Weekly Sync

5/11/24

Paper Review

Article

A Mathematical Investigation of Hallucination and Creativity in GPT Models

Minhyeok Lee

School of Electrical and Electronics Engineering, Chung-Ang University, Seoul 06974, Republic of Korea; mlee@cau.ac.kr

Abstract: In this paper, we present a comprehensive mathematical analysis of the hallucination phenomenon in generative pretrained transformer (GPT) models. We rigorously define and measure hallucination and creativity using concepts from probability theory and information theory. By introducing a parametric family of GPT models, we characterize the trade-off between hallucination and creativity and identify an optimal balance that maximizes model performance across various tasks. Our work offers a novel mathematical framework for understanding the origins and implications of hallucination in GPT models and paves the way for future research and development in the field of large language models (LLMs).

Keywords: generative pretrained transformers; large language model; LLM; GPT; ChatGPT; hallucination; creativity

MSC: 68T27

Paper Review

Wait... didn't we do this before...?

She's getting lazier...

Or is it dementia sia...

Article

A Mathematical Investigation of Hallucination and Creativity in GPT Models

Minhyeok Lee

School of Electrical and Electronics Engineering, Chung-Ang University, Seoul 06974, Republic of Korea; mlee@cau.ac.kr

Abstract: In this paper, we present a comprehensive mathematical analysis of the hallucination phenomenon in generative pretrained transformer (GPT) models. We rigorously define and measure hallucination and creativity using concepts from probability theory and information theory. By introducing a parametric family of GPT models, we characterize the trade-off between hallucination and creativity and identify an optimal balance that maximizes model performance across various tasks. Our work offers a novel mathematical framework for understanding the origins and implications of hallucination in GPT models and paves the way for future research and development in the field of large language models (LLMs).

Keywords: generative pretrained transformers; large language model; LLM; GPT; ChatGPT; hallucination; creativity

MSC: 68T27

Paper Review

Wait... didn't we do this before...?

Shhh.... Quiet.

Or is it dementia sia...

Article

A Mathematical Investigation of Hallucination and Creativity in GPT Models

Minhyeok Lee

School of Electrical and Electronics Engineering, Chung-Ang University, Seoul 06974, Republic of Korea; mlee@cau.ac.kr

Abstract: In this paper, we present a comprehensive mathematical analysis of the hallucination phenomenon in generative pretrained transformer (GPT) models. We rigorously define and measure hallucination and creativity using concepts from probability theory and information theory. By introducing a parametric family of GPT models, we characterize the trade-off between hallucination and creativity and identify an optimal balance that maximizes model performance across various tasks. Our work offers a novel mathematical framework for understanding the origins and implications of hallucination in GPT models and paves the way for future research and development in the field of large language models (LLMs).

Keywords: generative pretrained transformers; large language model; LLM; GPT; ChatGPT; hallucination; creativity

MSC: 68T27

Why and What This Paper is About?

- **Motivation**
  - How to define hallucination mathematically
  - How to define creativity mathematically
  - Is there a way to optimize for both hallucination and creativity mathematically?
- **Objectives**
  - Provide a mathematical framework to balance hallucination and creativity in LLMs (esp. transformer-based LLMs)
  - Introduce *hallucination loss* and *creativity score* metrics

Why and What This Paper is About?

- **Background & Current Gap**
  - Past methods have explored the hallucinatory behaviours
  - Past methods also have suggested ways to mitigate hallucinations
  - But few studies have provided a *quantitative approach* to balance hallucination and creativity
  - This paper addresses that gap, while also explaining how these elements interact

Approach & Results

- **Methods**
  - Very math-heavy. I almost died.
  - It uses advanced mathematical tools such as probability theory, information theory, and optimization strategies
  - Since I am not that smart, I will be explaining only the intuition behind each concept explained in the paper

1. Defining hallucination

Definition 6. *Let $P_{task}(x_{i+1}|x_1, x_2, \ldots, x_i)$ denote the probability distribution of the next token in the sequence, as conditioned on the specific task requirements. The performance metric of a GPT model is defined as the expected KL divergence between the task-specific distribution and the model's predicted distribution:*

$$\mathcal{P}(\Theta) = \mathbb{E}_{(x_1,\ldots,x_n)\sim P_{task}}[D_{KL}(P_{task}(x_{i+1}|x_1, x_2, \ldots, x_i)||P_{model}(x_{i+1}|x_1, x_2, \ldots, x_i; \Theta))]. \quad (21)$$

Approach & Results

1. Defining hallucination

Definition 6. *Let $P_{task}(x_{i+1}|x_1, x_2, \ldots, x_i)$ denote the probability distribution of the next token in the sequence, as conditioned on the specific task requirements. The performance metric of a GPT model is defined as the expected KL divergence between the task-specific distribution and the model's predicted distribution:*

$$\mathcal{P}(\Theta) = \mathbb{E}_{(x_1,\ldots,x_n)\sim P_{task}}[D_{KL}(P_{task}(x_{i+1}|x_1, x_2, \ldots, x_i)||P_{model}(x_{i+1}|x_1, x_2, \ldots, x_i; \Theta))]. \quad (21)$$

