TF}_t),$$

where $RL_l \in \mathbb{R}^N$ represents the logits of the router in the l-th block, $\mathrm{softmax}(\cdot)$ is the softmax function, and $\ell_{\mathrm{CE}}(\cdot, \cdot)$ is the cross-entropy loss. By training the router's parameters using $\mathcal{L}_{\mathrm{SvMoE}}$, the model is encouraged to assign tokens frequently appearing in training data to the corresponding expert.

Fine-tuning the MoE Model We train the entire MoE model, with the trained router, for the purpose of language modeling. The LB loss is not used.

4 Model Construction Experiment

4.1 Setup

We describe the data processing, training setup, and evaluation. For our experiments, we set $N = 8$ and use llm-jp/llm-jp-3-1.8b[1] for both the model and the tokenizer. This tokenizer is also used for calculating TF-IDF and token frequencies.

Dataset We conduct experiments using Japanese legal texts. The dataset $\mathcal{D}_0$ is derived from the legal texts obtained from e-Gov, with legal categories ($M = 50$). Using the tokenizer, we compute TF-IDF and perform equal-size spectral clustering[2] to divide the dataset into eight subdomains, resulting in the dataset $\mathcal{D}$. We split the dataset into training, validation, and test sets in an 8:1:1 ratio.

Model First, we train llm-jp-3-1.8b for language modeling on each of the eight legal subdomains to obtain eight expert models. These experts are merged into an MoE model following the procedure described in Section 3.2. The resulting model (**SvMoE**) undergoes the training of the router and subsequent fine-tuning. For comparison, we also evaluate a model where the LB loss is added during the fine-tuning stage.

Additionally, we prepare a baseline model (**BTX**) by merging the eight experts and performing entire fine-tuning. For this BTX baseline, we evaluate both models with and without the LB loss during fine-tuning.

Furthermore, we include another baseline with a dense Transformer model (**Dense**), which is not based on MoE. This model is trained on the entire dataset without subdomain distinction.

Evaluation We compare the baseline models to our proposed model using the dataset $\mathcal{D}$. To assess the degree to which knowledge gained from continual pre-training is retained, we evaluate the models separately on training, validation, and test sets. The following three metrics are used:

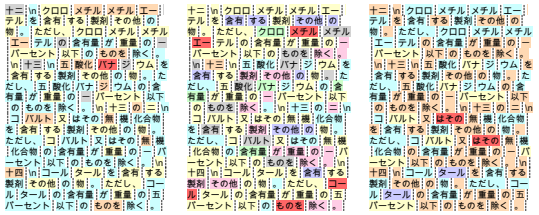
- PPL: The average perplexity of the model across given texts.
- CMR_k (Conditional Mean Rank)[3]: Given that the n-th expert's rank determined by the router logits (r_n^{RL}) equals k, the average of the n-th expert's rank determined by token frequencies (r_n^{TF}). i.e., $\mathbb{E}[r_n^{\mathrm{TF}} \mid r_n^{RL} = k]$. A value closer to k implies that routing follows token frequencies.
- S_{RL}: The average entropy of the router logits. A smaller value indicates greater confidence by the router.

4.2 Quantitative Results

Table 1 presents the evaluation results on the test set of the dataset $\mathcal{D}$.

First, regarding perplexity, Dense performs best, followed by BTX and SvMoE. In both BTX and SvMoE, the models without the LB loss yielded better outcomes. This aligns with previous findings (Wang et al., 2024), where the LB loss adversely affected model performance.

[1] A Japanese pre-trained model with 1.8B parameters.
[2] anamabo/Equal-Size-Spectral-Clustering
[3] We report values for $k = 1$. Results for $k = 2$ can be found in Appendix B.



(a) SvMoE (b) BTX (c) TF

(d) Legend for colors.

Figure 1: Comparison of the selected experts. In Figure 1a,1b, $l = 16$ is reported. Excerpt from the Specific Chemical Substances Hazard Prevention Regulations. Figure 1c shows the selection by the token frequencies.

For CMR_1, SvMoE outperformed BTX, indicating that training with token frequencies enables routing that aligns with the dataset on which the expert is trained. Similarly, for the entropy of the router logits S_{RL}, SvMoE showed superior results, suggesting that our training confidently activates the corresponding experts. Figure 3 in Appendix B also shows that SvMoE can select experts corresponding to the input token information.

