Mechanistic Interpretability Implementation Project (Language Specific Neurons)

Lang-Spec-Neurons is an implementation project for the "Mechanistic Interpretability" course offered by Frederick Riemenschneider in WS23/24.

It's based on the paper Language-Specific Neurons: The Key to Multilingual Capabilities in Large Language Models , with the forked repository linked here

This README serves as the project report and a PDF version is found here.

▯ File Contents

- Python scripts:
    - Loading and saving tensors of wikipedia datasets
    - [Original] Recording activation state (with code additions for GPT3)
    - [Original] Identifying language-specific neurons (with code additions for GPT3)
    - [Original] Open-ended generation when deactivating neurons (with code additions for GPT3)
- Results Analysis:
    - Jupyter Notebook
- Supplementary materials:
    - Llama2 generations with deactivated neurons in each language
    - ai-forever/mGPT generations with deactivated neurons in each language

Preparation

To create a conda environment and install dependencies, please run: `conda env create -f environment.yml` .

Introduction

The paper "Language-Specific Neurons: The Key to Multilingual Capabilities in Large Language Models" [1] explores multilingual capabilities of modern LLM's by identifying language specific neurons. Specifically, by introducing the language activation probability entropy (LAPE) method, they pinpoint a subset of language specific neurons in the model's bottom and top layers. Through various experiments, they probe the multilingual proficiency in representative models, such as LLaMa-2 [2], BLOOM and Mistral. Likewise, this project aims to explore the multilingual capabilities of two open-source representative models, LLaMa-2 and mGPT. Concretely, the research questions are formulated as:

Research Questions:

1. Will the deactivation of a neuron likewise result in performance degradation, i.e. for text generation?
2. Is likewise activating specific neurons able to "steer" the output language?

We aim to answer these questions using English, German and Turkish data for two autoregressive transformer decoder-only models: LLaMa-2 [2] and mGPT [3]. We choose mGPT for being trained besides the default English data specifically also on German and Turkish data. Alongside observations with regards to each specific language, we also hope to inspect whether the LAPE method for identifying "language neurons" is applicable to different model types and sizes. With that, we will also investigate, whether perturbing the neurons will result in different outcomes when comparing these models.

LAPE

For extensive explanation of the LAPE method, we refer the reader to the original paper [1]. In short: The LAPE score is calculated for each neuron j at the i-th FFN layer in the respective language. First, the activation probability of each neuron is computed by empirically estimating the likelikhoof of that neuron's activation exceeding zero in that language. The authors only regard neurons surpassing a defined threshold as "language specific" in the following step. After applying L1 normalization, the LAPE score is obtained as the entropy of that distribution. Neurons with a lower LAPE score indicate activation for one or two languages and stay unactivated for others, thus are deemed language specific. The opposite holds for constant activations across languages. Similarly to the initial threshold, only the lowest percentile of LAPE scores, specifically the bottom 1% are selected as the final language neurons.

Experiments

In line with the paper, experiments are conducted using the LLaMa-2 model [2], which we pair with the forever-ai/mGPT [3] model as a more language-aware multilingual transformer model.

Dataset Preparation

First, we downloaded English, German and Turkish wikipedia text data from https://huggingface.co/datasets/wikimedia/wikipedia. Each language is tokenized and concatenated using each pre-trained model's tokenizer. The resulting token lists are then converted to LongTensors and saved individually as files. The following displays the according tensor lengths:

1. German, [4242334]
2. Turkish, [4876973]
3. English, [6287540]

To adhere to mGPT's maximum sequence length, we additionally set padding and truncation to the model max length.

Activation states from each of the saved tensor list of each languge data are obtained by running `python activation.py` , which are then saved under the activations directory respectively. From these activations, according language specific neurons are identified with the LAPE method in `identify.py` . The resulting neuron activation files are saved individually per model under the activation_mask directory.

Results

The "language specific" neurons are identified with the following metrics. Please note that the top probability value denotes the 95th percentile of all activation probabilities for each language. This is for means of filtering out those neurons with no frequent activation in any language. The selected probabilities size then determines the number of the top neurons based on the specified activation frequency, which in our computation stays static across the languages.

