Mechanistic Interpretability on Irreducible Integers

Noah Syrkis
University of Copenhagen
noah@syrkis.com

Abstract

We apply the mechanistic interpretability framework to a transformer model trained on a dataset of irreducible integers. We show that the model has learned to perform modular addition, and we reverse-engineer the model to understand how it does so.

1 Introduction

Reverse-engineerings deep neural networks (DNN) is a relatively new field, but has already shown success. For example, reverse engineers a transformer model to understand how it performs modular addition. attempts to automate the reverse-engineering process, and is somewhat successful. Reverse-engineerings deep neural networks (DNN) is a relatively new field, but has already shown success. For example, reverse engineers a transformer model to understand how it performs modular addition. attempts to automate the reverse-engineering process, and is somewhat successful. Reverse-engineerings deep neural networks (DNN) is a relatively new field, but has already shown success. For example, reverse engineers a transformer model to understand how it performs modular addition. attempts to automate the reverse-engineering process, and is somewhat successful. Reverse-engineerings deep neural networks (DNN) is a relatively new field, but has already shown success. For example, reverse engineers a transformer model to understand how it performs modular addition. attempts to automate the reverse-engineering process, and is somewhat successful. Reverse-engineerings deep neural networks (DNN) is a relatively new field, but has already shown success. For example, reverse engineers a transformer model to understand how it performs modular addition. attempts to automate the reverse-engineering process, and is somewhat successful. Reverse-engineerings deep neural networks (DNN) is a relatively new field, but has already shown success. For example, reverse engineers a transformer model to understand how it performs modular addi-

tion. attempts to automate the reverse-engineering process, and is somewhat successful.

Mechanistic interpretability (MI) posits that deep neural networks (DNN) are circuits that can be reverse-engineered to understand their inner workings. MI is a relatively new field, but has already shown success. For example, reverse engineers a transformer model to understand how it performs modular addition. attempts to automate the reverse-engineering process, and is somewhat successful.

Mechanistic interpretability (MI) posits that deep neural networks (DNN) are circuits that can be reverse-engineered to understand their inner workings. MI is a relatively new field, but has already shown success. For example, reverse engineers a transformer model to understand how it performs modular addition. attempts to automate the reverse-engineering process, and is somewhat successful.

2 Background

2.1 Transformer models

As artificial intelligence systems Mechanistic interpretability (MI) posits that deep neural networks (DNN) are circuits that can be reverse-engineered to understand their inner workings. MI is a relatively new field, but has already shown success. For example, Nanda et al. [5] reverse engineers a transformer model [6] to understand how it performs modular addition. Cover and Thomas [3] attempts to automate the reverse-engineering process, and is somewhat successful. However, the process is still largely manual and requires a deep understanding of the model's architecture and training process. Leveraging the transformer model's attention mechanism, Conmy et al. [2] attempts to automate the reverse-engineering process, and is somewhat successful. Conmy et al. [2] attempts to automate the reverse-engineering process, and is somewhat successful, while Belcák et al. [1] diconfirms this.

In this paper, we apply the MI framework to a transformer model trained on a dataset of irreducible integers. We show that the model has learned to perform modular addition, and we reverse-engineer the model to understand how it does so.

2.2 Mechanistic interpretability

Mechanistic interpretability (MI) posits that deep neural networks (DNN) are circuits that can be reverse-engineered to understand their inner workings. MI is a relatively new field, but has already shown success. For example, Nanda et al. [5] reverse engineers a transformer model [6] to understand how it performs modular addition.

2.3 Irreducible integers

Irreducible integers are primes, though thet are described as such so as to allo the title of the paper to be a reference to Douglas Hofstades MIU puzzle.

3 Methodology

Our methodology consists of the following steps:

3.1 Data

Here's a table:

Table 1: Dataset

0	1	2	3	4	5	6	7
0	1	2	3	4	5	6	7
10001	10003	10007	10009	0	0	1	1
10011	10013	10017	10019	0	0	0	0
99981	99983	99987	99989	0	0	0	1
99991	99993	99997	99999	1	0	0	0

The dataset consists of four-digit integers and their labels. The labels are 1 if the integer is irreducible, and 0 otherwise. The dataset is generated by taking all four-digit integers and checking if they are irreducible. The dataset is then split into a training set and a test set.