- P(Θ) is called *performance metric*
- Calculated as the expected KL divergence between two distributions
    1. P_{task} → the **correct** probability distribution of the next token. It reflects the ground truth.
    2. P_{model} -> the model's **predicted** distribution of the next token, parameterized by Θ which is temperature

Approach & Results

1. Defining hallucination

Definition 6. *Let $P_{task}(x_{i+1}|x_1, x_2, \ldots, x_i)$ denote the probability distribution of the next token in the sequence, as conditioned on the specific task requirements. The performance metric of a GPT model is defined as the expected KL divergence between the task-specific distribution and the model's predicted distribution:*

$$\mathcal{P}(\Theta) = \mathbb{E}_{(x_1,\ldots,x_n)\sim P_{task}}[D_{KL}(P_{task}(x_{i+1}|x_1, x_2, \ldots, x_i)||P_{model}(x_{i+1}|x_1, x_2, \ldots, x_i; \Theta))]. \quad (21)$$

- KL divergence
  - It quantifies how much one distribution diverges from a second, true distribution
  - In this case, it estimates how GPT token distribution estimate the ground truth distribution

Approach & Results

1. Defining hallucination
   - KL divergence
     - P(x) is the true probability of the token x
     - Q(x) is the model's estimated probability of the token x
     - If P(x) and Q(x) are similar, then ln P(x) / Q(x) => 0
     - As such, KL divergence will be smaller
     - Essentially, it measures how much info is lost when we use Q to predict P

$$D_{\mathrm{KL}}(p(x)||q(x)) = \sum_{x \in X} p(x) ln \quad \frac{p(x)}{q(x)}$$

Approach & Results

1. Defining hallucination

Definition 6. *Let $P_{task}(x_{i+1}|x_1, x_2, \ldots, x_i)$ denote the probability distribution of the next token in the sequence, as conditioned on the specific task requirements. The performance metric of a GPT model is defined as the expected KL divergence between the task-specific distribution and the model's predicted distribution:*

$$\mathcal{P}(\Theta) = \mathbb{E}_{(x_1,\ldots,x_n)\sim P_{task}}[D_{KL}(P_{task}(x_{i+1}|x_1, x_2, \ldots, x_i)||P_{model}(x_{i+1}|x_1, x_2, \ldots, x_i; \Theta))]. \quad (21)$$

- Back to the formal definition of hallucination, it calculates the expectation of KL divergence over all possible tokens
- Minimizing P(Θ) would mean the model's prediction is closer to the ground truth => minimal hallucination
- So, P(Θ) can be used as a measure of hallucination loss, **the lower, the better**

Approach & Results

2. Defining creativity

Definition 4. *Let $p(x_{i+1}|x_1, x_2, \ldots, x_i; \Theta)$ denote the probability distribution of the next token in the sequence, as given by (2). The creativity associated with the GPT model's prediction at position $i+1$ is defined as the entropy of this distribution normalized by the maximum entropy:*

$$C(x_{i+1}|x_1, x_2, \ldots, x_i; \Theta) = \frac{H(x_{i+1}|x_1, x_2, \ldots, x_i; \Theta)}{H_{\max}(x_{i+1})}, \tag{19}$$

where $H_{\max}(x_{i+1})$ is the maximum entropy achievable for the given vocabulary $\mathcal{V}$, which occurs when all tokens have uniform probability.

- Creativity, on the other hand, is calculated by **normalizing entropy**
- First, H_{max}, which is the maximum entropy of distribution is calculated
  - H_{max} occurs when all tokens are equally likely
- Then, the entropy of model's predicted distribution over the next token is calculated

Approach & Results

2. Defining creativity

Definition 4. *Let $p(x_{i+1}|x_1, x_2, \ldots, x_i; \Theta)$ denote the probability distribution of the next token in the sequence, as given by (2). The creativity associated with the GPT model's prediction at position $i+1$ is defined as the entropy of this distribution normalized by the maximum entropy:*

$$C(x_{i+1}|x_1, x_2, \ldots, x_i; \Theta) = \frac{H(x_{i+1}|x_1, x_2, \ldots, x_i; \Theta)}{H_{\max}(x_{i+1})}, \tag{19}$$

where $H_{\max}(x_{i+1})$ is the maximum entropy achievable for the given vocabulary $\mathcal{V}$, which occurs when all tokens have uniform probability.

