Quantifying Robustness in White-Box Instruction-Tuned Large Language Models

Justin S. Lee
Independent
jsl2239@columbia.edu

Abstract

Instruction-tuned large language models (LLMs) are trained to generate outputs that adhere to requests in their input prompts. This work proposes and examines two candidate metrics for the sensitivity of instruction-tuned LLMs to semantically irrelevant perturbations: measurement of (1) the KL divergence between the distributions of the first output token for a perturbed and unperturbed prompt, and (2) the Euclidean distance between the mean-pooled token embeddings of the perturbed and unperturbed prompt. We test these metrics on various 7-billion parameter instruction-tuned LLMs against a small sample of MMLU questions. We find, for all models tested, that both metrics are higher on average when a perturbation results in a change in the meaning of the output, but that only some of these are statistically significant.

1 Introduction

Transformer-based (Vaswani et al., 2023) large language models (LLMs) have been noted for their ability to mimic human language and provide accurate answers to tasks that require reasoning. They are trained to do so by closely approximating the distribution of word sequences that a human might provide in response to an arbitrary input. Instruction-tuned LLMs are further trained to generate outputs that adhere to requests in their input prompts. In the instruction setting, an LLM should be robust to perturbations to its input that do not change its semantic intent, such as formatting, substitution of words with synonyms, phrasing, or noise. Robustness against such perturbations has real-world implications; it should be expected that users will phrase the same request in different ways according to factors such as personal writing style, language proficiency, dialect, and typographical errors. Normatively speaking, for a language model to be useful across use cases, it must be robust to these factors.

Because LLM model weights are fixed at inference time, it may be possible to measure the sensitivity of a given model to such perturbations. The objective of this work is to develop metrics for the robustness of white-box instruction-tuned large language lodels (LLMs) to various kinds of noisy perturbations to their prompts. Here, a "noisy perturbation" refers to any change to a prompt that, semantically, should not affect the corresponding completion. An analogous situation can be expressed as follows: given a function $f : \mathbb{R} \to \mathbb{R}$ and input x, we can measure how sensitive f is in the neighborhood of x by examining the derivative $f'(x)$. If $|f'(x)|$ is high, adding a small perturbation ϵ to x would result in large changes in the function output $f(x + \epsilon)$; likewise, a small value for $|f'(x)|$ would correspond to small changes in $f(x + \epsilon)$.

In the LLM setting, we can consider f as corresponding to a language model, x to a prompt, and ϵ to a noisy perturbation. As established above, perturbations that preserve the semantic contents of a prompt should result in a corresponding preservation of the semantic content of the model output. We hypothesize that perturbing a prompt so as to shift the semantic meaning of a completion will result in a measurable difference in some aspect of the model outputs; in the analogy above, this would correspond to measuring the derivative of f in the neighborhood of its input. The basis for our hypothesis is that if the output completion changes in meaning in response to a semantically meaningless perturbation, it indicates that the model is mistaking the perturbation for a meaningful signal.

We test our hypothesis by examining two candidate metrics for the sensitivity of instruction-tuned LLMs to semantically irrelevant perturbations: measurement of (1) the KL divergence between the distributions of the first output token for a perturbed and unperturbed prompt, and (2) the Euclidean distance between the mean-pooled to-

ken embeddings of the perturbed and unperturbed prompt. We choose these candidates because they both represent, in different ways, the model's semantic representation of the input.

We test our hypothesis on several instruction-tuned models by applying two types of perturbations to test set questions from the Massive Multitask Language Understanding (MMLU) dataset (Hendrycks et al., 2021). We then examine the model activations and and output probability distributions conditional on the presence or absence of a semantic shift in the model output.

2 Prior Literature

Broadly speaking, examining neural network features to understand their correspondence to semantic concepts falls under the domain of mechanistic intepretability (Elhage et al., 2022). One common approach in this area is to train "probe models" to extract interpretable information from intermediate model activations. For instance, activations might be used to label inputs as having a certain human-interpretable feature. The accuracy of the probe model would then correspond to the extent to which the model extracts the interpretable feature from its input. This approach can be applied across domains, from computer vision (Alain and Bengio (2018), Kim et al. (2018)), to embedding/autoregressive language models for natural language (Tenney et al. (2019), Wang et al. (2022), and the board games Othello (Li et al., 2023) and chess (Karvonen, 2024).

