LLM'S USE A SURPRISINGLY SIMPLE MECHANISM TO RETRIEVE STORED KNOWLEDGE

Summary by: Gene Olafsen

REFERENCES

- 2308.09124.pdf (arxiv.org)

OVERVIEW

- Researches from MIT and elsewhere found that complex large language machine-learning models use a simple mechanism to retrieve stored knowledge when they respond to a user prompt.
- Researches can leverage these simple mechanisms to see what the model knows about different subjects, and also possibly correct false information that it has stored.

INFORMATION RETRIEVAL

- Researchers at MIT studied the mechanisms at work within LLM's machine-learning models to retrieve stored knowledge.

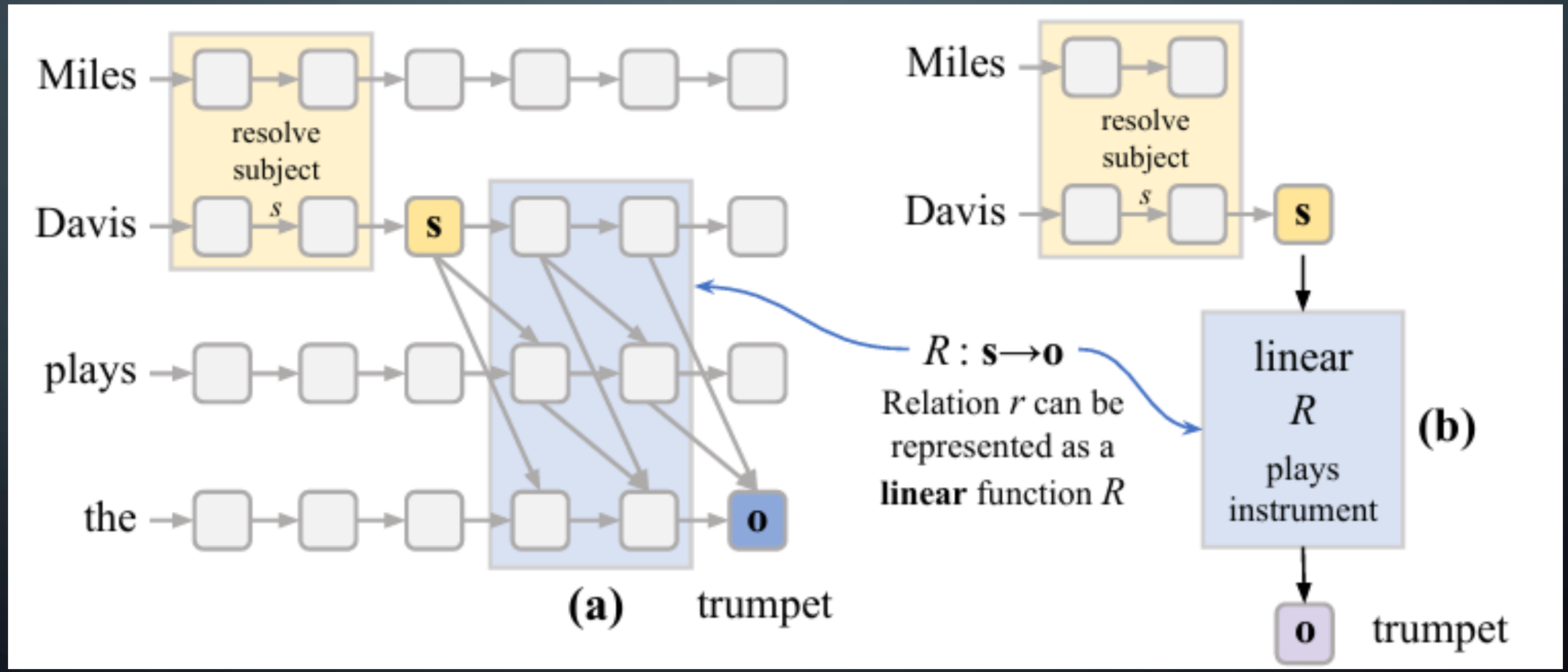
SURPRISING RESULT

- LLMs often use a very simple linear function to recover and decode stored facts. Moreover, the model uses the same decoding function for similar types of facts. Linear functions, equations with only two variables and no exponents, capture the straightforward, straight-line relationship between two variables.

