Do large language models use grammar to solve natural language tasks?

Ishan Shah
The University of Texas at Austin
ishan0102@utexas.edu

Abstract

Large language models are widely used to perform natural language tasks like summarization, translation, and generation. While they achieve remarkable performance, they are black boxes, and many questions remain as to how human-like their language processing is. One major question is whether they rely on syntactic structures like those argued to underlie human language. In this study, we investigate whether language models use an internal representation of grammatical subjecthood and objecthood in performing complex downstream tasks. By iteratively training classifiers to predict subjects and objects from embeddings, we aim to identify and systematically destroy a model's ability to classify grammatical subjecthood and objecthood, resulting in an ablated model. If an ablated model can still do complex tasks, then we know that these representations were not used. If it cannot, then this is evidence that these grammatical categories are used downstream in these tasks. By evaluating a RoBERTa model fine-tuned on the Multi-Genre Natural Language Inference (MNLI) corpus before and after this procedure, we find that accuracy drops on tasks requiring subject and object information. Our findings demonstrate how language models develop an internal representation of grammar through self-supervised learning that they actively use in complex natural language tasks.

1 Introduction and Prior Work

ChatGPT has taken the Internet by storm. It is the most recent in a series of large language models released over the last few years. Large language models can be thought of as systems designed to predict the next most probable word in a series. For example, given the prompt "I'm a student at the University of", OpenAI's GPT-3 (specifically `text-davinci-003`) (Brown et al., 2020) found the following words most probable:

Word	Probability
Washington	6.42%
Michigan	5.74%
California	4.59%
Toronto	4.34%
Texas	3.01%
Maryland	2.34%

Table 1: GPT-3's most probable completions for the prompt "I'm a student at the University of".

These models learn natural language by ingesting billions of pages of human-written text from the Internet. This text is crafted into training data by hiding text, letting the model guess the next words, and correcting the model with what the actual words should have been, a technique known as self-supervised learning (Liu et al., 2020). Doing this with enough text allows language models to learn the structure of language with a fairly high degree of accuracy. Language models are important today because there are a multitude of tasks that can be solved with next-word prediction, from code generation and translation to question answering and summarization. In addition, by doing this training, they seem to learn more than just next-word prediction — they learn syntax.

However, there is a debate about whether these models actually learn syntax. Trained language models are represented as large neural networks with parameters known as weights and biases. Instead of taking inputs as text directly, language models use embeddings, which are pieces of language encoded into a lower-dimensional vector space to make it easier to perform computations on them. Some research shows that linguistic hierarchical structure emerges in neural networks as sentence tree structures are able to be reconstructed from these learned embeddings (Manning et al., 2020). This is exciting because it illustrates that these models implicitly embed syntactic structure, essentially learning grammar on their own. However, other linguists argue that deep neural networks' understanding of syntax can be explained using non-grammatical factors (Linzen and Baroni, 2020). These conflicting views demonstrate how a lack of interpretability in neural networks makes it difficult to understand *how* they learn language.

One way researchers have tried to understand how language models learn is by probing their abilities on tasks like subject-object classification (Papadimitriou et al., 2021, 2022). In this task, a binary classifier is trained to predict whether a layer embedding is a subject or an object. This task is informative because subjecthood comprehension is a fundamental part of language understanding. Prior results demonstrate that learned embeddings of language models are in fact influenced by grammatical features like subjecthood. This suggests that language models are able to learn syntax.

With this in mind, we can explore further. If language models are able to accurately classify subjects and objects, does that imply that destroying a language model's internal representation of subjecthood would make it worse at classifying subjects and objects? Furthermore, would a language model without a representation of subjecthood perform worse on downstream evaluation tasks that implicitly require subjecthood comprehension?

In this paper, we explore these questions through two experiments, comparing the performance of a regular language model and a language model with its subjecthood representation removed. In the first experiment, we evaluate the performance of both models on a general task that does not specifically focus on subjecthood. In the second, we evaluate the models on a corpus which places a stronger emphasis on subjecthood understanding. We expect that the language model without a subjecthood representation will perform the same in the first experiment, and worse on the second.