In summary, while the proposed method facilitates token frequency-based routing, as intended, it does not translate to improved perplexity. This may be due to the method's emphasis on frequency information rather than syntactic cues, which are often used for expert selection in MoE (Jiang et al., 2024; Fan et al., 2024). Furthermore, it is known that lower perplexity does not necessarily equate to human-like performance (Kuribayashi et al., 2021).

4.3 Qualitative Results

Figure 1 illustrates an example of expert selection. The input sentence is taken from the test set of d_4.

In SvMoE, the experts E_5 and E_6, whose datasets have higher token frequencies for the input tokens, are frequently chosen. Conversely, in BTX, although E_4 is somewhat predominant, experts are selected more evenly overall. For instance, "クロロ" (Chloro) has high token frequencies for d_4 and d_5, and appears infrequently in d_1. In SvMoE, E_4 and E_6 are used for predicting "クロロ" and the subsequent word, respectively, whereas in BTX, E_3 and E_1 are employed.

Qualitatively, SvMoE makes selections based more on token frequency information than BTX, confirming its alignment with token frequencies in expert selection.

Model	LB loss	accuracy (%)
Dense	-	48.33
BTX	✓	**51.67**
	×	47.22
SvMoE	✓	48.33
	×	<u>51.11</u>

Table 2: Results for the Japanese bar exam.

5 Evaluation on a Downstream Task

We compare the proposed method with baselines in a downstream task. We focus on a Japanese bar exam benchmark constructed by Choi et al. (2024).

Setup Given that the bar exam includes complex formats such as selecting correct combinations, we use a light task format that involves determining the correctness of individual sentences, referred to as a binary judgment task. We evaluate the five models constructed in Section 4 in a few-shot setting. Five questions from the 2019 bar exam are chosen as the few-shot examples, and 180 questions from the 2023 bar exam are evaluated.

Results Table 2 shows the results for the binary judgment task. The BTX baseline with the LB loss achieved the highest score, and the score dropped by more than four points without the LB loss. Conversely, SvMoE achieved a high score even without the LB loss, nearly matching the BTX baseline with the LB loss. When the LB loss is applied to SvMoE, the score drops by about three points compared to when it is not used. This suggests that SvMoE can replace the need for the LB loss.

Although SvMoE did not outperform the baseline in terms of perplexity evaluation in Section 4.2, it demonstrated comparable performance in the downstream task. Given the influence of the LB loss on the model performance, as shown in Table 2, the proposed method is advantageous. The fact that it does not require hyperparameter tuning suggests its usefulness.

6 Conclusion

We proposed the SvMoE framework, which trains the router without relying on the LB loss, thereby avoiding its negative impact on model performance. We constructed an MoE model using the proposed method and confirmed that it enables routing based on the characteristics of the dataset on which each expert is trained. Additionally, in a downstream task, our approach achieved performance comparable to that of baselines with the LB loss.

Limitations

We focused on the continual learning for adaptation in the legal domain as a case study. In other words, it has not been verified in general MoE, including pre-training. However, we believe that the proposed method, which consists of the clustering using TF-IDF and the supervised learning based on token frequencies, has general applicability.

We only conducted the experiment in Japanese because we believe that the proposed method is not affected by the language. In addition, the exploration of model size is also future work.

The router architecture may require improvements. In this paper, following previous research, we used linear transformations to select experts. However, the router had relatively few parameters compared to other components, which may have been insufficient for effective learning.

Furthermore, we did not thoroughly examine which layers should be targeted for load balancing. As seen in Figure 1, the routing behavior varies greatly across layers. Therefore, it may be beneficial to limit the layers subjected to load balancing or to adjust the coefficients for each layer.

References

Jungmin Choi, Jungo Kasai, and Keisuke Sagaguchi. 2024. Evaluation of gpt models on the japanese bar exam. In *Proceedings of the Thirtieth Annual Meeting of the Association for Natural Language Processing*. In Japanese.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, and 181 others. 2024. Deepseek-v3 technical report. arXiv. Abs/2412.19437.

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 Bosma, Zongwei Zhou, Tao Wang, Yu Emma Wang, Kellie Webster, Marie Pellat, Kevin Robinson, and 8 others. 2021. Glam: Efficient scaling of language models with mixture-of-experts. arXiv. Abs/2112.06905.