PROBING SUBJECTS

- The researchers showed that, by identifying linear functions for different facts, they can probe the model to see what it knows about new subjects, and where within the model that knowledge is stored.
- Using a technique they developed to estimate these simple functions, the researchers found that even when a model answers a prompt incorrectly, it has often stored the correct information. Future researchers could use such an approach to find and correct falsehoods inside the model, which could reduce a model's tendency to sometimes give incorrect or nonsensical answers- commonly referred to as 'hallucinations'

MILES DAVIS PLAYS THE TRUMPET



Connecting 'Subject' with 'Object'

SIMPLE MECHANISMS

- Evan Hernandez, an electrical engineering and computer science (EECS) graduate student and co-lead author of a paper said "Even though these models are really complicated, nonlinear functions that are trained on lots of data and are very hard to understand, there are sometimes really simple mechanisms working inside them. This is one instance of that."
- Much of the knowledge stored in a transformer can be represented as relations that connect subjects and objects.
- As a transformer gains more knowledge, it stores additional facts about a certain subject across multiple layers. If a user asks about that subject, the model must decode the most relevant fact to respond to the query.

PAPER

- "In GPT and LLaMA models, we search for LREs encoding 47 different relations, covering more than 10k facts relating famous entities (The Space Needle, is located in, Seattle), commonsense knowledge (banana, has color, yellow), and implicit biases (doctor, has gender, man).
- In 48% of the relations we tested, we find robust LREs that faithfully recover subject–object mappings for a majority of the subjects.
- Furthermore, we find that LREs can be used to edit subject representations (Hernandez et al., 2023) to control LM output. "

FUNCTIONS FOR RELATIONSHIPS

- The researchers developed a method to estimate these simple functions, and then computed functions for 47 different relations, such as “capital city of a country” and “lead singer of a band.”

LINEAR RELATIONAL EMBEDDING (LRE)

constraints of geometry and the flexibility of the triplet representation. In many approaches, subject and object entities s and o are represented as vectors $\mathbf{s} \in \mathbb{R}^m, \mathbf{o} \in \mathbb{R}^n$; for a given relation r, we define a **relation function** $R : \mathbb{R}^m \rightarrow \mathbb{R}^n$, with the property that when (s, r, o) holds, we have $\mathbf{o} \approx R(\mathbf{s})$.

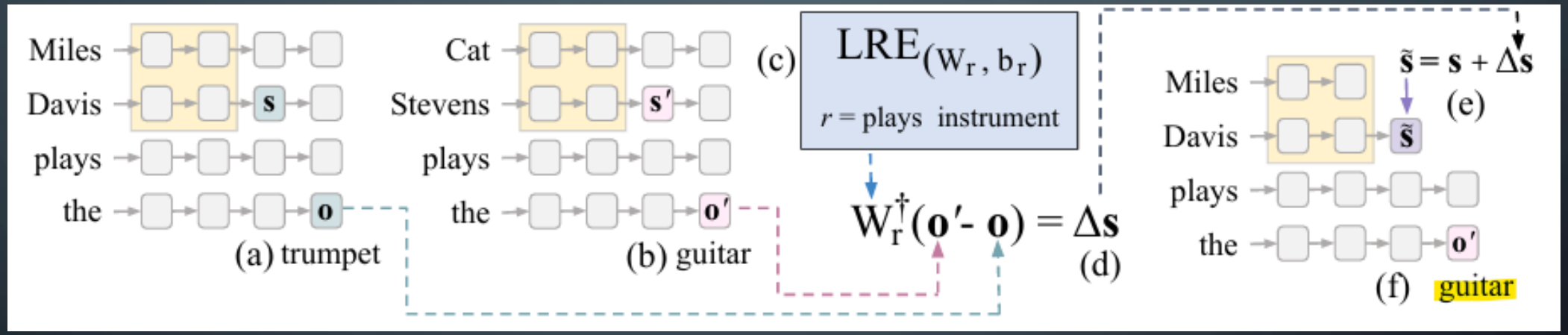
One way to implement R is to use linear transformations to represent relations. For instance, in *linear relational embedding* (Paccanaro & Hinton, 2001), the relation function has the form $R(\mathbf{s}) = W_r\mathbf{s}$ where W_r is a matrix depending on relation r. A modern example of this encoding can be seen in the positional encodings of many transformers (Vaswani et al., 2017). More generally, we can write R as an *affine* transformation, learning both a linear operator W_r and a translation b_r (Lin et al., 2015; Yang et al., 2021). There are multiple variations on this idea, but the basic relation function is:

$$R(\mathbf{s}) = W_r\mathbf{s} + b_r. \tag{1}$$

LRE – FAITHFULNESS AND CAUSALITY

- For a linear relation operator LRE is a good approximation of the transformer's decoding algorithm, it should satisfy two properties:
  - Faithfulness - When applied to new subjects s, the output of LRE(s) should make the same predictions as the transformer.
  - Causality - If a learned LRE is a good description of the LM's decoding procedure, it should be able to model causal influence of the relational embedding on the LM's predictions.

'SUBJECT' REPLACEMENT



ARE LRES FAITHFUL TO RELATIONS?

- From the paper: "Our method achieves over 60% faithfulness for almost half of the relations, indicating that those relations are linearly decodable from the subject representation."

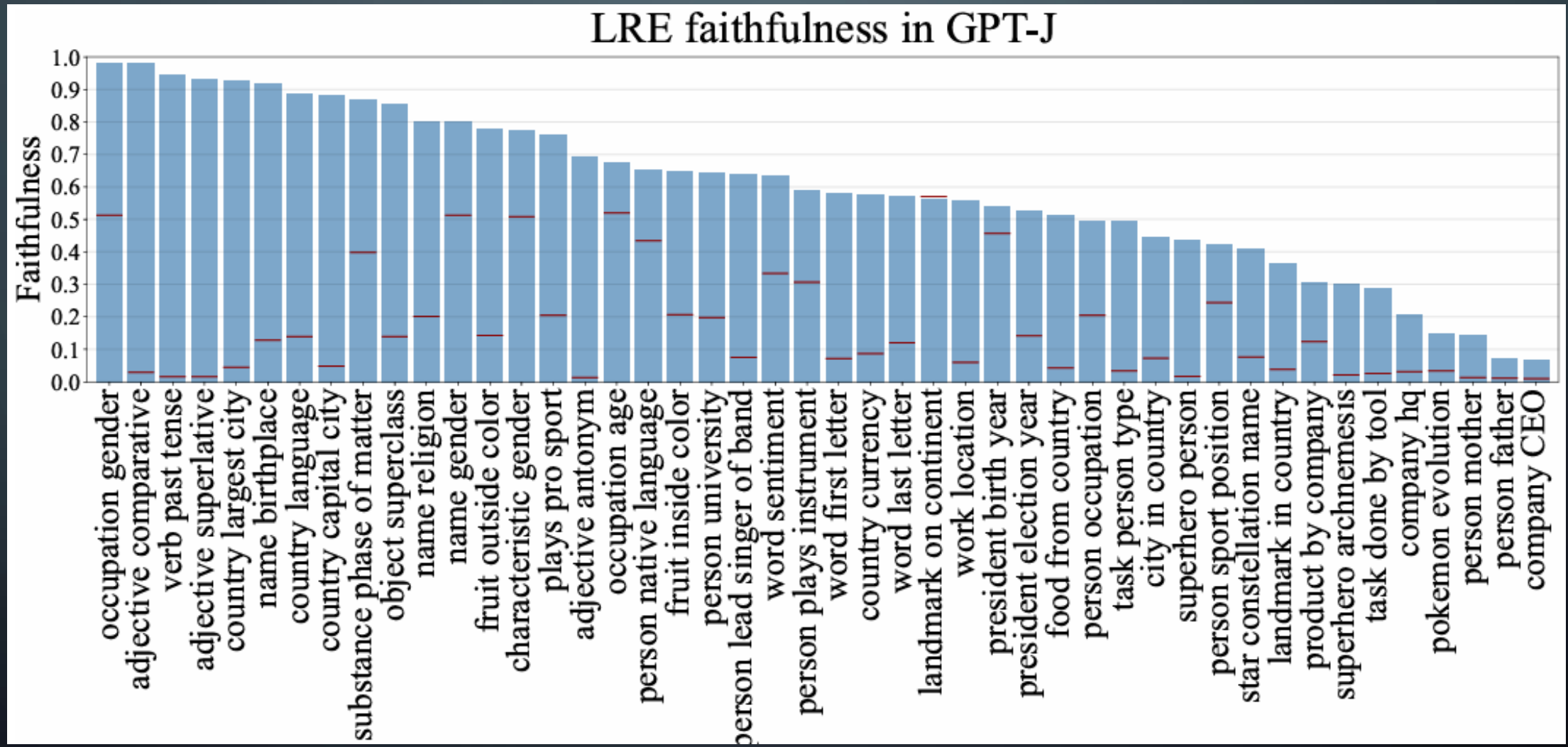

LRE faithfulness in GPT-J
Faithfulness
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
occupation gender
adjective comparative
verb past tense
adjective superlative
country largest city
name birthplace
country language
country capital city
substance phase of matter
object superclass
name religion
name gender