2 Methods

Our experimental procedure can be described in three steps and is illustrated in Figure 1.

1. **Training:** Take a regular RoBERTa model and destroy its knowledge of subjecthood to create a nulled RoBERTa model.

2. **Evaluation:** Evaluate the regular and nulled models on a general language understanding task.

3. **Evaluation:** Evaluate the regular and nulled models on a subjecthood-specific language understanding task.

We will cover each of these steps in more detail below.

2.1 Training INLP

In order to destroy a language model's representation of subjecthood, we turn to a method known as Iterative Nullspace Projection (INLP) (Ravfogel et al., 2020). Intuitively, we can think of INLP as a process that finds where the model stores its knowledge on subjecthood and objecthood and removes that knowledge until we believe the model is sufficiently ablated. It is a process that, when done correctly, can completely destroy a language model's understanding of any topic.

Mathematically, INLP neutralizes a language model's ability to predict a property, Z, from a set of representations, H, by iteratively training classifiers $c_1, c_2, \ldots, c_n$ to predict Z, and removing each classifier's contribution to the model's predictions. Effectively, this method keeps performing dimensionality reduction within latent space, each time removing the best classification subspace. The actual mechanics of removing a classifier's contribution are by using nullspace projection, where all data points are projected onto the nullspace of the classifier's weights, which is the classification subspace learned in training. This process is repeated for n classifiers, and the final representation is the nullspace of the last classifier's weights.

Measuring a language model's abilities before and after INLP is a method known as Amnesic Probing (Elazar et al., 2021), which dictates that we can measure the utility of a property by measuring the model's ability to predict it before and after INLP. For our experiments, we will train binary classifiers to distinguish the layer embeddings of nouns that are transitive or intransitive subjects from transitive objects. We will train separate classifiers for each layer of the model, each time performing 30 iterations of INLP. Through the process, we will measure the model's ability to predict the subjecthood of nouns before and after INLP.