3.2 Model

The model is a transformer model in the style of Hu et al. [4]. It is trained on the dataset above.

3.3 Reverse-engineering

We reverse-engineer the model by analyzing the attention weights. We show that the model has learned to perform modular addition. We reverse-engineer the model by analyzing the attention weights. We show that the model has learned to perform modular addition. We reverse-engineer the model by analyzing the attention weights. We show that the model has learned to perform modular addition. We reverse-engineer the model by analyzing the attention weights. We show that the model has learned to perform modular addition.

4 Results

We reverse-engineer the model by analyzing the attention weights. We show that the model has learned to perform modular addition. We reverse-engineer the model by analyzing the attention weights. We show that the model has learned to perform modular addition. We reverse-engineer the model by analyzing the attention weights. We show that the model has learned to perform modular addition.

4.1 Circuits

We see these circuits:

- Circuit 1
- Circuit 2
- Circuit 3

4.2 Attention weights

We see these attention weights:

- Attention weight 1
- Attention weight 2

4.3 Modular addition

We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular ad-

Table 2: Dataset

0	1	2	3	4	5	6	7
0	1	2	3	4	5	6	7
10001	10003	10007	10009	0	0	1	1
10011	10013	10017	10019	0	0	0	0
99981	99983	99987	99989	0	0	0	1
99991	99993	99997	99999	1	0	0	0

dition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition. We show that the model has learned to perform modular addition.

5 Analysis

Lorem Lorem ipsum dolor sit amet, consectetur adipisci elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

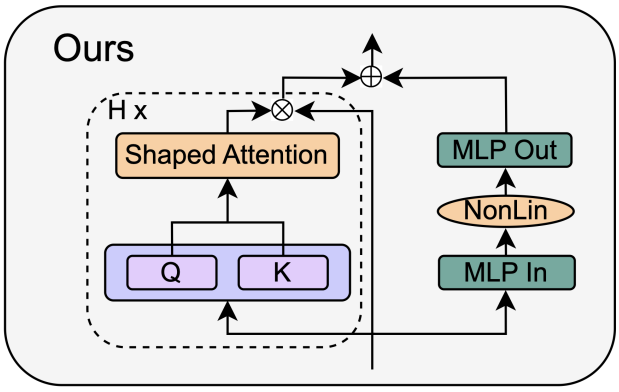

Figure 1: Attention

5.1 Interpretability

Lorem ipsum dolor sit amet, consectetur adipisci elit, sed eiusmod tempor incidunt ut labore et do-

lore magna aliqua. Ut enim ad minim veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

5.2 Generalization

Lorem ipsum dolor sit amet, consectetur adipisci elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

6 Conclusion

Lorem ipsum dolor sit amet, consectetur adipisci elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

References

[1] Peter Belcák et al. *FACT: Learning Governing Abstractions Behind Integer Sequences*. Sept. 2022. arXiv: 2209.09543 [cs].

[2] Arthur Conmy et al. *Towards Automated Circuit Discovery for Mechanistic Interpretability*. Oct. 2023. DOI: 10.48550/arXiv.2304.14997. arXiv: 2304.14997 [cs].

[3] T. M. Cover and Joy A. Thomas. *Elements of Information Theory*. 2nd ed. Hoboken, N.J: Wiley-Interscience, 2006. ISBN: 978-0-471-24195-9.

[4] Anthony Hu et al. *GAIA-1: A Generative World Model for Autonomous Driving*. Sept. 2023. DOI: 10.48550/arXiv.2309.17080. arXiv: 2309.17080 [cs].

[5] Neel Nanda et al. *Progress Measures for Grokking via Mechanistic Interpretability*. Oct. 2023. arXiv: 2301.05217 [cs].

[6] Ashish Vaswani et al. *Attention Is All You Need*. Dec. 2017. DOI: 10.48550/arXiv.1706.03762. arXiv: 1706.03762 [cs].