Specifically, Tenney, et al. study whether specific portions of pretrained BERT models output features corresponding to linguistic concepts, such as parts of speech or named entities. The authors train nonlinear classifiers on various per-token representations derived from the BERT model to classify single spans into various linguistic settings, and pairs of spans into pairwise relationship categories. They find that different layers tend to yield more suitable representations for different tasks. For instance, part-of-speech tags were best extracted from earlier layers. At a higher level, the authors found that basic aspects of language are extracted in earlier layers, and more complex aspects in later layers.

Novak et al. (2018) examines the sensitivity of neural network classifiers to their input. The authors propose that the robustness of a neural network, that is, the degree to which its output changes with respect to its input, is distinct from the functional complexity of the model. Further, they propose that networks will exhibit different sensitivity behaviors depending on the subspace that the input belongs to. The authors attempt to quantify these concepts into metrics for the case of a fully connected network for classification with piecewise-linear activations (e.g. ReLU). For nonlinear activation functions, approximations are made to split the outputs into discrete linear regions. The authors propose two sensitivity metrics:

1. the Frobenius norm of the Jacobian of the output class probabilities, referred to as "Jacobian norms"

2. transition counts, where model activations are mapped to discrete codes, and changes in the codes are counted along input trajectories

The authors' demonstrate for a set of image classification models that low sensitivity is correlated with high generalization. The same wis not found with the transition count, as larger models tend to yield higher transition counts without a corresponding increase in the generalization gap. However, techniques that are typically assumed to improve a model's generalization, such as ReLU nonlinearities and data augmentation, resulted in an observable decrease in both the Jacobian norm and transition count, and therefore the sensitivity of the model.

Jin et al. (2020) introduce a technique called TextFooler, which adversarially perturbs language model prompts to degrade model performance in a black-box setting. The models tested are for classification and textual entailment. The authors claim that TextFooler outperforms previous methods in (1) the success rate of its adversarial text in fooling models, (2) preservation of the original characteristics of the text pre-perturbation, and (3) computational complexity (being $O(N)$ with respect to the length of the original text). Across five classification tasks and two textual entailment tasks, the authors show that TextFooler reduces performance on almost all models tested to below 10% after modifying fewer than 20% of the words in the prompt.

Wu et al. (2024) examines the extent to which language models create representations of their input for the express purpose of being used after the immediate next token position. In light of previous research (Pal et al., 2023) demonstrating that

tokens after the immediate next token can be predicted from model hidden states, the authors offer and test two potential explanations: (1) the model creates features that are not useful for predicting the next token, but are useful for subsequent ones, and (2) the model creates input representations that are useful across time steps. In the autoregressive next token prediction setting, they find more evidence in support of second explanation.

3 Data

We created two perturbation datasets. First, a roughly 2% sample of the MMLU test set questions was taken from each of the 57 categories. MMLU was chosen because its multiple choice question (MCQ) format makes semantic shifts in model outputs simple to judge. The following perturbations were then applied to each question:

1. `SINGLECHAR` - a random non-punctuation symbol (one of "@", "#", "*", "{", "}", "‘", and "\"), was added to a random position in the question.

2. `SEMANTIC` - OpenAI's GPT-3.5 Turbo (OpenAI, 2022) was prompted to reword the question.

The GPT-3.5 perturbations were manually inspected against the original question for semantic match. If the reworded question did not match the original, it was either manually corrected, or the original question was discarded from both datasets. The final datasets consisted of 200 questions each. Because our interest was in isolating the ability of the model to understand the semantic content of a user's intent, the answers to the questions were not perturbed.

4 Methods

The following 7-billion parameter models were examined in this work:

- `Llama2-7b-instruct` (Touvron et al., 2023)
- `Mistral-7B-Instruct-v0.2` (Jiang et al., 2023)
- `Gemma 7B Instruct` (Team et al., 2024)

The models were 4-bit quantized due to resource constraints. For each model m, the original, `SINGLECHAR`, and `SEMANTIC` perturbed questions were given as input. Completions were requested to be JSON-formatted to reduce the likelihood of long-form answers, and generated via greedy decoding. For each question, the following data were recorded:

1. the probability distribution $\mathbf{p}_m$ over the vocabulary for the first output token

2. the mean-pooled final layer hidden state $\mathbf{h}_m$ representation of the input prompt

Let `ORIGINAL` represent data from the unperturbed question, and `PERTURBED` one of the perturbation schemes. The following scalar quantities, KL divergence and Euclidean distance, were computed as follows:

- $KL(\mathbf{p}_{m,\texttt{ORIGINAL}} || \mathbf{p}_{m,\texttt{PERTURBED}})$
- $||\mathbf{h}_{m,\texttt{ORIGINAL}} - \mathbf{h}_{m,\texttt{PERTURBED}}||_2$

Finally, GPT-3.5 Turbo was used to generate a binary label for whether the respective completions for the original and perturbed inputs were in semantic agreement or disagreement. These labels were manually vetted for accuracy and corrected where necessary.