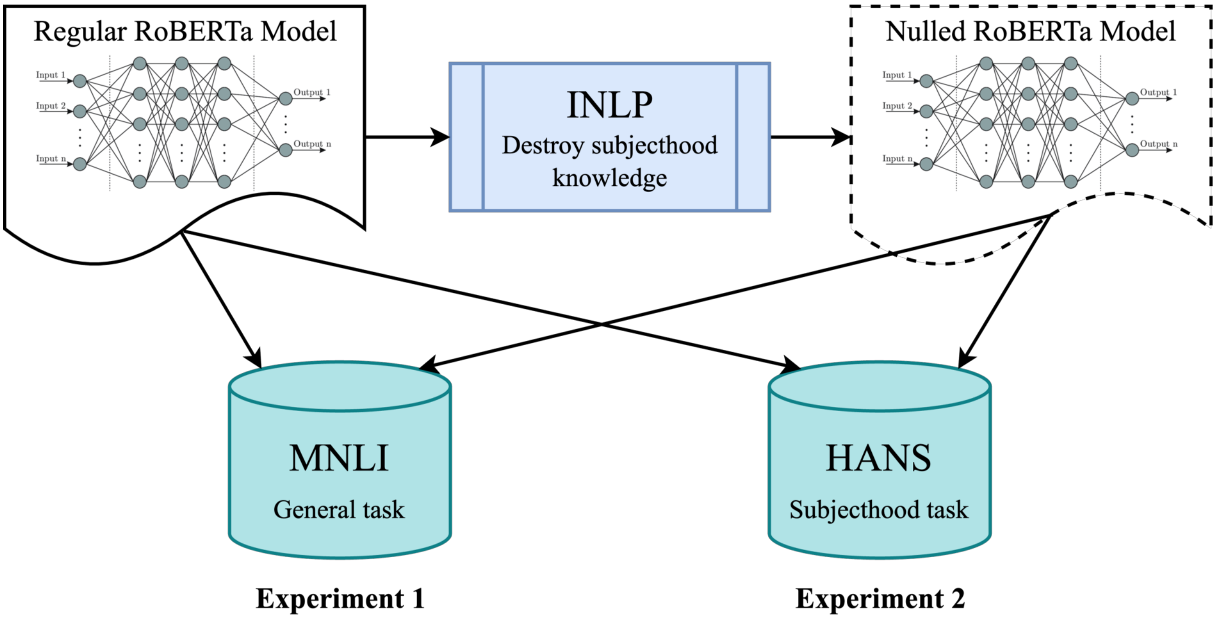

Figure 1: A depiction of the entire experiment. We start with a regular `roberta-large-mnli` model, perform INLP on it to remove subjecthood information, and evaluate the regular and nulled models against the MNLI and HANS datasets. Critically, we use the MNLI dataset to verify that we are not degrading the model's performance on non-subjecthood tasks.

The training data for the INLP classifiers comes from the Universal Dependencies treebank. Layer embeddings are annotated with information on whether it represents a transitive subject, intransitive subject, transitive object, or none of the above. The language model embeddings we use are `roberta-large-mnli`, a RoBERTa model (Liu et al., 2019) with 24 hidden layers fine-tuned on the Multi-Genre Natural Language Inference (MNLI) corpus.

2.2 Model Evaluation

For our downstream evaluation tasks, we use the MNLI corpus (Williams et al., 2018) and the Heuristic Analysis for NLI Systems (HANS) corpus (McCoy et al., 2019). Natural Language Inference (NLI) involves training a language model to predict whether a premise entails or contradicts a hypothesis. For example, “A soccer game with multiple people playing” *entails* that “People are playing a sport” and *contradicts* that “People are sleeping”. MNLI applies this task to multiple genres and there are about 10,000 examples in the test evaluation set. HANS is similar as it is designed to evaluate MNLI systems but we are interested in the 2,000 passive sentences from this dataset as they allow us to directly probe subjecthood. This dataset contains examples like “The managers were introduced by the scientists” *contradicts* “The managers introduced the scientists”.

What we expect from our experiments is to find that MNLI tasks do not have a significant drop in accuracy after performing INLP to null out subjecthood, since MNLI does not always depend on subjecthood. There are, however, cases when MNLI uses subjecthood information, such as “My ankle” *entails* “My body part”. In a case like this we would expect a drop in performance, though only a fraction of MNLI tasks depend on subjecthood information.

On the other hand, we expect worse performance on the HANS dataset since the examples generally depend on subjecthood. Model performance on the MNLI and HANS datasets prior to performing INLP should also be relatively high since the `roberta-large-mnli` model is fine-tuned on an MNLI dataset. All experiments were run on the UT Computational Linguistics Research Group's cluster, which consists of 4 Nvidia A40 GPUs.

3 Results

Through the process of training classifiers for INLP, we can visualize the model's ability to classify subjects and objects before (Figure 2) and after (Figure 3) performing INLP. Prior to INLP, the model is best able to accurately classify subjects and objects in the middle hidden layers. As the number of classifiers increases to 30, evaluation accuracy decreases for all layers, since we are gradually removing more and more subjecthood information.

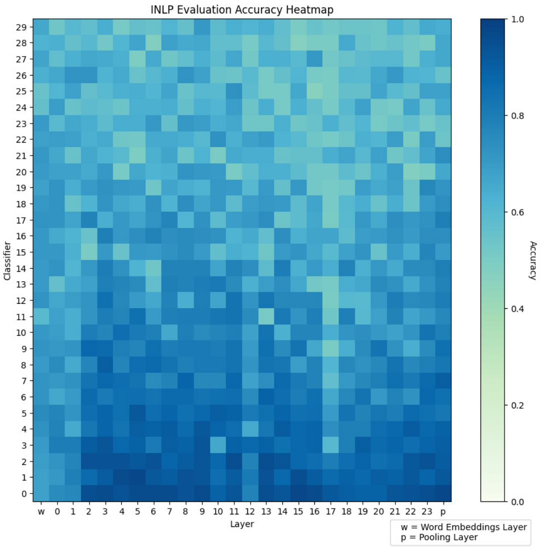

Figure 2: Heatmap of evaluation accuracy on a subject-object task while performing INLP. The x-axis is the number of layers of the model, and the y-axis is the number of classifiers. We expect that as the number of classifiers increases, accuracy should decrease because we are nulling out more information.