Model	Language	Top_prob_value	Selected_probs_size
LLaMa-2	EN	0.5280590057373047	3523
	GE	0.6435436010360718	"-"
	TR	0.7302824854850769	"-"
mGPT	EN	0.6470092535018921	1966
	GE	0.7583200335502625	"-"
	TR	0.7775343656539917	"-"

Findings:

LLaMa-2

- Perturbing the LLaMa-2 model generally causes the model to increasingly generate in English for both German and Turkish data
    - The biggest effect of perturbing neurons is recorded for English data, with nearly a double increase in length of model output in case of the perturbed Turkish neuron (780600/1569071 ~ 0.5). Interestingly, again perturbing the Turkish neurons has the second biggest effect with 60% of reduced output in case of German data (833957/1327300 ~ 0.63)
    - Remarkably, perturbing any language neuron on English data results in increased lengths of model output (Compare (non-)perturbation), while this does not hold for German and Turkish data, with a reduction of the output in the majority cases. Namely for German data (English (1252.5), German (1295.4) and Turkish(834)), as well as 2/3 cases of Turkish data: German (314.0), Turkish (453.4). As noted above, still for these reduced outputs Llama-2 increasingly generates in English.

mGPT

- Perturbing mGPT language specific neurons doesn't seem to have an effect on the language proficiency of the GPT model, since the specific neuron's perturbation still results in the same statistics across different generation data (i.e. Data_lang).
    - Also, perturbation generally does not seem to have any effect whatsoever on the output generation of the model, with the overlapping statistics for both settings of (non-)perturbed neurons.

General

- LLaMa - in comparison to mGPT - predominantly generates sensible outputs for the prompted questions
    - For a substantial amount of cases, the model answers in a different provided language, such as the below case for English with off-turned Turkish neurons. We note also, that the desired political correctness of the model is non-apparent for this case:

```
 "input": "Q: Aztekler, İspanyol fatihleri başarıyla püskürtmüş olsaydı ne olurdu? Lütfen Türkçe cevap ver.\nA:",
"output": "If the Spanish had won at Lepanto, they would have been able to conquer the Ottoman Empire and Islam would not be a
```

-
- mGPT as well as LLaMa, in many cases, solely repeat the question from the given prompt when perturbed, predominantly on Turkish and German data. We mark this as a special case of hallucination.
    - Typical hallucination appears, when models generate completely disassociated and out-of-context outputs, which we also observe for mGPT[3].
- Lastly, the stop word ratios are naturally increasing with increased model output lengths.

LLaMa-2

Perturbation

Data_lang	Perturb_lang	Average Output Length	Stop Word Ratio
en	en	940.629	0.390
en	de	1345.214	0.432
en	tr	1569.071	0.316
de	en	1252.457	0.195
de	de	1295.400	0.205
de	tr	833.957	0.102
tr	en	536.686	0.029

Data_lang	Perturb_lang	Average Output Length	Stop Word Ratio
tr	de	314.014	0.002
tr	tr	453.400	0.003

Normal

Lang	Average Output Length	Stop Word Ratio
en	780.600	0.408
de	1327.300	0.203
tr	471.114	0.008

mGPT

Perturbation

Data_lang	Perturb_lang	Average Output Length	Stop Word Ratio
en	en	953.471	0.366
en	de	953.471	0.366
en	tr	953.471	0.366
de	en	874.357	0.396
de	de	874.357	0.396
de	tr	874.357	0.396
tr	en	1007.143	0.112
tr	de	1007.143	0.112
tr	tr	1007.143	0.112

Normal

Lang	Average Output Length	Stop Word Ratio
en	953.471	0.366
de	874.357	0.396
tr	1007.143	0.112

Conclusion

When perturbing identified language neurons in LLaMa-2 [2] and mGPT [3], we found that this generally causes LLaMa-2 to increasingly generate in English for both German and Turkish data, while simultaneously reducing its ouput.These observations may expose LLaMa-2's insecurity to these languages when perturbed. For mGPT we found no effect whatsoever of perturbing identified language neurons generally and across languages. Regarding the research questions, we can answer them in the following way:

1. Deactivating the language neurons did indeed result in performance degradation, since more cases of the LLaMa-2 model repeating questions (in the other languages) and hallucinating appear. Though it remains open, whether language-specific neurons are specifically responsible for this effect (or other "non-language" neurons could have had the same effect)
2. (Re-)activating distinguished language neurons requires an implementation that similarly sets the activation values to defined top values. Due to limited resources, we were unable to facilitate this and thus leave this point of investigation for future work.

References

1. 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.

2. Meta, Llama 2: Open Foundation and Fine-Tuned Chat Models.
3. Shliazhko, Oleh ; Fenogenova, Alena ; Tikhonova, Maria ; Mikhailov, Vladislav ; Kozlova, Anastasia ; Shavrina, Tatiana April 2022 mGPT: Few-Shot Learners Go Multilingual Transactions of the Association for Computational Linguistics, 2024

✏ Maintenance

Author: Raziye Sari - sari@cl.uni-heidelberg.de
Last updated: May 21th 2025