Note that we did not check whether the completions were correct or incorrect. This is because the performance of the models against ground truth is not relevant to how sensitive a model is to perturbation. That is, even if a model outputs an incorrect answer to a question, it should do so consistently if the prompt is perturbed in a way that keeps the semantic content of the question intact.

All code to generate the data and run the experiments are shared in a GitHub repository [1].

5 Analysis

For each model, the mean and standard deviation of each datapoint were computed, conditional on the semantic agreement/disagreement of the perturbed and unperturbed completions. The summary statistics for KL divergence and Euclidean distance are shown in Tables 1 and 2, respectively. We find that our results align with basic intuition. Across all models tested and with both types of perturbation, the KL divergence and Euclidean distance had a higher mean for semantic agreement compared

[1] https://github.com/jsylee/llm-robustness

Table 1: KL Divergence Summary Statistics

Model	Perturbation	Completion Agreement	Mean	SD	Support
Gemma	SINGLECHAR	Yes	5.465e-4	1.098e-3	190
	SINGLECHAR	No	5.670e-4	5.463e-4	10
	SEMANTIC	Yes	1.220e-3	5.128e-3	165
	SEMANTIC	No	2.155e-3	4.868e-3	35
Llama2	SINGLECHAR	Yes	1.129e-5	2.499e-5	177
	SINGLECHAR	No	5.393e-5	2.191e-4	23
	SEMANTIC	Yes	2.971e-5	7.103e-5	150
	SEMANTIC	No	2.986e-5	4.514e-5	50
Mistral	SINGLECHAR	Yes	9.062e-3	4.292e-2	174
	SINGLECHAR	No	6.757e-2	1.995e-1	26
	SEMANTIC	Yes	6.574e-2	3.237e-1	151
	SEMANTIC	No	2.033e-1	5.720e-1	49

to disagreement. Further, conditional on semantic agreement, SEMANTIC perturbation yielded a higher average KL divergence and Euclidean distance compared to the simpler SINGLECHAR perturbation.

To determine whether the distributional differences between semantic agreement and disagreement populations were significant, Welch's t-tests were conducted on the Euclidean distance results, the results of which are shown in Table 3. The null hypothesis, H_0, was that the distributions had the same mean, while the alternative hypothesis, H_1 was that the average Euclidean distance is higher when the completions disagree. For Gemma 7B and Llama2 7B, we find that with a significance level of 0.05, we are able to reject H_0 for SEMANTIC perturbation. In all other cases, there was not sufficient evidence to reject H_0 at the chosen significance level.

Welch's t-tests were not performed for the KL divergence statistics, as the distributions exhibited high skew, which would violate the normality assumption for that test.

6 Conclusion

In this work, we explore two metrics for measuring the sensitivity of an instruction-tuned LLM to various degrees of input perturbation. For all three models examined above, we find that perturbation yields a higher average score for both metrics when that perturbation elicits a change in the meaning of the output. We find that some of these distributional differences are statistically significant.

Potential areas for future exploration include creating sensitivity metrics based on the gradients of the models (à la Novak, et al.), expanding the work to a larger number of MMLU questions, and more types of perturbation. Another area of interest is the extent to which the robustness of a model in the MCQ setting correlates with robustness in long-form settings. Finally, if future work more concretely supports the hypothesis for either of the candidates discussed in this work, it may be possible to convert it from a metric to a loss function - that is, to fine-tune models after pretraining to align the model's representations of semantically equivalent phrasings of a request.

Known Project Limitations

Due to time and resource constraints, all analyses done in this paper were based on 4-bit quantized models. It is not known whether this had an effect on the results in this paper.

This analysis was also done on a relatively small number of data points, and therefore the extent to which the results shown here are reproducible on other MMLU subsets is unknown. The method by which the final 200 MMLU questions were determined also introduces a risk of selection bias. GPT-3.5 Turbo struggled to rephrase questions that contained long passages, and so these were disproportionately rejected. This characteristic was also strongly correlated with certain MMLU subjects.

Another area for potential improvement is the fact that only the first token KL divergences were analyzed. This is at odds with the fact that the models were instructed to provide outputs in JSON, meaning the first token is almost always an opening curly bracket ("{").