After performing INLP, we observe that the model accuracy is significantly lower and there is not a clear pattern of accuracy across layers. The model is unable to accurately classify subjects and objects even in the middle hidden layers, despite having the ability to do so prior to INLP. This confirms that the INLP method is able to null out subjecthood information in the model.

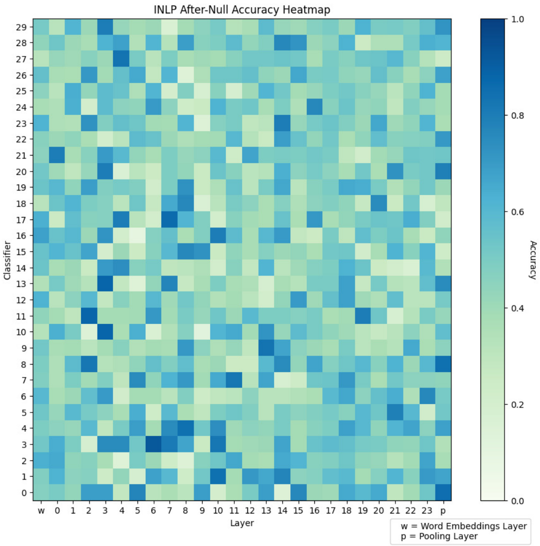

Figure 3: Heatmap of after-null accuracy on a subject-object task while performing INLP. The x-axis is the number of layers of the model, and the y-axis is the number of classifiers. We see that accuracy drops across all layers and that there is not a general trend to this plot, which is what we expected.

Since our primary reason to use the MNLI dataset was to make sure that the model still retained performance, we expected to not see a significant drop in accuracy. Our results show that the model's accuracy remains within 2% of the baseline accuracy for all layers. This confirms that the model still retains performance on the MNLI dataset after performing INLP, which we expected since the dataset does not always depend on subjecthood.

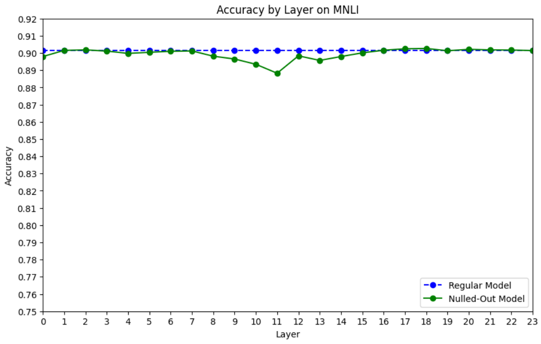

Figure 4: Accuracy on the MNLI dataset with models before and after having layers nulled out.

On our HANS dataset, we saw quite different results. On 6 of the hidden layers, the model's accuracy dropped by more than 6%, and 3 layers dropped more than 9%. This implies that the model's performance is significantly worse on the HANS dataset after performing INLP, which we expected since the dataset does depend on subjecthood.

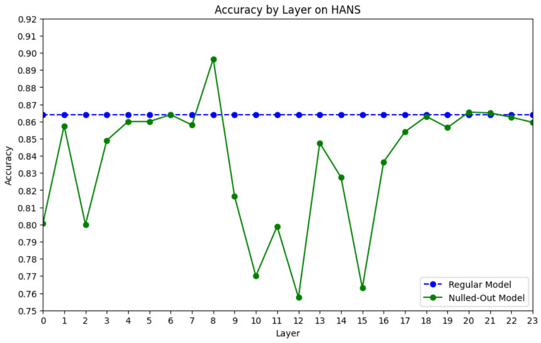

Figure 5: Accuracy on the HANS dataset with models before and after having layers nulled out.