David Eigen, Marc'Aurelio Ranzato, and Ilya Sutskever. 2013. Learning factored representations in a deep mixture of experts. arXiv. Abs/1312.4314.

Dongyang Fan, Bettina Messmer, and Martin Jaggi. 2024. TOWARDS AN EMPIRICAL UNDERSTANDING OF MOE DESIGN CHOICES. In *ICLR 2024 Workshop on Mathematical and Empirical Understanding of Foundation Models*.

William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39.

Suchin Gururangan, Margaret Li, Mike Lewis, Weijia Shi, Tim Althoff, Noah A. Smith, and Luke Zettlemoyer. 2023. Scaling expert language models with unsupervised domain discovery. *Preprint*, arXiv:2303.14177. Abs/2303.14177.

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. 1991. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, and 7 others. 2024. Mixtral of experts. arXiv. Abs/2401.04088.

M.I. Jordan and R.A. Jacobs. 1993. Hierarchical mixtures of experts and the em algorithm. In *Proceedings of 1993 International Conference on Neural Networks (IJCNN-93-Nagoya, Japan)*, volume 2, pages 1339–1344 vol.2.

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*.

Tatsuki Kuribayashi, Yohei Oseki, Takumi Ito, Ryo Yoshida, Masayuki Asahara, and Kentaro Inui. 2021. Lower perplexity is not always human-like. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5203–5217, Online. Association for Computational Linguistics.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. 2021. {GS}hard: Scaling giant models with conditional computation and automatic sharding. In *International Conference on Learning Representations*.

Margaret Li, Suchin Gururangan, Tim Dettmers, Mike Lewis, Tim Althoff, Noah A. Smith, and Luke Zettlemoyer. 2022. Branch-train-merge: Embarrassingly parallel training of expert language models. In *First Workshop on Interpolation Regularizers and Beyond at NeurIPS 2022*.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff

Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*.

Sheng Shen, Le Hou, Yanqi Zhou, Nan Du, Shayne Longpre, Jason Wei, Hyung Won Chung, Barret Zoph, William Fedus, Xinyun Chen, Tu Vu, Yuexin Wu, Wuyang Chen, Albert Webson, Yunxuan Li, Vincent Y Zhao, Hongkun Yu, Kurt Keutzer, Trevor Darrell, and Denny Zhou. 2024. Mixture-of-experts meets instruction tuning: A winning combination for large language models. In *The Twelfth International Conference on Learning Representations*.

Sainbayar Sukhbaatar, Olga Golovneva, Vasu Sharma, Hu Xu, Xi Victoria Lin, Baptiste Roziere, Jacob Kahn, Shang-Wen Li, Wen tau Yih, Jason E Weston, and Xian Li. 2024. Branch-train-mix: Mixing expert LLMs into a mixture-of-experts LLM. In *First Conference on Language Modeling*.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2022. No language left behind: Scaling human-centered machine translation. arXiv. Abs/2207.04672.

Lean Wang, Huazuo Gao, Chenggang Zhao, Xu Sun, and Damai Dai. 2024. Auxiliary-loss-free load balancing strategy for mixture-of-experts. *Preprint*, arXiv:2408.15664. Abs/2408.15664.

ID	Train	Val.	Test	Total
1	23.0M	3.2M	1.9M	28.1M
2	18.0M	2.0M	2.6M	22.6M
3	24.0M	3.5M	2.4M	29.9M
4	32.1M	3.8M	4.7M	40.5M
5	31.3M	4.3M	4.1M	39.7M
6	73.4M	5.9M	8.1M	87.4M
7	21.0M	2.9M	2.5M	26.4M
8	39.5M	2.0M	5.3M	46.9M

Table 3: The number of tokens per subdomain.

Model	LB loss	CMR_2
Dense	-	-
BTX	✓	4.742±2.206
	×	4.058±2.152
SvMoE	✓	**3.503±2.374**
	×	3.518±2.365

Table 4: CMR_1 results for the test set.

Figure 2: Loss curves for the proposed method and baselines.

A Details of the Clustering

As mentioned in the main text, we perform clustering on categories obtained from e-Gov[4] using TF-IDF as features. Due to the uneven distribution of the data, the categories 'National Tax,' 'Finance and Insurance,' and 'Local Finance' are fixed as individual subdomains. In other words, we cluster the remaining 47 categories into 5 clusters. Table 3 shows the number of tokens per subdomain and the results of the clustering are shown in Table 5.