Table 2: Euclidean Distance Summary Statistics

Model	Perturbation	Completion Agreement	Mean	SD	Support
Gemma	SINGLECHAR	Yes	6.218	2.596	190
	SINGLECHAR	No	9.016	5.411	10
	SEMANTIC	Yes	12.308	4.415	165
	SEMANTIC	No	14.228	3.528	35
Llama2	SINGLECHAR	Yes	5.047	1.695	177
	SINGLECHAR	No	5.429	1.834	23
	SEMANTIC	Yes	7.052	2.335	150
	SEMANTIC	No	8.084	2.067	50
Mistral	SINGLECHAR	Yes	13.809	4.997	174
	SINGLECHAR	No	14.086	5.666	26
	SEMANTIC	Yes	21.955	7.909	151
	SEMANTIC	No	24.024	7.620	49

Table 3: Euclidean Distance Welch's T-test Results

Model	Perturbation	t-statistic	p-value	Reject H_0?	Degrees of Freedom
Gemma	SINGLECHAR	-1.543	0.078	No	9.198
	SEMANTIC	-2.757	0.004	Yes	58.390
Llama2	SINGLECHAR	-0.929	0.180	No	26.911
	SEMANTIC	-2.933	0.002	Yes	93.376
Mistral	SINGLECHAR	-0.232	0.409	No	30.878
	SEMANTIC	-1.622	0.054	No	83.624

Authorship Statement

Justin Lee was solely responsible for this work.

References

Guillaume Alain and Yoshua Bengio. 2018. Understanding intermediate layers using linear classifier probes.

Nelson Elhage, Tristan Hume, Catherine Olsson, Neel Nanda, Tom Henighan, Scott Johnston, Sheer ElShowk, Nicholas Joseph, Nova DasSarma, Ben Mann, Danny Hernandez, Amanda Askell, Kamal Ndousse, Andy Jones, Dawn Drain, Anna Chen, Yuntao Bai, Deep Ganguli, Liane Lovitt, Zac Hatfield-Dodds, Jackson Kernion, Tom Conerly, Shauna Kravec, Stanislav Fort, Saurav Kadavath, Josh Jacobson, Eli Tran-Johnson, Jared Kaplan, Jack Clark, Tom Brown, Sam McCandlish, Dario Amodei, and Christopher Olah. 2022. Softmax linear units. *Transformer Circuits Thread*. Https://transformer-circuits.pub/2022/solu/index.html.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is bert really robust? a strong baseline for natural language attack on text classification and entailment.

Adam Karvonen. 2024. Emergent world models and latent variable estimation in chess-playing language models.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. 2018. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav).

Kenneth Li, Aspen K. Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023. Emergent world representations: Exploring a sequence model trained on a synthetic task.

Roman Novak, Yasaman Bahri, Daniel A. Abolafia, Jeffrey Pennington, and Jascha Sohl-Dickstein. 2018. Sensitivity and generalization in neural networks: an empirical study.

OpenAI. 2022. Introducing chatgpt. https://

openai.com/blog/chatgpt, Last accessed on 2024-04-21.

Koyena Pal, Jiuding Sun, Andrew Yuan, Byron Wallace, and David Bau. 2023. Future lens: Anticipating subsequent tokens from a single hidden state. In *Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL)*, pages 548–560, Singapore. Association for Computational Linguistics.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, Justin Mao-Jones, Katherine Lee, Kathy Yu, Katie Millican, Lars Lowe Sjoesund, Lisa Lee, Lucas Dixon, Machel Reid, Maciej Mikuła, Mateo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Oscar Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko Yotov, Rahma Chaabouni, Ramona Comanescu, Reena Jana, Rohan Anil, Ross McIlroy, Ruibo Liu, Ryan Mullins, Samuel L Smith, Sebastian Borgeaud, Sertan Girgin, Sholto Douglas, Shree Pandya, Siamak Shakeri, Soham De, Ted Klimenko, Tom Hennigan, Vlad Feinberg, Wojciech Stokowiec, Yu hui Chen, Zafarali Ahmed, Zhitao Gong, Tris Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, Oriol Vinyals, Jeff Dean, Koray Kavukcuoglu, Demis Hassabis, Zoubin Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy. 2024. Gemma: Open models based on gemini research and technology.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. Bert rediscovers the classical nlp pipeline.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need.

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. 2022. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small.

Wilson Wu, John X. Morris, and Lionel Levine. 2024. Do language models plan ahead for future tokens?