fruit outside color
characteristic gender
plays pro sport
adjective antonym
occupation age
person native language
fruit inside color
person university
person lead singer of band
word sentiment
person plays instrument
word first letter
country currency
word last letter
landmark on continent
work location
president birth year
president election year
food from country
person occupation
task person type
city in country
superhero person
person sport position
star constellation name
landmark in country
product by company
superhero archnemesis
task done by tool
company hq
pokemon evolution
person mother
person father
company CEO

FIGURE ANALYSIS SUMMARY

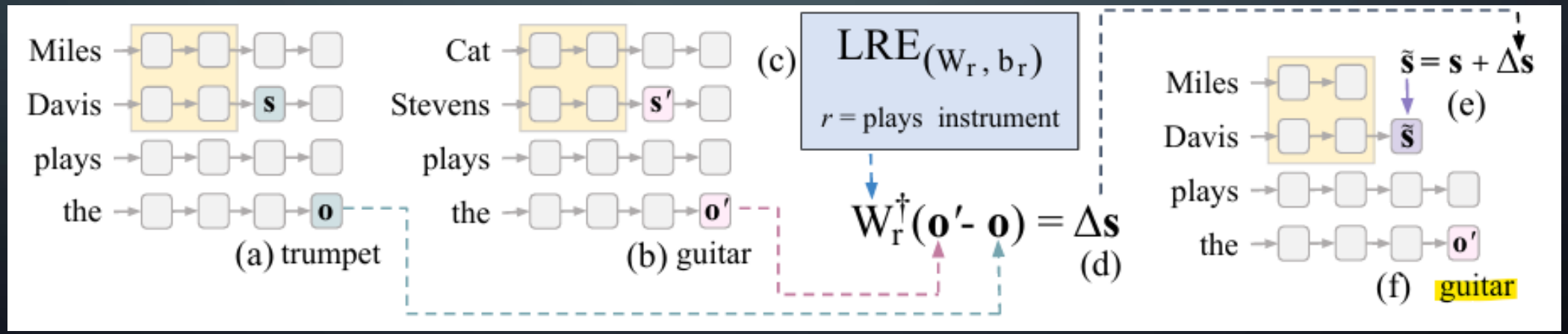
- Relation-wise LRE faithfulness to LM computation F. Horizontal red lines per relation indicate accuracy of a random-guess baseline. LRE is consistently better than random guess and is predictive of the behavior of the transformer on most relations. However, for some relations such as "company CEO" or "task done by tool", the transformer LM deviates from LRE, suggesting non-linear model computation for those relations.