B Details of the Model Construction

Setup For a training batch b, the LB loss $\mathcal{L}_{\mathrm{LB}}$ is formulated as:

$$\mathcal{L}_{\mathrm{LB}} = N \sum_{i=1}^{N} \mathcal{D}_i \mathcal{P}_i,$$

$$\mathcal{D}_i = \frac{1}{|b|} \sum_{t \in b} \mathbb{1}\{\mathrm{argmax}\, \mathcal{G}(x_t) = i\},$$

$$\mathcal{P}_i = \frac{1}{|b|} \sum_{t \in b} \mathcal{G}(x_t),$$

where x_t is the hidden state for token t and $\mathcal{G}(\cdot)$ is the router function including the top-k process. $\mathcal{D}_i$ and $\mathcal{P}_i$ represent the proportion of tokens assigned to E_i and the proportion of the routing probability for E_i, respectively. Using this $\mathcal{L}_{\mathrm{LB}}$ and the language model loss $\mathcal{L}_{\mathrm{LM}}$, the objective loss used in the entire training of MoE models $\mathcal{L}$ is defined as:

$$\mathcal{L} = \mathcal{L}_{\mathrm{LM}} + \alpha \mathcal{L}_{\mathrm{LB}},$$

where α is a hyperparameter which determines the LB loss's weight.

[4] https://laws.e-gov.go.jp/bulkdownload

For all models with an MoE architecture, we set the coefficient for the LB loss α to 0.01, following the BTX setup. The number of experts selected per inference is fixed at 2.

Results The loss curve during model training is shown in Figure 2. In the initial stages of training, the pre-trained router in SvMoE displayed lower loss than BTX, which was initialized randomly. However, the final training loss showed the best results for BTX without the LB loss. As indicated in Section 5, loss does not necessarily directly correlate with model performance.

Next, CMR_2 for the test set is reported in Figure 4. The trend was the same as CMR_1 mentioned in Section 4.2.

The evaluation results on the training and validation sets are presented in Table 6, which follow the same methodology used for the test set results in Table 1. Overall, a tendency similar to the test set results was observed, but better outcomes were shown for the training set compared to the others.

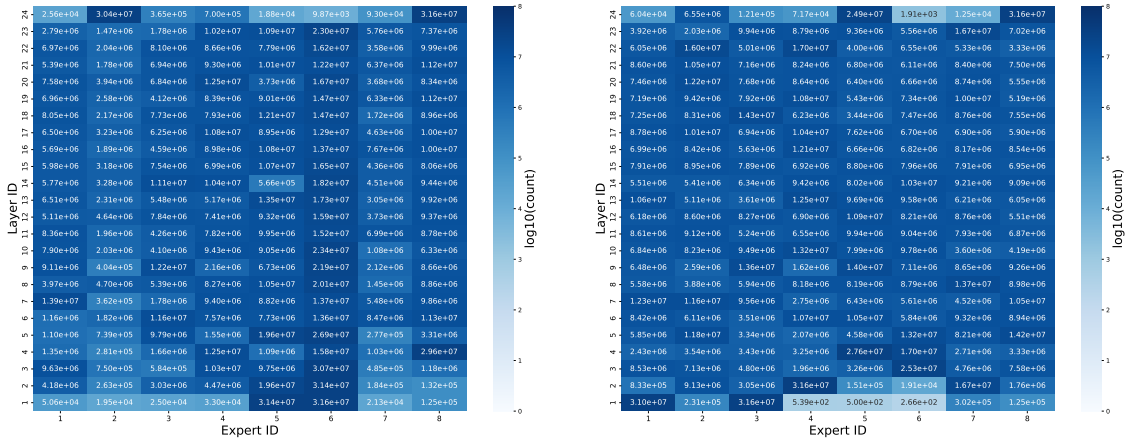
Additionally, Figure 3 displays the results of counting the frequency of expert selections on the test set. As shown in Table 3, there is a imbalance in the number of input data, and SvMoE predominantly selects experts corresponding to this presence. On the other hand, BTX selects experts relatively evenly, especially when the LB loss is used, irrespective of the input data distribution. It is also observed that near the input and output layers, a bias in selections arises under all conditions.