Interestingly, when we look at some of the examples of errors on the MNLI dataset, we see that they tend to be cases where subject and object information is important. For example, one error was on the example “The FAA cleared the airspace” *entails* “Airspace was cleared by the FAA”. This is reminiscent of the passive sentences from our HANS dataset, which are examples where subjecthood is important. Another example exhibiting the same phenomenon is “Laura has! declared Sophie, glancing at me” *entails* “Sophie glanced at me as she declared, Laura has!”. Upon closer inspection, it appears that about 40% of the errors on the MNLI dataset involve subjecthood in some form.

4 Discussion and Conclusion

Our experimental results demonstrate that language models do indeed use grammar to derive their representations of subjecthood and objecthood. We were able to null out subjecthood information in the model by performing INLP, which resulted in a significant drop in accuracy on the HANS dataset. We also observed that the model's accuracy on the MNLI dataset was not significantly affected by performing INLP, as we expected since the dataset does not always depend on subjecthood. This is a promising result for future research in interpretability, as it shows that language models can be used to discover grammar from unstructured data.

In future research, we can apply other techniques like AlterRep (Ravfogel et al., 2021) to attempt to invert a language model's representation of subjecthood and objecthood rather than destroying it. Another new technique is Linear Adversarial Concept Erasure (R-LACE) (Ravfogel et al., 2022), designed to be a replacement for INLP. It may be interesting to see whether we can attain the same results using different intermediary methods. We could also try using other variations of the HANS dataset to compare performance against more control groups.

While this paper strengthens the answer to the question of *if* language models use grammar, future interpretability research should explore *how* language models derive their representations of grammar to begin with. It is intriguing that human language can be discovered from unstructured data, and there is a lot of research to be done in understanding how this process works.

References

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *CoRR*, abs/2005.14165.

Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. 2021. Amnesic probing: Behavioral explanation with amnesic counterfactuals. *Transactions of the Association for Computational Linguistics*, 9:160–175.

Tal Linzen and Marco Baroni. 2020. Syntactic structure from deep learning. *CoRR*, abs/2004.10827.

Xiao Liu, Fanjin Zhang, Zhenyu Hou, Zhaoyu Wang, Li Mian, Jing Zhang, and Jie Tang. 2020. Self-supervised learning: Generative or contrastive. *CoRR*, abs/2006.08218.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Christopher D. Manning, Kevin Clark, John Hewitt, Urvashi Khandelwal, and Omer Levy. 2020. Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proceedings of the National Academy of Sciences*, 117(48):30046–30054.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Isabel Papadimitriou, Ethan A. Chi, Richard Futrell, and Kyle Mahowald. 2021. Deep subjecthood: Higher-order grammatical features in multilingual BERT. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2522–2532, Online. Association for Computational Linguistics.

Isabel Papadimitriou, Richard Futrell, and Kyle Mahowald. 2022. When classifying arguments, BERT doesn't care about word order...except when it matters. In *Proceedings of the Society for Computation in Linguistics 2022*, pages 203–205, online. Association for Computational Linguistics.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. 2020. Null it out: Guarding protected attributes by iterative nullspace projection. In *Proceedings of the 58th Annual*

Meeting of the Association for Computational Linguistics, pages 7237–7256, Online. Association for Computational Linguistics.

Shauli Ravfogel, Grusha Prasad, Tal Linzen, and Yoav Goldberg. 2021. Counterfactual interventions reveal the causal effect of relative clause representations on agreement prediction. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 194–209, Online. Association for Computational Linguistics.

Shauli Ravfogel, Michael Twiton, Yoav Goldberg, and Ryan Cotterell. 2022. Linear adversarial concept erasure. *CoRR*, abs/2201.12091.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.