- Intuition => entropy measure how spread out the probability of the next token
- Higher entropy = more creative outputs
- Normalized so that it is scaled between 0 and 1 consistently, making it easy to interpret across different models with varying vocab sizes

Approach & Results

3. Introducing a trade off parameter α

Definition 5. *Let $\mathcal{M}(\alpha)$ be a GPT model parametrized by $\alpha \in [0,1]$. We define the hallucination–creativity trade-off parameter α as the weighting factor that balances the contribution of the hallucination-related prediction error and the creativity of the model in the model's objective function:*

$$\begin{aligned} J(\Theta,\alpha) =& (1-\alpha)\cdot \mathbb{E}_{(x_1,\ldots,x_n)\sim P_{true}}[D_{KL}(P_{true}(x_{i+1}|x_1,x_2,\ldots,x_i)||P_{model}(x_{i+1}|x_1,x_2,\ldots,x_i;\Theta))] \\ & -\alpha\cdot \mathbb{E}_{(x_1,\ldots,x_n)\sim P_{true}}[C(x_{i+1}|x_1,x_2,\ldots,x_i;\Theta)], \end{aligned} \tag{20}$$

where D_{KL} denotes the KL divergence and C denotes the creativity measure as defined in (19).

Approach & Results

3. Introducing a trade off parameter α

Definition 5. *Let $\mathcal{M}(\alpha)$ be a GPT model parametrized by $\alpha \in [0,1]$. We define the hallucination–creativity trade-off parameter α as the weighting factor that balances the contribution of the hallucination-related prediction error and the creativity of the model in the model's objective function:*

$$\begin{aligned} J(\Theta,\alpha) =& (1-\alpha)\cdot \mathbb{E}_{(x_1,\ldots,x_n)\sim P_{true}}[D_{KL}(P_{true}(x_{i+1}|x_1,x_2,\ldots,x_i)||P_{model}(x_{i+1}|x_1,x_2,\ldots,x_i;\Theta))] \\ & -\alpha\cdot \mathbb{E}_{(x_1,\ldots,x_n)\sim P_{true}}[C(x_{i+1}|x_1,x_2,\ldots,x_i;\Theta)], \end{aligned} \tag{20}$$

where D_{KL} denotes the KL divergence and C denotes the creativity measure as defined in (19).

To solve (22), we first examine the relationship between the objective function $J(\Theta,\alpha)$ in (20) and the performance metric $\mathcal{P}(\Theta)$ in (21).

For a fixed Θ, the objective function can be written as follows:

$$J(\Theta,\alpha) = (1-\alpha)\cdot J_{\text{hallucination}}(\Theta) - \alpha\cdot J_{\text{creativity}}(\Theta), \tag{23}$$

where $J_{\text{hallucination}}(\Theta)$ and $J_{\text{creativity}}(\Theta)$ represent the hallucination-related prediction error and the creativity of the model, respectively.

Approach & Results

3. Introducing a trade off parameter α

To solve (22), we first examine the relationship between the objective function $J(\Theta, \alpha)$ in (20) and the performance metric $\mathcal{P}(\Theta)$ in (21).

For a fixed Θ, the objective function can be written as follows:

$$J(\Theta, \alpha) = (1 - \alpha) \cdot J_{\text{hallucination}}(\Theta) - \alpha \cdot J_{\text{creativity}}(\Theta), \tag{23}$$

where $J_{\text{hallucination}}(\Theta)$ and $J_{\text{creativity}}(\Theta)$ represent the hallucination-related prediction error and the creativity of the model, respectively.

- High α would encourage creativity, while low α would minimize hallucination
- If the model is trained to optimize this objective function, we can finally find that “sweet spot” between hallucination and creativity

Approach & Results

- **Key Findings**
  - Provide a mathematical proof that trade off between minimizing hallucination and maximizing creativity exists
  - Proof the existence of an optimal trade off parameter α^* depending on the requirements of the task

Analysis

- **Strengths**
  - Mathematically rigorous. Honestly, I only touched the surface of the math behind this paper.
  - Having an adjustable parameter α makes the model versatile and adaptable to different needs
  - Justified that hallucination and creativity is just two sides of the same coin

Analysis

- **Limitations**
  - Too theoretical. No implementation.
  - Non-convex optimization challenges. Hard to find the global optima, resulting in difficulties in finding the ideal α
  - Paper admitted that, but offer no resolution ☹
- **Impact**
  - Potentially a good way in measuring creativity => good for my use case!
  - Reliable and versatile framework that can be implemented relatively easily!

Questions

1. How do you get P_{task} (refer to slide 7, definition 21)?
2. How relevant is the calculation of hallucination loss via KL divergence to what I am doing right now?
3. Most of the time, entropy is being used as a measure of “uncertainty”. However, this paper introduced is as a measure of creativity instead. Is it valid?
4. Can I use the normalized entropy calculation as creativity score for my paper?
5. For the objective function (refer to equation 20), must I train an LLM that optimize this objective function? How does it work, exactly?