ID	Category Names
1	‘Criminal’, ‘National Diet’, ‘Mining’, ‘Enterprise’, ‘Commerce’, ‘Land’, ‘Culture’, ‘Judiciary’, ‘Civil’, ‘Foreign Affairs’
2	‘Fisheries’, ‘Government Bonds’, ‘Local Autonomy’, ‘General Industry Provisions’, ‘Disaster Response’
3	‘Constitution’, ‘General Finance Provisions’, ‘Postal Services’, ‘Administrative Procedures’, ‘Urban Planning’, ‘Roads’, ‘Social Welfare’, ‘Social Insurance’, ‘Forestry’, ‘Freight Transport’
4	‘National Property’, ‘Administrative Organization’, ‘National Public Employees’, ‘National Land Development’, ‘Labor’, ‘Statistics’, ‘Education’, ‘Maritime’, ‘Agriculture’, ‘Defense’
5	‘Tourism’, ‘Police’, ‘Firefighting’, ‘Industry’, ‘Telecommunications’, ‘Environmental Protection’, ‘Foreign Exchange and Trade’, ‘Health’, ‘Land Transport’, ‘Rivers’, ‘Aviation’, ‘Building and Housing’
6	‘National Tax’
7	‘Finance and Insurance’
8	‘Local Finance’

Table 5: Categories of legal classifications assigned to each subdomain. The original category names are in Japanse.

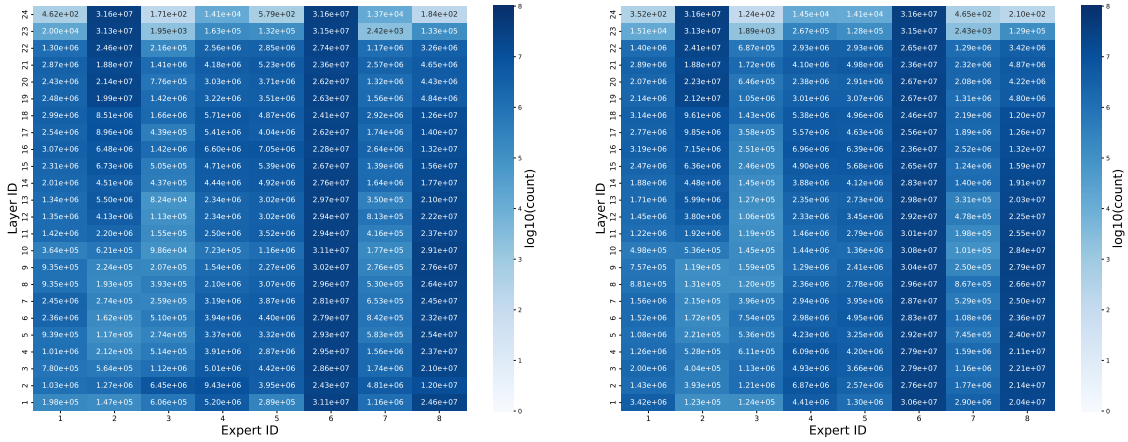
Split	Model	LB loss	PPL	CMR_1	CMR_2	S_{RL}
Train	Dense	-	**1.042±0.087**	-	-	-
	BTX	✓	1.089±0.159	4.043±2.254	4.755±2.205	1.802±0.201
		×	1.069±0.145	2.802±2.035	4.087±2.150	1.691±0.269
	SvMoE	✓	1.141±0.196	**2.473±2.078**	**3.543±2.369**	**0.449±0.316**
		×	1.099±0.202	2.477±2.079	3.563±2.360	0.450±0.318
Valid.	Dense	-	**1.154±0.331**	-	-	-
	BTX	✓	1.188±0.320	4.145±2.214	4.749±2.201	1.803±0.199
		×	1.169±0.316	3.057±2.086	4.086±2.152	1.716±0.256
	SvMoE	✓	1.243±0.348	2.687±2.164	**3.558±2.397**	0.477±0.327
		×	1.201±0.340	**2.681±2.161**	3.569±2.385	**0.476±0.329**

Table 6: The evaluation results on the data used for training. The best and second-best values within each set are highlighted in **bold** and underlined, respectively.



(a) BTX (without the LB loss)

(b) BTX (with the LB loss)



(c) SvMoE (without the LB loss)

(d) SvMoE (with the LB loss)

Figure 3: Frequencies of expert selection per layer for each model.