POSSIBLE MULTIPLE LAYER ENCODINGS

- No method reaches over 6% faithfulness on the Company CEO relation, despite GPT-J accurately predicting the CEOs of 69 companies when prompted.
- Indicating that a more involved, non-linear decoding approach is employed by the model to make those predictions. Interestingly, the relations that exhibit this behavior the most are those where the range is the names of people or companies.
- One possible explanation is that these ranges are so large that the LM cannot reliably linearly encode them at a single layer, and relies on a more complicated encoding procedures possibly involving multiple layers.

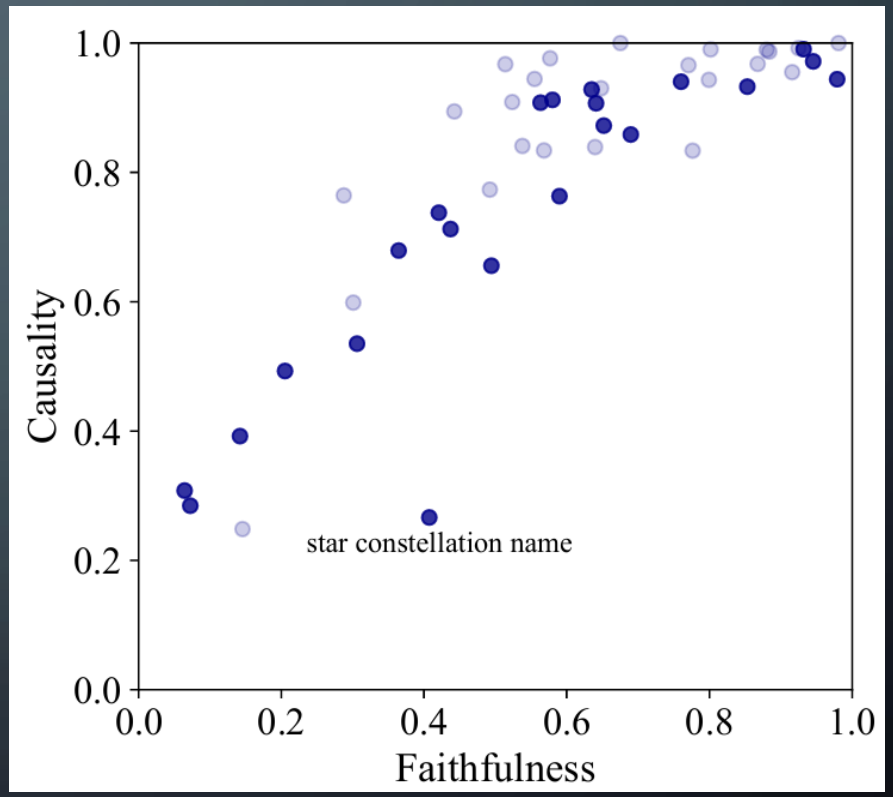
DO LRES CAUSALLY CHARACTERIZE MODEL PREDICTIONS?

- To show that LREs causally influence LM predictions, we follow the procedure described in the figure below to use the inverse of LRE to change the LM's predicted object for a given subject.



CAUSALITY VS FAITHFULNESS

- This figure depicts a strong linear correlation between the metrics when the hyperparameters were selected to achieve best causal influence.



ERASING AN EMBEDDING